

MPRA

Munich Personal RePEc Archive

Particle Markov Chain Monte Carlo Techniques of Unobserved Component Time Series Models Using Ox

Nima Nonejad

1. May 2014

Online at <http://mpra.ub.uni-muenchen.de/55662/>

MPRA Paper No. 55662, posted 1. May 2014 15:44 UTC

Particle Markov Chain Monte Carlo Techniques of Unobserved Component Time Series Models Using Ox

Nima Nonejad^{†*}

Aarhus University and CREATES

Abstract

This paper details particle Markov chain Monte Carlo (PMCMC) techniques for analysis of unobserved component time series models using several economic data sets. PMCMC provides a very compelling, computationally fast and efficient framework for estimation and model comparison. For instance, we estimate a stochastic volatility model with leverage effect and one with Student-t distributed errors. We also model time series characteristics of US inflation rate by considering a heteroskedastic ARFIMA model where heteroskedasticity is specified by means of a Gaussian stochastic volatility process.

Keywords: Bayes, Metropolis-Hastings, Particle filter, Unobserved components

(*JEL*: C11, C22, C63)

*The author acknowledges support from CREATES-Center for Research in Econometric Analysis of Time Series (DNRF78), funded by the Danish National Research Foundation. † Department of Economics and Business, Aarhus University, Fuglesangs Alle 4, 8210 Aarhus V, Denmark. Phone: +45 87165325. E-mail: nnonejad@creates.au.dk, nimanonejad@gmail.com. I would like to thank Asger Lunde and Tommaso Proietti for useful comments and suggestions. This version 05/01/2014.

1 Introduction

In this article we analyze different economic data sets using particle Markov chain Monte Carlo (PMCMC) techniques, see Andrieu et al. (2010) implemented in Ox (Doornik 2009). The aim of this paper is not to focus on the properties of PMCMC nor to provide thorough analysis of empirical data. Instead, we aim to describe the basic steps of PMCMC together with details on implementation of some of the key algorithms in Ox. These algorithms are chosen for the insights that they provide. They are not always the most advanced or quickest way of programming in Ox. Rather, we show that Ox provides a very compelling and computationally fast framework for estimating rather advanced econometric models. We illustrate these methods on three problems, producing rather generic methods.

The main difference between MCMC and PMCMC is that where in traditional MCMC, in particular Gibbs sampling problems we resort to converting nonlinear or non-Gaussian models to linear and Gaussian state space models in order to draw the latent states, in the PMCMC framework we integrate these latent states out directly using the particle filter and thereafter sample the model parameters using Metropolis-Hastings. This is due to the results in Andrieu et al. (2010) where they show that when an unbiasedly estimated likelihood is used inside a Metropolis-Hastings algorithm then the estimation error makes no difference to the equilibrium distribution of the algorithm. We believe that applying the PMCMC methodology to unobserved component models is in fact the most important contribution that we make. As we shall see for these type of models, PMCMC requires limited design effort on the user's part especially if one desires to change some features in a particular model. It allows us straightforwardly to develop a sampling algorithm by changing only few lines in our codes. On the other hand, estimating the same type of models using "pure" Gibbs sampling would require relatively more programming effort.

We briefly introduce the main concepts of PMCMC in section 2. The initial model in Section 3 is the standard stochastic volatility (SV) model with Gaussian errors applied to a financial data set concerning daily OMXC20 returns. Next, we consider different well-known extensions of the SV model. The first extension is a SV model with Student-t errors. In the second extension we incorporate a leverage effect by modeling a correlation parameter between measurement and state errors. In the third extension we implement a model that has both stochastic volatility and moving average errors, see for example Chan (2013). The fourth extension is PMCMC implementation of the stochastic volatility in mean model of Koopman and Hol Uspensky (2002). In this specification the unobserved volatility process appears in both the conditional mean and the conditional variance. Finally, we consider a two factor stochastic volatility model as in Harvey et al. (1994) and Shephard (1996). We show that PMCMC provides a straightforward procedure for estimation and marginal likelihood calculation of these models. Specifically, within the PMCMC framework computing the

marginal likelihood is relatively easy using the method of Gelfand and Dey (1994) as the integrated likelihood is easily available from the particle filter.

Thereafter, we reconsider the unobserved components model of the US inflation rate, see Stock and Watson (2007), Grassi and Proietti (2010). We estimate different specifications of the unobserved components model using PMCMC. Model selection is again carried out by comparing marginal likelihoods between models. Results indicate that the specification in which the volatility of both the regular and irregular components of inflation evolve according to SV processes performs best in terms of ML. Finally, we show that it is also relatively easy to estimate more complicated models using PMCMC. We do this by estimating an autoregressive fractionally integrated moving average (ARFIMA) model with time-varying volatility modeled as a Gaussian SV process using monthly postwar US core inflation data from 1960 to 2013.

The main concepts of PMCMC with focus on Metropolis-Hastings and the particle filter are presented in section 2. In section 3 we present several applications to demonstrate the performance of the algorithms using different economic data sets and finally the last section concludes.

2 Markov Chains and Particle Filters

Consider the simplest formulation of the stochastic volatility (SV) model

$$y_t = \exp(\alpha_t/2) \varepsilon_t, \quad \varepsilon_t \sim N(0, 1) \quad (2.1)$$

$$\alpha_{t+1} = \mu + \phi(\alpha_t - \mu) + \sigma \eta_t, \quad \eta_t \sim N(0, 1) \quad (2.2)$$

where y_t is the observed data, $(\alpha_1, \dots, \alpha_T)'$ are the unobserved log-volatilities, μ is the drift in the state equation, σ is the volatility of log-volatility and ϕ is the persistence parameter. Typically, we would impose that $|\phi| < 1$ so that we have a stationary process with the initial condition, $\alpha_1 \sim N(\mu, \sigma^2 / (1 - \phi^2))$. Let $\theta = (\mu, \phi, \sigma^2)'$ and $Y_T = (y_1, \dots, y_T)'$. This model has been heavily analyzed in time series econometrics, see for example Kim et al. (1998). The major challenge of estimating this model is that while sampling $p(\theta | \alpha_1, \dots, \alpha_T, Y_T)$ is relatively easy, sampling $\alpha_1, \dots, \alpha_T \sim p(\alpha_1, \dots, \alpha_T | \theta, Y_T)$ is often difficult.

Within the Gibbs sampling framework, the most popular approach for estimating (2.1)-(2.2) is the so-called auxiliary mixture sampler, see Kim et al. (1998). The idea is to approximate the SV model using a mixture of linear Gaussian models. Specifically, we can square both sides of (2.1) and take the logarithm such that $y_t^* = \alpha_t + \varepsilon_t^*$ where $y_t^* = \log y_t^2$ and $\varepsilon_t^* = \log \varepsilon_t^2$. Kim et al. (1998) show that ε_t^* can be approximated by a seven-component Gaussian mixture density. We can then write this mixture density using an auxiliary random variable, $z_t \in \{1, \dots, 7\}$ that serves as the mixture component indicator. Hence, $\varepsilon_t^* | z_t \sim N(m_i, s_i^2)$ where $p(z_t = i) = \omega_i$. The values of m_i ,

s_i^2 and ω_i , $i = 1, \dots, 7$ are all fixed and given in Kim et al. (1998). Using this approximation the SV model can be expressed as a linear Gaussian state space model. Bayesian estimation can then be performed using standard Gibbs sampling techniques for linear Gaussian state space models, see for instance Kim and Nelson (1999). Finally, notice that using this specification we sample from the augmented posterior, $p(\theta, \alpha_1, \dots, \alpha_T, z_1, \dots, z_T | Y_T)$.

Within the PMCMC framework we need not use the above approximation. On the contrary, we approach estimating (2.1)-(2.2) directly by first integrating out $(\alpha_1, \dots, \alpha_T)'$ and thus obtaining $p(Y_T | \theta)$ where

$$p(Y_T | \theta) = \prod_{t=1}^T p(y_t | Y_{t-1}, \theta) \quad (2.3)$$

In more general cases we do not have a closed form expression for $p(y_t | Y_{t-1}, \theta)$ and can therefore only approximate it. In this paper, we use simulations to unbiasedly estimate each term on the right side of (2.3). This is carried out using the particle filter, see Kim et al. (1998), Doucet et al. (2000) and Creal (2012). We then use the main result of Andrieu et al. (2010) which states that when an unbiasedly estimated likelihood is used inside a MCMC algorithm then the estimation error makes no difference to the equilibrium distribution of the algorithm, the posterior distribution, $p(\theta | Y_T) \propto p(Y_T | \theta) p(\theta)$. Thus, conditional on $p(Y_T | \theta)$ and the prior, $p(\theta)$ we can sample θ from $p(\theta | Y_T)$ by using Metropolis-Hastings (M-H). M-H follows the steps:

1. Initialize, start with a vector of parameters, $\theta^{(0)}$, set $i = 1$.
2. Draw a candidate value $\theta^* \sim q(\theta | \theta^{(i-1)})$.
3. Accept θ^* with probability

$$a_{\text{MH}}(\theta^*, \theta^{(i-1)}) = \min \left\{ 1, \frac{p(\theta^* | Y_T) q(\theta^{(i-1)} | \theta^*)}{p(\theta^{(i-1)} | Y_T) q(\theta^* | \theta^{(i-1)})} \right\} \quad (2.4)$$

4. If θ^* is accepted, set $\theta^{(i)} = \theta^*$, else $\theta^{(i)} = \theta^{(i-1)}$.
5. Set $i = i + 1$ and repeat from Step 2.

The candidate density, $q(\theta | \theta^{(i-1)})$ can be chosen freely, though a density which is related to the target density would lead to better acceptance rates. We start by using the random walk Metropolis-Hastings algorithm, see Koop (2003). Thus, we generate θ^* from $q(\theta | \theta^{(i-1)}) \sim N(\theta^{(i-1)}, \Sigma_q)$ where Σ_q is chosen by the researcher in a manner to ensure a sufficient acceptance rate. A sufficient acceptance rate ensures that the chain moves in the appropriate direction. We follow Koop

(2003, page 98) and adjust Σ_q to get acceptance rates roughly around 30 to 40%. We do this by experimenting with different values of Σ_q until we find one which yields a reasonable acceptance rate probability.

To allow for faster convergence and even better mixing we follow So et al. (2005) and do the following: we perform the random walk M-H algorithm for the first N_1 of the total N PMCMC iterations and form the sample mean, $\hat{\theta}$ and sample covariance, $\hat{\Sigma}$ for $\{\theta^{(i)}\}_{i=1}^{N_1}$. Then, using the Gaussian proposal with mean $\hat{\theta}$ and covariance $\hat{\Sigma}$, we perform the following independent kernel M-H algorithm for the remaining $N_2 = N - N_1$ iterations. Hence,

1. At iteration i where $i \geq (N_1 + 1)$, generate θ^* from $N(\hat{\theta}, \hat{\Sigma})$
2. Accept θ^* with probability

$$a_{\text{MH}}(\theta^*, \theta^{(i-1)}) = \min \left\{ 1, \frac{p(\theta^* | Y_T) q(\theta^{(i-1)})}{p(\theta^{(i-1)} | Y_T) q(\theta^*)} \right\} \quad (2.5)$$

where $q(\theta^{(i-1)})$ is the Gaussian proposal density with mean $\hat{\theta}$ and covariance $\hat{\Sigma}$ evaluated at $\theta^{(i-1)}$.

3. Set $i = i + 1$ and repeat from Step 2.

Finally, for the SV model we specify the priors for μ , ϕ and σ^2 as

$$\mu \sim N(0, 1) \quad , \quad \frac{\phi + 1}{2} \sim \text{Beta}(20, 1.5) \quad , \quad \sigma^2 \sim \text{IG}\left(\frac{4}{2}, \frac{0.02}{2}\right)$$

where $\text{IG}\left(\frac{s_0}{2}, \frac{v_0}{2}\right)$ denotes the Inverse-gamma density, see Kim and Nelson (1999).

2.1 Particle filter

The particle filter is a sequential simulation device for filtering of non-Gaussian, nonlinear state space models. It can be thought of as a generalization of the Kalman filter, which is only able to analyze linear, Gaussian state space models. Both the particle and Kalman filters produce filtered estimates of $(\alpha_1, \dots, \alpha_T)'$ and $p(y_t | Y_{t-1}, \theta)$ for $t = 1, \dots, T$. In the Kalman case all these quantities are exact whereas in the particle filter case they are simulation-based estimates.

The main idea of the particle filter is to sample a cloud of particles, $\alpha_t^{(j)}$, $j = 1, \dots, M$ such that they together describe the density of the state variable at time t conditional on Y_t . At each t we propagate the particles $\alpha_t^{(j)}$ and update their associate weights. This way we prevent accumulation of errors by eliminating unpromising particles. In the following we give a brief description of a

very general particle filter that we use throughout this paper. For more details on particle filtering the reader is referred to Doucet et al. (2000) and Creal (2012). Our particle filter scheme is as follows:

1. Set $t = 1$ and $l_0 = 0$. Draw $\alpha_t^{(1)}, \dots, \alpha_t^{(M)}$ from $\alpha_t \mid \alpha_0, \theta$.
2. Compute $\tau_t^{(j)} = p\left(y_t \mid \alpha_t^{(j)}, Y_{t-1}, \theta\right)$ and $w_t^{(j)} = \tau_t^{(j)} / \left(\sum_{k=1}^M \tau_t^{(k)}\right)$ for $j = 1, \dots, M$.
3. Resample $\left\{\alpha_t^{(1)}, \dots, \alpha_t^{(M)}\right\}$ with probabilities $w_t^{(1)}, \dots, w_t^{(M)}$. First, draw $u \sim U(0, 1)$. Let $x^{(j)} = \frac{u}{M} + (j-1)/M$ for $j = 1, \dots, M$ and find indices i^1, \dots, i^M such that $\sum_{k=1}^{i^j-1} w_t^{(k)} < x^{(j)} \leq \sum_{k=1}^{i^j} w_t^{(k)}$. We refer to this step as the ‘‘Resampling step’’ for future references.
4. Sample $\alpha_{t+1}^{(j)} \sim \alpha_{t+1} \mid \alpha_t^{(i^j)}, \theta$ for $j = 1, \dots, M$.
5. Compute $l_t(\theta) = l_{t-1}(\theta) + \log\left(M^{-1} \sum_{j=1}^M \tau_t^{(j)}\right)$. Set $t = t + 1$ and goto step 2.

As an estimate for $E[\alpha_t \mid \alpha_1, \dots, \alpha_{t-1}, Y_{t-1}, \theta]$ we use $\hat{\alpha}_t = \sum_{j=1}^M w_t^{(j)} \alpha_t^{(j)}$. A basic implementation of the particle filter is provided in Table 1.

Del Moral (2004, Theorem 7.4.2) shows that $p(\widehat{Y_T} \mid \theta) = E[\exp(l_T(\theta))] = p(Y_T \mid \theta)$. Therefore, since the particle filter provides us with an unbiased estimate of $p(Y_T \mid \theta)$ we can use the result in Andrieu et al. (2010) and replace $p(Y_T \mid \theta)$ with $p(\widehat{Y_T} \mid \theta)$ inside a MCMC sampling scheme. Thereafter, we sample θ which has a relatively lower dimension using Metropolis-Hastings. Basic code for evaluating $a_{\text{MH}}(\theta^*, \theta^{(i-1)})$ itself could be implemented as in Table 2. The routine `funcMetropolisHasting()` draws θ^* , evaluates $a_{\text{MH}}(\theta^*, \theta^{(i-1)})$ using (2.4) for $i = 1, \dots, N_1$ and (2.5) for $i = N_1 + 1, \dots, N$. For $i = 1, \dots, N_1$ we specify the random walk variance, Σ_q as `vtune`. As stated before, we experiment with several values for `vtune` in order to get reasonable acceptance rate probabilities. For instance, for the Gaussian SV model we set: $\Delta\mu^{(i)} = 0.2828\varepsilon_1^{(i)}$, $\Delta\phi^{(i)} = 0.01\varepsilon_2^{(i)}$ and $\Delta\sigma^{2(i)} = 0.01\varepsilon_3^{(i)}$ where $\varepsilon_k \sim N(0, 1)$ for $k = 1, \dots, 3$. Thereafter, we follow section 2 and perform independent kernel Metropolis-Hastings for the remaining N_2 draws.

The function `funcPosterior()` evaluates $p(\theta^* \mid Y_T)$ as θ^* is generated. In order to evaluate $p(\theta^* \mid Y_T)$ we need to run the particle filter using θ^* . The denominator is evaluated using $\theta^{(i-1)}$ and $p(\widehat{Y_T} \mid \theta^{(i-1)})$ which are available from the previous iteration. We thereafter complete the M-H step by drawing u from the standard Uniform distribution. If $a_{\text{MH}}(\theta^*, \theta^{(i-1)}) \geq u$ we set $\theta^{(i)} = \theta^*$ and $p(\widehat{Y_T} \mid \theta^{(i)}) = p(\widehat{Y_T} \mid \theta^*)$, else we retain $\theta^{(i-1)}$ and $p(\widehat{Y_T} \mid \theta^{(i-1)})$. Thereafter, we take another MCMC iteration and move along the chain.

2.2 Bayes factors and marginal likelihood computation

The main output from the particle filter is the loglikelihood contribution of each observation. The sum of the loglikelihood contributions delivers the estimated loglikelihood of the data, $p(\widehat{Y_T} | \theta)$, with the unobserved states integrated out. This quantity can then be used to compute the marginal likelihood (ML) for a model \mathcal{M} . The marginal likelihood is defined as

$$p(Y_T | \mathcal{M}) = \int_{\Theta} p(Y_T | \theta, \mathcal{M}) p(\theta | \mathcal{M}) d\theta$$

In the following steps the model index, \mathcal{M} is suppressed for conciseness. Gelfand and Dey (1994) propose a very compelling and general method to calculate ML. It is efficient and utilizes the same routines when calculating ML for different models. The Gelfand-Dey (G-D) estimate of the marginal likelihood is based on

$$\frac{1}{N} \sum_{i=1}^N g(\theta^{(i)}) / [p(Y_T | \theta^{(i)}) p(\theta^{(i)})] \rightarrow p(Y_T)^{-1} \quad \text{as } N \rightarrow \infty \quad (2.6)$$

whereas before, N is the number of PMCMC iterations. G-D show that if $g(\theta^{(i)})$ is thin-tailed relative to $p(Y_T | \theta^{(i)}) p(\theta^{(i)})$ then (2.6) is bounded and the estimator is consistent. Following Geweke (2005) a truncated Normal distribution, $N(\theta^*, \Sigma^*)$ is used for $g(\theta)$. θ^* and Σ^* are the posterior sample moments calculated as

$$\theta^* = \frac{1}{N} \sum_{i=1}^N \theta^{(i)} \quad \text{and} \quad \Sigma^* = \frac{1}{N} \sum_{i=1}^N (\theta^{(i)} - \theta^*) (\theta^{(i)} - \theta^*)'$$

whenever $\theta^{(i)}$ is in the domain of the truncated Normal. The domain, Θ is defined as

$$\Theta = \left\{ \theta : (\theta^{(i)} - \theta^*)' (\Sigma^*)^{-1} (\theta^{(i)} - \theta^*) \leq \chi_a^2(z) \right\}$$

where z is the dimension of the parameter vector and $\chi_a^2(z)$ is the a th percentile of the Chi-squared distribution with z degrees of freedom. In practice, 0.75, 0.95 and 0.99 are popular selections for a . Once the marginal likelihood for different specifications has been calculated, we can compare them using Bayes factors, BF. The relative evidence for \mathcal{M}_A versus \mathcal{M}_B is $BF_{\mathcal{M}_{AB}} = p(Y_T | \mathcal{M}_A) / p(Y_T | \mathcal{M}_B)$. This odds ratio is the factor by which the data considers \mathcal{M}_A more probable than \mathcal{M}_B . Kass and Raftery (1995) recommend considering twice the logarithm of the Bayes factor for model comparison and suggest a rule-of-thumb of support for \mathcal{M}_A based on $2 \log BF_{\mathcal{M}_{AB}}$: 0 to 2 not worth more than a bare mention, 2 to 6 positive, 6 to 10 strong, and greater than 10 as very strong.

3 Applications

3.1 Stochastic volatility models

In this section we estimate the standard stochastic volatility (SV) model along with different extensions. The first of these extensions we label the SVt model where $\varepsilon_t \sim St(\nu)$ and St stands for the Student-t distribution with $\nu > 2$ degrees of freedom. In the second extension we incorporate a leverage effect by letting ρ denote the correlation between ε_t and η_t . We shall refer to this model as SVL. Notice that in both cases we need only to make small adjustments in the codes. With regards to the SVt model, for $p(y_t | \alpha_t, Y_{t-1}, \theta)$ we use

$$p(y_t | \alpha_t, Y_{t-1}, \theta) = \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2}) \sqrt{(v-2)} \pi \sigma_t} \frac{1}{\sigma_t} \left(1 + \frac{y_t^2}{(v-2) \sigma_t^2} \right)^{-(v+1)/2}$$

We then follow the sampling steps as before. On the other hand, if we were to use pure Gibbs sampling to estimate the SVt model then we would be forced to convert the model into a conditionally Gaussian state space model by letting $\varepsilon_t = \lambda_t^{-1/2} e_t$ where $e_t \sim N(0, 1)$ and $\lambda_t \sim G(\nu/2, \nu/2)$. We would then follow the steps in Chib et al. (2002) and sample from the augmented posterior, $p(\theta, \nu, \alpha_1, \dots, \alpha_T, z_1, \dots, z_T, \lambda_1, \dots, \lambda_T | Y_T)$ whereas before, z_t serves as the mixture component indicator.

For the SVL model we need only to rewrite (2.2) as

$$\alpha_{t+1} = \mu + \phi(\alpha_t - \mu) + \sigma \rho y_t \exp(-\alpha_t/2) + \sigma \sqrt{(1-\rho^2)} \xi_t$$

Here, we use that $\eta_t = \rho \varepsilon_t + \sqrt{(1-\rho^2)} \xi_t$ where $\xi_t \sim N(0, 1)$ and we need to sample an additional parameter, ρ . We choose the same priors as in section 2 for μ , ϕ and σ^2 . With regards to the additional parameters we let $\nu \sim Exp(0.2)$ where Exp denotes the Exponential distribution and we set $\rho \sim N(0, 1)$ truncated such that $-1 < \rho < 1$. The prior on ρ assumes that ρ lies between -1 and 1 . Furthermore, we ensure that $|\rho^{(i)}| < 1$ by resampling $\rho^{(i)}$ until $|\rho^{(i)}| < 1$.

We can also expand the plain stochastic volatility model by allowing the errors in the measurement equation to follow a moving average (MA) process of order m , see for example Chan (2013). This means that the errors in the measurement equation are no longer serially independent as for the plain SV model. Here, we choose a more simple specification and set $m = 1$. Hence, our model is given as

$$\begin{aligned} y_t &= e^{\alpha_t/2} \varepsilon_t + \psi e^{\alpha_{t-1}/2} \varepsilon_{t-1}, & \varepsilon_t &\sim N(0, 1) \\ \alpha_{t+1} &= \mu + \phi(\alpha_t - \mu) + \sigma \eta_t, & \eta_t &\sim N(0, 1) \end{aligned}$$

As before, we impose that $|\phi| < 1$, $\alpha_1 \sim N(\mu, \sigma^2 / (1 - \phi^2))$ and we also ensure that the root of the characteristic polynomial associated with the MA coefficient, ψ is outside the unit circle. Notice that under the standard stochastic volatility model, the unconditional variance of y_t is simply e^{α_t} . However, under the moving average variant, the unconditional variance of y_t is given by $e^{\alpha_t} + \psi^2 e^{\alpha_{t-1}}$. Hence, in this case the unconditional variance is time-varying through two channels: the moving average term of e^{α_t} and α_t which evolves according to (2.2). Estimating this model is straightforward as again we only need to make small adjustments in the codes. With regards to ψ , we let $p(\psi) \sim N(0, 1)$ truncated such that $-1 < \psi < 1$.

The flexibility of PMCMC can be used to model other attractive specifications of the stochastic volatility model. For instance, consider the popular stochastic volatility in mean (SVM) model of Koopman and Hol Uspensky (2002) where $e^{\alpha_t/2}$ appears in both the conditional mean and the conditional volatility. We follow the same notation as before and define the SVM model as

$$y_t = \beta \exp(\alpha_t/2) + \exp(\alpha_t/2) \varepsilon_t, \quad \varepsilon_t \sim N(0, 1)$$

where α_t follows (2.2). Estimation of this extension is nontrivial using pure Gibbs sampling. This is because drawing $\alpha_1, \dots, \alpha_T \sim p(\alpha_1, \dots, \alpha_T | \theta, \beta, Y_T)$ is computationally more demanding since the model cannot be written in linear state space form. However, within the PMCMC context estimating the SVM model is straightforward. In fact, we note that $p(y_t | \alpha_t, Y_{t-1}, \theta) \sim N(\beta \exp(\alpha_t/2), \exp(\alpha_t))$. Incorporating this specification is very easy in the particle filter as we only need to modify step 2 of the algorithm and thus use $\tau_t^{(j)} = N(\beta e^{\alpha_t^{(j)}/2}, e^{\alpha_t^{(j)}})$, $j = 1, \dots, M$. Furthermore, in the M-H step we sample an additional parameter, β where $p(\beta) \sim N(0, 1)$.

It was suggested to provide a simulation example of SV-MA(1) and SVM. We present simulation results in appendix A where we also experiment with different values of M to see its impact on the results, especially for low values of M , for example $M = 1$ or $M = 10$. Finally, we estimate a two factor SV model, TFSV. We define TFSV as

$$\begin{aligned} y_t &= \exp\left(\frac{\mu}{2} + \frac{\alpha_t + \alpha_{2t}}{2}\right) \varepsilon_t, \quad \varepsilon_t \sim N(0, 1) \\ \alpha_t &= \phi \alpha_{t-1} + \sigma \eta_t, \quad \eta_t \sim N(0, 1) \\ \alpha_{2t} &= \phi_2 \alpha_{2t-1} + \sigma_2 \eta_{2t}, \quad \eta_{2t} \sim N(0, 1) \\ |\phi| < 1 &, \quad \alpha_1 \sim N(0, \sigma^2 / (1 - \phi^2)) \\ |\phi_2| < 1 &, \quad \alpha_{21} \sim N(0, \sigma_2^2 / (1 - \phi_2^2)) \end{aligned}$$

Estimating the two factor SV model using PMCMC is straightforward. First, we collect all the parameters in $\theta = (\mu, \phi, \phi_2, \sigma^2, \sigma_2^2)'$. Then, we only need to modify the particle filter such that we draw two sets of particles (one for α_t and one for α_{2t}) instead of one. This part is also very

easy and does not cost anything in terms of computation. We specify the increments in the M-H part as $\Delta\mu^{(i)} = 0.3162\varepsilon_1^{(i)}$ where $\varepsilon_1^{(i)} \sim N(0, 1)$. For ϕ and ϕ_2 we find that M-H is very sensitive with regards to their values. Therefore, we try get as close as possible to the center of the posterior. Hence, we perform the following, see also Fouque et al. (2010): First, we put the restriction $\phi > \phi_2$. This restriction is needed for identification. We then draw ϕ and ϕ_2 from a truncated Normal density. For instance, $\phi^* \sim N(b, B^{-1})$ truncated such that $|\phi^*| < 1$ where $b = (1/B)\sum_{t=1}^T \alpha_t^{(i-1)} \alpha_{t-1}^{(i-1)}$ and $B = \sum_{t=1}^T \alpha_{t-1}^{2(i-1)}$. We draw μ , σ^2 and σ_2^2 as before and thereafter accept or reject θ^* . For the priors, we set $\mu \sim N(0, 1)$, $\phi \sim N(0, 1)$, $\phi_2 \sim N(0, 1)$, truncated such that $|\phi| < 1$, $|\phi_2| < 1$, $\sigma^2 \sim IG(4/2, 0.02/2)$ and $\sigma_2^2 \sim IG(4/2, 0.02/2)$.

The top panel in Figure 1 displays the OMX Copenhagen 20 (OMXC20) index for the period 1/2/2006-12/30/2010 followed by the returns and the filtered estimates of $\hat{\sigma}_t = \exp(\hat{\alpha}_t/2)$ for $t = 1, \dots, T$. From these figures strong differences in return and volatility are apparent². For instance, the top left panel shows a sharp decrease in the OMXC20 index towards the end of 2008. At the same time the third panel in the top row of Figure1 shows an increase in the conditional volatility of SVL for that time period.

We set $M = 1000$ and run our sampler for $N = 20000$ M-H iterations. After discarding the first 10000 iterations we collect the final sample and compute the posterior mean, $\bar{\theta}$, 95% credibility intervals (indicated inside the brackets), the inefficiency measures, R_B , the loglikelihood that results from the particle filter, the logarithm of the marginal likelihood, $\log(\text{ML})$ for $a = 0.75$, $a = 0.99$ and the M-H acceptance ratio. The inefficiency measures display the relative variance of the posterior sample draws when adapting for correlation between iterations, as compared to the variance without accounting for correlation. In these calculations, we follow Bos (2011) and choose a bandwidth, B of 100, see also Kim et al. (1998) for a further background on this measure. Results are summarized in Table 3.

We find that the Gaussian SVL model performs best in terms of the marginal likelihood criterion³. The $2\log BF$ of SVL versus SV in Table 3 is 7.8 and this indicates strong evidence in favor of the SVL model. Compared to SVt the $2\log BF$ in favor of SVL is 12.8 which is also very strong evidence. The distribution of the parameters, $\theta = (\mu, \phi, \sigma^2, \rho)'$ are also concentrated around their means. The values of ϕ close to one confirm strong daily volatility persistence, in accordance with typical estimates reported in the literature. Notice that the persistence increases slightly as the fat-tailed error distribution is introduced ($\phi = 0.9810$ and 0.9852 for the SV and SVt model,

¹We could also choose a $Beta(20, 1.5)$ prior for ϕ and ϕ_2 . However, we find that ϕ_2 is very sensitive with regards to the hyperparameter values of $Beta(., .)$. Therefore, we choose a more uninformative prior, i.e. $N(0, 1)$ truncated to ensure that $|\phi| < 1$.

²OMX Copenhagen 20 (OMXC20) is the Copenhagen Stock Exchange's leading share index. The index consists of the 20 most actively traded shares on the Copenhagen Stock Exchange.

³An advantage of using Bayes factors is that they automatically include Occam's razor effect in that they penalize highly parametrized models that do not deliver improved content.

respectively) and drops from $\phi = 0.9810$ for SV to $\phi = 0.9777$ for SVL. In the SVt model, the distribution of the degrees of freedom parameter is centered around 14.20 with a standard deviation of 2.28. For the SVL model the posterior mean of ρ is -0.22 and negative as expected. We also find that $\psi = 0.0045$ for the SV-MA(1) model. Compared to SV-MA(1) the $2\log BF$ in favor of SVL is 15.8. On the other hand, we estimate β at 0.5636 with a standard deviation of 0.1688. However, compared to SV or SVL, SVM does not offer any improvements in terms of ML. Finally, TFSV performs worst in terms of ML. The estimates of μ , ϕ and σ^2 are relatively close to those for the plain SV model. On the other hand, the estimates of ϕ_2 and σ_2^2 show that the second factor, α_{2t} is very close to being a white noise process.

In Figure 1 we report the Markov chain output for $\mu | Y_T$, $\phi | Y_T$, $\sigma^2 | Y_T$, $\rho | Y_T$ along with histograms and the evolution of ACFs for these parameters. The chain mixes well with relatively fast decaying autocorrelation functions.

3.2 Unobserved components model of inflation

In this section we reconsider the unobserved components model of Stock and Watson (2007) and provide a computational framework using PMCMC. This model provides a simple but yet sufficient framework for discussing the main stylized facts concerning inflation. Specifically, the model postulates the decomposition of observed inflation into two components: the regular component which captures the trend in inflation and the irregular component which captures the deviations of inflation from its trend value. We start from a specification where components are driven by disturbances whose variance are constant over time. Thereafter, we consider specifications in which the components are driven by disturbances whose variance evolves over time according to a stationary stochastic volatility process.

We focus on quarterly inflation rates constructed from the seasonally adjusted consumer price index, made available by Fred (Federal Reserve Economic Data). We denote the quarterly series by CPI_t . The annualized quarterly inflation rate, denoted y_t , $t = 1, \dots, T$ is computed as $y_t = 400 \log(CPI_t/CPI_{t-1})$. For the analysis we use data from 1952:q1-2013:q1. In the following the most general specification of the unobserved component (UC) model is defined as

$$y_t = \alpha_t + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma_\varepsilon^2) \quad (3.1)$$

$$\alpha_{t+1} = \alpha_t + \eta_t, \quad \eta_t \sim N(0, \sigma_\eta^2) \quad (3.2)$$

This model contains two parameters, $\theta = (\sigma_\varepsilon^2, \sigma_\eta^2)'$ and a vector of the unobserved states, $(\alpha_1, \dots, \alpha_T)'$. We let $p(\sigma_\varepsilon^2) \sim IG(\frac{4}{2}, \frac{0.02}{2})$ and $p(\sigma_\eta^2) \sim IG(\frac{4}{2}, \frac{0.02}{2})$. PMCMC implementation of this model follows the same procedure as in section 3.1. In this model it is important to make sure that $\theta \geq 0$. Implementation of this restriction in Ox is nothing more than using the `while()` condition. This

routine is presented in Table 4.

We also provide extensions of (3.1) and (3.2) by incorporating stochastic volatility effects in σ_ε^2 or both in σ_ε^2 and σ_η^2 . First, let $h_{1t} = \log \sigma_{\varepsilon t}^2$ where

$$h_{1t+1} = \mu_1 + \phi_1 (h_{1t} - \mu_1) + \sigma_1 \eta_{1t}, \quad \eta_{1t} \sim N(0, 1)$$

Hence, in this case we add a second unobserved state which describes the evolution of the log-volatility to the irregular component of inflation. We shall refer to this model as UC-SVm. Finally, we add a third unobserved state which describes the log-volatility of α_t . Henceforth, we refer to this model as UC-SV⁴.

UC-SVm and UC-SV both have a special structure. For instance, for the UC-SV model, conditional on h_{kt} , $k = 1, 2$ the remaining model is a linear Gaussian state space model where $(\alpha_1, \dots, \alpha_T)'$ can be integrated out analytically using the Kalman filter. This is known as Rao-Blackwellization in the literature because it is an implication of the Rao-Blackwell Theorem, see Robert and Casella (2004). When this is possible, the state vectors can be separated. Particles are only simulated for $h_{kt}^{(j)}$ $k = 1, 2$ and conditional on these we can integrate α_t , $t = 1, \dots, T$ out analytically using the prediction and updating steps of the Kalman filter.

In the PMCMC procedure we modify the particle filter using the approach of Creal (2012). In the following we provide the steps of the modified particle filter for the UC-SV model.

1. For $t = 1$, draw $\alpha_1^{(1)}, \dots, \alpha_1^{(M)}, P_1^{(1)}, \dots, P_1^{(M)}$ where P_t is the covariance of α_t and it is obtained from the Kalman filter, see Kim and Nelson (1999). Draw $h_{k1}^{(1)}, \dots, h_{k1}^{(M)}$ for $k = 1, 2$ and set $\tau_1^{(1)}, \dots, \tau_1^{(M)} = 1/M$.
2. for $t = 2, \dots, T$ use the prediction step of the Kalman filter and obtain prediction errors and variances, $\left\{ v_t^{(j)}, f_t^{(j)} \right\}_{j=1}^M$.
3. Compute $\tau_t^{(j)} \sim N\left(v_t^{(j)}, f_t^{(j)}\right)$ and normalize $w_t^{(j)} = \tau_t^{(j)} / \sum_{k=1}^M \tau_t^{(k)}$.
4. Resample M particles with probabilities $w_t^{(1)}, \dots, w_t^{(M)}$ using the ‘‘Resampling step’’ and set $w_t^{(j)} = 1/M$.
5. Draw $h_{kt}^{(1)}, \dots, h_{kt}^{(M)}$, $k = 1, 2$ and run the updating step of the Kalman filter on each of these particles to obtain $\left\{ \alpha_t^{(j)}, P_t^{(j)} \right\}_{j=1}^M$.

Finally, we compare the performance of UC, UC-SVm and UC-SV using the marginal likelihoods. The implementation of G-D for the UC model is provided in Table 5. Notice that calculating the

⁴The specification of the stochastic volatility processes differs only slightly from Stock and Watson (2007) who assume a random walk process for the log-variances, h_{kt} , $k = 1, 2$. Here, we follow Grassi and Proietti (2010).

marginal likelihood of UC-SVM or UC-SV is very straightforward. Results, reported in Table 6, point out that with regards to ML the UC-SV model performs best.

The filtered estimate of $(\alpha_1, \dots, \alpha_T)'$ and the filtered estimates of the volatilities are all available from the particle filter. They are pictured in Figure 2 together with y_t . These estimates confirm largely the results of Stock and Watson (2007) and Grassi and Proietti (2010). The volatility of the irregular component, $\exp(h_{1t}/2)$ increases during the high periods of inflation in the 1970s, while the volatility of the regular component, $\exp(h_{2t}/2)$ is relatively more stable. Specifically, $\exp(h_{2t}/2)$ has been decreasing substantially after 1982. The decrease in $\exp(h_{kt}/2)$, $k = 1, 2$ since the early 1980s and throughout the 1990s has been documented in a range of studies and it is often labeled “The Great Moderation”. Finally, $\exp(h_{1t}/2)$ shows that the increase in volatility of the inflation rate during the last recession is mainly concentrated in the irregular component.

3.3 Long memory with stochastic volatility

In this section we model changing time series characteristics of monthly US core inflation rate by considering a heteroskedastic ARFIMA model similar to Bos et al. (2012). The heteroskedasticity is specified by a Gaussian SV process, see section 2.

The ARFIMA(0, d ,0) model for a time series, y_t with time-varying volatility, $\sigma_{\varepsilon t}$, $t = 1, \dots, T$ is given by

$$(1-L)^d(y_t - \tau) = \sigma_{\varepsilon t}\varepsilon_t, \quad \varepsilon_t \sim N(0, 1) \quad (3.3)$$

The fractional difference operator $(1-L)^d$ with $d \in \mathbb{R}$ is given by

$$(1-L)^d = \sum_{j=0}^{\infty} \binom{d}{j} (-L)^j$$

Here, we assume that $0 < d < 0.5$ and specify $\sigma_{\varepsilon t} = \exp(\alpha_t/2)$ where α_t evolves according to (2.2). We let $\theta = (\tau, d, \mu, \phi, \sigma^2)'$ and proceed in a very similar fashion as in section 3.1. For the priors of τ and d we specify $p(\tau) \sim N(0, 1)$ and $p(d) \sim N(0, 1)$ truncated such that $0 < d < 0.5$.

We simulate a data set of $T = 1000$ observations. We set $\tau = 0.7$, $d = 0.35$, $\mu = 1.2$, $\phi = 0.97$ and $\sigma^2 = 0.05$. We first generate the volatility sequence according to (2.2) with $\alpha_1 \sim N(\mu, \sigma^2/(1-\phi^2))$. Thereafter, we use $\exp(\alpha_t/2)\varepsilon_t$ and generate y_t through ARFIMA dynamics using d and τ . We estimate (3.3) using PMCMC and compare the estimates with the true parameters. For comparison, we also estimate a plain ARFIMA model using Gibbs sampling. Results are summarized in Table 7. Overall, PMCMC works very well as parameter estimates are close to their respective true values.

We then apply our model to a monthly time series of inflation, using the core consumer price index (CPILFESL) downloaded from FRED' s database. This series excludes the direct effects of price changes for food and energy and it is seasonally adjusted. We denote the price index by P_t . From this price index series, we construct the monthly US core inflation as $y_t = 100\log(P_t/P_{t-1})$. It can be interpreted as the percentage price change per month. Our series starts in 1960:1 and runs until 2013:4, for a total of 639 months.

We estimate (3.3) along with an ARFIMA($1,d,0$)-SV model. We also estimate an ARFIMA($1,d,1$)-SV model (not reported) but do not find the MA coefficient to be significantly different from zero. For further comparison we also estimate a homoscedastic ARFIMA($0,d,0$) model using Gibbs sampling. Results are summarized in Table 8. For ARFIMA($0,d,0$)-SV the order of integration, d , is estimated at 0.35 and significantly different from 0. This implies that US core inflation exhibits long memory behavior. The average inflation rate, τ is estimated at 0.13% per month. When we estimate an additional parameter, namely, ρ in the ARFIMA($1,d,0$)-SV model we find that d increases from 0.35 to 0.46. At the same time the AR coefficient, ρ is estimated at -0.28 and is significantly different from zero. The stochastic volatility component itself is nearly nonstationary as the autoregressive coefficient of volatility, ϕ , is close to one and the conditional volatility of volatility, σ is well-identified and estimated at 0.14. The average volatility, $\exp(\mu/2)$ is at 0.18% per month for both models

The marginal likelihood criterion shows that there is strong evidence in favor of ARFIMA-SV specifications. The $2\log BF$ in favor of ARFIMA($0,d,0$)-SV compared to ARFIMA($0,d,0$) is 212.12. We note further improvements in terms of ML for the ARFIMA($1,d,0$)-SV model. We plot the inflation rate along with the filtered estimates of $\exp(\alpha_t/2)$ for ARFIMA($1,d,0$)-SV in Figure 3. The volatility decrease in the early 1980s is noticeable and persistent. As stated in section 3.2, this period is labeled as the Great Moderation. We report the Markov chain output for $d | Y_T$ and $\rho | Y_T$ along with the evolution of ACFs for these parameters in the bottom rows of Figure 3. Clearly, the Markov chain mixes well with relatively fast decaying autocorrelation functions.

We also perform a recursive out-of-sample forecasting exercise to evaluate the performance of ARFIMA-SV models. For each model in Table 9 we produce h -month-ahead point forecasts with $h = 1$, $h = 4$ and $h = 8$ using a rolling window with a width of 200 months. We choose the out-of-sample period from 1976:9 to the end of the sample for a total of 439 observations. Specifically, given Y_t , $t \geq 200$ we implement our sampling scheme, obtain posteriors draws of θ and compute $E[y_{t+h} | Y_t]$ using at each step $h = 2, \dots$ previously obtained forecasts until $h-1$. As a new observation enters the information set, the posterior is updated through a new round of sampling and the forecasting procedure is implemented.

Table 9 reports mean absolute error (MAE) and root mean squared error (RMSE) for the predictive mean. The ARFIMA-SV models are estimated using PMCMC while AR(1), AR(4),

ARMA(1,1), ARFIMA(0,d,0) and ARFIMA(1,d,0) are estimated using Gibbs sampling.

Overall, we find that ARFIMA(1,d,0)-SV performs very well against the other models. It is the top performer for $h = 4$, $h = 8$ in terms of MAE and RMSE. For instance, the RMSE of the ARFIMA(1,d,0)-SV model is 10% lower than the RMSE of the AR(1) model for $h = 1$, 23% for $h = 4$ and 27% for $h = 8$. ARFIMA(1,d,0)-SV also offers improvements in terms of out-of-sample point forecasts compared to ARFIMA(1,d,0). However, these improvements are relatively modest.

In order to perform a joint evaluation of the forecasts and find out if ARFIMA-SV models generate significant improvements in terms of forecasting the methodology of Hansen et al. (2011), termed the Model Confidence Set (MCS) is applied. The appealing feature of the MCS approach is that it allows for a user-defined criterion of “best” and does not require a benchmark model for comparison. Overall, we see that ARFIMA-SV specifications perform very well compared to their homoscedastic counterparts and the AR models. Both models belong to the 5% MCS for $h = 1$ while ARFIMA(1,d,0)-SV is the only model that belongs to the 5% MCS for $h = 4$ and $h = 8$, i.e. ARFIMA(1,d,0)-SV performs significantly better than all the other models.

Finally, in order to get a better understanding of changes in the dynamics of US inflation we follow Bos et al. (2012) and perform a sensitivity analysis using rolling estimates (with a width of 200 months) of the parameters for the ARFIMA(1,d,0)-SV model. We show recursive estimates of τ , d , ρ and the unconditional volatility of volatility of inflation, $\sqrt{\sigma^2/1 - \phi^2}$ along with their respective one-standard-error credibility intervals in Figure 4. The values for 1960:1 correspond to the estimation period 1960:1-1976:8.

Panel (a) shows a clear structural break in τ . Recursive estimates of τ fluctuate around 0.35 and 0.4 until the early 1980s and thereafter drop to about 0.25 fluctuating around this value till the end of the sample. More importantly, post-break estimates of τ are relatively more stable confirming a certain degree of success in achieving long run inflation stability since the Great Moderation. Recursive estimates of d in panel (b) show that d gradually drops from 0.45 at the start of the sample to about 0.3 towards the end of the sample. The estimate of d is 0.31 for the last subsample which runs from 1996:9 to 2013:4. It is cautiously evident that long memory characteristics of inflation might not have remained significant after the Great Moderation as evidenced by a smaller d . On the other hand, ρ increases from -0.4 to -0.1 and then subsequently falls back to around -0.3 after the Great Moderation until around 1995. Finally, we find a significant one time drop in $\sqrt{\sigma^2/1 - \phi^2}$ in the beginning of the 1980s confirming the effects of the Great Moderation also in the unconditional volatility of volatility of inflation.

4 Conclusion

In this paper we present algorithms and implementations for analyzing different data sets using Ox in combination with PMCMC techniques. We provide some short Ox programs that show how to implement the main part of PMCMC. These programs can easily be extended to different models.

In section 3 we provide several empirical examples. We show how to estimate stochastic volatility models with different specifications. Thereafter, we focus on estimating the unobserved components model with time-varying volatility for US inflation data using PMCMC. Results using quarterly inflation data show that extending the unobserved components model towards a model with time-varying volatility both in the irregular and regular component of inflation provides improvements in terms of the marginal likelihood. Finally, we estimate two heteroskedastic ARFIMA models where heteroskedasticity is specified by a Gaussian SV process. We apply these models to the monthly US core inflation data from 1960:1 to 2013:4. We find that US core inflation exhibits long memory behavior. Comparing the ARFIMA-SV models with their homoscedastic counterparts using the marginal likelihood criterion shows that there is strong evidence in favor of the ARFIMA-SV models. Sensitivity analysis using rolling estimates of the parameters provides a clear distinction between parameter changes in the level, long run dynamics and changes in the unconditional volatility of volatility of inflation.

References

- [1] Andrieu, C., and A. Doucet. 2002. "Particle filtering for partially observed Gaussian state space models." *Journal of the Royal Statistical Society B* 64(4): 827-836.
- [2] Andrieu, C., A. Doucet, and R. Holenstein. 2010. "Particle Markov chain Monte Carlo methods (with discussion)." *Journal of the Royal Statistical Society B* 72(3): 1-33.
- [3] Baillie, R. T., C. F. Chung, M. A. Tieslau. 1996. "Analysing inflation by the fractionally integrated ARFIMA-GARCH model." *Journal of Applied Econometrics* 11(1): 23-40.
- [4] Beran, J. 1994. *Statistics for Long-Memory Processes*. Chapman and Hall.
- [5] Bos, C. S. 2011. "A Bayesian Analysis of Unobserved Component Models Using Ox." *Journal of Statistical Software* 41(13): 1-24.
- [6] Bos, C. S. 2013. *GnuDraw*. URL <http://www.tinbergen.nl/~cbos/gnudraw.html>.
- [7] Bos, C. S., S. J. Koopman, and M. Ooms. 2012. "Long memory with stochastic variance model: A recursive analysis for U.S. inflation." *Computational Statistics and Data Analysis*, forthcoming.
- [8] Chan, J. 2013. "Moving Average Stochastic Volatility Models with Application to Inflation Forecast." *Journal of Econometrics* 176(2): 162-172.
- [9] Chib, S., E. Greenberg. 1995. "Understanding the Metropolis-Hastings Algorithm." *The American Statistician* 49(4): 327-335.
- [10] Chib, S., F. Nadari, and N. Shephard. 2002. "Markov chain Monte Carlo methods for stochastic volatility models." *Journal of Econometrics* 108(2): 281-316.
- [11] Creal, D. 2012. "A survey of sequential Monte Carlo methods for economics and finance." *Econometric Reviews* 31(3): 245-296.
- [12] Del Moral, P. 2004. *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. Springer.
- [13] Doornik, J. A. 2009. *An Object-Oriented Matrix Language Ox 6*. Timberlake Consultants Press.
- [14] Doucet, A., S. J. Godsill, and C. Andrieu. 2000. "On sequential Monte Carlo sampling methods for Bayesian filtering." *Statistics and Computing* 10(3): 197-208.

- [15] Flury, T., and N. Shephard. 2011. "Bayesian inference based only on simulated likelihood: particle filter analysis of dynamic economic models." *Econometric Theory* 27(5): 933-956.
- [16] Gelfand, A., and D. Dey. 1994. "Bayesian Model Choice: Asymptotics and Exact Calculations." *Journal of the Royal Statistical Society B* 56(3): 501-514.
- [17] Geweke, J. 2005. *Contemporary Bayesian Econometrics and Statistics*. Wiley.
- [18] Grassi, S., and T. Proietti. 2010. "Has the Volatility of U.S. Inflation Changed and How?" *Journal of Time Series Econometrics* 2(1): Article 6.
- [19] Hansen, P. R., A. Lunde, and J. M. Nason. 2011. "The Model Confidence Set." *Econometrica* 79(2): 453-497.
- [20] Jacquier, E., N. G. Polson, and P. E. Rossi. 1994. "Bayesian Analysis of Stochastic Volatility Models." *Journal of Business & Economic Statistics* 12: 371-417.
- [21] Kass, R. E., and A. E. Raftery. 1995. "Bayes Factors." *Journal of the American Statistical Association* 90: 773-795.
- [22] Kim, C. J., and C. R. Nelson. 1999. *State Space Models with Regime Switching Classical and Gibbs Sampling Approaches with Applications*. MIT Press.
- [23] Kim, S., N. Shephard, and S. Chib. 1998. "Stochastic Volatility: Likelihood Inference and Comparison with ARCH Models." *Review of Economic Studies* 65(3): 361-393.
- [24] Koop, G. 2003. *Bayesian Econometrics*. John Wiley & Sons Ltd.
- [25] Koopman, S. J., and E. H. Uspensky. 2002. "The stochastic volatility in mean model: empirical evidence from international stock markets." *Journal of Applied Econometrics* 17(6): 667-689.
- [26] Liu, J. S., and R. Chen. 1998. "Sequential Monte Carlo Methods for Dynamic Systems." *Journal of the American Statistical Association* 93(443): 1032-1044.
- [27] Pitt, M. K., and N. Shephard. 1999. "Filtering via Simulation: Auxiliary Particle Filters." *Journal of the American Statistical Association* 94: 590-599.
- [28] So, M. K. P., C. W. S. Chen, and M. Chen. 2005. "A Bayesian Threshold Nonlinearity Test for Financial Time Series." *Journal of Forecasting* 24(1): 61-75.
- [29] Stock, J. H., and M. W. Watson. 2007. "Why Has U.S. Inflation Become Harder to Forecast?" *Journal of Money, Credit, and Banking* 39(1): 3-34.

A Appendix

In this appendix we present simulation results for SV-MA(1) and SVM. We simulate $T = 1000$ observations from these models and report the true DGP parameters along with PMCMC parameter estimates in Table 10. In each case we also estimate a plain SV model for comparison. Overall, we see that PMCMC works very well as parameter estimates are close to their respective true values. Not surprisingly, in each case the corresponding model outperforms the plain SV model in terms of ML.

Finally, we analyze the performance of PMCMC with respect to the number of particles, M . We do this by estimating SV-MA(1) using $M = 1$, $M = 10$, $M = 100$ and $M = 1000$. In all of these cases we choose $N = 20000$. From the columns of Figure 5 we see that using very low values of M appears to be insufficient. For instance, for $M = 1$ the chain gets stuck on a specific parameter value almost throughout the sample. For $M = 100$ we get better results but we still see that the chain gets stuck for a considerable time. However, we see drastic improvements in the performance of the algorithm for $M = 1000$. We also run the PMCMC algorithm for $M = 2000$ and get almost identical results.

Table 1: The particle filter in Ox

```

funcParticleFilter(const vhpf, const vESS, const vlogpdf, const vparam,
const vy, const iN)
{
    ...
    vw=ones(iN,1)*(1/iN);
    vlogpdf[0][0]=0; // No likelihood contribution
    [vh]=funcInParticles(vparam,iN); // Initial particles
    for(i=1; i<it; i++)
    {
        vA=funcResample(vw',iN); // Resample
        vh=funcDrawP(vh[vA'],vparam,vy[i][0],iN); // Draw from q()
        vE=vy[i]./(exp(0.5*vh));
        vtau=exp(-0.5*vh).*densn(vE); // Compute likelihood
        vlogpdf[0][i][0]=log(meanc(vtau)); // Take logs
        vw=vtau./sumc(vtau); // Normalize
        if (ismissing(vw)) // Reset weights if missing
            vw=ones(iN,1)/iN;
        vhpf[0][i][0]=sumc(vw.*vh); // Store mean
        vESS[0][i][0]=1/sumc(vw.^2); // Store ESS
        if(vESS[0][i][0]<0.5*iN)
        {
            // If vESS[0][i][0]<0.5*iN resample
            vA=funcResample(vw',iN);
            vh=funcDrawP(vh[vA'],vparam,vy[i][0],iN);
            vw=ones(iN,1)*(1/iN);
        }
    }
    return 1;
}

```

Table 2: PMCMC scheme in Ox

```

funcMetropolisHastings(const mprior, const vy, const vparamo, const
vparamq,
    const mcov, const dlogpdfo, const iN, const idumth)
{
    ...
    [vparamp]=funcDrawparam(vparamq,mcov,idumth);
    // Draw the candidate, vparamp
    [dnum,dden,dlogpdfp]=
funcPosterior(vparamp,vparamo,dlogpdfo,vy,mcov,mprior,iN);

    idum=0;
    vparamnew=vparamo; // vparamo is the old value
    dlogpdfnew=dlogpdfo;
    // Prepare these quantities
    // Set the new parameters equal to the old
    // If dalpha>du they will be replaced

    dalpha=min(1,exp(dnum-dden)); // Calculate aMH
    du=ranu(1,1); // Draw a random uniform number
    if (dalpha>du) // accept
        {
            vparamnew=vparamp;
            dlogpdfnew=dlogpdfp;
            idum=1; // Count that the draw was accepted
        }

    return {vparamnew,dlogpdfnew,idum};
}

```

Table 3: Estimation results, stochastic volatility models

Parameter	SV $\bar{\theta}$	SVt $\bar{\theta}$	SVL $\bar{\theta}$	SV-MA(1) $\bar{\theta}$	SVM $\bar{\theta}$	TFSV $\bar{\theta}$
μ	0.3526 (7.54) [0.1112,0.5895]	0.2562 (7.38) [-0.0482,0.5508]	0.4221 (6.89) [0.2046,0.6436]	0.3779 (7.78) [0.0831,0.6703]	0.3754 (7.43) [0.0685,0.6708]	0.3598 (6.23) [0.0821,0.6332]
ϕ	0.9810 (8.27) [0.9743,0.9887]	0.9852 (5.33) [0.9783,0.9914]	0.9777 (6.97) [0.9716,0.9838]	0.9817 (8.86) [0.9730,0.9893]	0.9805 (9.63) [0.9727,0.9882]	0.9776 (7.13) [0.9671,0.9881]
ϕ_2						0.0458 (10.66) [-0.0642,0.1576]
σ^2	0.0296 (9.13) [0.0199,0.0394]	0.0221 (6.11) [0.0149,0.0297]	0.0319 (7.36) [0.0231,0.0401]	0.0285 (7.97) [0.0183,0.0395]	0.0311 (9.16) [0.0203,0.0421]	0.0373 (6.59) [0.0236,0.0512]
σ_2^2						0.0981 (6.31) [0.0276,0.1728]
ρ			-0.2216 (7.04) [-0.3202,-0.1226]			
ψ				0.0045 (7.63) [-0.0296,0.0376]		
β					0.0563 (6.37) [0.0223,0.0892]	
ν		14.2019 (8.06) [9.9591,18.8712]				
log(L)	-2113.6	-2114.2	-2106.1	-2113.4	-2112.1	-2113.2
log(ML), $a = 0.75$	-2125.4	-2127.9	-2121.5	-2129.1	-2128.0	-2186.3
log(ML), $a = 0.99$	-2125.1	-2127.6	-2121.2	-2129.1	-2127.7	-2186.1
M-H ratio	0.35	0.36	0.35	0.33	0.33	0.34

This table reports estimation results for different stochastic volatility models. R_B is indicated inside the parentheses. log(L): loglikelihood, log(ML): log-marginal likelihood for the corresponding value of a . M-H ratio: Metropolis-Hastings acceptance ratio.

Table 4: Drawing $\theta^{(i)}$ in the UC model

```
funcDrawparam(const vparamo, const vtune)
{
    ...
    vparamp=zeros(sizer(vparamo),1);
    // Declare this vector for storage
    // We use the while command to make sure
    // that the variances are positive
    while (vparamp[0][0]<=0 || vparamp[1][0]<=0)
    {
        // vparamo is the parameters at the old state
        vparamp[0][0]=vparamo[0][0]+sqrt(vtune[0][0])*rann(1,1);
        vparamp[1][0]=vparamo[1][0]+sqrt(vtune[1][0])*rann(1,1);
    }
    return vparamp;
}
```

Table 5: Marginal likelihood computation using Gelfand-Dey method in Ox

```

funcGD(const mtheta, const mprior, const vlogpdf, const isim, const dalpha)
{
    ...
    // mtheta is isimx2 vector of parameters
    // Make the loop in order to evaluate the pdf
    mpdf=zeros(isim,sizec(mtheta));
    for (j=0; j<isim; j++)
        {
            // Compute pdf at each draw
            mpdf[j][0]=funcIGpdfx(mtheta[j][0],mprior[0][0],mprior[0][1]);
            mpdf[j][1]=funcIGpdfx(mtheta[j][1],mprior[1][0],mprior[1][1]);
        }

    // Calculate mean and variance of the draws
    vmean=meanc(mtheta);    mcov=variance(mtheta);
    vtrunc=zeros(isim,1);

    // Calculate g
    vf=zeros(isim,1);
    for (i=0; i<isim; i++)
        {
            // funcMVNpdfx is multivariate Normal
            density (already in log)
            vtrunc[i][0]=
            (mtheta[i][0]-vmean)*invert(mcov)*
            (mtheta[i][0]-vmean)';
            vf[i][0]=funcMVNpdfx(mtheta[i][0],vmean,mcov)-log(dalpha);
        }

    dcritical=quanchi(dalpha,sizec(mtheta));
    vtrunci=vtrunc>dcritical;
    // Set this to basically zero if we
    // extend the truncation limit
    // i.e discard these draws
    vf[vtrunci][0]=- .Inf;
    vlogGD=vf-sumr(mpdf)-vlogpdf;
    dconstant=max(vlogGD);
    // Calculate marginal likelihood
    dml=-dconstant-log(meanc(exp(vlogGD-dconstant)));
    return dml;
}

```

Table 6: Estimation results of unobserved component (UC) models

Parameter	UC		UC-SVm		UC-SV	
	$\bar{\theta}$	R_B	$\bar{\theta}$	R_B	$\bar{\theta}$	R_B
μ_1			0.0602	4.08	-0.0764	8.47
			[-0.4104, 0.5197]		[-0.3733, 0.2223]	
μ_2					-0.5886	4.95
					[-0.9465, -0.2391]	
ϕ_1			0.9456	4.54	0.9541	7.84
			[0.9067, 0.9792]		[0.9278, 0.9779]	
ϕ_2					0.9815	7.21
					[0.9676, 0.9938]	
σ_1^2	3.2280	5.09	0.1153	3.93	0.0844	8.97
	[2.9026, 3.5680]		[0.0408, 0.2215]		[0.0383, 0.1406]	
σ_2^2	2.0021	4.57	0.5090	3.59	0.0141	7.75
	[1.2772, 2.7866]		[0.3449, 0.6826]		[0.0051, 0.0255]	
log(L)	-592.28		-469.35		-465.07	
log(ML), $a = 0.75$	-638.30		-497.77		-484.86	
log(ML), $a = 0.99$	-638.03		-497.50		-484.58	
M-H ratio	0.42		0.48		0.40	

This table reports estimation results for different UC models. log(L): loglikelihood, log(ML): log-marginal likelihood for the corresponding value of a . M-H ratio: Metropolis-Hastings acceptance ratio.

Table 7: Simulation evidence, ARFIMA-SV

Parameter	true	ARFIMA(0,d,0)-SV		ARFIMA(0,d,0)	
		$\hat{\theta}$	R_B	$\hat{\theta}$	R_B
τ	0.70	0.5783 [0.0270,1.1633]	6.19	2.0887	7.45
d	0.35	0.3301 [0.2970,0.3623]	5.36	0.3406	6.37
μ	1.20	1.4317 [1.1736,1.6831]	5.37		
ϕ	0.97	0.9666 [0.9551,0.9781]	4.67		
σ^2	0.05	0.0523 [0.0361,0.0699]	5.26	6.0434	1.02
	log(L)		-2203.3		-2321.9
	log(ML), $a = 0.75$		-2221.4		-2353.4
	log(ML), $a = 0.99$		-2221.1		-2354.1
	M-H ratio		0.38		

This table reports estimation results for different ARFIMA(0,d,0) models using simulated data. log(L): loglikelihood, log(ML): log-marginal likelihood for the corresponding value of a . M-H ratio: Metropolis-Hastings acceptance ratio.

Table 8: Estimation results, ARFIMA-SV and ARFIMA

Parameter	ARFIMA(0,d,0)-SV		ARFIMA(1,d,0)-SV		ARFIMA(0,d,0)	
	$\bar{\theta}$	R_B	$\bar{\theta}$	R_B	$\bar{\theta}$	R_B
τ	0.1331	5.42	0.1086	5.80	0.2657	6.69
	[0.0763,0.1886]		[0.0324,0.1827]		[0.1261,0.4009]	
d	0.3509	6.22	0.4609	8.06	0.3968	4.44
	[0.3206,0.3829]		[0.4281,0.4926]		[0.3491,0.4453]	
ρ			-0.2835	6.75		
			[-0.3335,-0.2328]			
μ	-3.5845	5.38	-3.4799	6.46		
	[-4.0077,-3.1675]		[-3.9046,-3.0321]			
ϕ	0.9871	5.74	0.9870	7.04		
	[0.9828,0.9898]		[0.9831,0.9897]			
σ^2	0.0175	5.78	0.0191	6.61	0.0293	1.02
	[0.0108,0.0249]		[0.0124,0.0267]		[0.0263,0.0326]	
log(L)	337.77		353.37		218.70	
log(ML), $a = 0.75$	315.00		327.08		209.25	
log(ML), $a = 0.99$	315.28		327.36		209.22	
M-H ratio	0.39		0.39			

This table reports estimation results for different ARFIMA(p,d,q) models using US core inflation rate data. log(L): loglikelihood, log(ML): log-marginal likelihood for the corresponding value of a . M-H ratio: Metropolis-Hastings acceptance ratio.

Table 9: Out-of-sample forecast results, y_{t+h}

Model	MAE			RMSE		
	$h = 1$	$h = 4$	$h = 8$	$h = 1$	$h = 4$	$h = 8$
AR(1)	0.1173	0.1633	0.1786	0.1656	0.2187	0.2350
AR(4)	0.1075	0.1296	0.1423	0.1552 ^(*)	0.1851	0.1889
ARMA(1,1)	0.1026	0.1185	0.1291	0.1500 ^(*)	0.1751	0.1771
ARFIMA(0,d,0)	0.1010 ^(*)	0.1167	0.1257	0.1480 ^(*)	0.1717	0.1787
ARFIMA(1,d,0)	0.1028	0.1154	0.1255	0.1491 ^(*)	0.1697	0.1766
ARFIMA(0,d,0)-SV	0.1002 ^(*)	0.1139	0.1205	0.1474 ^(*)	0.1695	0.1753
ARFIMA(1,d,0)-SV	0.1011 ^(*)	0.1105 ^(*)	0.1164 ^(*)	0.1479 ^(*)	0.1665 ^(*)	0.1707 ^(*)

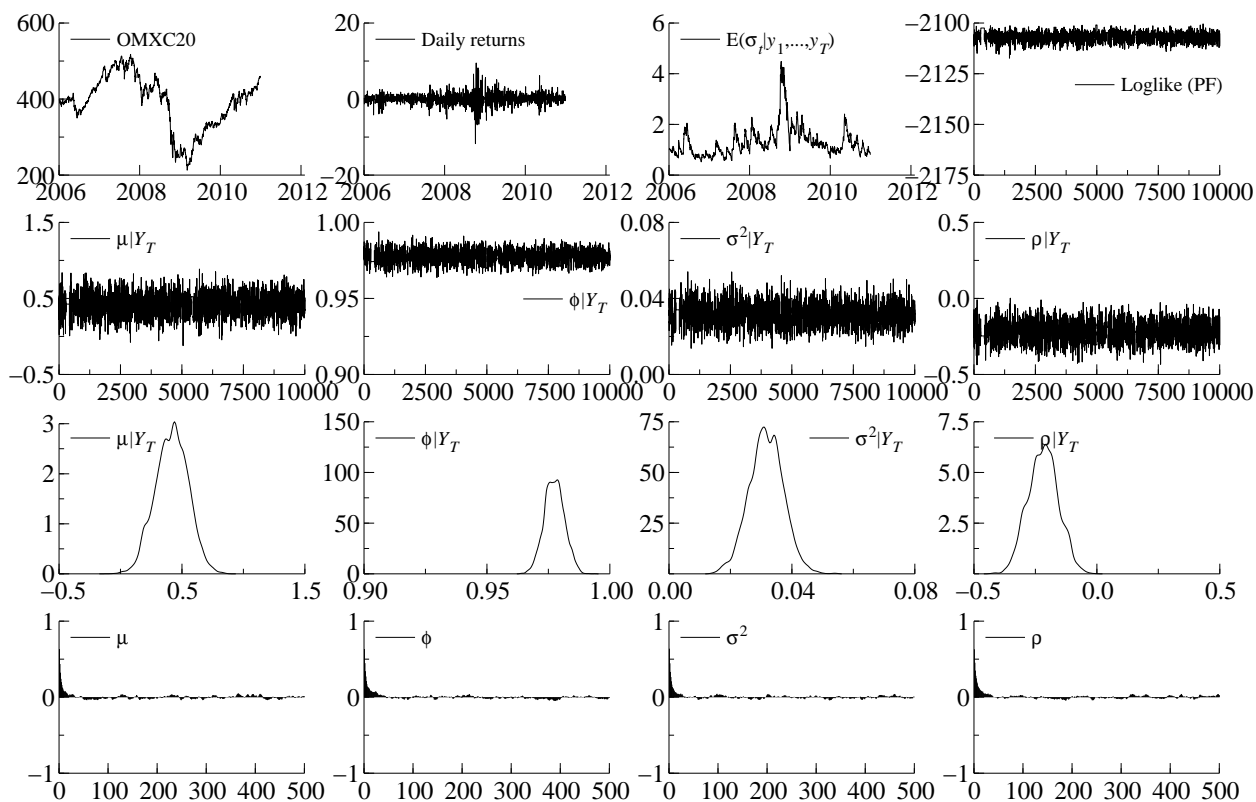
This table reports out-of-sample mean absolute error (MAE) and root mean squared error (RMSE) for the predictive mean. The out-of-sample period is from 1976:9 to the end of the sample. An asterisk, (*) signifies that the model belongs to the 5% MCS of Hansen et al. (2011).

Table 10: Simulation evidence

Parameter	DGP: stochastic volatility with MA(1) errors, SV-MA(1)			DGP: stochastic volatility in mean, SVM			
	SV $\bar{\theta}$	R_B	$\bar{\theta}$	SV-MA(1) $\bar{\theta}$	R_B	SV $\bar{\theta}$	SVM $\bar{\theta}$
μ	0.20	4.31	0.2756	0.20	4.40	0.7287	0.2883
ϕ	0.98	5.07	0.9781	0.98	4.55	0.9815	0.9790
σ^2	0.01	4.86	0.0104	0.01	5.22	0.0087	0.0108
ψ	0.40		0.0048,0.0172]	0.80	5.57	0.0043,0.0159]	0.0055,0.0174]
β			0.4591				
			[0.4221,0.4963]				[0.7533
							[0.7090,0.8001]
$\log(L)$							
$\log(ML), a = 0.75$			-1581.0			-1811.0	-1586.8
$\log(ML), a = 0.99$			-1592.4			-1817.5	-1598.5
M-H ratio			-1592.2			-1817.2	-1598.2
			0.38			0.33	0.37

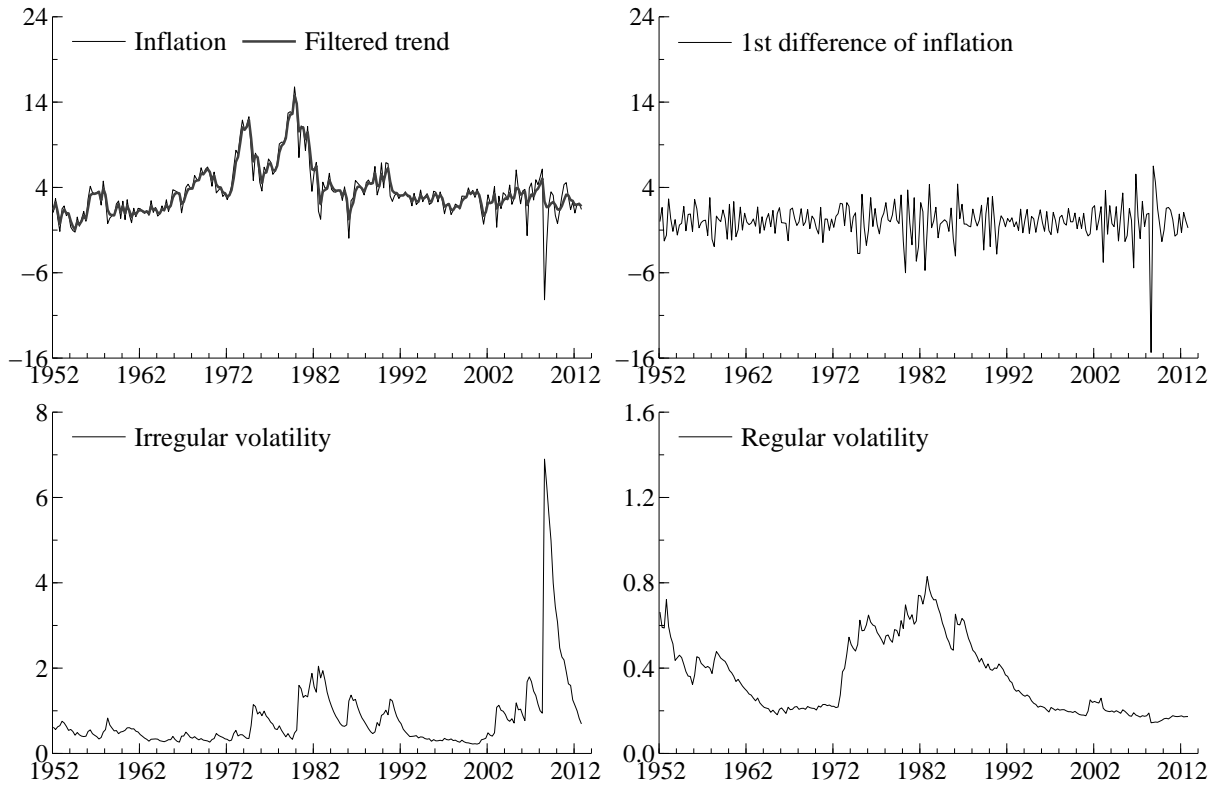
This table reports estimation results for SV-MA(1) and SVM using simulated data. $\log(L)$: loglikelihood, $\log(ML)$: log-marginal likelihood for the corresponding value of a . M-H ratio: Metropolis-Hastings acceptance ratio.

Figure 1: Estimation results, SVL model on OMXC20 daily returns



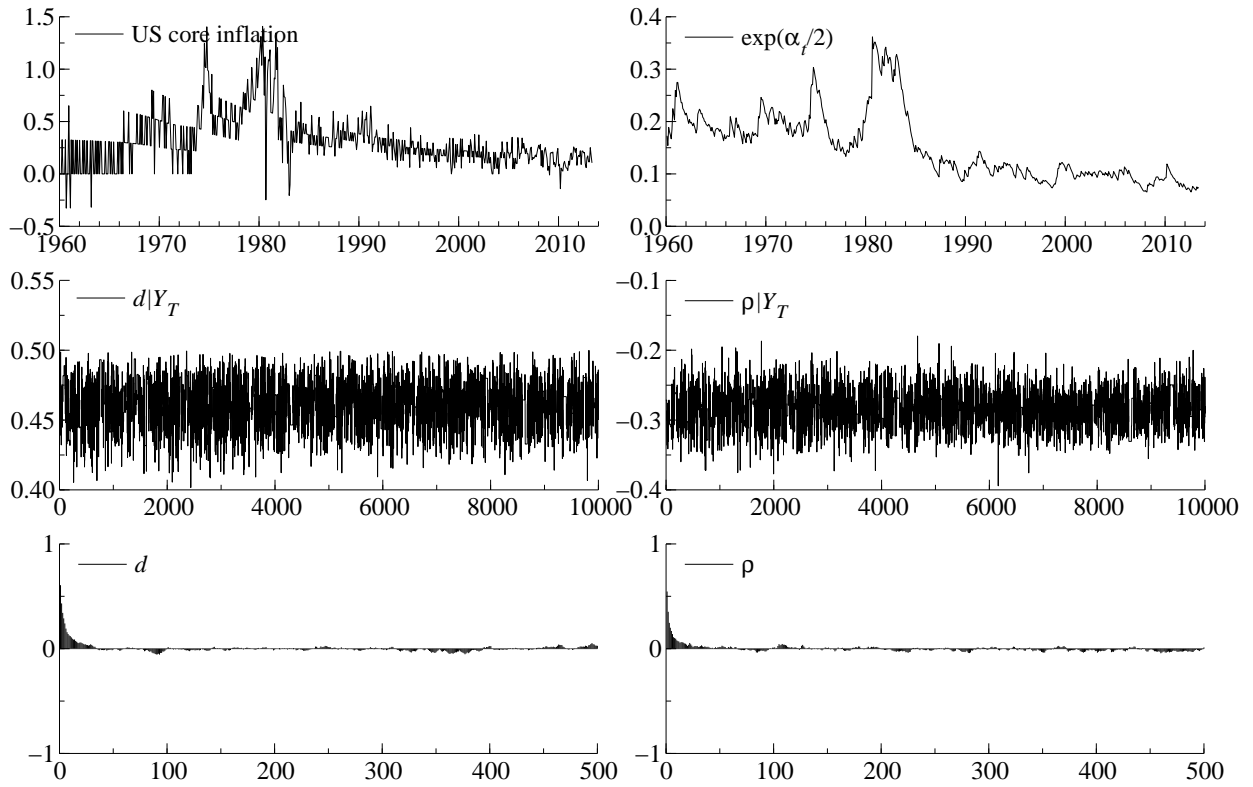
Markov chains, histograms and autocorrelation functions for the parameters of the Gaussian SVL model. Notice that the first 10000 iterations are considered as the burn-in period and therefore are discarded. Note: for graphical output, the professional version of Ox is needed. Alternatively, the updated **GnuDraw** package of Bos (2013) can be used.

Figure 2: Estimation results, UC-SV



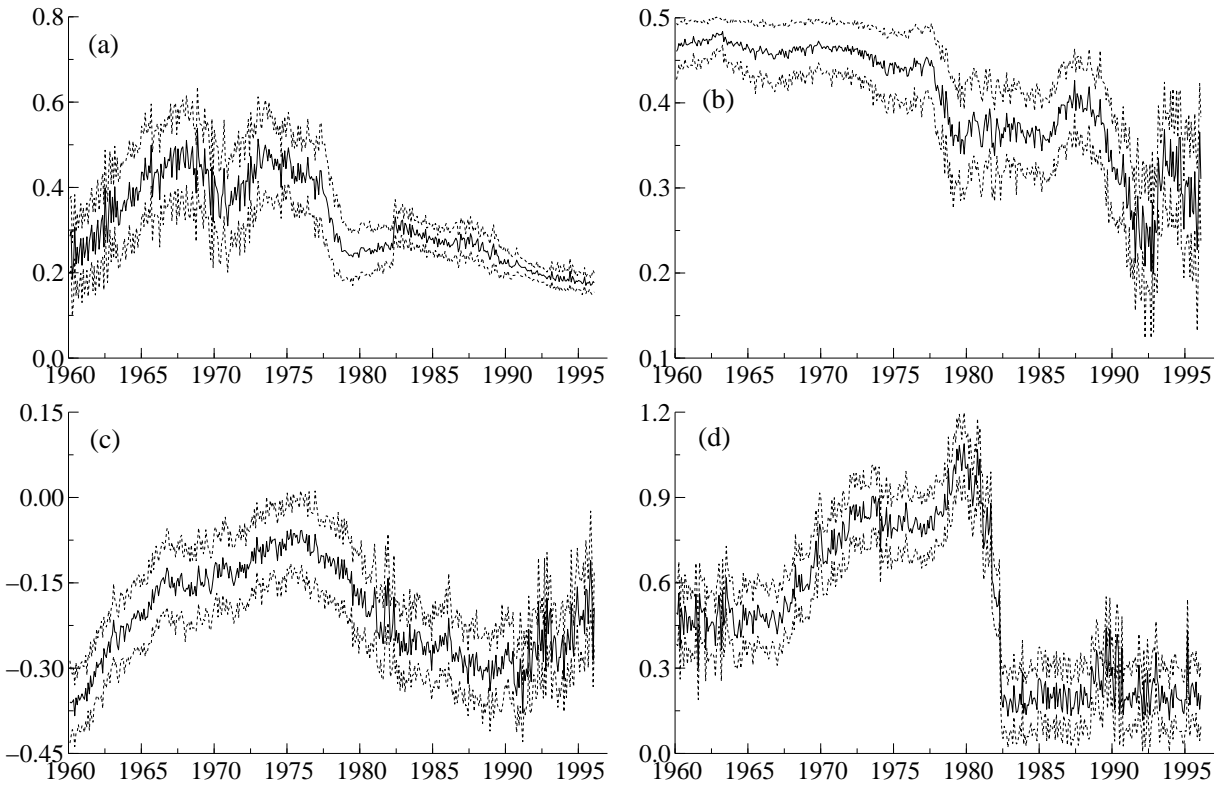
Top panels: inflation, filtered estimates of the trend inflation and the 1st difference of inflation.
Bottom panels: volatility of the irregular and regular component of inflation.

Figure 3: Estimation results, ARFIMA(1,d,0)-SV



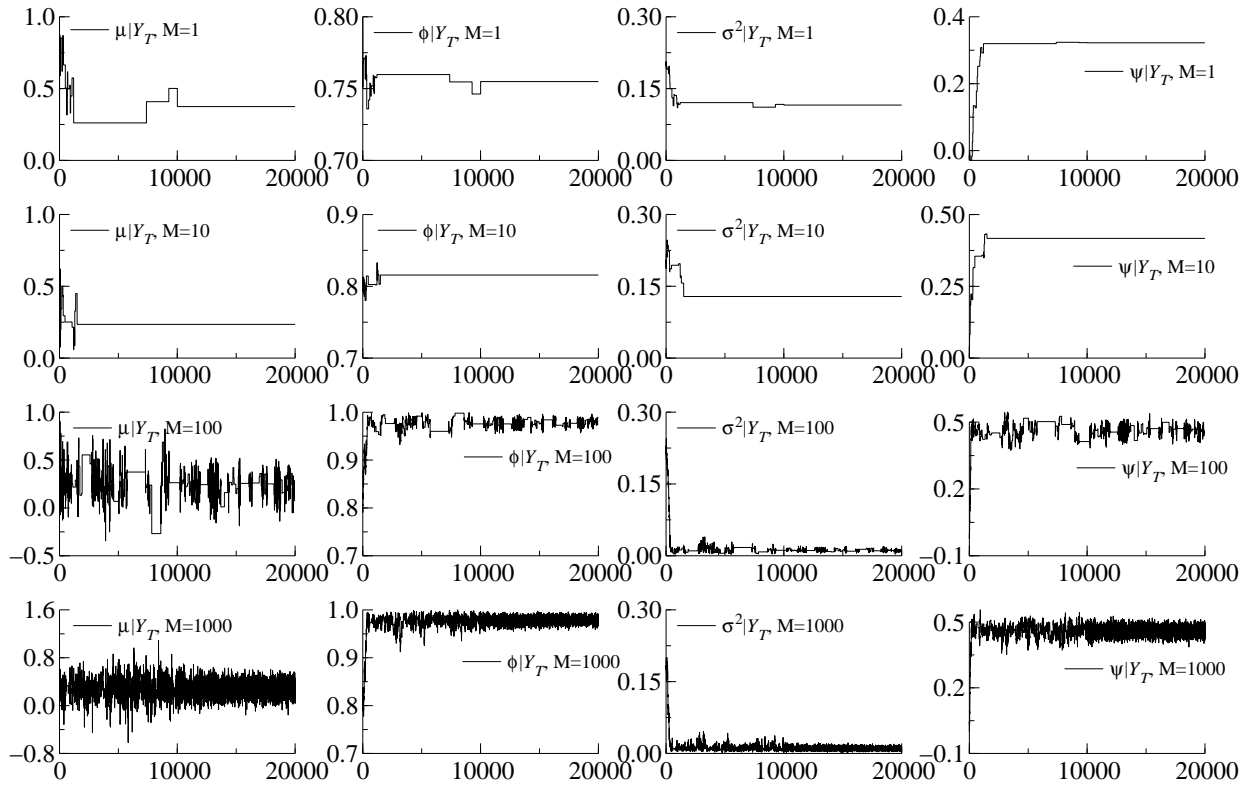
Top panels: inflation and filtered estimates of volatility of the inflation rate. Bottom panels: Markov chain and autocorrelation functions (after burn-in) of d and ρ .

Figure 4: Rolling window parameter estimates, ARFIMA(1,d,0)-SV



Panel (a): τ , (b): d , (c): ρ and (d): unconditional volatility of volatility of inflation. Window width: 200 months. First period: 1960:1-1976:8, last period 1996:9 to 2013:4. The solid lines represent parameter estimates and the dashed lines denote the one-standard-error credibility intervals.

Figure 5: Estimation results, SV-MA(1) model



Each column shows posterior draws of the parameter of interest for different number of particles, M .