Discussion Paper Series

On the initialization of adaptive learning algorithms: A review of methods and a new smoothing-based routine

By

Michele Berardi and Jaqueson K. Galimberti

Centre for Growth and Business Cycle Research, Economic Studies,
University of Manchester, Manchester, M13 9PL, UK

October 2012
Number 175

On the initialization of adaptive learning algorithms: A review of methods and a new smoothing-based routine

Michele Berardi

The University of Manchester

Jaqueson K. Galimberti*

The University of Manchester and The Capes Foundation

October 2, 2012

Abstract

We provide a critical review on the methods previously adopted into the literature of learning and expectations in macroeconomics in order to initialize its underlying learning algorithms either for simulation or empirical purposes. We find that none of these methods is able to pass the sieve of both criteria of coherence to the algorithm long run behavior and of feasibility within the data availability restrictions for macroeconomics. We then propose a smoothing-based initialization routine, and show through simulations that our method meets both those criteria in exchange for a higher computational cost. A simple empirical application is also presented to demonstrate the relevance of initialization for beginning-of-sample inferences.

Keywords: adaptive learning, algorithms, initialization, smoothing.

JEL codes: C63, D84, E37.

1 Introduction

Adaptive learning algorithms have been proposed to provide a procedural rationality view on agents process of expectations formation. Reopening a long standing debate on how should expectations be modeled in macroeconomic models, the heuristics provided by learning algorithms come at the cost of introducing new degrees of freedom into the analysis. One open node relates to how these recursive mechanisms should be initialized in order to be representative of agents' learning-to-forecast behavior.

In this paper we investigate this issue with particular attention at the applied literature in macroeconomics. Here applied is taken to encompass both theoretical simulations as well as exercises of empirical

*Corresponding author. E-mail: jaqueson.galimberti@postgrad.manchester.ac.uk.

estimation and calibration. Examples can be found in Sargent (1999); Marcet and Nicolini (2003), or more recently in Eusepi and Preston (2011); Milani (2011), between many others cited throughout the paper. The main distinctive feature of these works consists in the replacement of unrealistic assumptions implying an instantaneous adjustment of agents expectations, inherent in the rational expectations hypothesis, with a characterization of agents as adaptive learners of their own environment. More generally, our study will be relevant for scholars interested in the actual implementation of the learning algorithms here considered.

We start by reviewing the literature in order to pool together the initialization methods previously adopted into an archetypal classification. In spite of the obvious relevance of such issue, surprisingly, we did not find many other attempts to systematically assess these methods. In economics, one exception is provided by Carceles-Poveda and Giannitsarou (2007), although their contribution is to a great extent restricted to theoretical applications. With the freedom to develop our own assessment framework, we compare initialization methods on the basis of two main criteria. First, we argue that a good initialization should provide estimates coherent to the long run behavior of the algorithm; second, achieving this coherence must be feasible within the usual data availability restrictions in macroeconomics. Our review indicates that none of the initialization methods found in the literature is able to satisfy both these criteria.

Motivated by this critical finding, we propose a new method of initialization based on smoothing within a sample of training data. Our point of departure is a unified framework under which the main learning algorithms considered in the literature, namely the Least Squares (LS) and the Stochastic Gradient (SG) ones, are obtained as special cases of the Kalman filter associated to a time-varying parameters model of the economy (Ljung and Soderstrom, 1983; Sargent, 1999; Evans et al., 2010). More specifically, Berardi and Galimberti (2012) 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 initiliazations. 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.

We then evaluate our procedure in comparison with two of the reviewed alternative methods with respect to their convergence performance in a simulation exercise. We show that our approach is able to deal with the conflict between coherence and feasibility, present in the other methods, at the same time that it has the advantage of being implementable in any algorithm that can be encompassed into the Kalman unifying framework. This solution, however, comes at the cost of an increased computational burden. To further enhance our understanding on the relevance of these different initialization methods for applied macroeconomics, we also present an empirical exercise of learning-to-forecast. Using US inflation and growth data, results are again found to favour our new smoothing routine. An ulti-

mate judgement on the relevance of these results, however, would require going beyond our simplified application, and we leave this issue open for future research.

The remainder of this paper proceeds as follows. In section §2 we establish the estimation framework and the specific recursions assumed by the algorithms whose initialization we are interested in analyzing. There, we also present a discussion on what is required from an initial estimate for these algorithms, so as to provide the criteria through which we can critically evaluate the methods we review from the literature. This review is presented in section §3, where we also describe our own proposal of a new smoothing-based routine. We then proceed to present a simulation exercise, in section §4, and an empirical application, in section §5, both aiming at a comparative analysis between different methods of initialization. Finally, we conclude this paper with some remarks in section §6.

2 Adaptive Learning Algorithms

2.1 The algorithms

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[1]

$$y_t = \mathbf{x}_t' \boldsymbol{\theta}_t + \varepsilon_t, \tag{2.1}$$

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[2]) 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 (2.1).

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.

[1] Our simulation and empirical applications presented later in this paper will focus on (vector) autoregression specifications that can be straightforwardly translated into the form of (2.1).

[2] Distributional assumptions such as Gaussianity are not strictly necessary for our purposes, but are 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.

Algorithm 1 (LS)**.** *Under the estimation context of (2.1), the* LS *algorithm assumes the form of*

$$\hat{\boldsymbol{\theta}}_t^{LS} = \hat{\boldsymbol{\theta}}_{t-1}^{LS} + \gamma_t \mathbf{R}_t^{-1} \mathbf{x}_t \left(y_t - \mathbf{x}_t' \hat{\boldsymbol{\theta}}_{t-1}^{LS} \right), \tag{2.2}$$

$$\mathbf{R}_t = \mathbf{R}_{t-1} + \gamma_t \left(\mathbf{x}_t \mathbf{x}_t' - \mathbf{R}_{t-1} \right), \tag{2.3}$$

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 (2.1), the* SG *algorithm is given by*

$$\hat{\boldsymbol{\theta}}_t^{SG} = \hat{\boldsymbol{\theta}}_{t-1}^{SG} + \mu_t \mathbf{x}_t \left(y_t - \mathbf{x}_t' \hat{\boldsymbol{\theta}}_{t-1}^{SG} \right), \tag{2.4}$$

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 (2.4) 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).

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 Learning and statistical rationales

Once a learning mechanism is specified, one of the main issues in the theoretical literature on adaptive learning has been to single out the conditions under which this learning process converges towards a target equilibrium (see Marcet and Sargent, 1988, for an earlier overview). This equilibrium is often well defined by the Rational Expectations (RE) hypothesis and it is represented by a fixed-point in the dynamics of a self-referential structural model, where agents expectations play a role in determining the economy's outcomes. The key feature of such analyses lies on the use of a stochastic approximation

approach in order to assess the asymptotic dynamics of the stochastic system through a deterministic differential equation (see Evans and Honkapohja, 2009, for a recent account).

Different questions have emerged from the applied side of this literature, where people have tried to understand how much of macroeconomic persistence can actually be attributed to learning (Orphanides and Williams, 2005b; Milani, 2007),and what part of business cycle fluctuations can be explained by a model with learning and expectational shocks (Eusepi and Preston, 2011; Milani, 2011). In contrast to the theoretical literature, where the interest is on stability and the eventual convergence of the learning algorithm to a given (equilibrium) target, here the key feature is represented by the actual behavior of the learning algorithms in finite samples. Such shift in interest motivates a statistical analysis of these estimators.

Recursive estimation algorithms are statistically characterized by undergoing through two main distinct phases: a transient and a steady state one. In defining the criteria we use to assess initializations, further below, we suggest that the purpose of an initialization method, from an empirical perspective, is to provide estimates as close as possible to the algorithm's steady state operation, since such beliefs should reflect the continuation of an estimation process that was already running prior to the sample beginning.

The separating frontier between the transient and the steady state 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 (2.1),* $\boldsymbol{\theta}_t$*, and the algorithms estimates,* $\hat{\boldsymbol{\theta}}_t$*, is given by*

$$\mathcal{D}_t = E\left[\Delta_t^2\right], \tag{2.5}$$

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).

We shall observe such a trade-off in the MSD learning curves that we compute for various gain calibrations, but notice that these are steady state features, while our main interest lies on the transient behavior that follows the algorithms' initializations[3]. In our context, hence, the MSD measure serves to

[3]Also notice that the MSD is a second moment measure, such that its convergence to different levels for different gains is not in conflict with the algorithm's convergence in distribution to estimates around the true coefficients' values.

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*

$$\mathcal{M}_t = \frac{\left|\mathcal{D}_t - \overline{\mathcal{D}_t}\right|}{\widehat{\mathcal{D}_t}}, \tag{2.6}$$

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

Clearly, 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 simulational purposes, (2.5) and (2.6) can be readily evaluated by computing their sample counterparts.

2.3 Requirements on initializations

From the recursive form of both learning algorithms, the initializations clearly take the form of estimates for $\hat{\boldsymbol{\theta}}_0$ and $\mathbf{R}_0$, although the latter is dispensed with in the SG case. 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[4].

Within the context of learning and expectations, the estimates provided by the learning algorithms are taken to represent agent's beliefs about the economy. With this in mind, inquiring about the values to assign to $\hat{\boldsymbol{\theta}}_0$ should lead to the question: (i) what were agent's beliefs at the beginning of our sample of data? It is to answer to this question that an initialization method is purposefully designed. But from a statistical point of view, whatever the initial estimate we assume, it would be just a matter of time until the algorithm achieves convergence. Then the relevant question becomes: (ii) how long will it take for the algorithm to converge to an appropriate estimate of agent's beliefs?

In the assessment that follows, we associate two evaluation criteria to the above questions in order to qualify the initialization methods previously adopted in the literature. The first is the COHERENCE of initial estimates relative to the steady state behavior associated with the gain used to calibrate the algorithm. As we have argued, however, at an applied level it becomes difficult to distinguish between transient and steady state dynamics. Our assessment on this criterion, therefore, is restricted to a relative scaling of how the different methods perform in terms of their MISALIGNMENT.

Criterion 1 (COHERENCE)**.** *An initialization method is said to provide more* coherent *initial estimates than another method if the* MISALIGNMENT *of the former initial estimates, as measured by (2.6), is smaller than the* MISALIGNMENT *of the latter.*

[4]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. When it comes to our applied exercises we shall be precise about this rule and how we use it.

The second criterion is the FEASIBILITY of the initialization method in the context of macroeconomic data. Even though in this case we could provide a definition in absolute terms, such as establishing a limit on the amount of data required by a feasible method, we opt for another relative form that is suitable for our later comparative exercises.

Criterion 2 (FEASIBILITY)**.** *An initialization method is said to be more* feasible *than another method if the amount of data required by the former to obtain the aimed initial estimates is smaller than the amount of data required by the latter to achieve the same purpose.*

3 Initialization Methods

A summary of how different initialization methods compare with each other is proposed in table 1 and a discussion is provided in the following section. In order to address the shortcomings we find in the methods traditionally used in the literature, we then propose a new initialization routine based on smoothing within a training sample of data.

3.1 A review of previous methods

From an applied perspective, initialization methods can be classified between two extreme ends depending on their suitability for simulation or empirical purposes. Their distinction in that respect reflects the amount of information available about the system to which the algorithms are applied. While in simulation studies the true system is known by the researcher, in empirical applications most knowledge incorporated into an initialization represents the assumptions that qualify the study in its own.

Starting from the theory-guided methods, one first way to initialize learning algorithms for simulation studies is that obtained from the use of full knowledge about the law of motions generating the data. This method is referred as *Exact* in table 1, and it first[5] appeared into the seminal applied contribution of Sargent (1999). Its usage has since been prominent in studies that replace the assumption of frictionless REE with the sticky process of expectations formation through adaptive learning. For lack of a better guess, this method proposes to take the coefficient values corresponding to the REE as initial estimates for the algorithm's recursion.

Clearly, the main benefits of the *Exact* method of initialization relates to its theoretical support, as well as its exemption from pre-forecasting data requirements. These advantages, however, come at the cost of its unsuitability for empirical applications, where the information about the true system under analysis is often the object of study.

One closer alternative is provided by the method we denote as the *Ad-hoc* initialization in table 1, where the initials are hand-picked by the researcher. When taking the *Exact* initials as reference, this

[5]Earlier simulation works, such as Bray and Savin (1986), also followed a similar approach, but in the context of a decreasing-gain LS algorithm.

method provides a way to validate the sensitivity of results obtained under the former. It lacks, however, the objectivity of the previous method given that there is usually no guidance on the magnitude of variations on the initials, and the researcher's degrees of freedom increase rapidly with the system's dimension. Both the *Exact* and the *Ad-hoc* methods may, furthermore, lead to initials still incoherent to agents' beliefs in terms of the underlying algorithm and its calibration, given that both provide initials incorporating information from the system, but not from the learning algorithm itself.

Shifting now our focus to approaches less theoretically-grounded, but which in contrast are favored for their empirical suitability, there are two main methods adopted in the literature to initialize the estimates of recursive learning algorithms. In the engineering literature (see Ljung and Soderstrom, 1983, pp. 299-303, e.g.), it is often suggested that the coefficients should be initialized with the value of zero (known as a diffuse prior) and an initial sample should be left aside to let the algorithm adjust its estimates according to the underlying calibration. This is especially recommended for the cases where there is not enough previous knowledge about the system under estimation so as to allow an educated guess. It is referred in table 1 as the *Diffuse-track* method.

Clearly, under the *Diffuse-track* method of initialization the criterion of COHERENCE is satisfied. As long as the algorithm and its calibration are appropriate for the application, it can be expected that convergence will eventually take place, and, therefore, estimates representing the steady state behavior of the algorithm can be obtained as initials. However, two problems arise with this method. First, it is up to the researcher's wisdom to recognize how many observations are needed to get past of the initial transient phase, an aspect that increases the method's subjectivity. The second problem with the *Diffuse-track* method relates to its FEASIBILITY, which may become critical in the macroeconomic context where availability of data is usually restricted as compared to the engineering context under which this method was originally proposed. This last drawback is of special relevance for the case of learning gains calibrated to small values, as usual in economic applications, where the algorithms tend to show rather slow rates of convergence. We shall return to this point later in our simulation experiments, but notice that for simulation purposes, as in Huang et al. (2009); Eusepi and Preston (2011), this point is not restrictive as long as enough observations are generated for the algorithm to achieve convergence within the pre-forecasting initialization sample.

A second empirically-grounded method of initialization involves the use of the decreasing gain LS block estimation counterpart, namely the OLS estimator, within a pre-specified initial sample of data. Essentially, this method corresponds to an initialization of the coefficients from zero, and then updating the estimates within an initial sample using the LS estimator with decreasing gains given by[6] $\gamma_t = 1/t$. It therefore represents an hybrid of the *Diffuse-track* method, implemented with the LS algorithm under a decreasing gain. This method also seems to be the quite popular in empirical studies on learning in

[6]To prevent instabilities into the first estimates we set the decreasing gains as $\gamma_t = \bar{\gamma}/t$. See also the discussion on the design of our simulation experiments.

macroeconomics, perhaps due to the prominence of the LS algorithm and its popularity between applied researchers. In table 1 we denote it by the *Diffuse-ordinary* method.

As with the previous method, the *Diffuse-ordinary* initialization suffers from the same lack of objectivity regarding the determination of how much of the available sample of data should be set aside for the initialization routine. Nevertheless, the fact that a relatively higher gain value is used in the first iterations of the initialization tends to improve the convergence speed considerably, and so favoring the FEASIBILITY of this method[7]. The *Diffuse-ordinary* initialization can, therefore, be expected to require a lower number of initial observations to achieve convergence in relation to the *Diffuse-track* method.

The most important drawback of the *Diffuse-ordinary* initialization method, however, is its lack of COHERENCE with respect to the algorithm's gain calibration, especially in the primitive sense that this parameter stands for learning. 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, this initial estimate will tend to be biased in relation to the algorithm's steady state estimates. By using a decreasing gain the *Diffuse-ordinary* method provides the same initial estimate irrespective of the gain calibration for which this is required. Thus, even though it tends to attain a quicker convergence, the estimates to which this initialization method converges may be incoherent with the subsequent performance of the algorithm, a fact that ends up requiring further adaptations of the algorithm outside of the initialization sample in order to get to its steady state.

3.2 A new method based on smoothing

The main difficulty that the initialization methods reviewed above face for their use in empirical applications relates to the trade-off between their FEASIBILITY and the COHERENCE they can achieve. These criteria can be seen to represent antagonistic requirements due to the effects that the size of the training sample of data has over them. Namely, while devoting additional data to the initialization procedure tends to favour COHERENCE, by expanding the room for the algorithm's convergence to play, the method's FEASIBILITY becomes impaired.

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.

In order to understand the concept of smoothing, it is important to first define the concept of filtering, from which the former departs. In our context, filtered estimates are those obtained from the (forward) recursions associated to the learning algorithms in (2.2)-(2.3) and (2.4). For clarity, we add another subscript to our previous notation: $\hat{\boldsymbol{\theta}}_{t|k}$, where t indicates the period the estimate stand for and k indicates the information period on which the estimate is based. Then, the filtered estimates are

[7] Also notice that this method can be adjusted to use the SG algorithm with a decreasing gain.

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 (see Anderson and Moore, 1979).

Due to the use of more information, one can expect the smoothed estimates to be more accurate than the filtered ones, and this is the reason we propose their usage for the initialization of learning algorithms. This gain in accuracy, however, comes at the cost of more computations and a delay incurred by waiting for the arrival of new observations. Such a delay, obviously, prevents the use of the smoothed estimates for learning-to-forecast applications, given that these estimates make use of information not available to real-time learning agents. Therefore, a sample of initial data, say of N observations, is required to be set aside for the smoothing initialization procedure, just as it is required by the diffuse initialization methods.

Within this initial sample of data, then, one can start the computation of the learning algorithms from a diffuse prior, such as $\hat{\boldsymbol{\theta}}_0 = 0$, and obtain not only the algorithm's filtered estimates up to $\hat{\boldsymbol{\theta}}_N$, but also its smoothed[8] estimates of $\hat{\boldsymbol{\theta}}_{0|N}$. 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[9] criterion based on the Euclidean distance between filtered and smoothed estimates, under which the above process is repeated until $\left\| \hat{\boldsymbol{\theta}}_0 - \hat{\boldsymbol{\theta}}_{0|N} \right\| < \epsilon$.

It is important to note that these repetitions are not worthless with respect to their computational cost. Even though the same set of data is supplied to the algorithms throughout these repetitions of filtering/smoothing estimations, the information provided is not the same. Namely, by changing the initial estimates, $\hat{\boldsymbol{\theta}}_0$, the whole stream of subsequent filtered estimates is affected, and so is the information on the system dynamics that is incorporated into the smoothed estimates.

To obtain the smoothed estimates associated to the learning algorithms in (2.2)-(2.3) and (2.4), we follow the literature (Ljung and Soderstrom, 1983; Sargent, 1999; Evans et al., 2010; Berardi and Galimberti, 2012) drawing a parallel between these algorithms and the Kalman filter applied to the estimation of a time-varying parameters models. More specifically, we start from the exact correspondences drawn in Berardi and Galimberti (2012) for both the LS and the SG algorithms under an unifying state-space

[8] Smoothing is usually carried out into one of three forms: (i) as fixed-point, fix t, and update the estimates of $\hat{\boldsymbol{\theta}}_{\bar{t}|k}$ as k increases; (ii) as fixed-lag, set $k = t + l$, with l fixed, and obtain $\hat{\boldsymbol{\theta}}_{t|t+l}$ as t increases; and, (iii) as fixed-interval, fix the information set k, and obtain $\hat{\boldsymbol{\theta}}_{t|\bar{k}}$ for $t \leq \bar{k}$. For our purposes only (i) and (iii) are sensible, but given that our interest rests solely on an initial estimate we adopt the former, avoiding the need of a "backward pass" as in the latter.

[9] To avoid halting the computations for longer than necessary, we also imposed a stopping limit to the number of these repetitions.

framework[10]. Then, we obtain the smoother associated to each of these algorithms by direct substitution of their correspondences with the Kalman filter into 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 appropriately initialized to represent the fixed-point smoothed estimates[11].

4 Comparative Simulation Analysis

In this section we present simulation evidence comparing two of the initialization methods reviewed above, namely the *diffuse-track* and the *diffuse-ordinary*, to our own new method based on *smoothing*. The comparison is made with respect to their initial transient behavior in estimating the parameters of a given (known) non-stationary environment.

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 initializations adopted. Consistent to our discussion in section 2.2, our focus in on the MSD mesure of the algorithms' performance, as given by (2.5). 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 (1st order) autoregression of the form

$$y_t = \theta_t y_{t-1} + \varepsilon_t, \tag{4.1}$$

where the autoregressive parameter evolves according to

$$\left(\theta_{t+1} - \bar{\theta}\right) = \beta\left(\theta_t - \bar{\theta}\right) + \omega_{t+1}, \tag{4.2}$$

and the random disturbances ε_t and ω_{t+1} are zero mean mutually independent distributed as Gaussian[12] with variances given by σ_ε^2 and σ_ω^2, respectively. Note that, in spite of simplifying our analysis by focusing into a one-parameter specification, we are adopting a time-varying model similar to that of Doan et al. (1984) and Hamilton (1994, pp. 400-3). In particular, notice that if $|\beta| < 1$, then $\bar{\theta}$ may be viewed as the steady-state value of the autoregressive coefficient in (4.1). 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

[10] For convenience, we reproduce these correspondences in Appendix A.1.
[11] Key steps in this derivation are also reproduced for convenience in Appendix A.2.
[12] See footnote 2.

of resembling a random walk, this assumption prevents the dynamics of the autoregressive 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. We also carry out a sensitivity analysis over the values assumed for σ_{ε}^2 and $\bar{\theta}$. Our focus on these parameters, obviously, stems from their role in determining the most notable statistical features distinguishing macroeconomic series of data, such as their variance and dynamic persistence. To be consistent with our later empirical application, our interest in the generation of artificial series is to mimic these properties for data on inflation and output growth. Using standard econometric tools, then, we fitted a univariate fixed-coefficient AR(1) process to each of these series (the data shall be described in the next section), and obtained calibrations for them as $\sigma_{\varepsilon}^2 = 0.9\hat{\sigma}^2$ and $\bar{\theta} = \hat{\theta}$, where $\hat{\sigma}^2$ stands for the above estimation residuals variance for each variable, and $\hat{\theta}$ is the associated autoregressive coefficient estimate. All these calibrations are summarized in Table 2.

For these given calibrations, we drew 1,000 different samples of the random disturbances and used them with the DGP given by (4.1)-(4.2) 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 $\theta_1 = \bar{\theta}$. Examples of such artificially generated series of data, one for each calibration considered in table 2, are presented in figure 1. Remarkably, the inflation and growth-like characterization of these series is clear from the higher degree of persistence in the former, and the higher degree of volatility in the latter.

Apart from these calibrations for the artificial series, we also need to specify how we calibrated the algorithms' 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). Without extending any longer on this issue here, 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 (4.1) is simply given by the variance of y_t. The LS gain calibrations specified in table 2 by $\bar{\gamma}_i$, for $i = 1, \ldots, 4$, 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 autoregressive coefficient as fixed to its long run value, $\theta_t = \bar{\theta}$.

For the case of the LS algorithm, it also remains to specify how we proceed to initialize the matrix

of moments associated to (2.3). We follow Moustakides (1997) rule, under which

$$\mathbf{R}_0 = \gamma_0^{\alpha} \boldsymbol{\Theta}_x^2, \tag{4.3}$$

where $\boldsymbol{\Theta}_x^2$ is the variance (matrix) of the regressors in (2.1) and α is a parameter to be calibrated according to the signal-to-noise ratio of (2.1). For our purposes we simply set $\alpha = 1$, and compute $\boldsymbol{\Theta}_x^2$ on the basis of (2.1) as an autoregression with $\boldsymbol{\theta}_t$ fixed to its (expected) long run value.

The set of generated series, totalizing $2 \times 1,000$, were then supplied to different combinations of algorithm/gain/initialization to obtain the associated $\hat{\theta}_t$ estimates. These latter were then used for the computation of the MSD learning curves, following (2.5), averaged over the one thousand replications of data samples.

4.2 Simulation Results

The MSD learning curves obtained from the application of the LS and the SG algorithms to inflation and growth-like data are presented in figures 2 and 3, respectively for these series. For each of the methods considered, we have fixed the number of observations taken for training to the first 75, which are highlighted in the figures by shaded areas. Most importantly, the criterion under which each initialization is evaluated is the MISALIGNMENT of the initial estimates from their corresponding algorithm/gain long run behavior, as defined in (2.6). Here, firstly, we evaluate this criterion visually leaving its quantification to the end.

The initial MISALIGNMENT incurred by each initialization method depends on the gain calibration. This is evident in figures 2 and 3 by the jumps undertaken by the MSD estimates from their after-initialization level to their stable long run level. These jumps are more remarkable for the *diffuse-ordinary* initializations. This observation corroborates our previous point that the *diffuse-ordinary* method tends to violate the requirement on the initials' COHERENCE.

The *diffuse-track* also seems to perform poorly with respect to that COHERENCE criterion, but this is more clearly evident from its application to the SG algorithm. The lack of a normalization step in the operation of this algorithm seems to be reflected into its slowly rate of convergence to steady state. The number of observations left aside for the *diffuse-track* initialization of the SG algorithm is clearly too small to permit convergence of the smaller gain calibrations, corroborating our statement that this method lacks on FEASIBILITY.

The only method that seems to be performing consistently for all the above criteria and throughout the different algorithms and gain calibrations is our own *smoothing* procedure. In any of the cases considered in figures 2 and 3, this method tended to provide initial estimates that were closer to the algorithm/gain steady state behavior compared to those provided by the other two methods. 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.

Such failures of the methods previously used in the literature may, nevertheless, be questioned for their relevance in economic applications. This might especially be the case for the LS algorithm, which is predominant in this literature, given that this algorithm is usually calibrated to small values of learning gain parameters. To enhance our understanding under these circumstances we also present a comparison of MSD learning curves obtained from different initialization methods for a given gain calibration, labeled in table 2 as "usual" for its proximity to values used into the literature. This is presented in figure 4.

Results in figure 4 indicate that for a "usual", economically meaningful, calibration of the LS algorithm, the three methods of initialization here evaluated tend to lead to similar results in terms of initial MISALIGNMENTS to its steady state behavior. This is remarkably true for the *diffuse-track* and the *smoothing* methods, while for the *diffuse-ordinary* one there is some evidence indicating a higher degree of MISALIGNMENT for the case of the inflation-like series in panel (a) of figure 4. Notice, however, that the use of less than 75 observations for the initialization of this algorithm/calibration would unequivocally drive the initial estimates obtained from the *diffuse-track* and the *diffuse-ordinary* methods into incoherent estimates. Hence, the FEASIBILITY of these methods may still be impaired under cases of tight data availability.

Pictures similar to those we found previously emerge for the SG cases depicted in panels (b) and (d) of figure 4. In both cases, for inflation and growth-like artificial data, results point out that only the *smoothing* initialization method achieves COHERENCE of initial estimates, while the *diffuse-ordinary* method is again performing poorly. Notice, also, that it takes about 200 observations for the *diffuse-track* and the *diffuse-ordinary* methods to achieve a level of coherence close to that attained by the *smoothing* method within the training sample of only 75 observations.

To add precision to the observations made above, we complement the visual analysis with a look over the associated statistics. For this purpose we present in tables 3-6 average MSD statistics, with the first two focusing on inflation-like data, and the remaining on growth-like data. For each of these tables, the averaged statistics are segmented in several subsamples after the initializations, in an attempt to obtain short run measures corresponding to the transient phase undertaken by the algorithms after the initials.

Focusing first on the results for inflation-like data, we can see that our main observations from the visual inspection are here corroborated: (i) the *diffuse-ordinary* method is overall outperformed by the others, being the only method for which the magnitude of initial MISALIGNMENTS persist to affect the first short run measures, for each 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. These observations are also corroborated for the growth-like

data, although with some deterioration of the short run MSD measures.

5 Empirical Application

In this section we attempt to check the evidence about different initialization methods that was derived in the previous section from simulations, by means of an empirical application with US macroeconomic data on inflation and growth. It is not our purpose to be exhaustive in this application, but only to provide an assessment of how these initialization methods perform, comparatively, under a simplistic context of applied macroeconomics. Before going through the results, we begin with a brief description of the data and model specification to be used in conjunction to the LS and the SG algorithms to assess the effects that three different initial estimates have over the properties of the forecasts for inflation and growth. For consistency, we evaluate the same methods of initialization, namely, the *diffuse-track*, the *diffuse-ordinary*, and the *smoothing* methods.

5.1 Data, model, and algorithms' calibrations

We use quarterly data on the US real GDP and its price index from 1947q1 to 2011q4. Our data on this series comes from the Philadelphia's Fed Real-Time Data Research Center[13] from which we used the series observed at the vintage of 2012q1. For simplicity, we are neglecting real-time data issues by focusing on a unique snapshot of the realization of these series. As our interest is in modeling inflation and output growth, we construct these rates from the above data on levels computing their associated annual growth rates by compounding their simple quarterly growth factors. This gives us a total of 259 observations for each variable, which from now on we denote by π_t and $\dot{y}_t$, for inflation and growth, respectively.

We then use a simple unrestricted VAR(1) specification to model these series, where coefficients are updated recursively for each new observation made available through time. These estimates are obtained separately for each algorithm, i.e., the LS and the SG, and also for each different initialization method. The model specification we are using can then be expressed as

$$\begin{bmatrix} \pi_t \\ \dot{y}_t \end{bmatrix} = \begin{bmatrix} \hat{\theta}_{\pi,t} & \hat{\theta}_{\dot{y},t} \\ \hat{\phi}_{\pi,t} & \hat{\phi}_{\dot{y},t} \end{bmatrix} \begin{bmatrix} \pi_{t-1} \\ \dot{y}_{t-1} \end{bmatrix} + \begin{bmatrix} e_{\pi,t} \\ e_{\dot{y},t} \end{bmatrix}, \tag{5.1}$$

where $\hat{\theta}$ and $\hat{\phi}$ stand for the coefficients' estimates associated to the equations having inflation and growth as endogenous, respectively, and their subscripts indicate the explanatory variable to which they are attached and the period from which their estimates are made. The residual terms $e_{\pi,t}$ and $e_{\dot{y},t}$ stand for estimation errors.

[13] See http://www.philadelphiafed.org/research-and-data/real-time-center/.

The initializations are computed on the basis of a training sample of 75 initial observations, as we did in our simulation exercise. In terms of the time span of our sample, this means we left aside data from 1947q2 to 1965q4, which is (approximately) in line with the previous applied literature on adaptive learning in macroeconomics. Orphanides and Williams (2005a), for example, use data up to 1965q4 to initialize the learning algorithms, while Branch and Evans (2006) and Milani (2011) expand this information set with a few more observations, using data up to 1969q4 and 1968q4, respectively.

It remains to specify how we calibrate the learning gain of the LS and the SG estimation algorithms. In the spirit of the calibration we used for the simulation exercises, we adopt three alternative values of gain for each algorithm, where we first define the gains for the LS and adjust these for the data-dependent context of gain calibration for the SG algorithm.

To be consistent with the previous literature, however, the reference calibrations we adopt for the LS are less dispersed than those we used for the simulation exercise, and will be referred with alphabetic subscripts for clarity. Specifically, we adopt the following values for the LS gains: $\bar{\gamma}_a = 0.01$, $\bar{\gamma}_b = 0.025$, and $\bar{\gamma}_c = 0.05$. For the SG algorithm, we use expressions similar to those we adopted for the simulation exercises, though with the regressors covariance matrix being estimated using the initialization data[14]. Doing that we obtain the following calibrations for the SG gains: $\bar{\mu}_a = 0.0018$, $\bar{\mu}_b = 0.0046$, and $\bar{\mu}_c = 0.0092$.

5.2 Empirical Results

Our analysis is based on statistics of 1-period ahead forecasts that can be straightforwardly computed from the VAR specification in (5.1) for inflation and growth, together with coefficients estimates obtained from each combination of algorithm, gain calibration, and initialization. Our focus on forecasts is naturally due to the learning-to-forecast rationale we are adopting. Also, our lack of knowledge about the true system coefficients prevents us from evaluating MSD measures directly.

It is instructive to have a look over the coefficient estimates that each algorithm provides for the two-variables VAR(1) model of (5.1), and how these estimates are affected by the method used to initialize the algorithms' operations. These estimates are presented in figures 5 and 6, for the coefficients associated to the inflation and the growth equations, respectively.

One first observation is that the differences between the initial estimates presented by each method of initialization are more remarkable for the coefficient associated to the inflation variable. Corroborating our simulation findings, these differences are also more pronounced for the *diffuse-ordinary* method. As expected, these initial differences tend to die out as observations accumulate over time, with only small

[14] Other than for the calibration of the SG gains, we also used the covariance matrix estimated on the basis of initialization data for the initialization of $\mathbf{R}_0$ in the LS algorithm. See equation (4.3). Our use of only initialization data is an attempt to prevent any form of "cheating in forecasting" by the learning algorithms in what they stand as representative of agents' expectations.

differences remaining after say observation 150, which in our sample stands for 1984q2[15].

A deeper understanding of these differences in estimates can be obtained through statistics computed over the forecasts associated to them. Here our focus goes to their variances and mean squared errors (MSE), presented in tables 7 and 8. While the MSE stands for a useful measure of performance comparisons, it is mainly in the forecasts' variance that lies our interest. Learning-to-forecast behavior provides the channel through which shocks may persist over time, rather than perishing instantaneously as implied by RE. Recent studies have found support to this view by incorporating forecasts generated through learning algorithms as an additional explanatory variable in business cycle modeling (see, e.g., Orphanides and Williams, 2005a; Milani, 2011, between others cited above). Hence, it is important to assess whether a given initialization method is distorting the variance of the forecasts associated to each learning algorithm in relation to its long run behavior.

In accordance with the previous applied literature we can further narrow our focus to the LS case, presented in table 7. Focusing on the first subsamples we see that, for both inflation and growth, the *diffuse-ordinary* method rendered forecasts with lower variances than the other methods. Compared to the *smoothing* method, e.g., the forecasts obtained departing from the *diffuse-ordinary* initials presented variances around[16] 20% lower for both inflation and growth during the first subsample (1966q1-1971q4), and 5% and 16% lower for inflation and growth, respectively, during the second subsample (1972q1-1984q2). It should be emphasized that the result that the *diffuse-ordinary* method delivers forecasts with a lower variance than the others does not constitute a point in favour of this method, and neither, necessarily, against it. Bear in mind that we are interested in assessing the initials coherence to the algorithm's long run behavior, rather than in a minimization of forecasts variance or their associated errors[17].

Before drawing a conclusion on this point, one additional observation is useful. A smaller difference is observed between the variances of the forecasts obtained from the *diffuse-track* and the *smoothing* methods, although this depends on the variable forecasted. For inflation, the forecasts' variance from the former initialization procedure was found to be around 8% and 15% higher than that from the latter, during, respectively, the first and the second subsamples. For growth, in contrast, the forecasts associated to the *diffuse-track* method mostly presented a lower variance than those for the *smoothing* method, though this difference here amounted to only around 3% .

In contrast to our previous simulation exercise, though, our knowledge about the true nature of the series under estimation is restricted at the empirical level. This prevents us from obtaining a clear-cut answer to which of the initialization methods provides a higher degree of coherence between the initial

[15] Another observation is that the SG estimates appear to be more volatile than those of the LS, which could well be explained either as a difference between the algorithms behavior, as well as a gain calibration feature. As their comparison is not our main purpose here, we leave this open for future research.

[16] These are averages over the three gain calibrations presented.

[17] In that respect, notice that the *diffuse-ordinary* presented slightly higher MSEs than the other methods during the first two subsamples for inflation, and the second subsample for growth.

estimate and the algorithm's long run behavior. Nonetheless, our simulation results suggested that the *diffuse-ordinary* method tends to be more distortionary than the other methods. Together with the empirical results found here, then, our study indicates that in order to initialize learning algorithms: (i) there is not much difference between using the *diffuse-tracking* or the *smoothing* methods to initialize learning algorithms, although the latter has the advantage of being robust to changes in algorithms/gain calibrations; (ii) using the *diffuse-ordinary* method results in initial forecasts with a lower variance than those obtained from the other methods, and our simulations indicate that this difference represents a distortion in relation to the algorithm's long run operation.

6 Concluding Remarks

In this paper we provided a critical review on the several methods previously proposed in the literature of learning and expectations in macroeconomics in order to initialize its learning algorithms either for simulation or for empirical purposes. Most importantly, we have also provided one of the first attempts in the literature to evaluate how these methods compare to each other, and how their performance may be evaluated with respect to their learning and expectations rationale. To delineate the scope of our analysis, we focused on two of the main algorithms found in this literature, namely, the Least Squares (LS) and the Stochastic Gradient (SG) algorithms.

Before pooling the initialization methods in a classification exercise, we provided a discussion on what it is required from them, arguing for the use of two main criteria. First, an initialization should provide initial estimates coherent to the algorithm's long run behavior. Second, such a coherence must be feasible within the data availability restrictions of usual macroeconomic applications. Our finding is that none of the previous methods reviewed is able to pass the sieve of both criteria, and this motivated us to propose a new method.

Departing from exact correspondences between the learning algorithms and the Kalman filter associated to a time-varying hypermodel, as recently drawn by Berardi and Galimberti (2012), we proposed the use of a smoothing routine to obtain the initial estimates for each algorithm. This routine makes use of a sample of initial training data, and it is designed to satisfy the above requirements of coherence and feasibility in exchange for additional computational costs. In order to evaluate its success, we undertake both a simulation exercise and an empirical application, comparing our new smoothing-based initialization to two of the methods found in the previous applied literature.

From the simulation exercise, the main conclusion we can draw is that our method is successful in satisfying what we required from an initialization while the previous ones achieved only partial success in that respect. Namely, our smoothing-based routine was the only method performing consistently throughout the various applications we have explored, under different algorithms, calibrations and sta-

tistical environments. We interpret this finding as a natural result from the unified design we adopted for the derivation of our smoothing initialization method.

A different question, however, is how much the differences we found across methods are relevant for actual applications of these algorithms into macroeconomic contexts. To shed some light on this issue, we compared the initialization methods in a simplified empirical application of learning-to-forecast US data on inflation and output growth. Using a sample of quarterly data from 1947 to 2011, where data up to 1965 is left aside for the initialization of the algorithms, our results indicate that the effects of the different initialization methods last no longer than mid-1980s. For the preceding sample periods, nonetheless, our results indicate that the initialization method can distort the variances of the forecasts constructed using the learning algorithms. Even though we have quantified these effects for our simplified application, their actual relevance will depend on the issue under scrutiny.

A Detailed derivations

A.1 Correspondences between learning algorithms and Kalman filter

To establish the state-space framework assume, in addition to the linear model in (2.1), that the coefficients vector evolves according to

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

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

$$\begin{aligned}
\hat{\boldsymbol{\theta}}_t &= \hat{\boldsymbol{\theta}}_{t-1} + \mathbf{K}_t \left(y_t - \mathbf{x}_t' \hat{\boldsymbol{\theta}}_{t-1} \right), && \text{(A.2)} \\
\mathbf{K}_t &= \frac{\mathbf{P}_{t-1}\mathbf{x}_t}{\mathbf{x}_t' \mathbf{P}_{t-1} \mathbf{x}_t + \sigma_t^2}, && \text{(A.3)} \\
\mathbf{P}_t &= \left(\mathbf{I} - \frac{\mathbf{P}_{t-1}\mathbf{x}_t\mathbf{x}_t'}{\mathbf{x}_t' \mathbf{P}_{t-1} \mathbf{x}_t + \sigma_t^2} \right) \mathbf{P}_{t-1} + \boldsymbol{\Omega}_t, && \text{(A.4)}
\end{aligned}$$

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 (2012), the LS and the SG learning algorithms, as given by (2.2)-(2.3) and (2.4), respectively, can be obtained as special cases of the Kalman filter when

$$\begin{aligned}
\sigma_t^2 &= \frac{\gamma_{t-1}}{\gamma_t} \left(1 - \gamma_t \right), && \text{(A.5)} \\
\boldsymbol{\Omega}_t &= \left(\frac{1 - \sigma_t^2}{\sigma_t^2} \right) \left(\mathbf{I} - \frac{\mathbf{P}_{t-1}\mathbf{x}_t\mathbf{x}_t'}{\mathbf{x}_t' \mathbf{P}_{t-1} \mathbf{x}_t + \sigma_t^2} \right) \mathbf{P}_{t-1}, && \text{(A.6)}
\end{aligned}$$

and

$$\begin{aligned}
\sigma_t^2 &= \mu_t^{-1} - \mathbf{x}_t' \mathbf{x}_t, && \text{(A.7)} \\
\boldsymbol{\Omega}_t &= \mathbf{I} - \left(\mathbf{I} - \frac{\mathbf{P}_{t-1}\mathbf{x}_t\mathbf{x}_t'}{\mathbf{x}_t' \mathbf{P}_{t-1} \mathbf{x}_t + \sigma_t^2} \right) \mathbf{P}_{t-1}, && \text{(A.8)}
\end{aligned}$$

respectively.

A.2 Kalman fixed-point smoother

Following Anderson and Moore (1979, pp. 170-6), consider replacing the state-space framework of (2.1) and (A.1) by

$$y_t = \begin{bmatrix} \mathbf{x}_t' & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta}_t \\ \boldsymbol{\theta}_t^a \end{bmatrix} + \varepsilon_t, \tag{A.9}$$

$$\begin{bmatrix} \boldsymbol{\theta}_t \\ \boldsymbol{\theta}_t^a \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta}_{t-1} \\ \boldsymbol{\theta}_{t-1}^a \end{bmatrix} + \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \boldsymbol{\omega}_t, \tag{A.10}$$

with the state vector at a fixed $t = j$ satisfying $\begin{bmatrix} \boldsymbol{\theta}_j' & \boldsymbol{\theta}_j^{a'} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\theta}_j' & \boldsymbol{\theta}_j' \end{bmatrix}$. 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 in the right hand side of these equalities are clearly in accordance to what we have defined as fixed-point smoothed estimates in the main text (see footnote 8), 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 (A.9)-(A.10) 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 (A.2)-(A.4), 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 found, from Anderson and Moore (1979), to be given by

$$\hat{\boldsymbol{\theta}}_{j|t} = \hat{\boldsymbol{\theta}}_{j|t-1} + \mathbf{K}_t^a \left(y_t - \mathbf{x}_t' \hat{\boldsymbol{\theta}}_{t-1} \right), \tag{A.11}$$

$$\mathbf{K}_t^a = \frac{\boldsymbol{\Sigma}_{t-1} \mathbf{x}_t}{\mathbf{x}_t' \mathbf{P}_{t-1} \mathbf{x}_t + \sigma_t^2}, \tag{A.12}$$

$$\boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_{t-1} \left(\mathbf{I} - \mathbf{K}_t \mathbf{x}_t' \right)', \tag{A.13}$$

$$\mathbf{P}_{j|t} = \mathbf{P}_{j|t-1} - \boldsymbol{\Sigma}_{t-1} \mathbf{x}_t \mathbf{K}_t^{a'}, \tag{A.14}$$

where $\boldsymbol{\Sigma}_j = \mathbf{P}_j$, and the conditional covariance matrix of the coefficients smoothed estimates errors is here given by (A.14), i.e., $\mathbf{P}_{j|t} = E\left[\left(\boldsymbol{\theta}_j - \hat{\boldsymbol{\theta}}_{j|t} \right) \left(\boldsymbol{\theta}_j - \hat{\boldsymbol{\theta}}_{j|t} \right)' \right]$. It is also important to note the use of terms from the filtering recursions, $\mathbf{K}_t$ and $\mathbf{P}_{t-1}$. The smoother associated to each learning algorithm, thence, follows automatically from what the different assumptions on (A.5)-(A.6) and (A.7)-(A.8) imply for these recursions.

B Tables

Table 2: Calibration of parameters for simulation.

Parameters	Description	Calibrations values	
		Inflation-like	Growth-like
(a) For artificial series:			
σ_ε^2	Variance of ε_t in (4.1).	2.25	13.00
σ_ω^2	Variance of ω_{t+1} in (4.2).	7×10^{-5}	7×10^{-5}
$\bar{\theta}$	Steady-state value of θ_t.	0.80	0.40
β	Persistence of deviations from $\bar{\theta}$.	0.999	0.999
(b) For algorithms:			
$\bar{\gamma}_1$	LS "low" constant learning gains.	0.01	0.01
$\bar{\gamma}_2$	LS "usual" constant learning gains.	0.02	0.02
$\bar{\gamma}_3$	LS "median" constant learning gains.	0.10	0.10
$\bar{\gamma}_4$	LS "high" constant learning gains.	0.40	0.40
$\bar{\mu}_1$	SG "low" constant learning gain.	0.05×10^{-2}	0.08×10^{-3}
$\bar{\mu}_2$	SG "usual" constant learning gain.	0.10×10^{-2}	0.17×10^{-3}
$\bar{\mu}_3$	SG "median" constant learning gain.	0.51×10^{-2}	0.83×10^{-3}
$\bar{\mu}_4$	SG "high" constant learning gain.	2.05×10^{-2}	3.34×10^{-3}
(c) For initialization methods:			
N	Size of initial sample of training data.	75	75
ϵ	Convergence tolerance for smoothed initials.	0.01	0.01
$\bar{S}$	Maximum number of smoothing repetitions.	100	100

The calibrations for the artificial series follow those recommended by Doan et al. (1984) and Hamilton (1994, pp. 400-3), (roughly) adjusted to the variables of interest. 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.

Table 1: Comparative of initialization methods on their applied appropriateness.

Initialization methods	Benefit(s)	Drawback(s)	Previously applied in economics
Exact: initials obtained from theoretical (known) REE.	- Most *theoretically-grounded* between these methods. - Low *computational cost*.	- *Unsuitable* for empirical applications, unless accurate previous estimates are available. - May be lead to *incoherent* estimates under non-stationary setups.	Sargent (1999); Marcet and Nicolini (2003); Bullard and Eusepi (2005)[1]; Orphanides and Williams (2005b); Carceles-Poveda and Giannitsarou (2007)
Ad-hoc: initials hand-picked by researcher. Usually taking values closer to those from the *Exact* method.	- *Useful* for sensitivity analysis. - Low *computational cost*.	- *Unsuitable* for empirical applications. - May be *incoherent* with the algorithm/gain of interest. - High degree of *subjective* freedom.	Milani (2007); Carceles-Poveda and Giannitsarou (2007, 2008).
Diffuse-track: initials estimated from a training pre-sample of actual/simulated data, using the algorithm/gain of interest.	- Provides initials *coherent* with algorithm/gain of interest. - Easiness of *implementation*, as it follows directly from the algorithm's usual operation.	- May require *unfeasible* amount of training obs. for convergence, specially for small gain values. - *Subjective* determination of amount of training obs.	Milani (2007, 2008)[2]; Huang et al. (2009); Pfajfar and Santoro (2010)[3]; Eusepi and Preston (2011); Milani (2011).
Diffuse-ordinary: same as *Diffuse-track*, but using a decreasing gain algorithm, e.g., the ordinary least squares.	- Accelerated *convergence*. - Appropriate for a *time-invariant* context, e.g., a REE path.	- May (and usually does) converge to an estimate *incoherent* with the algorithm/gain of interest. - *Subjective* determination of amount of training obs.	Williams (2003); Orphanides and Williams (2005a); Carceles-Poveda and Giannitsarou (2007, 2008).
Smoothing: initials estimated from a training pre-sample by smoothing the *Diffuse-track* estimates backwards repeatedly.	- Provides initials *coherent* with algorithm/gain of interest. - *Feasible* with the amount of macro data usually available. - Exchanges training obs. for computational cost.	- Highest *computational cost* between these methods. - Trickier *implementation*.	The present work.

The initialization methods are assessed in accordance to the results and discussions presented in the text. Some of the referred works applied more than one form of initialization for robustness purposes, and so figure in more than one of the above classifications. Also notice that most of the works classified under the same classification are not, strictly speaking, identical with respect to their initialization method(s), but share the main characteristic defining the corresponding classification. (1) In Bullard and Eusepi (2005) the initialization method is not explicitly stated, but it is understood that the estimates are set to depart from the values implied by the model REE. (2) In Milani (2007, 2008) a *Diffuse-* (under our terminology) initialization is used, but there is no explicit reference to whether this follows the *track* or the *ordinary* method. The use of the former is later acknowledged by the same author in Milani (2011), and for this reason we classify all these papers under the *Diffuse-track* method. (3) In Pfajfar and Santoro (2010) the initialization is determined jointly with the calibration of the gain parameter using the whole sample of data instead of a pre-sample.

Table 3: Average MSDs after initializations - LEAST SQUARES on INFLATION-like data.

Gains	Initializations	Average MSDs after initializations - samples					Average long run MSDs
		76-100	101-150	151-200	201-250	251-300	750-1000
$\bar{\gamma}_1$	*Diffuse-track*	**0.0051** **[4.9]**	0.0043 [0.7]	0.0040 [-1.1]	0.0040 [-0.9]	0.0043 [0.4]	0.0042 (0.0002)
	Diffuse-ordinary	**0.0060** **[10.1]**	**0.0048** **[3.6]**	0.0041 [-0.2]	0.0041 [-0.3]	0.0043 [0.7]	0.0042 (0.0002)
	Smoothing	**0.0052** **[5.6]**	**0.0046** **[2.3]**	0.0042 [-0.0]	0.0042 [0.3]	0.0044 [1.2]	0.0042 (0.0002)
$\bar{\gamma}_3$	*Diffuse-track*	0.0175 [-1.3]	0.0175 [-1.4]	0.0183 [-0.6]	0.0184 [-0.6]	0.0188 [-0.2]	0.0190 (0.0011)
	Diffuse-ordinary	**0.0064** **[-11.5]**	**0.0139** **[-4.7]**	0.0182 [-0.7]	0.0183 [-0.6]	0.0188 [-0.2]	0.0190 (0.0011)
	Smoothing	0.0174 [-1.3]	0.0174 [-1.4]	0.0183 [-0.6]	0.0183 [-0.6]	0.0187 [-0.2]	0.0189 (0.0011)
$\bar{\gamma}_4$	*Diffuse-track*	0.1011 [-0.3]	0.0990 [-0.6]	0.1030 [-0.1]	0.1024 [-0.2]	0.1009 [-0.3]	0.1037 (0.0084)
	Diffuse-ordinary	**0.0603** **[-5.3]**	0.0999 [-0.6]	0.1038 [-0.1]	0.1032 [-0.2]	0.1020 [-0.3]	0.1049 (0.0084)
	Smoothing	0.0891 [-0.1]	0.0865 [-0.4]	0.0900 [0.0]	0.0891 [-0.1]	0.0876 [-0.3]	0.0897 (0.0081)

The average statistics refer to the mean-square deviation of coefficient estimates from their true counterparts, as defined in (2.5). The second line of headers indicate the samples of observations used to compute the average statistics. The values in round brackets, (...), are standard deviations of the statistic from the corresponding long run average. The values in square brackets, [...], refer to the number of (long run) standard deviations by which the corresponding short run average deviates from the long run average. Emphasis is given in **bold** to those short run averages that deviate by more than two standard deviations from the corresponding long run average. Gain calibrations follow from specifications in table 2.

Table 4: Average MSDs after initializations - STOCHASTIC GRADIENT on INFLATION-like data.

Gains	Initializations	Average MSDs after initializations - samples					Average long run MSDs
		76-100	101-150	151-200	201-250	251-300	750-1000
$\bar{\mu}_1$	*Diffuse-track*	**0.2280** **[333.1]**	**0.1802** **[259.9]**	**0.1307** **[184.2]**	**0.0976** **[133.5]**	**0.0737** **[97.0]**	0.0102 (0.0007)
	Diffuse-ordinary	**0.0843** **[292.4]**	**0.0690** **[233.6]**	**0.0513** **[165.7]**	**0.0393** **[119.7]**	**0.0306** **[86.4]**	0.0081 (0.0003)
	Smoothing	**0.0085** **[5.8]**	**0.0085** **[5.5]**	**0.0079** **[3.6]**	**0.0077** **[3.1]**	**0.0076** **[2.6]**	0.0067 (0.0003)
$\bar{\mu}_3$	*Diffuse-track*	**0.0157** **[20.9]**	**0.0088** **[6.4]**	0.0062 [0.9]	0.0056 [-0.4]	0.0057 [-0.3]	0.0058 (0.0005)
	Diffuse-ordinary	**0.0551** **[104.2]**	**0.0191** **[28.2]**	**0.0080** **[4.8]**	0.0060 [0.5]	0.0058 [-0.0]	0.0058 (0.0005)
	Smoothing	0.0058 [-0.0]	0.0056 [-0.4]	0.0055 [-0.6]	0.0054 [-0.9]	0.0056 [-0.4]	0.0058 (0.0005)
$\bar{\mu}_4$	*Diffuse-track*	0.0224 [-0.1]	0.0223 [-0.1]	0.0279 [1.5]	0.0210 [-0.5]	0.0238 [0.3]	0.0226 (0.0034)
	Diffuse-ordinary	**0.0481** **[7.4]**	0.0240 [0.3]	**0.0303** **[2.2]**	0.0213 [-0.5]	0.0242 [0.4]	0.0228 (0.0034)
	Smoothing	0.0180 [-0.5]	0.0193 [-0.1]	0.0247 [1.5]	0.0179 [-0.5]	0.0208 [0.3]	0.0197 (0.0034)

See notes to table 3.

Table 5: Average MSDs after initializations - LEAST SQUARES on GROWTH-like data.

Gains	Initializations	Average MSDs after initializations - samples					Average long run MSDs
		76-100	101-150	151-200	201-250	251-300	750-1000
$\bar{\gamma}_1$	*Diffuse-track*	**0.0122** **[25.3]**	**0.0098** **[13.2]**	**0.0087** **[7.5]**	**0.0079** **[3.7]**	**0.0076** **[2.4]**	0.0071 (0.0002)
	Diffuse-ordinary	**0.0128** **[28.1]**	**0.0099** **[13.9]**	**0.0085** **[6.5]**	**0.0078** **[3.1]**	**0.0076** **[2.0]**	0.0071 (0.0002)
	Smoothing	**0.0127** **[27.6]**	**0.0103** **[15.5]**	**0.0088** **[8.0]**	**0.0079** **[4.0]**	**0.0076** **[2.1]**	0.0071 (0.0002)
$\bar{\gamma}_3$	*Diffuse-track*	0.0387 [-0.2]	0.0390 [0.0]	0.0407 [1.0]	0.0399 [0.5]	0.0401 [0.6]	0.0390 (0.0018)
	Diffuse-ordinary	**0.0145** **[-13.4]**	**0.0328** **[-3.4]**	0.0407 [0.9]	0.0399 [0.5]	0.0401 [0.6]	0.0390 (0.0018)
	Smoothing	0.0385 [-0.2]	0.0389 [0.0]	0.0407 [1.0]	0.0398 [0.5]	0.0400 [0.6]	0.0389 (0.0019)
$\bar{\gamma}_4$	*Diffuse-track*	0.2080 [0.0]	0.2069 [-0.1]	0.2109 [0.2]	0.2112 [0.3]	0.2069 [-0.1]	0.2077 (0.0133)
	Diffuse-ordinary	**0.1314** **[-5.7]**	0.2069 [-0.1]	0.2109 [0.2]	0.2112 [0.3]	0.2069 [-0.1]	0.2077 (0.0133)
	Smoothing	0.1895 [0.1]	0.1873 [-0.1]	0.1920 [0.3]	0.1911 [0.2]	0.1883 [0.0]	0.1881 (0.0123)

See notes to table 3.

Table 6: Average MSDs after initializations - STOCHASTIC GRADIENT on GROWTH-like data.

Gains	Initializations	Average MSDs after initializations - samples					Average long run MSDs
		76-100	101-150	151-200	201-250	251-300	750-1000
$\bar{\mu}_1$	*Diffuse-track*	**0.1404** **[55.5]**	**0.1289** **[50.0]**	**0.1116** **[41.8]**	**0.0969** **[34.7]**	**0.0844** **[28.8]**	0.0242 (0.0021)
	Diffuse-ordinary	**0.0819** **[57.9]**	**0.0763** **[52.7]**	**0.0669** **[44.0]**	**0.0589** **[36.6]**	**0.0523** **[30.5]**	0.0194 (0.0011)
	Smoothing	**0.0242** **[16.0]**	**0.0244** **[16.4]**	**0.0236** **[14.7]**	**0.0227** **[13.0]**	**0.0220** **[11.6]**	0.0163 (0.0005)
$\bar{\mu}_3$	*Diffuse-track*	**0.0260** **[75.6]**	**0.0158** **[33.7]**	**0.0108** **[13.0]**	**0.0088** **[4.9]**	**0.0082** **[2.3]**	0.0076 (0.0002)
	Diffuse-ordinary	**0.0630** **[228.0]**	**0.0308** **[95.4]**	**0.0151** **[31.0]**	**0.0102** **[10.5]**	**0.0086** **[4.2]**	0.0076 (0.0002)
	Smoothing	**0.0121** **[18.7]**	**0.0096** **[8.3]**	**0.0085** **[4.0]**	0.0079 [1.4]	0.0078 [0.9]	0.0076 (0.0002)
$\bar{\mu}_4$	*Diffuse-track*	0.0214 [-0.7]	0.0215 [-0.7]	0.0226 [0.6]	0.0214 [-0.8]	0.0220 [-0.1]	0.0221 (0.0009)
	Diffuse-ordinary	**0.0482** **[28.5]**	**0.0249** **[3.0]**	0.0228 [0.8]	0.0214 [-0.8]	0.0220 [-0.1]	0.0221 (0.0009)
	Smoothing	0.0192 [-1.1]	0.0194 [-0.9]	0.0204 [0.3]	0.0193 [-0.9]	0.0201 [-0.0]	0.0201 (0.0009)

See notes to table 3.

Table 7: Variances and mean squared errors of forecasts - LEAST SQUARES on US data.

Gains	Initializations	Statistics for INFLATION - samples				Statistics for GROWTH - samples			
		76-100	101-150	151-200	201-250	76-100	101-150	151-200	201-250
$\bar{\gamma}_a$	*Diffuse-track*	0.73	4.29	0.60	0.83	1.94	3.06	0.44	1.31
		(2.14)	(2.87)	(0.62)	(0.90)	(15.49)	(20.82)	(3.71)	(7.88)
	Diffuse-ordinary	0.64	3.16	0.50	0.73	1.76	2.77	0.45	1.31
		(2.23)	(3.50)	(0.61)	(0.87)	(14.88)	(21.11)	(3.72)	(7.74)
	Smoothing	0.72	3.37	0.51	0.75	2.03	3.32	0.48	1.36
		(2.21)	(3.26)	(0.61)	(0.87)	(15.45)	(20.74)	(3.71)	(7.79)
$\bar{\gamma}_b$	*Diffuse-track*	0.82	5.51	0.65	0.90	1.73	2.78	0.45	2.59
		(2.09)	(2.69)	(0.64)	(0.93	(15.63)	(21.20)	(3.73)	(7.50)
	Diffuse-ordinary	0.61	4.66	0.62	0.88	1.47	2.27	0.44	2.55
		(2.17)	(2.78)	(0.64)	(0.92)	(14.72)	(21.46)	(3.73)	(7.49)
	Smoothing	0.74	4.79	0.62	0.89	1.82	2.82	0.46	2.58
		(2.14)	(2.78)	(0.64)	(0.92)	(15.48)	(21.11)	(3.73)	(7.49)
$\bar{\gamma}_c$	*Diffuse-track*	0.94	6.67	0.63	0.90	1.58	2.93	0.61	5.19
		(2.04)	(2.80)	(0.65)	(0.93)	(16.10)	(21.95)	(3.82)	(7.51)
	Diffuse-ordinary	0.59	6.15	0.63	0.90	1.15	2.55	0.61	5.19
		(2.09)	(2.83)	(0.65)	(0.93)	(14.69)	(22.10)	(3.82)	(7.51)
	Smoothing	0.85	6.55	0.63	0.90	1.57	2.91	0.61	5.19
		(2.03)	(2.79)	(0.65)	(0.93)	(15.98)	(21.93)	(3.82)	(7.51)

The statistics presented here refer to variances of forecasts (first row for each gain/initial) and mean squared forecast error (in round brackets at second row for each gain/initial). The second line of headers indicates the samples of observations used to compute these statistics. Gain calibrations follow from specifications in the text.

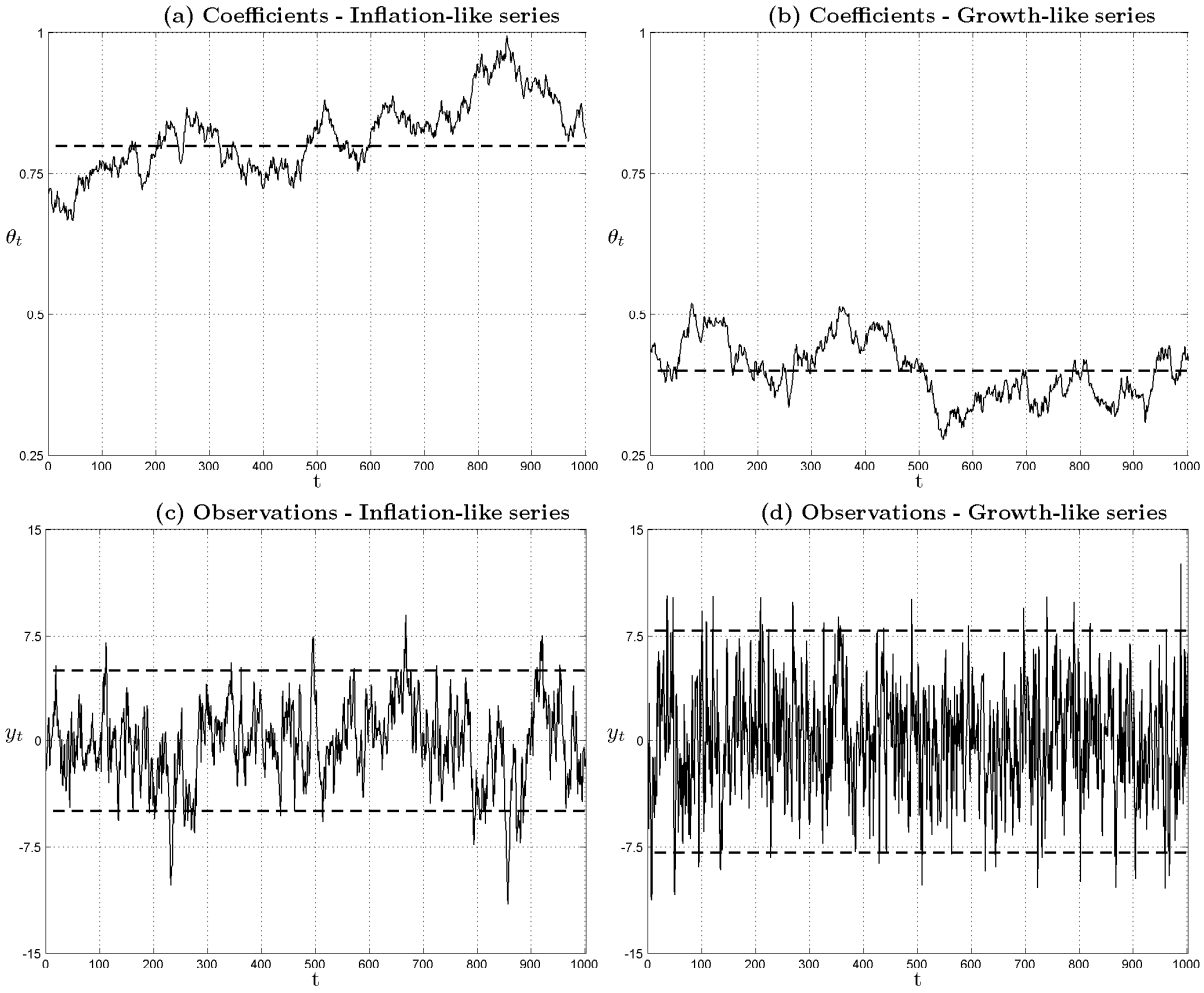
Table 8: Variances and mean squared errors of forecasts - STOCHASTIC GRADIENT on US data.

Gains	Initializations	Statistics for INFLATION - samples				Statistics for GROWTH - samples			
		76-100	101-150	151-200	201-250	76-100	101-150	151-200	201-250
$\bar{\mu}_a$	*Diffuse-track*	0.31	3.91	0.54	0.76	1.44	1.89	0.53	1.58
		(1.90)	(3.87)	(0.63)	(0.92)	(15.02)	(21.31)	(3.74)	(7.93)
	Diffuse-ordinary	0.34	3.95	0.55	0.77	1.36	1.86	0.53	1.58
		(1.90)	(3.72)	(0.63)	(0.92)	(14.94)	(21.30)	(3.74)	(7.93)
	Smoothing	0.57	4.43	0.57	0.80	1.46	1.92	0.53	1.58
		(1.92)	(3.10)	(0.63)	(0.91)	(15.10)	(21.26)	(3.74)	(7.94)
$\bar{\mu}_b$	*Diffuse-track*	0.62	5.87	0.49	0.83	1.50	2.60	0.95	3.00
		(1.93)	(2.99)	(0.62)	(0.96)	(15.67)	(22.60)	(3.93)	(7.86)
	Diffuse-ordinary	0.35	5.66	0.49	0.83	0.94	2.53	0.95	3.00
		(1.94)	(3.28)	(0.62)	(0.96)	(14.73)	(22.88)	(3.92)	(7.86)
	Smoothing	0.68	5.92	0.49	0.83	1.50	2.60	0.95	3.00
		(1.95)	(2.93)	(0.62)	(0.96)	(15.64)	(22.62)	(3.93)	(7.86)
$\bar{\mu}_c$	*Diffuse-track*	0.83	7.16	0.35	0.95	2.72	5.69	1.45	4.84
		(2.05)	(2.82)	(0.64)	(1.24)	(17.22)	(25.69)	(4.32)	(7.58)
	Diffuse-ordinary	0.41	7.01	0.35	0.95	1.26	5.75	1.45	4.84
		(2.05)	(3.01)	(0.64)	(1.24)	(15.24)	(26.02)	(4.32)	(7.58)
	Smoothing	0.83	6.49	0.35	0.95	2.01	5.09	0.59	4.65
		(2.05)	(2.76)	(0.64)	(1.24)	(16.40)	(26.02)	(3.91)	(7.57)

See notes to table 7.

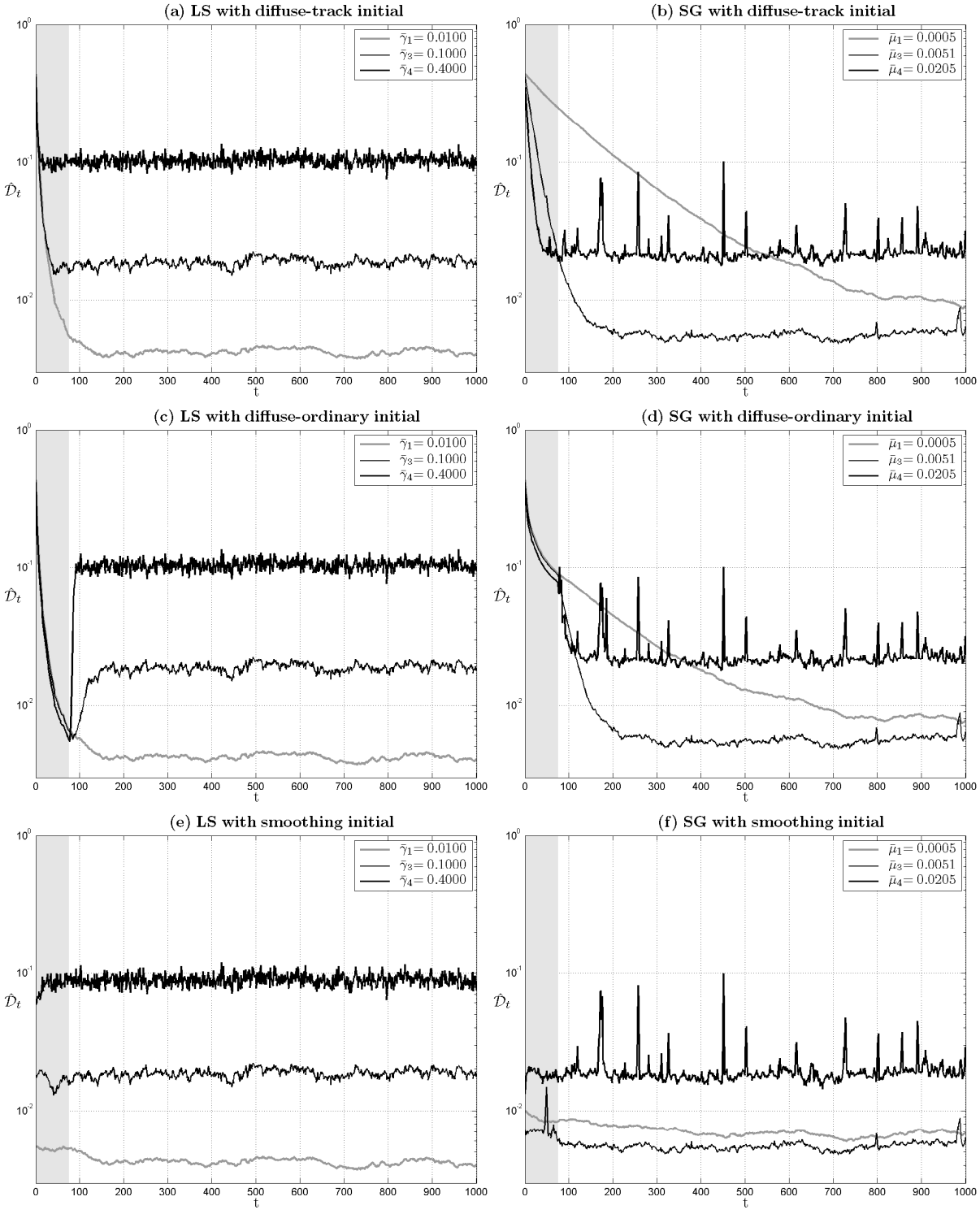
C Figures

Figure 1: Example of artificially generated series.



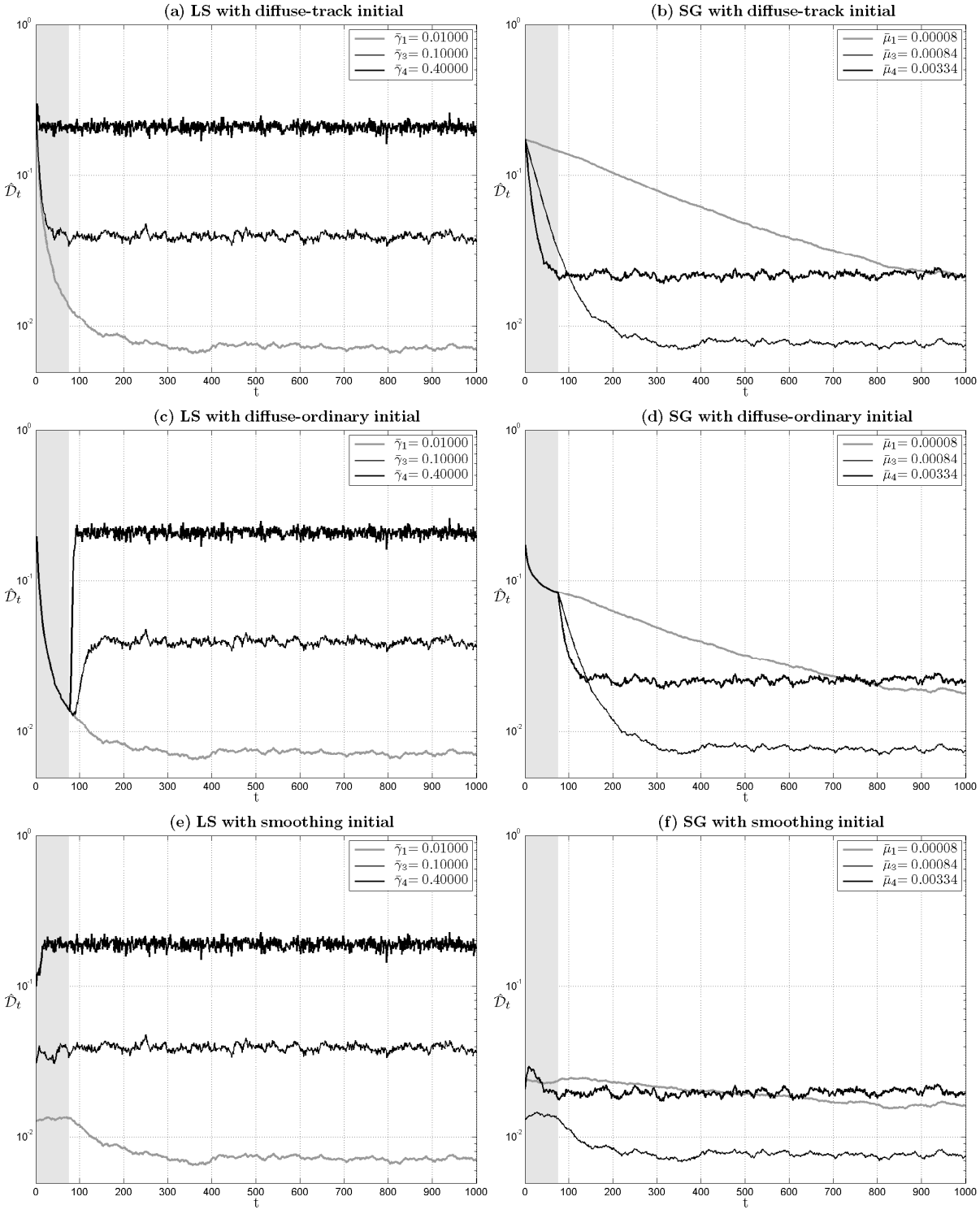
The dashed lines in the top panels represent the coefficients' steady-state values, $\bar{\theta}$, according to table 2. The dashed lines in the bottom panels represent ± 2 standard deviations bands around zero, as computed from (4.1) assuming $\theta_t = \bar{\theta}$, and the calibrations in table 2.

Figure 2: MSD learning curves for INFLATION-like artificial data.



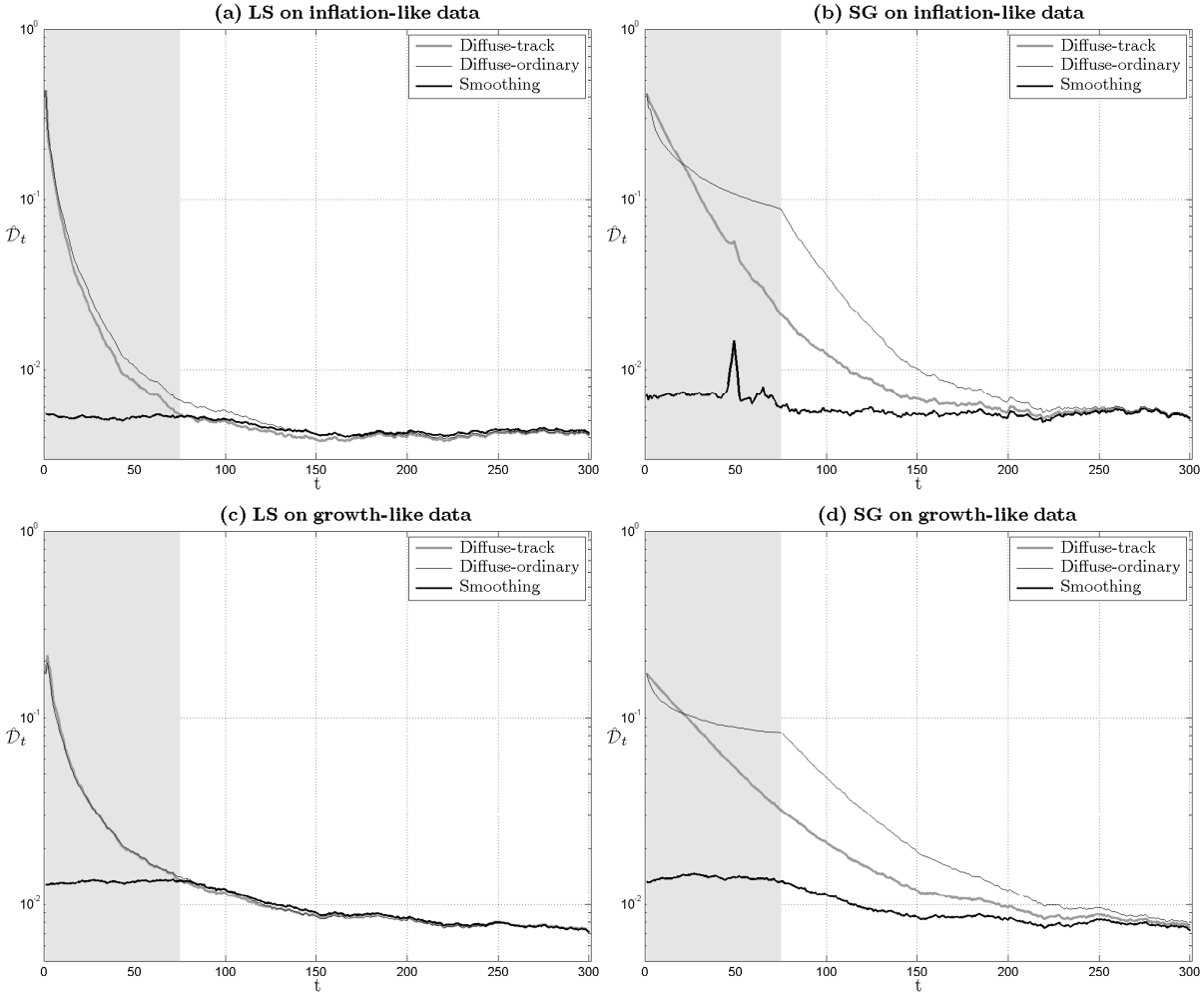
The shaded areas indicate the portion of observation left aside for use by the initialization methods. $\hat{\mathcal{D}}_t$ stands for the sample correspondent to the mean-square deviation (MSD) as defined in (2.5). Notice that all the vertical axes are on the same logarithmic scale for comparative purposes. See the text for further details on how these curves were computed.

Figure 3: MSD learning curves for GROWTH-like artificial data.



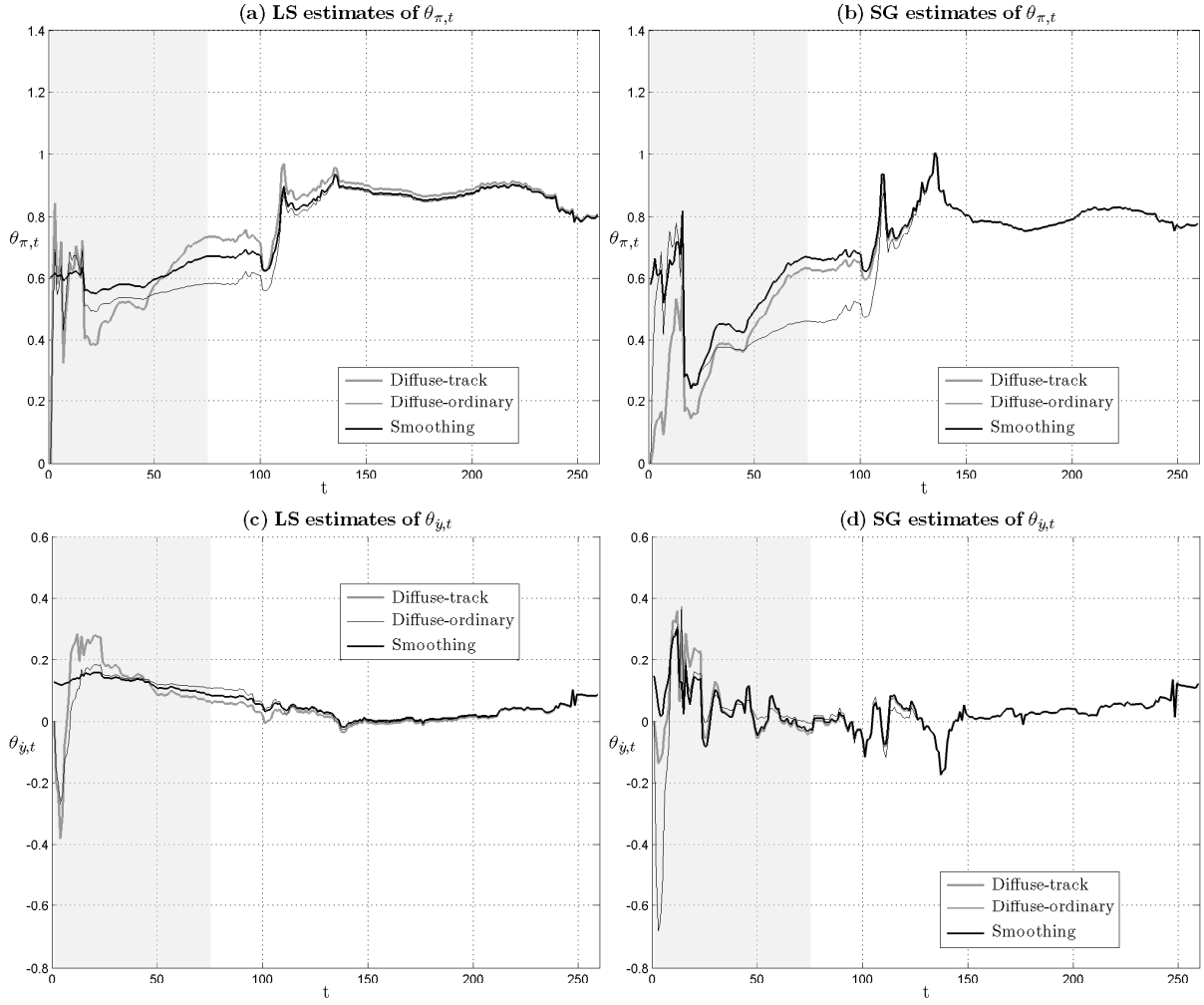
See notes to figure 2.

Figure 4: MSD learning curves by initialization for a given gain.



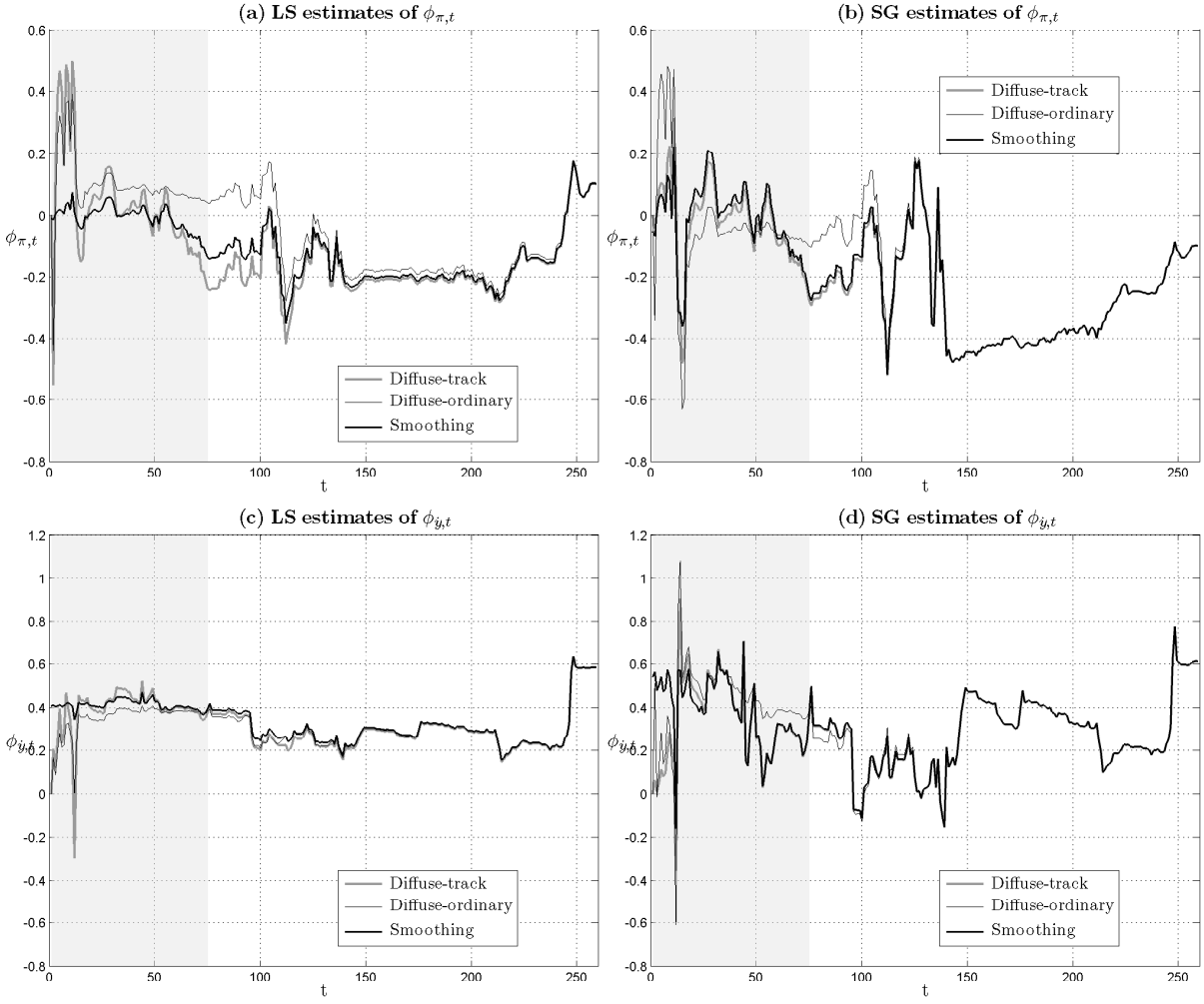
The plotted learning curves refer to a unique learning gain for each algorithm, namely $\bar{\gamma}_2$ and $\bar{\mu}_2$, as specified in table 2. See also the notes to figure 2.

Figure 5: Coefficients estimates from empirical application with US data - equation on INFLATION.



The plotted estimates refer to those obtained with a unique learning gain for each algorithm, namely $\bar{\gamma}_b$ and $\bar{\mu}_b$, as specified in the text. The shaded areas indicate the portion of observation left aside for use by the initialization methods. Notice that the vertical axes of each row of plots are on the same scale to facilitate comparisons between algorithms' estimates for the same parameter.

Figure 6: Coefficients estimates from empirical application with US data - equation on GROWTH.



See notes to figure 5.

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. A note on exact correspondences between adaptive learning algorithms and the kalman filter. Economics Letters (Forthcoming).

Branch, W.A., Evans, G.W., 2006. A simple recursive forecasting model. Economics Letters 91, 158–166.

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

Bray, M.M., Savin, N.E., 1986. Rational expectations equilibria, learning, and model specification. Econometrica 54, 1129–1160.

Bullard, J., Eusepi, S., 2005. Did the great inflation occur despite policymaker commitment to a taylor rule? Review of Economic Dynamics 8, 324 – 359.

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

Carceles-Poveda, E., Giannitsarou, C., 2008. Asset pricing with adaptive learning. Review of Economic Dynamics 11, 629 – 651.

Doan, T., Litterman, R., Sims, C., 1984. Forecasting and conditional projection using realistic prior distributions. Econometric Reviews 3, 1–100.

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., 2009. Learning and macroeconomics. Annual Review of Economics 1, 421–49.

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. fourth edition edition.

Huang, K.X., Liu, Z., Zha, T., 2009. Learning, adaptive expectations and technology shocks. The Economic Journal 119, 377–405.

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

Marcet, A., Nicolini, J.P., 2003. Recurrent hyperinflations and learning. American Economic Review 93, 1476–1498.

Marcet, A., Sargent, T.J., 1988. The fate of systems with "adaptive" expectations. The American Economic Review 78, 168–172.

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.

Milani, F., 2007. Expectations, learning and macroeconomic persistence. Journal of Monetary Economics 54, 2065–2082.

Milani, F., 2008. Learning, monetary policy rules, and macroeconomic stability. Journal of Economic Dynamics and Control 32, 3148 – 3165.

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

Orphanides, A., Williams, J.C., 2005b. Inflation scares and forecast-based monetary policy. Review of Economic Dynamics 8, 498 – 527.

Pfajfar, D., Santoro, E., 2010. Heterogeneity, learning and information stickiness in inflation expectations. Journal of Economic Behavior & Organization 75, 426–444.

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

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