Smoothing-based initialization for learning-to-forecast algorithms

DOI:
10.1017/S1365100517000128

Document Version
Accepted author manuscript

Smoothing-based initialization for learning-to-forecast algorithms*

MICHELE BERARDI
University of Manchester

JAQUESON K. GALIMBERTI
ETH Zurich

*An earlier version of this paper was presented at the 2013 Computing in Economics and Finance conference in Vancouver. We thank to our discussants for helpful comments. We also gratefully acknowledge the comments provided by one Associate Editor and two referees. Finally, we thank the Editor Professor William A. Barnett for the quick responsiveness and handling of our submission. Any remaining errors are ours.

Proposed running head: Smoothing initials for learning-to-forecast algorithms

Corresponding author: Jaqueson K. Galimberti

E-mail: galimberti@kof.ethz.ch

Phone/fax: +41 44 6328529 / +41 44 6321218

Mailing address: KOF Swiss Economic Institute, ETH Zürich, LEE G 116, Leonhardstrasse 21, 8092 Zürich, Switzerland

Abstract

Under adaptive learning, recursive algorithms are proposed to represent how agents update their beliefs over time. For applied purposes these algorithms require initial estimates of agents perceived law of motion. Obtaining appropriate initial estimates can become prohibitive within the usual data availability restrictions of macroeconomics. To circumvent this issue we propose a new smoothing-based initialization routine that optimizes the use of a training sample of data to obtain initials consistent with the statistical properties of the learning algorithm. Our method is generically formulated to cover different specifications of the learning mechanism, such as the Least Squares and the Stochastic Gradient algorithms. Using simulations we show that our method is able to speed up the convergence of initial estimates in exchange for a higher computational cost.

Keywords: learning algorithms, initialization, smoothing, expectations.

JEL codes: C63, D84, E37.

1 Introduction

Under adaptive learning agents are assumed to act like econometricians, forming their expectations with forecast functions that are adjusted as new data becomes available. As the beliefs inherent in such forecasting models are updated according to explicit recursive mechanisms, or learning algorithms, an initial estimate of agents beliefs is required to commence this learning-to-forecast process. In this paper we propose a smoothing-based initialization method, designed to optimize the use of information available in feasible training samples of initial data. Our main contribution is to show that the smoothing method is capable of accelerating the convergence of the estimates of a forecasting model within a restricted training sample of data, and that this method is also robust to changes in the specification of the learning algorithm and its calibration.

The need of learning initials is particularly relevant for applied purposes, where adaptive learning provides a new channel through which the effects of expectation shocks can dissipate dynamically over the economy (e.g., Sargent, 1999; Orphanides and Williams, 2005; Eusepi and Preston, 2011; Milani, 2011). Frequently, these applications involve the computational implementation of the learning algorithms, for which the initial estimates are indispensable. Nevertheless, few attempts have been done in the previous literature to rationalize an initialization method[1].

Here we take up this issue focusing on learning-to-forecast exercises, where observations are generated exogenously to the learning process, thence representing the limiting case of atomistic agents whose individual forecasts and decisions cannot affect aggregate outcomes. This allows us to obtain a clearer evaluation of the accuracy of initialization methods than one would obtain with the joint estimation of other macroeconomic model structural parameters, which is known to be plagued by weak identification issues (see Chevillon et al., 2010)[2].

Our point of departure is a unified framework, presented in Section 2, under which the main learning algorithms considered in the literature, namely the Least Squares (LS) and the Stochastic Gradient (SG), are obtained as special cases of the Kalman filter associated to a time-varying parameters model of the economy. More specifically, Berardi and Galimberti (2013) have recently shown how to extend the asymptotic correspondences between these algorithms

to hold exactly in transient phases too, hence allowing for a unified approach to initializations. From these correspondences, long standing Kalman smoothing results can be readily translated into smoothing routines for the estimates obtained from each of the above learning algorithms, and we develop our routine using these premises in Section 3.

We then conduct a simulation exercise, presented in Section 4, generating artificial data that mimic the statistical properties of inflation[3], and evaluate our procedure in comparison with two other training sample-based methods found in the previous literature, . Our guiding principle in this evaluation is the idea that an initial estimate should reflect the beliefs implied by a learning process that was already in motion prior to the sample beginning. We show that our approach is able to speed up the learning algorithms convergence to their long run operation without impairing the feasibility of the learning analysis by requiring too many observations for the initials training. This solution, however, comes at the cost of an increased, but feasible, computational burden. The alternative methods, in contrast, are found to lack the robustness provided by our unifying framework: the accuracy of the initial estimates provided by the traditional methods is dependent on the specification of the learning algorithm and on the gain calibration.

We conclude this paper with some remarks in Section 5.

2 Learning-to-forecast Framework

Consider an estimation context faced by a real-time agent wishing to obtain inferences about the law of motion of a variable of interest, say y_t. From an economic perspective, these inferences can be thought of as the middle step agents undertake in a process of learning-to-forecast in order to form their expectations.

To narrow down our focus, we assume this agent attempts to construct such inferences assuming that y_t is statistically related to other observed variables, say a vector of (pre-determined) variables $\mathbf{x}_t = (x_{1,t}, \ldots, x_{K,t})'$, through a linear regression of the form

$$y_t = \mathbf{x}_t' \boldsymbol{\theta}_t + \varepsilon_t, \tag{1}$$

where $\boldsymbol{\theta}_t = (\theta_{1,t}, \ldots, \theta_{K,t})'$ stands for a vector of (possibly time-varying) coefficients, and ε_t denotes a (Gaussian[4]) white noise disturbance with variance given by σ_t^2. Both coefficients and disturbances are assumed not to be directly observable by the agent.

Under this context, a technique for estimation of $\boldsymbol{\theta}_t$ is required to allow the agent to construct inferences for y_t on the basis of (1). In the literature of learning and expectations in macroeconomics (see Evans and Honkapohja, 2001) recursive algorithms have been proposed for this task. Two of the main forms adopted are the LS and the SG specifications.

2.1 Learning algorithms

Algorithm 1 (LS)**.** *Under the estimation context of* (1)*, the* LS *algorithm assumes the form of*

$$\hat{\boldsymbol{\theta}}_t^{LS} = \hat{\boldsymbol{\theta}}_{t-1}^{LS} + \gamma_t \mathbf{R}_t^{-1} \mathbf{x}_t \left(y_t - \mathbf{x}_t' \hat{\boldsymbol{\theta}}_{t-1}^{LS} \right), \tag{2}$$

$$\mathbf{R}_t = \mathbf{R}_{t-1} + \gamma_t \left(\mathbf{x}_t \mathbf{x}_t' - \mathbf{R}_{t-1} \right), \tag{3}$$

where γ_t is a learning gain parameter, and $\mathbf{R}_t$ stands for an estimate of regressors matrix of second moments.

Algorithm 2 (SG)**.** *Under the estimation context of* (1)*, the* SG *algorithm is given by*

$$\hat{\boldsymbol{\theta}}_t^{SG} = \hat{\boldsymbol{\theta}}_{t-1}^{SG} + \mu_t \mathbf{x}_t \left(y_t - \mathbf{x}_t' \hat{\boldsymbol{\theta}}_{t-1}^{SG} \right), \tag{4}$$

with μ_t standing for the learning gain parameter.

Since the seminal works of Bray (1982); Marcet and Sargent (1989) the LS algorithm has been taken as the natural choice to represent agents mechanism of adaptive learning. This was due to its widespread popularity between econometricians. The SG algorithm, on the other hand, provides a computationally simpler alternative, a feature clearly apparent in (4) for the absence of the LS "normalization" step given by the inverse of the matrix of second moments. For this reason some authors have advocated for its use as a more plausible learning device from a bounded rationality standpoint (Barucci and Landi, 1997; Evans and Honkapohja, 1998; Christev and Slobodyan, 2014).

Both the LS and the SG algorithms require the specification of a sequence of learning gains. The learning gain stands for a parameter determining how quickly a given information is incorporated into the algorithm's coefficients estimates. Three of the main alternatives for the specification of this learning gain are those of a time-decreasing, a time-constant, and a time-varying (not restricted to be decreasing) sequence of values. Our focus in this study will be on the constant gain specification, which has been in the spotlight of most applied research since Sargent (1999). Such a choice naturally sprouts from the tracking capabilities associated to the constant gain specification and its suitability for time-varying environments.

2.2 Statistical rationale of learning initials

Recursive estimation algorithms are statistically characterized by undergoing through two main distinct phases: a transient and a steady state one. Since initial beliefs should reflect the continuation of an estimation process that was already running prior to the sample beginning, we argue that an initialization method should be purposefully designed to provide estimates as close as possible to the algorithm's steady state operation. The separating frontier between these phases, nevertheless, is not clear-cut. To obtain an assessment, it is common practice (see Haykin, 2001, p. 266) to focus on a statistical measure of interest and construct the algorithm's learning curves, which represent how that measure evolves through time. Roughly, one can then visually lay up bare these phases by identifying the steady state when the statistic settles down. One measure of interest is the Mean-Square Deviation (MSD).

Definition 1 (MSD)**.** *The* MSD *between the actual vector of coefficients in* (1)*,* $\boldsymbol{\theta}_t$*, and the algorithms estimates,* $\hat{\boldsymbol{\theta}}_t$*, is given by*

$$\mathcal{D}_t = E\left[\Delta_t^2\right], \tag{5}$$

where $\Delta_t = \left\|\boldsymbol{\theta}_t - \hat{\boldsymbol{\theta}}_t\right\|$ *stands for the Euclidean norm of the vector of coefficients deviations.*

The MSD is intended to capture the (average) accuracy of the algorithm's estimates. Its evolution through time is also associated with the speed at which the algorithm is able to adjust its estimates to the time-varying system. Optimization of tracking performance is mainly done

through control of the gain parameter, giving rise to a well known trade-off between the tracking speed and the accuracy of estimates (see Benveniste et al., 1990, Part I, Chapters 1 and 4). In our context, the MSD measure serves to the purpose of defining a metric that will be the basis of our main evaluation criterion of initializations.

Definition 2 (MISALIGNMENT)**.** *The* MISALIGNMENT *of an algorithm estimates at period* t*, with respect to its* MSD*, can be measured by*

$$\mathcal{M}_t = \frac{\left|\mathcal{D}_t - \overline{\mathcal{D}}\right|}{\widetilde{\mathcal{D}}_t}, \tag{6}$$

where $\overline{\mathcal{D}} = \lim_{t\to\infty} \mathcal{D}_t$ *stands for the steady state level of the algorithm's* MSD*, and* $\widetilde{\mathcal{D}}_t = \sqrt{E\left[\left(\Delta_t^2 - \mathcal{D}_t\right)^2\right]}$ *stands for its standard deviation.*

Our measure of MISALIGNMENT has the appealing interpretation of representing the distance between the algorithm's current MSD and its steady state level in terms of standard deviations. For simulation purposes, (5) and (6) can be readily evaluated by computing their sample counterparts.

3 Learning Initialization

Our analysis will focus on initializations obtained on the basis of a training sample of observations[5]. This is especially recommended for the cases where there is not enough previous knowledge about the system under estimation, such as in empirical applications, so as to allow an educated guess. The main difficulty that initialization methods based on training samples face for their use in empirical applications relates to the trade-off between the degree of convergence of the algorithm estimates to its long run operation and the number of observations required to achieve such convergence. Namely, while devoting additional data to the initialization procedure tends to favor its adherence to the ongoing estimation process, by expanding the room for the algorithm's convergence to play, the number of observations left for the post-initialization analysis is reduced. We now propose a new method aimed at mitigating this trade-off through an increase in the computational burden required for the initialization. The

main idea draws upon the use of a smoothing procedure within a training sample of data.

3.1 Smoothing-based initialization

Let $\hat{\boldsymbol{\theta}}_{t|k}$ stand for an estimate of period t vector of coefficients, $\boldsymbol{\theta}_t$, where k indicates the information period on which the estimates are based. Under this notation, the estimates obtained with the (forward) recursions of the learning algorithms in (2)-(3) and (4) stand for filtered estimates and are given by $\hat{\boldsymbol{\theta}}_{t|t} \equiv \hat{\boldsymbol{\theta}}_t$. The smoothed estimates, on the other hand, stand for (backward-looking) updated inferences on the filtered estimates , i.e., $\hat{\boldsymbol{\theta}}_{t|k}$ with $k \geq t$. Clearly, while the filtered estimates stand for the inferences made on the basis of information available at the period the estimates stand for, the smoothed estimates are inferences obtained as new information about the system becomes available. Due to the use of more information, one can expect the smoothed estimates to be more accurate than the filtered ones. To take advantage of this gain in accuracy to the estimation of learning initials we propose the following procedure.

Using a training sample of N observations, one can start the computation of the learning algorithms from an initial guess, say $\hat{\boldsymbol{\theta}}_0 = 0$, and obtain not only the algorithm's filtered estimates up to $\hat{\boldsymbol{\theta}}_N$, but also its smoothed estimates of $\hat{\boldsymbol{\theta}}_{0|N}$ (we explain how to obtain these further below). This process is also known as the fixed-point smoother, since the updates are applied only to the estimates of a particular period in the past[6]. With these latter at hand, then, one re-starts the estimation process, within the same sample of data, but now assigning the initial in accordance to the smoothed estimate, i.e., $\hat{\boldsymbol{\theta}}_0 = \hat{\boldsymbol{\theta}}_{0|N}$. A new sequence of filtered and smoothed estimates is in this way obtained, and this process can be repeated a few more times until a given convergence criterion is met. For this latter, here we adopted an ϵ-convergence criterion based on the Euclidean distance between filtered and smoothed estimates, under which the above process is repeated until $\left\| \hat{\boldsymbol{\theta}}_0 - \hat{\boldsymbol{\theta}}_{0|N} \right\| < \epsilon$, with ϵ determined experimentally.

3.2 Smoothing recursions

To obtain the smoothed initials associated to the learning algorithms, we make use of a parallel drawn in Berardi and Galimberti (2013) between these algorithms and the Kalman filter applied to the estimation of a time-varying parameters model (see also McGough, 2003). More

specifically, we start by establishing a state-space framework where the coefficients vector of the linear model in (1) evolves according to

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} + \boldsymbol{\omega}_t, \tag{7}$$

where $\boldsymbol{\omega}_t$ is assumed to be (Gaussian) white noise with variances (and covariances) given by $\boldsymbol{\Omega}_t = E\left[\boldsymbol{\omega}_t \boldsymbol{\omega}_t'\right]$. The Kalman filter recursion for estimation of $\hat{\boldsymbol{\theta}}_t \equiv \hat{\boldsymbol{\theta}}_{t+1|t}$ then is given by

$$\begin{aligned}
\hat{\boldsymbol{\theta}}_t &= \hat{\boldsymbol{\theta}}_{t-1} + \mathbf{K}_t \left(y_t - \mathbf{x}_t' \hat{\boldsymbol{\theta}}_{t-1} \right), && (8)\\
\mathbf{K}_t &= \frac{\mathbf{P}_{t-1}\mathbf{x}_t}{\mathbf{x}_t' \mathbf{P}_{t-1}\mathbf{x}_t + \sigma_t^2}, && (9)\\
\mathbf{P}_t &= \left(\mathbf{I} - \frac{\mathbf{P}_{t-1}\mathbf{x}_t\mathbf{x}_t'}{\mathbf{x}_t' \mathbf{P}_{t-1}\mathbf{x}_t + \sigma_t^2} \right) \mathbf{P}_{t-1} + \boldsymbol{\Omega}_t, && (10)
\end{aligned}$$

where $\mathbf{P}_t$ stands for the conditional covariance matrix of the coefficients estimates errors, i.e., $\mathbf{P}_t = E\left[\left(\boldsymbol{\theta}_t - \hat{\boldsymbol{\theta}}_t \right) \left(\boldsymbol{\theta}_t - \hat{\boldsymbol{\theta}}_t \right)' \right]$. Following Berardi and Galimberti (2013), the LS and the SG learning algorithms, as given by (2)-(3) and (4), respectively, can be obtained as special cases of the Kalman filter when

$$\begin{aligned}
\sigma_t^2 &= \frac{\gamma_{t-1}}{\gamma_t} \left(1 - \gamma_t \right), && (11)\\
\boldsymbol{\Omega}_t &= \left(\frac{1 - \sigma_t^2}{\sigma_t^2} \right) \left(\mathbf{I} - \frac{\mathbf{P}_{t-1}\mathbf{x}_t\mathbf{x}_t'}{\mathbf{x}_t' \mathbf{P}_{t-1}\mathbf{x}_t + \sigma_t^2} \right) \mathbf{P}_{t-1}, && (12)
\end{aligned}$$

and

$$\begin{aligned}
\sigma_t^2 &= \mu_t^{-1} - \mathbf{x}_t'\mathbf{x}_t, && (13)\\
\boldsymbol{\Omega}_t &= \mathbf{I} - \left(\mathbf{I} - \frac{\mathbf{P}_{t-1}\mathbf{x}_t\mathbf{x}_t'}{\mathbf{x}_t' \mathbf{P}_{t-1}\mathbf{x}_t + \sigma_t^2} \right) \mathbf{P}_{t-1}, && (14)
\end{aligned}$$

respectively.

Finally, smoothed estimates of the initials can be obtained using the Kalman fixed-point smoother of Anderson and Moore (1979, pp. 170-6). Essentially, these authors have shown how the fixed-point smoothing problem can be solved through the application of the standard Kalman filtering expressions to the original state space model augmented with a state appropri-

ately initialized to represent the fixed-point smoothed estimates. To that end, consider replacing the state-space framework of (1) and (7) by

$$y_t = \begin{bmatrix} \mathbf{x}_t' & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta}_t \\ \boldsymbol{\theta}_t^a \end{bmatrix} + \varepsilon_t, \tag{15}$$

$$\begin{bmatrix} \boldsymbol{\theta}_t \\ \boldsymbol{\theta}_t^a \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta}_{t-1} \\ \boldsymbol{\theta}_{t-1}^a \end{bmatrix} + \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \boldsymbol{\omega}_t, \tag{16}$$

with the state vector at a fixed $t = j$ satisfying $\begin{bmatrix} \boldsymbol{\theta}_j' & \boldsymbol{\theta}_j^{a\prime} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\theta}_j' & \boldsymbol{\theta}_j' \end{bmatrix}$. Thus, we are essentially augmenting the former system with an additional state vector which, due to the assumed "initialization" at period j, will satisfy $\boldsymbol{\theta}_t^a = \boldsymbol{\theta}_j,\ \forall t \geq j$. It follows from this latter observation and the definition of conditional estimates that $\hat{\boldsymbol{\theta}}_{t|t-1}^a = \hat{\boldsymbol{\theta}}_{j|t-1}$, $\hat{\boldsymbol{\theta}}_{t+1|t}^a = \hat{\boldsymbol{\theta}}_{j|t}$, and so on. The coefficients on the right hand side of these equalities are clearly in accordance to what we have defined as fixed-point smoothed estimates, i.e., keeping j fixed we evaluate how the coefficients estimates get updated as time goes on and new observations become available. Furthermore, the state-space system in (15)-(16) is conformable to the application of the Kalman filter, where the updating recursions for $\hat{\boldsymbol{\theta}}_t \equiv \hat{\boldsymbol{\theta}}_{t+1|t}$ will still be given by (8)-(10), and those for $\hat{\boldsymbol{\theta}}_t^a \equiv \hat{\boldsymbol{\theta}}_{t+1|t}^a$ will represent the fixed-point smoothing recursions of $\hat{\boldsymbol{\theta}}_{j|t}$. These latter are given by

$$\hat{\boldsymbol{\theta}}_{j|t} = \hat{\boldsymbol{\theta}}_{j|t-1} + \mathbf{K}_t^a \left(y_t - \mathbf{x}_t' \hat{\boldsymbol{\theta}}_{t-1} \right), \tag{17}$$

$$\mathbf{K}_t^a = \frac{\boldsymbol{\Sigma}_{t-1} \mathbf{x}_t}{\mathbf{x}_t' \mathbf{P}_{t-1} \mathbf{x}_t + \sigma_t^2}, \tag{18}$$

$$\boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_{t-1} \left(\mathbf{I} - \mathbf{K}_t \mathbf{x}_t' \right)', \tag{19}$$

$$\mathbf{P}_{j|t} = \mathbf{P}_{j|t-1} - \boldsymbol{\Sigma}_{t-1} \mathbf{x}_t \mathbf{K}_t^{a\prime}, \tag{20}$$

where $\boldsymbol{\Sigma}_j = \mathbf{P}_j$, and the conditional covariance matrix of the coefficients smoothed estimates errors is here given by (20), i.e., $\mathbf{P}_{j|t} = E\left[\left(\boldsymbol{\theta}_j - \hat{\boldsymbol{\theta}}_{j|t} \right) \left(\boldsymbol{\theta}_j - \hat{\boldsymbol{\theta}}_{j|t} \right)' \right]$. The smoother associated to each learning algorithm, thence, follows automatically from what the different assumptions on (11)-(12) and (13)-(14) imply for these recursions.

3.3 Benchmark initializations

In order to benchmark our evaluation of the smoothing-based initialization method, we consider two alternatives commonly found in the applied literature (see Carceles-Poveda and Giannitsarou, 2007; Berardi and Galimberti, 2016, for comprehensive reviews), both based on training samples. The first is inspired by the engineering literature (e.g., see Ljung and Soderstrom, 1983, pp. 299-303), where it is often suggested that the coefficients should be initialized with the value of zero and the algorithm should be allowed to adjust its estimates over the training sample according to the underlying gain calibration. Given our focus on constant gain specifications of the learning algorithms, we denote this initialization as *Tracking* method.

A second method of initialization involves the use of the decreasing gain block estimation counterpart of the learning algorithms. For the case of the LS algorithm this method is equivalent to the well known Ordinary Least Squares, and has been often adopted in the literature for robustness purposes (see, e.g., Williams, 2003; Orphanides and Williams, 2005; Carceles-Poveda and Giannitsarou, 2007; Slobodyan and Wouters, 2012); hence, we denote this method as *Ordinary* initialization. In the recursive setup, this method corresponds to an initialization of the coefficients from zero, and then updating the estimates within the initial sample using the learning algorithm with decreasing gains. To prevent instabilities into the first estimates we set the decreasing gains as $\gamma_t = \bar{\gamma}/t$, for the LS, and $\mu_t = \bar{\mu}/t$ for the SG, where the upper bounds are derived from stability considerations. In spite of the obvious inconsistency between the gains used in the training and the post-initialization samples, the use of a higher gain in the first training observations tends to accelerate the convergence of the algorithm estimates towards its steady state operation.

4 Comparative Simulation Analysis

4.1 Setup

Our purpose here is to construct the (averaged) learning curves of the algorithms during their initial transient phase and evaluate how their statistical properties are affected by the initial-

izations adopted. Given the stochastic environment under which these algorithms operate, in simulation studies these curves are computed as an average over repeated samples of generated data.

The artificial data is generated according to a linear auto-regression of the form

$$y_t = \theta_t y_{t-1} + \varepsilon_t, \tag{21}$$

where the auto-regressive parameter evolves according to

$$\left(\theta_{t+1} - \bar{\theta}\right) = \beta\left(\theta_t - \bar{\theta}\right) + \omega_{t+1}, \tag{22}$$

and the random disturbances ε_t and ω_{t+1} are zero mean mutually independent distributed as Gaussian with variances given by σ_ε^2 and σ_ω^2, respectively[7]. Notice that if $|\beta| < 1$, then $\bar{\theta}$ may be viewed as the steady-state value of the auto-regressive coefficient in (21). Yet, in order to avoid too quick variations in the statistical properties of the data, the value of β is usually assumed to be very close to unity. In spite of resembling a random walk, this assumption prevents the dynamics of the auto-regressive coefficient to be dominated by the noise variations in its stochastic disturbances.

For the calibration of σ_ε^2, σ_ω^2, $\bar{\theta}$, and β, we take the recommendations of Hamilton (1994, pp. 401-3) as a reference, though adjusting them to our context. One of these adjustments refers to the use of a higher σ_ω^2 in order to accentuate the variations in the estimation environment, and further justify the use of constant-gain algorithms. For the parameters determining the variance and the dynamic persistence of the artificial series, σ_ε^2 and $\bar{\theta}$, respectively, we attempt to obtain a calibration that mimics the statistical properties of inflation data (see Berardi and Galimberti, 2012, for results under output growth-like data). These calibrations are summarized in Table 1. We drew 1,000 different samples of the random disturbances and used them with the DGP given by (21)-(22) for the generation of artificial series with a time dimension of 1,250 observations. We discarded the first 250 of these observations for each sample to avoid sensitivity to the series initializations, for which we used $y_0 = 0$ and $\theta_1 = \bar{\theta}$.

[Table 1 goes here, please]

Apart from these calibrations for the artificial series, we also need to specify how we calibrated the algorithms constant learning gains. Here we first define a set of different values for the LS, which is not sensitive to the scale of the data, and then adjust these gains for the SG case. In order to do this conversion, we need to compute estimates for the upper bounds on the gain calibrations that still ensure stability for each algorithm. The main issue here lies on the determination of this upper bound for the SG algorithm, which is known to be sensitive to the scale of the data (see Evans et al., 2010). We follow the recommendations of Haykin (2001, pp. 258-74) and compute the SG upper bound as $\overline{\mu}_{max} = 2/\lambda_{max}$, where λ_{max} stand for the maximum eigenvalue of the regressors covariance matrix, which for the case of (21) is simply given by the variance of y_t. The LS gain calibrations, specified in Table 1 by $\bar{\gamma}_1$ and $\bar{\gamma}_2$, are then converted to the SG as $\bar{\mu}_i = \overline{\mu}_{max}\left(\bar{\gamma}_i/\sigma_y^2\right)$, where the variance of y_t is approximated taking the auto-regressive coefficient as fixed to its long run value, $\theta_t = \bar{\theta}$. This same variance is used to initialize the matrix of second moments associated to (3) in the LS case. Finally, for the smoothing-based initialization we set the convergence criterion to $\epsilon = 0.01$, a value that according to our experimentation provides initials close to the steady state level of the algorithms' MSDs without requiring too many smoothing repetitions (we discuss the associated computational cost in subsection 4.3 below).

4.2 Simulation Results

The MSD learning curves obtained from the application of the LS and the SG algorithms to inflation-like data are presented in Figures 1 and 2, respectively. We have fixed the number of observations taken for training to the first 75,. and evaluated the MISALIGNMENT of the initial estimates from their corresponding algorithm/gain long run behavior, as defined in (6); visually, the initials MISALIGNMENT can be assessed by looking at the distance between their associated MSD at the end of the training sample and the MSD level the learning algorithm eventually settles down.

[Figures 1 and 2 go here, please]

The initial MISALIGNMENT incurred by each initialization method depends on the gain calibration, and the same dependency is observed with respect to the algorithm's MSD steady

state level, as expected. Different gain values engender different steady state behaviors of the algorithm's estimates. So, if the initialization for a given gain calibration is obtained by using a different gain value, as in the case of the *ordinary* initializations, this initial estimate will tend to be biased in relation to the algorithm's steady state estimates. This is evident in Figures 1 and 2 by the jumps undertaken by the *ordinary* MSD estimates from their after-initialization level to their stable long run level. The *tracking* method also performs poorly in its application to the SG algorithm. The lack of a normalization step in the operation of this algorithm seems to be reflected into its slow rate of convergence to steady state. The number of observations left aside for the *tracking* initialization of the SG algorithm is clearly too small to permit convergence under the smaller gain calibration.

The only method that seems to be performing consistently throughout the different algorithms and gain calibrations is our own *smoothing* procedure. The *smoothing* procedure has, in special, presented a better performance for the cases where the other methods have failed, namely: (i) for higher gain calibrations in the LS, where resulting estimates were less accurate; and (ii) for lower gain calibrations in the SG, where the rate of convergence tended to be slower. Besides, the *smoothing* method achieves a faster convergence within the training sample than the other methods, increasing its relevance for applications with tight data availability.

We complement the visual analysis with a look over the associated statistics in Tables 2 and 3, where averaged MSDs and MISALIGNMENTS are segmented in several subsamples of the algorithms' transient phases after the initializations. In short, we corroborate our conclusions from the visual inspection, observing that: (i) the *ordinary* method is overall outperformed by the others, with initial MISALIGNMENTS persisting to affect the first short run measures accross every combination of algorithm and gain calibration; (ii) the SG rate of convergence is slower than the one attained by the LS, and the *smoothing* method is the only one providing initializations closer to the algorithm/calibrations steady states.

[Tables 2 and 3 go here, please]

4.3 Smoothing computational cost

Apart from the extra computation of smoothed estimates, the additional computational cost of the smoothing-based initialization clearly depends on the number of repetitions necessary to satisfy the ϵ-convergence criterion. The smaller ϵ the tighter the convergence requirement, and the greater the number of repetitions, thence increasing the computational burden. Nevertheless, a loose convergence criterion may hinder the gains in accuracy obtained from extra smoothing passes.

In our simulations above we find that most of the improvements in accuracy are exhausted after about half a dozen repetitions of the smoothing routine for the LS algorithm and about three times that number for the SG. The time-varying properties of the system under estimation also matter in that respect: under the higher constant gain calibration, which imply noisier estimates of the model parameters than those obtained under the low constant gain, a higher number of repetitions of the smoothing routine is required to achieve convergence. Of course, the computational time associated to these numbers are practically negligible beside the computational power of technologies currently available to researchers.

5 Concluding Remarks

In this paper we proposed the use of a smoothing routine to obtain the initial estimates of the learning algorithms adopted for applied purposes in the literature of adaptive learning and expectations in macroeconomics. This routine is designed to speed up the convergence of the learning algorithms over a sample of initial training data, so as to minimize the amount of data required to obtain proper learning initials. Particularly, we assumed the target is to mimic the beliefs associated to a learning process that was already in motion before the sample beginning. In order to evaluate its success, we undertook a simulation exercise comparing our new smoothing-based initialization to two of the main methods found in the previous applied literature.

Our smoothing-based routine was the only method performing consistently under the different algorithms and calibrations we have considered. We interpret this finding as a natural result

from the unified state space framework we adopted for the derivation of our smoothing initialization method. The robustness attained by the smoothing-based method comes at the cost of a higher computational burden, though arguably small for the computational power currently available to researchers. We have quantified the effects of initials misspecification in a rather stylized learning-to-forecast framework; hence, their actual relevance should be assessed individually for particular applications. We hope to have provided some methodological guidance on that matter.

Notes

[1]The most notorious exception is provided by Carceles-Poveda and Giannitsarou (2007), where three alternative initialization methods were proposed and shown to affect the behavior of macroeconomic variables in simulation analysis. An empirical analysis of initializations is also provided by Slobodyan and Wouters (2012), though focusing on their joint estimation with other model parameters.

[2]We also discuss issues with the joint estimation of initials in a companion paper: see Berardi and Galimberti (2016) and references therein.

[3]We have also carried out a sensitivity analysis with artificial series mimicking output growth. Overall, our main conclusions were not affected by these differences, and these results are available in Berardi and Galimberti (2012).

[4]Gaussianity is only required to guarantee the optimality of the Kalman filter estimator associated to this non-stationary context. This latter is the basis under which a unifying smoother is derived later for the initialization of different learning algorithms.

[5]To keep up with the generality of our analysis here we focus solely on the initialization of the coefficients estimates, $\hat{\boldsymbol{\theta}}_0$, common to both algorithms. Moustakides (1997) provides a study on how to optimally initialize $\mathbf{R}_0$ in the LS algorithm, proposing a simple rule based on the data signal-to-noise ratio.

[6]There are two alternative forms of smoothing: (i) as fixed-lag, set $k = t + l$, with l fixed, and obtain $\hat{\boldsymbol{\theta}}_{t|t+l}$ as t increases; and, (ii) as fixed-interval, fix the information set k, and obtain $\hat{\boldsymbol{\theta}}_{t|\bar{k}}$ for $t \leq \bar{k}$ (see Anderson and Moore, 1979).

[7]This data generating process (DGP) is taken only as an approximation to time series data typically found in macroeconomic applications. It does not correspond to the specifications of the Kalman filter that would render the learning algorithms optimal, which is consistent with the bounded rationality view of adaptive learning in macroeconomics.

References

Anderson, B.D.O., Moore, J.B., 1979. Optimal Filtering. Prentice-Hall, Englewood Cliffs, NJ.

Barucci, E., Landi, L., 1997. Least mean squares learning in self-referential linear stochastic models. Economics Letters 57, 313–317.

Benveniste, A., Metivier, M., Priouret, P., 1990. Adaptive Algorithms and Stochastic Approximations. Springer-Verlag.

Berardi, M., Galimberti, J.K., 2012. On the initialization of adaptive learning algorithms: A review of methods and a new smoothing-based routine. Centre for Growth and Business Cycle Research Discussion Paper Series 175. Economics, The Univeristy of Manchester.

Berardi, M., Galimberti, J.K., 2013. A note on exact correspondences between adaptive learning algorithms and the kalman filter. Economics Letters 118, 139–142.

Berardi, M., Galimberti, J.K., 2016. On the Initialization of Adaptive Learning in Macroeconomic Models. Technical Report. KOF Working Papers. Zürich.

Bray, M., 1982. Learning, estimation, and the stability of rational expectations. Journal of Economic Theory 26, 318–339.

Carceles-Poveda, E., Giannitsarou, C., 2007. Adaptive learning in practice. Journal of Economic Dynamics and Control 31, 2659–2697.

Chevillon, G., Massmann, M., Mavroeidis, S., 2010. Inference in models with adaptive learning. Journal of Monetary Economics 57, 341–351.

Christev, A., Slobodyan, S., 2014. Learnability of e-stable equilibria. Macroeconomic Dynamics 18, 959–984.

Eusepi, S., Preston, B., 2011. Expectations, learning, and business cycle fluctuations. American Economic Review 101, 2844–72.

Evans, G.W., Honkapohja, S., 1998. Stochastic gradient learning in the cobweb model. Economics Letters 61, 333–337.

Evans, G.W., Honkapohja, S., 2001. Learning and expectations in macroeconomics. Frontiers of Economic Research, Princeton University Press, Princeton, NJ.

Evans, G.W., Honkapohja, S., Williams, N., 2010. Generalized stochastic gradient learning. International Economic Review 51, 237–262.

Hamilton, J.D., 1994. Time Series Analysis. Princeton University Press.

Haykin, S.S., 2001. Adaptive Filter Theory. Prentice Hall Information and System Sciences Series, Prentice Hall, New Jersey, USA. 4th edition.

Ljung, L., Soderstrom, T., 1983. Theory and Practice of Recursive Identification. The MIT Press.

Marcet, A., Sargent, T.J., 1989. Convergence of least squares learning mechanisms in self-referential linear stochastic models. Journal of Economic Theory 48, 337–368.

McGough, B., 2003. Statistical learning with time-varying parameters. Macroeconomic Dynamics 7, 119–139.

Milani, F., 2011. Expectation shocks and learning as drivers of the business cycle. The Economic Journal 121, 379–401.

Moustakides, G., 1997. Study of the transient phase of the forgetting factor rls. Signal Processing, IEEE Transactions on 45, 2468 –2476.

Orphanides, A., Williams, J.C., 2005. The decline of activist stabilization policy: Natural rate misperceptions, learning, and expectations. Journal of Economic Dynamics and Control 29, 1927–1950.

Sargent, T.J., 1999. The Conquest of American Inflation. Princeton University Press, Princeton, NJ.

Slobodyan, S., Wouters, R., 2012. Learning in an estimated medium-scale dsge model. Journal of Economic Dynamics and Control 36, 26 – 46.

Williams, N., 2003. Adaptive learning and business cycles. Mimeo.

A Tables

Table 1: Calibration of parameters for simulation with inflation-like artificial data.

Parameters	Description	Calibrations values
(a) For artificial series:		
σ_{ε}^2	Variance of ε_t in (21).	2.25
σ_{ω}^2	Variance of ω_{t+1} in (22).	7×10^{-5}
$\bar{\theta}$	Steady-state value of θ_t.	0.80
β	Persistence of deviations from $\bar{\theta}$.	0.999
(b) For algorithms:		
$\bar{\gamma}_1$	LS "low" constant learning gains.	0.02
$\bar{\gamma}_2$	LS "high" constant learning gains.	0.10
$\bar{\mu}_1$	SG "low" constant learning gain.	0.001
$\bar{\mu}_2$	SG "high" constant learning gain.	0.0205

The learning gain calibrations are first set for the LS, and then adjusted for the SG according to $\bar{\mu}_i = 2\bar{\gamma}_i / \left(\sigma_{\varepsilon}^2/1-\bar{\theta}^2\right)^2$ in order to account for the scale dependency of this latter to the data variance, but for illustrative purposes $\bar{\mu}_2$ is set based on $\bar{\gamma}_{2'} = 0.40$ instead of $\bar{\gamma}_2$.

Table 2: Average statistics after initializations - Least Squares on inflation-like data.

Gains	Initials	Samples after initializations					Steady state
		76-100	101-150	151-200	201-250	251-300	750-1000
$\bar{\gamma}_1 = 0.02$	*Tracking*	**0.0054**	0.0047	0.0045	0.0044	0.0047	0.0046
		[4.3]	[0.4]	[-0.8]	[-1.0]	[0.5]	(0.0002)
	Ordinary	**0.0060**	0.0047	**0.0042**	0.0043	0.0046	0.0046
		[7.9]	[0.6]	**[-2.4]**	[-1.8]	[0.1]	(0.0002)
	Smoothing	**0.0051**	0.0043	**0.0042**	0.0043	0.0046	0.0046
		[2.6]	[-1.7]	**[-2.6]**	[-1.7]	[0.2]	(0.0002)
$\bar{\gamma}_2 = 0.10$	*Tracking*	0.0175	0.0175	0.0183	0.0184	0.0188	0.0190
		[-1.3]	[-1.4]	[-0.6]	[-0.6]	[-0.2]	(0.0011)
	Ordinary	**0.0064**	**0.0139**	0.0182	0.0183	0.0188	0.0190
		[-11.5]	**[-4.7]**	[-0.7]	[-0.6]	[-0.2]	(0.0011)
	Smoothing	0.0174	0.0174	0.0183	0.0183	0.0187	0.0189
		[-1.3]	[-1.4]	[-0.6]	[-0.6]	[-0.2]	(0.0011)

The average statistics refer to the mean-square deviation (MSD) of coefficient estimates from their true counterparts, as defined in (5), and the MISALIGNMENT (in square brackets, [...]) of these average MSDs in relation to their steady state average, calculated according to (6). The second line of headers indicate the samples of observations used to compute the average statistics. The steady state averages are calculated over the last subsample, 750-1000, and the values in round brackets, (...), are standard deviations of the statistic from the corresponding steady state average. Emphasis is given in **bold** to those short run averages that deviate by more than two standard deviations from the corresponding steady state average.

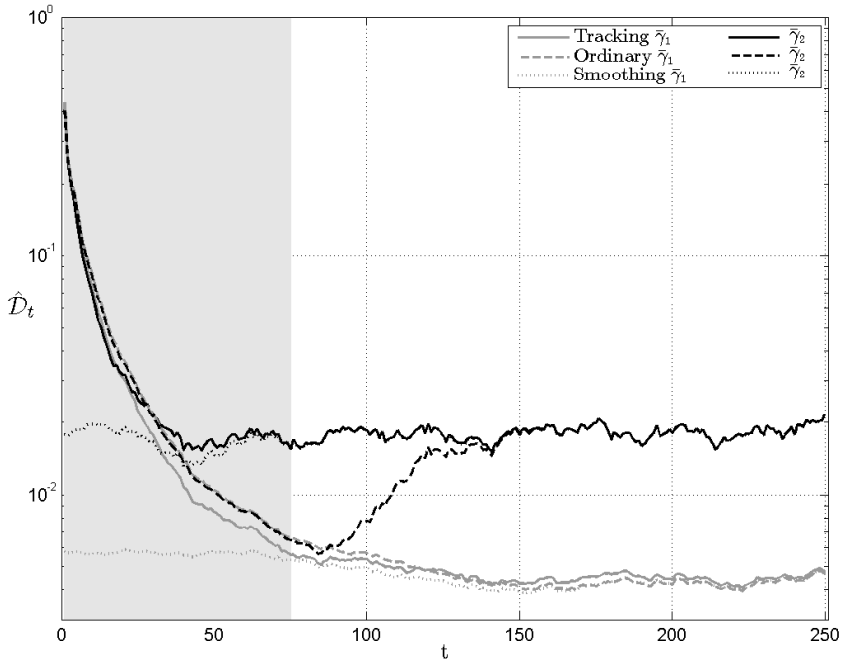
Table 3: Average statistics after initializations - Stochastic Gradient on inflation-like data.

Gains	Initials	Samples after initializations					Steady state
		76-100	101-150	151-200	201-250	251-300	750-1000
$\bar{\mu}_1 = 0.001$	*Tracking*	**0.1407**	**0.0944**	**0.0552**	**0.0344**	**0.0222**	0.0048
		[680.5]	**[448.5]**	**[252.5]**	**[148.1]**	**[87.2]**	(0.0002)
	Ordinary	**0.0790**	**0.0544**	**0.0327**	**0.0211**	**0.0143**	0.0047
		[358.4]	**[240.0]**	**[135.1]**	**[79.1]**	**[46.5]**	(0.0002)
	Smoothing	**0.0061**	**0.0057**	**0.0051**	**0.0050**	**0.0049**	0.0045
		[7.6]	**[5.7]**	**[2.8]**	**[2.3]**	**[2.1]**	(0.0002)
$\bar{\mu}_2 = 0.0205$	*Tracking*	0.0224	0.0223	0.0279	0.0210	0.0238	0.0226
		[-0.1]	[-0.1]	[1.5]	[-0.5]	[0.3]	(0.0034)
	Ordinary	**0.0481**	0.0240	**0.0303**	0.0213	0.0242	0.0228
		[7.4]	[0.3]	**[2.2]**	[-0.5]	[0.4]	(0.0034)
	Smoothing	0.0180	0.0193	0.0247	0.0179	0.0208	0.0197
		[-0.5]	[-0.1]	[1.5]	[-0.5]	[0.3]	(0.0034)

See notes to Table 2.

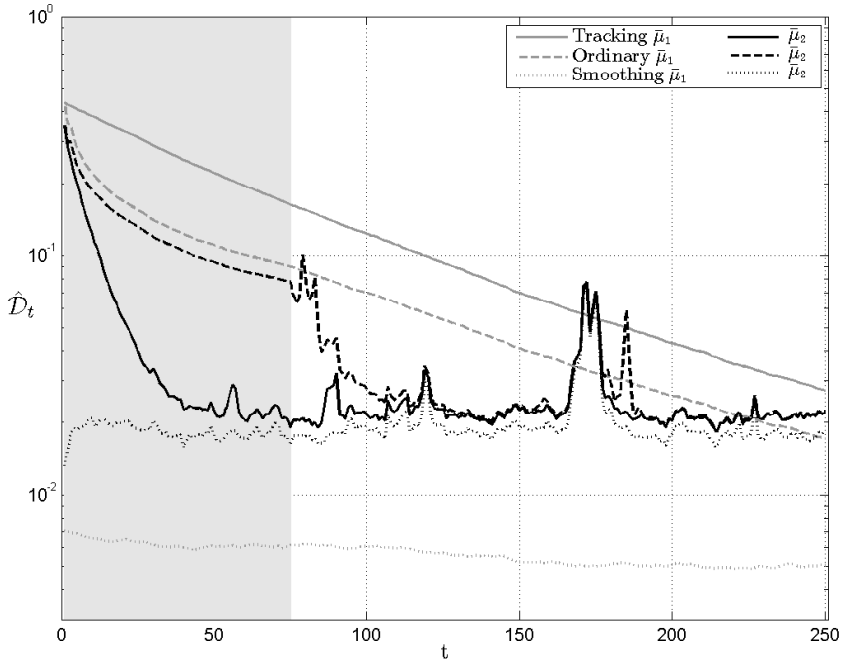
B Figures

Figure 1: Least Squares MSD learning curves for inflation-like artificial data.



$\hat{\mathcal{D}}_t$ stands for the sample correspondent to the mean-square deviation (MSD) as defined in (5). The shaded areas indicate the portion of observation left aside for use by the initialization methods, and we restrict the presentation to only the first quarter of our sample in order to obtain a clear picture on the after-initials periods (the MSDs remain relatively constant around their corresponding steady states in the remaining periods). The vertical axis is on logarithmic scale.

Figure 2: Stochastic Gradient MSD learning curves for inflation-like artificial data.



See the notes to Figure 2.