Forecasting Time Series with Long Memory and Level Shifts: A Bayesian Approach

Silvestro Di Sanzo
Forecasting Time Series with Long Memory and Level Shifts: 
A Bayesian Approach

Silvestro Di Sanzo
Universidad de Alicante

Abstract
Recent studies have showed that it is troublesome, in practice, to distinguish between long memory and nonlinear processes. Therefore, it is of obvious interest to try to capture both features of long memory and non-linearity into a single time series model to be able to assess their relative importance. In this paper we put forward such a model, where we combine the features of long memory and Markov nonlinearity. A Markov Chain Monte Carlo algorithm is proposed to estimate the model and evaluate its forecasting performance using Bayesian predictive densities. The resulting forecasts are a significant improvement over those obtained by the linear long memory and Markov switching models.

Keywords
Markov-Switching models, Bootstrap, Gibbs Sampling

JEL Codes
C11, C15, C22

Address for correspondence:
Silvestro Di Sanzo
Department of Economics
University of Alicante
Campus de San Vicente
03080 Alicante - Spain
Phone: (+34) 965 90 33 49
Fax: (+34) 965 90 38 98
e-mail: silvio@merlin.fae.ua.es

This Working Paper is published under the auspices of the Department of Economics of the Ca’ Foscari University of Venice. Opinions expressed herein are those of the authors and not those of the Department. The Working Paper series is designed to divulge preliminary or incomplete work, circulated to favour discussion and comments. Citation of this paper should consider its provisional character.
1 Introduction

Long memory processes have received considerable attention among researchers from various disciplines, including Hydrology, Meteorology, Economics and Finance. For recent surveys of the literature on this topic, we refer the reader to Baillie (1996). A commonly used model for such processes is the autoregressive fractionally integrated moving-average (ARFIMA) model, introduced by Granger and Joyeaux (1980) and Hosking (1981).

However, only recently, there has been a considerable interest in the links between long memory and nonlinearity, stressing the memory properties that arise in models with parameter changes. For example, Granger and Terasvirta (1999) shows that nonlinear models generate data to which some linear long memory models could be fitted. At the same time, data from long memory models seem nonlinear enough to try and fit certain nonlinear models to these observations; see Andersson et al. (1999). Granger and Hyung (2004), Diebold and Inoue (2001) and Gourieroux and Jasiak (2001) provide both theoretical justification and Monte Carlo evidence that models with structural breaks exhibit spurious long memory properties and, on the other hand, that a spurious break-type phenomenon appears when trying to estimate the number of breaks of a long memory process with no breaks. Due to the slowly decaying correlation structure of a long memory process, test statistics commonly used to test for the stability of the model over time encounter some difficulties.

To summarize, given the potential confusion between long memory and nonlinearity that arises in many empirical situations, it is worthwhile to try to capture both features into a single time series model to be able to assess their relative
importance. Within the context of such a model, we investigate whether nested alternative models perform better on measures of fit and forecasting. With this in mind, the aim of this paper is to put forward such a model by means of the combination of the concepts of fractional integration and Markov nonlinearity. We employ this model, which is named as Markov switching long memory (MS-LM) model, to describe the dynamic properties of the logarithmic realized volatility of the S&P 500 stock index. To estimate the model, we use a Bayesian method, which is based on estimating consistently the probability and the size of a shift at each point, together with the other model parameters using Markov Chain Monte Carlo (MCMC) techniques. We not only consider in-sample fit, but also compare the forecasting performance of the MS-LM model with that of ARFIMA and Markov switching (MS) models using Bayesian predictive densities obtained by means of the MCMC algorithm.

This is the first study to propose a model for realized volatilities that simultaneously captures long memory and Markov nonlinearity. To the best of our knowledge, the only study that considers nonlinearities and long memory into a single model is Franses and Hyung (2004). It proposes a time series that allows for both long memory and occasional level shifts, but it employs the traditional approach that separates estimation from detection of the breaks. This approach is fast in computation, but it may produce spurious detection when multiple breaks are present. Moreover, an occasional-break model amount to a rather stylized nonlinear model, while our model allows for richer dynamics.

The organization of the paper is as follows. In the next two sections, we introduce the ARFIMA, MS and MS-LM models. Section 4 explains the MCMC algorithms used to estimate the models and performs two simple Monte Carlo
experiments to check our methods. In Section 5, we briefly describe how to do forecasting using the MCMC algorithm. The empirical and forecasting analyses are provided in Section 6. Some conclusion are given in Section 7. Tables and Figures are relegated to Appendix.

2 The ARFIMA model

An ARFIMA (p,d,q) process \( \{ y_t \} \) is defined by

\[
\Phi(L)(1 - L)^d(y_t - \mu) = \Theta(L)\varepsilon_t,
\]

where \( \varepsilon_t \) is a Gaussian white noise process with variance \( \sigma^2 \), \( \mu \) is the mean of the process, \( \Phi(L) = 1 - \phi_1 L - \phi_2 L^2 - \ldots - \phi_p L^p \) and \( \Theta(L) = 1 - \theta_1 L - \theta_2 L^2 - \ldots - \theta_q L^q \) are polynomials in the backward shift operator \( L \) of degrees \( p \) and \( q \), respectively.

The order of integration \( (d) \) is allowed to take non-integer values in the difference operator, \( (1 - L)^d = \sum_{j=0}^{\infty} \pi_j L^j \) with \( \pi_j = \Gamma(j - d)/\Gamma(j + 1)\Gamma(-d) \), and \( \Gamma \) denoting the Gamma function.

The process is stationary and invertible if all roots of \( \Phi(L) \) and \( \Theta(L) \) lie outside the unit circle and \(-0.5 < d < 0.5 \). Long memory is usually defined in time domain, characterized by a hyperbolically decaying autocorrelation function with \( \rho_y(k) = ak^{2d-1} \) as \( k \to \infty \), or alternatively, in the frequencies domain, where in the lowest frequencies the spectrum is \( f_y(\omega) \sim c\omega^{2d} \) when \( \omega \to 0 \). The process exhibits long memory for \( d \in (0,1) \), being covariance-stationary if \( d < 0.5 \), and still mean reverting if \( d < 1 \). This contrasts with stationary, \( I(0) \), ARMA, or short memory processes, where the dependence tends to be dissipated geometrically.
with time, meaning that shocks have a temporary effect in the process. In its turn, I(1) processes are not mean-reverting, wherefore shocks have permanent effects. Therefore, ARFIMA models are an intermediate and flexible form of analyzing time series. In this paper, we focus on a particular case of an ARFIMA model: the fractional white noise process, which is described by

\[(1 - L)^d(y_t - \mu) = \varepsilon_t. \tag{1}\]

Only the case with \(d \in (0, 1/2)\) is considered here. In this case, the process is stationary, invertible and possesses long-range dependence.

The infinite-order autoregressive representation of a fractional white noise is given by

\[y_t = \mu + \sum_{j=1}^{\infty} \pi_j (y_{t-j} - \mu) + \varepsilon_t. \tag{2}\]

In terms of practical implementation, this form needs truncation after \(l\) lags, but there is no obvious way of doing it. In the empirical analysis, we propose to use the Schwarz Information Criteria (BIC) to select this value \(l\). To estimate the parameters of the model, we employ an efficient Gibbs sampling algorithm.

\section{The MS-LM model}

In this section, we introduce a new time series model that combines long memory and nonlinearity. To capture nonlinear features in a time series \(y_t\), one can choose from a wide variety of nonlinear models, see Franses and Van Dijk (2000) for a
A model which enjoys a fair amount of popularity, mainly due to its empirical tractability, is the MS model, that is,

\[ y_t = \mu_{s_t} + \varepsilon_t, \]  

\[ (3) \]

with level shifts described by

\[ \mu_{s_t} = \mu_0 + \mu_1 s_t, \]

where \( \{\varepsilon_t\} \) is a sequence of independent and identically distributed (i.i.d.) real-valued random variables with mean zero and unit variance, \( s_t \) is a random variable which takes values in the finite set \( \Upsilon = \{0, 1\} \) and indicates the unobservable state of the system at time \( t \), \( \mu_i \) \( (i = 0, 1) \) are real constants. The process \( \{s_t\} \) is assumed to form a strictly stationary, time-homogenous, first-order Markov chain on \( \Upsilon = \{0, 1\} \) with transition probability matrix \( P = (p_{ij})_{i,j \in \Upsilon} \), where

\[ p_{ij} = \Pr(s_t = j/s_{t-1} = i), \quad i, j \in \Upsilon, \]

and \( \sum_{j=1}^{2} p_{ij} = 1 \) for \( i \in \Upsilon \). It is also assumed that \( \{s_t\} \) is independent of \( \{u_t\} \) and that \( P \) is ergodic.

In this paper, we combine both (1) and (3) into the following new time series model,

\[ (1 - L)^d(y_t - \mu_{s_t}) = \varepsilon_t, \]

\[ (4) \]
which hereafter will be called MS-LM.

To estimate the parameters of this model, we extend the MCMC algorithm proposed by Albert and Chib (1993) for MS models to take into account the additional parameter $d$.

4 MCMC algorithms

We now describe the Gibbs sampling algorithms, which are used to estimate the ARFIMA and the MS-LM models. This entails the derivation of the conditional posterior distribution for each one of the process parameters, given appropriate conditional prior distributions. For an excellent expository treatment of the Gibbs sampling, see Casella and George (1992).

4.1 A MCMC algorithm for the ARFIMA model

For an ARFIMA model, several methods for estimating the fractional difference parameter $d$ are available in the literature. The R/S algorithm, which is proposed by Hurst (1951) in Hydrology, is analyzed by Mandelbrot and Wallis (1969) and Mandelbrot and Taqqu (1979). A regression method based on the spectral density of the fractional ARIMA process is proposed by Geweke and Porter-Hudak (1983). This method provides an asymptotically-biased estimate of $d$, which is less efficient than the estimate obtained using maximum likelihood (ML) procedures; see, for example, Hurvich and Beltrao (1993). Semiparametric modifications to this method are proposed by Robinson (1995).

For the Gaussian ML estimator, several asymptotic results are available: Fox and Taqqu (1986), Dahlhaus (1989) and Giraitis and Surgailis (1990) show that
the ML estimator of an ARFIMA model is consistent, asymptotically normal and asymptotically efficient. Despite all these desiderate properties, implementing exact ML procedures presents computational problems in practice. First, since the correlations of a long memory model decay slowly, all autocorrelations, including those with large lags, have to be taken into account in the computation. Therefore, the procedure is computationally demanding. Second, contrary to the standard ARIMA model, for the ARFIMA model, there are not available recursive algorithms that allow us to calculate the log likelihood function in an easy way. An excellent discussion of the computational problems that arise from implementing ML procedures in this class of models can be found in Li and McLeod (1986) and Sowell (1992). To avoid these problems, we estimate the parameters of the model in the Bayesian framework. We employ a Gibbs sampling algorithm, which is straightforward to implement and has a very small computational cost. We next describe the steps of the algorithm.

4.1.1 Generating $d$

The values of $d$ are chosen using a Metropolis-Hasting algorithm with a $N(0, \sigma_d^2)$ proposal. Rejection sampling is used to enforce $0.0 < d < 0.5$. After $d$ is drawn, the series $y_t$ is fractionally differenced using a binomial expansion for $(1 - L)^d$ with some limited number $l$ of lags. The prior of $d$ is uniformly distributed over the set of real values such that $0 < d < 0.5$.

4.1.2 Generating $\mu$ conditional on $d$ and $\tilde{y}_T$

Rearranging equation (2), we obtain $y_t^* = \mu x_t^* + \varepsilon_t$, where
\[ y_t^* = y_t - \sum_{j=1}^{l} \pi_j y_{t-j}, \quad \text{and} \quad x_t^* = 1 - n \sum_{j=1}^{l} \pi_j. \]

In matrix notation, we have

\[ Y^* = \mu X^* + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2 I_T). \]

We assume a normal prior for \( \mu \):

**Prior:** \( \mu \sim N(b_0, B_0) \), where \( b_0 \) and \( B_0 \) are known.

Then, the posterior distribution is given by

**Posterior:** \( \mu | \sigma^2, \tilde{y}_T \sim N(b_1, B_1) \), where \( \tilde{y}_T = [y_1, ..., y_T]' \) is the vector of observed data in the sample, \( b_1 = (B_0^{-1} + \sigma^{-2} X^* X^*)^{-1}(B_0^{-1} b_0 + \sigma^{-2} X^* Y^*) \), and \( B_1 = (B_0^{-1} + \sigma^{-2} X^* X^*)^{-1} \).

This is a standard result of normal distribution with a conjugate prior, see DeGroot (1970).

### 4.1.3 Generating \( \sigma^2 \) conditional on \( d \), \( \mu \) and \( \tilde{y}_T \)

We assume the following prior:

**Prior:** \( \sigma^2 \sim IG(v_0, \delta_0) \), where \( v_0 \) and \( \delta_0 \) are known hyperparameters, and \( IG \) refers to the inverted Gamma distribution.

**Posterior:** \( \sigma^2 | \tilde{y}_T, d, \mu \sim IG(v_1, \delta_1) \), where \( v_1 = v_0 + T \) and \( \delta_1 = \delta_0 + \sum_{t=1}^{T} \varepsilon_t^2 \).

This is also a standard conjugate result (see DeGroot, 1970).
4.2 A MCMC algorithm for the MS-LM model

In this framework, it is not attractive to employ ML techniques for two important reasons. First, we have the computational problems described in the previous section because of the long memory property of the model. Second, it is well-known that ML estimation of regime-switching models is plagued for complicated likelihood functions with numerous local maxima, and thus there are strong convergence problems (see Boldin (1996) for details). Therefore, to avoid these computational problems, we extend the MCMC algorithm proposed by Albert and Chib (1993) for MS models to take into account the additional parameter $d$. We discuss the algorithm in the next subsections.

4.2.1 Generating $d$

The derivation of the values of $d$ is exactly the same as that of the linear long memory model. After $d$ is drawn, the MCMC algorithm steps are identical to the steps in the MCMC proposed by Albert and Chib (1993). Here, The only difference is that we employ a Multi-move Gibbs sampling to generate $s_t$, $t = 1, 2, ..., T$.

4.2.2 Generating $\tilde{s}_T$ conditional on $\mu_0, \mu_1, \sigma^2, p, q, d$ and $\tilde{y}_T$

Differently to Albert and Chib (1993), to generate $\tilde{s}_T = [s_1, ..., s_T]'$, we use the Multi-move Gibbs sampling proposed by Carter and Kohn (1994) in the context of a state-space model. The algorithm allows us to generate $s_t$, $t = 1, 2, ..., T$, as a block from the joint conditional distribution $f(\tilde{s}_T|d, \mu_0, \mu_1, \sigma^2, p, q, \tilde{y}_T)$.

Suppressing the conditioning on the parameters of the model, the conditional
joint distribution can be written as follows:

\[
   f(s_T | y_T) = f(s_1, ..., s_T | y_T) = f(s_T | y_T) * f(s_{T-1}, ..., s_1 | s_T, y_T)
\]

\[
   = f(s_T | y_T) * f(s_{T-1} | s_T, y_T) * f(s_{T-2}, ..., s_1 | s_T, s_{T-1}, y_T)
\]

\[
   = f(s_T | y_T) * f(s_{T-1} | s_T, y_T) * f(s_{T-2} | s_T, s_{T-1}, y_T)
\]

\[...
   = f(s_{t+1} | s_t, s_{t-1}, ..., s_2, y_T)
\]

\[
   = f(s_{T-1} | s_T, y_T-1) f(s_{T-2} | s_{T-1}, y_{T-2}) * ... * f(s_1 | s_2, y_1)
\]

\[
   = f(s_T | y_T) \prod_{t=1}^{T-1} f(s_t | s_{t+1}, y_T).
\]

(5)

The above derivation depends on the Markov property of \( s_t \) conditional on \( s_{t+1} \). For example, \((s_{t+2}, ..., s_T)\) and \((y_{t+1}, ..., y_T)\) contain no information beyond \( s_{t+1} \). Equation (5) suggests that we first generate \( s_T \) conditional on \( y_T \). Second, for \( t = T - 1, T - 2, ..., 1 \), we generate \( s_t \) conditional on \( y_t \) and, finally, we generate \( s_{t+1} \). To do that, we employ the following steps:

**S1:** Run Hamilton’s (1989) filter to get \( f(s_t | y_T) \) for \( t = 1, 2, ..., T \). From the last iteration, we obtain \( f(s_T | y_T) \), from which \( s_T \) is generated.

**S2:** To generate \( s_t \) conditional on \( s_{t+1} \) and \( y_t \) for \( t = T - 1, T - 2, ..., 1 \), we use the following result:

\[
   f(s_t | y_t, s_{t+1}) = \frac{f(s_t, s_{t+1} | y_T)}{f(s_{t+1} | y_T)} = \frac{f(s_{t+1} | s_t, y_T) f(s_t | y_T)}{f(s_{t+1} | y_T)}
\]

\[
   = \frac{f(s_{t+1} | s_t) f(s_t | y_T)}{f(s_{t+1} | y_T)} \propto f(s_{t+1} | s_t) f(s_t | y_T),
\]

where \( f(s_{t+1} | s_t) \) is the transition probability, and \( f(s_t | y_T) \) is saved from S1.

Therefore, to generate \( s_t \), we first calculate

\[
   \Pr[s_t = 1 | s_{t+1}, y_t] = \frac{f(s_{t+1} | s_t = 1) f(s_t = 1 | y_T)}{\sum_{j=0}^{1} f(s_{t+1} | s_t = j) f(s_t = j | y_T)}.
\]

10
Then, we draw random numbers from a uniform distribution to generate $s_t$. For example, we generate a random number from a uniform distribution $U(0, 1)$. If the generated number is less than or equal to the calculated value $\Pr[s_t = 1|s_{t+1}, \tilde{y}_t]$, we set $s_t = 1$. Otherwise, we set $s_t = 0$.

### 4.2.3 Generating transition probabilities $p$ and $q$ conditional on $\tilde{s}_T$

Note that, conditional on $\tilde{s}_T$, $p$ and $q$ are independent of the data set $\tilde{y}_T$ and the other parameters of the model. We assume independent beta distributions for the priors of $p$ and $q$:

$$p \sim beta(u_{11}, u_{10}), \quad \text{and} \quad q \sim beta(u_{00}, u_{01})$$

with

$$f(p, q) \propto p^{u_{11}-1}(1 - p)^{u_{10}-1}q^{u_{00}-1}(1 - q)^{u_{01}-1},$$

where $u_{ij}, i, j = 0, 1$, are known hyperparameters of the priors.

The likelihood function for $p$ and $q$ is given by:

$$L(p, q|\tilde{s}_T) = p^{n_{11}}(1 - p)^{n_{10}}q^{n_{00}}(1 - q)^{n_{01}},$$

where $n_{ij}$ refers to the transition for $i$ to $j$, which can be easily counted for given $\tilde{s}_T = [s_1, ..., s_T]$.

Combining the prior distribution and the likelihood function, we obtain the following posterior distribution:
\begin{align*}
P(p, q|\tilde{s}_T) &= f(p, q)L(p, q|\tilde{s}_T) \\
&\propto p^{u_{11}-1}(1-p)^{u_{10}-1}q^{u_{00}-1}(1-q)^{u_{01}-1}p^{n_{11}}(1-p)^{n_{10}}q^{n_{00}}(1-q)^{n_{01}} \\
&= p^{n_{11}+u_{11}-1}(1-p)^{n_{10}+u_{10}-1}q^{n_{00}+u_{00}-1}(1-q)^{n_{01}+n_{01}-1}.
\end{align*}

Therefore, the posterior distributions are given by two independent beta distributions:

\[ p|\tilde{s}_T \sim \text{beta}(u_{11}+n_{11}, u_{10}+n_{10}), \quad \text{and} \quad q|\tilde{s}_T \sim \text{beta}(u_{00}+n_{00}, u_{01}+n_{01}), \]

from which \( p \) and \( q \) are drawn.

4.2.4 Generating \( \mu_0 \) and \( \mu_1 \), conditional on \( \sigma^2 \), \( d \), \( \tilde{s}_T \) and \( \tilde{y}_T \)

Rearranging (3), we obtain \( y_t^* = \mu_0 + \mu_1 s_t^* + \varepsilon_t \), where

\[ y_t^* = y_t - \sum_{j=1}^{l} \pi_j y_{t-j}, \quad \text{and} \quad s_t^* = s_t - \sum_{j=1}^{l} \pi_j s_{t-j}. \]

We write equation (3) in matrix form:

\[ \tilde{y}_T^* = \tilde{s}_T^* \tilde{\mu} + \nu, \quad \nu \sim N(0, I_{T-2}). \]

Now, it is straightforward to derive the posterior distribution of \( \tilde{\mu} = [\mu_0, \mu_1] \)
given an appropriate prior distribution:

**Prior:** \( \tilde{\mu} \sim N(a_0, A_0)I[\mu_1 > 0] \), where \( a_0 \) and \( A_0 \) are known hyperparameters of the prior distribution, and \( I[\cdot] \) is an indicator function.

**Posterior:** \( \tilde{\mu}|\sigma^2, d, \tilde{s}_T, \tilde{y}_T \sim N(a_1, A_1)I[\mu_1 > 0] \), where \( a_1 = (A_0^{-1} + \tilde{s}_T^* \tilde{y}_T^*)^{-1}(A_0^{-1}a_0 + \tilde{s}_T^* \tilde{y}_T^*)^{-1} \), and \( A_1 = (A_0^{-1} + \tilde{s}_T^* \tilde{y}_T^*)^{-1} \).
This is a standard result of normal distribution with a conjugate prior; see DeGroot (1970). To fulfill the constraint \( \mu_1 > 0 \), we use the rejection sampling.

4.2.5 Generating \( \sigma^2 \) conditional on \( \mu_0, \mu_1, \tilde{s}_T, \tilde{y}_T, d \)

In this case, we suppose the following prior distribution:

**Prior:** \( \sigma^2 \sim IG\left(\frac{\nu_0}{2}, \frac{\delta_0}{2}\right) \), where \( \nu_0 \) and \( \delta_0 \) are known hyperparameters.

**Posterior:** \( \sigma^2|\mu_0, \mu_1, \tilde{s}_T, \tilde{y}_T, d \sim IG\left(\frac{\nu_1}{2}, \frac{\delta_1}{2}\right) \), where \( \nu_1 = \nu_0 + T \), and \( \delta_1 = \delta_0 + (\tilde{y}_T - \tilde{s}_T \hat{\mu})' (\tilde{y}_T - \tilde{s}_T \hat{\mu}) \).

This is also a standard conjugate result; see DeGroot (1970).

4.2.6 Implementation issues

Based on these conditional posterior distributions, the Gibbs sampling can be readily implemented given the hyperparameters and some initial values of the unknown parameters. The algorithm runs as follows:

**P1** Draw sequentially random realizations from the conditional posterior distributions.

**P2** Once the realizations are drawn, treat them as new parameter values and iterate the random drawing in (P1) for \( M + N \) times, where \( M \) and \( N \) are prespecified positive integers.

**P3** Use the realizations of the last \( N \) iterations to form a random sample of the joint posterior distributions of the unknown parameters on which statistical inference can be deduced.
In practise, selection of \( M \) and \( N \) requires certain attention to ensure the convergence of the Gibbs sampling. Gelman and Rubin (1992) and McCulloch and Tsay (1993) suggest trying various different sets of starting values. Thus, we repeated the Gibbs sampling with different initial values, but obtained similar results. The Gibbs sampling appears to have converged. We choose \( M = 2000 \) and \( N = 10000 \). The priors are specified as almost uninformative and are reported in Tables 1-3 in Appendix. We select similar priors to those used by Albert and Chib (1993). To see if our results are sensitive to start-up values, we tried for different starting values. None significant difference was found. As the results are not dependent on the priors selected, we do not provide a sensitivity analysis with respect to the prior inputs. As a check of our methods, we report in Table 1 a simple simulation experiment with a sample size of \( T = 1324 \). We change \( l \) and make sure that the posterior probability distribution function of \( d \) is insensitive to \( l \) lags. We used \( l = 10, 20, 30 \). The final results are reported for \( l = 10 \). We can clearly see that the posterior means are generally close to the true values that generated the data.

5 Forecasting using predictive densities within MCMC

Forecasting under the MCMC framework can be done easily. The procedure is simply to use the fitted model in each Gibbs iteration to generate samples for the forecasting period. In a sense, forecasting here is done by using the fitted model

\[1\] More experiments were run using different sets of parameters, but results are always quite similar. They are available from the author upon requests.
to simulate realizations for the forecasting period. We now describe briefly the procedure, for more details Albert and Chib (1993) and Tsay (2005). Samples from Bayes prediction density can be obtained within the MCMC algorithm. Let $\theta$ be the vector of parameters. For each draw of $(s_{T+1-n}^T, \theta)$ made available via the Gibbs sampling, we sample

\[(a) \, s_{T+1} \text{ from } \Pr(s_{T+1}/s_T),\]
\[(b) \, y_{T+1} \text{ from (2)}.\]

These two steps can be easily implemented along with the regular Gibbs cycle. Therefore, at the end of the algorithm we obtain samples from both the parameter posterior and the prediction posterior. It is easy to generalize this procedure. Specifically, consider $y_{T+2}$. We obtain a draw from the prediction density as follows:

\[(a^*) \, s_{T+2} \text{ from } \Pr(s_{T+2}/s_{T+1}), \text{ using } s_{T+1} \text{ drawn in (a)},\]
\[(b^*) \, s_{T+2} \text{ from (2), using } y_{T+1} \text{ drawn in (b)}.\]

We run MCMC for $M+N$ iterations in model estimation and generate forecasts for the last $N$ iterations. Note that, in this way, we incorporate both parameter uncertainty and state uncertainty in predictive inferences. Various forecasting criteria can be used to assess the predictive accuracy of the models considered. Two popular loss functions are used to evaluate and compare the various models: the Mean Squared Error (MSE) and the Mean Absolute Error (MAE). Using
MCMC algorithms, we obtain the predictive density of MSE as well as that of the MAE:

\[
MSE^{(i)} = \frac{1}{m} \sum_{j=1}^{m} (y_{T+j}^{(i)} - y_{T+j})^2; \quad MAE^{(i)} = \frac{1}{m} \sum_{j=1}^{m} |y_{T+j}^{(i)} - y_{T+j}|,
\]

where \(y_{T+j}^{(i)}\) is the \(i\)-th draw of the predicted value of \(y_{T+j}\), \(j = 1, ..., m\), and \(y_{T+j}\) is the realized value at \(T + j\).

It is clear that we cannot conclude that the forecasting performance of a model is superior to that of another one on the basis of just one criterion and just one sample. Several testing strategies have been proposed in the literature that allow us to determine whether a particular model is outperformed by another one; see Diebold and Mariano (1995), West (1996) and White (2000). Recently, Hansen (2001), building in the work of White (2000), proposes a superior predictivity ability (SPA) test. He also shows that SPA has good power properties and is robust. In this paper, we employ the SPA test to study the relative performance of various volatility models. In the following, we briefly describe the Hansen testing strategy, for more details see Hansen (2001) and Hansen and Lude (2005). We consider \(K + 1\) models \(M_k\) for \(k = 0, 1, 2, ..., K\). For each model, we calculate \(m\) forecasts and for every forecast we calculate an appropriate loss function \(L_{m,k}\). We choose a particular model \(M_0\) as the benchmark model. The loss function relative to the benchmark model is defined as

\[
X_{k,m} = L_{m,0} - L_{m,k}.
\]

We assume that the models under study can be ranked consistently, in this way, \(\lambda_k = E(X_{k,m})\) is well defined. When the benchmark model \(M_0\) outperforms
all the other models, we have $\lambda_k < 0$ for all models $k = 1, 2, \ldots, K$. Therefore, the null hypothesis of interest is the following

$$\max_{k=1,2,\ldots,K} \lambda_k \leq 0.$$ 

If we accept the null hypothesis, the benchmark is not outperformed. The test proposed by Hansen (2001) is given by

$$T_H = \max_{k=1,2,\ldots,K} \sqrt{m} \frac{X_k}{\hat{\omega}_{kk}},$$

where $\hat{\omega}_{kk}^2$ is a consistent estimate of $\omega_{kk}^2$, and

$$X_k = \frac{1}{m} \sum_{i=1}^{m} X_{k,m},$$

$$\omega_{kk}^2 = \lim_{m \to \infty} \text{var}(\sqrt{m} X_k).$$

We employ the stationary bootstrap of Politis and Romano (1994) to calculate the p-value of the test statistic $T_H$ and obtain a consistent estimator of $\omega_{kk}$. We denote the bootstrap samples with $X_{k,m}^i$ for $i = 1, 2, \ldots, B$. For each bootstrap, the sample mean is computed

$$\bar{X}_k = \frac{1}{m} \sum_{i=1}^{m} X_{k,m}, \quad i = 1, \ldots, B.$$ 

The bootstrap estimate of $\omega_{kk}$ is given by

$$\hat{\omega}_{kk} = 1/B \sum_{i=B}^{B} (\bar{X}_k^i - \bar{X}_k)^2, \quad \bar{X}_k = \sum_{i=B}^{B} X_{k,m}^i.$$ 

Thus, using $\hat{\omega}_{kk}$, we are able to calculate the test statistic $T_H$.

Finally, we identify the empirical distribution of $T_H$ under the null hypothesis as follows. Define $Z_k^i$ as
\[ Z_k^i = (X_k^i - \overline{X}_k) \times I(\overline{X}_k > A_k), \]

where \( A_k = \frac{1}{4M^4 \hat{\omega}_{kk}} \), and \( I \) is an indicator function. Hansen (2001) shows that the empirical distribution of

\[ T^i_H = \max_{k=1,2,\ldots,K} \frac{\sqrt{m} Z_k^i}{\hat{\omega}_{kk}} \]

converges to the distribution of \( T_H \) under the null hypothesis. Therefore, we can compute the p-value of the test as

\[ 1/B \sum_{i=1}^{B} I(T^i_H > T_H). \]

### 6 Empirical results

In this section, we compare the estimation and forecast performance of ARFIMA, MS, MS-LM models for realized volatility of the S&P 500 series for the sample period that runs from January 3, 2000 to February 28, 2005. Realized volatility is a popular measure of volatility proposed by Andersen and Bollerslev (1997). Since its introduction, there has been a considerable interest in its distribution, modelling and forecasting (see Andersen et al., 2003). The realized volatility is defined as the sum of squared intra-day returns taken at fixed small periods \( \Delta \) within a day, e.g. 5-minute returns are accepted by many studies as a norm:

\[ R^2_t = \sum_{\tau} r^2_{t+\tau\Delta}, \]

where \( r^2_{t+\tau\Delta} \) is the return for a small fixed period \( \Delta \) within a day. We estimated the models using the log of realized volatility of 5-minute stocks returns, which we denote as \( \tilde{R}_t \), i.e. \( \tilde{R}_t = \log(R^2_t) \).
Figure 1 shows the time series of the realized volatility and its counterpart in logs, together with their respective correlograms and Kernel density estimates. The realized volatility is heavily skewed and exhibits excess kurtosis. By contrast, $\tilde{R}_t$ is much more symmetrically distributed and has much lower kurtosis. The autocorrelations are all positive and decay to zero slowly, indicating a possible long-range dependent behavior. However, the time series plot suggests that there may be structural changes in the series.

Tables 2-4 show the marginal posterior distributions of the parameters of the corresponding models. Several conclusions can be drawn from these tables. First, the order of integration $d$ is 0.3 for the linear model and 0.4 for the nonlinear one, which is in line with estimates reported in previous studies (see Andersen et al. 2003). Second, comparing the estimate of $d$ of the different models, makes clear that allowing for nonlinearity lowers the order of integration, confirming that neglecting level shifts may spuriously suggests fractional integration (see Granger and Terasvirta, 1999). Note, however, that the posterior standard errors of $d$ are not large, indicating little uncertainty in the estimates of $d$. Hence, the level shift cannot fully account for the long-memory of the volatility. Third, the estimated transition probabilities of the MS model are quite large, indicating that the regimes are very persistent. Therefore, it is not surprising to find long-memory in the series, considering the results in Diebold and Inoue (2001). Fourth, the AIC and BIC of the different models show that incorporating nonlinearity enhances the in-sample fit. Allowing for long memory appears to be most important in this respect, because the MS-LM model, which includes this effect, is preferred to the others. In both the nonlinear models, we have one phase of high volatility characterized by a negative value of $\mu_0$, and the other phase of low volatility.
characterized by a positive value of $\mu_1$.

Figures 2-4 give marginal posterior density estimates for the parameters of interest. These plots were constructed using normal Kernel density estimates. They show that the densities of some parameters are distinctly skewed. Thus, the maximum likelihood assumption of normality appears to be inaccurate for these parameters.

The out-of-sample period that runs from 17-October-2003 to 28-February-2005 is used for forecasting purposes. The empirical forecast accuracy comparison is based on series of rolling forecasts. The models are estimated once, and for all of them in the sub-sample 1/3/2000-16/10/2003, a sequence of 1, 5, 10 and 20 days ahead forecasts are generated using 6/10/2003 as the forecast origin. The forecast origin is then rolled forward one period, and another sequence of 5, 10 and 20 days ahead forecasts is generated. This is repeated until we achieve the end of the out-of-sample forecast period. Two popular out-of-sample forecast performance measures are used to evaluate and compare the various models: the MSE and the MAE. The results for 1, 5, 10 and 20 days ahead forecasts are presented in Table 5.

For both loss functions and for every forecast horizons, the best forecast performance is attained by the MS-LM model. For example, for the MS-LM, MS and ARFIMA models, the MSE for the 1-day ahead forecast is 0.709, 0.922 and 1.304, respectively. It is interesting to highlight the performance of the MS model, which ranks second for both the shorter and longer horizons. These results are in accordance with those obtained by Gabriel and Martins (2004). The benefits of modelling the nonlinearities explicitly become less for longer horizons in the case of the MSE. For example, the MSE of the MS-FI model is 0.709 for the 1-day
horizon, and increases to 0.783 in the 20-day horizon. Although, the forecasting results have clearly shown that the MS-FI model for realized volatility is the best model for prediction, these results only apply to a one selected sample. In order to circumvent the specificity of these results, we apply the hypothesis testing theory proposed by Hansen (2001). In Table 6, we report the SPA results for the MS-LM, MS and ARFIMA models. The individual $X_k$ is reported for each model that is considered as a base model $M_0$, while the alternative models are treated as rival models $M_1$. All models under study are consecutively taken as a base model. The last column of the table reports the p-values of the SPA tests for each base model. A high p-value suggests that the null hypothesis of ”the base model is not outperformed” cannot be rejected. As the results for the several horizons considered are qualitatively similar, we present only results for 1-day horizon. The average of daily squared forecast error differences between ARFIMA and MS models is 2.42, which means that the squared forecast error of ARFIMA was in average higher that of MS. The average values for ARFIMA in relation to the rival MS-LM is 4.02.

This results are not good for the ARFIMA, as it is also reflected by its p-value of zero for the SPA test. Overall, Table 6 shows that for both loss functions, it can be concluded that MS-LM forecasts are to be preferred. These results confirm the earlier findings from Table 5. Our analysis puts in evidence the importance of modeling long memory and nonlinearities jointly in order to obtain better forecasts of future volatility. We confirm the findings of Franses and Hyung (2004), which obtained the same results using a long memory model with structural breaks.
7 Conclusion

In this paper, we propose a new time series model, which can jointly capture long memory and Markov nonlinearities. We applied the model to S&P 500 realized volatility, and compare its performance in terms of fit and forecasting with those of MS and ARFIMA models. All models were estimated using an MCMC algorithm and predictive densities were obtained within such an algorithm. The in-sample results show that all nonlinearities are highly significant and improve the description of the data. The out-sample results show that modelling these nonlinearities produces superior volatility forecasts over those obtained from a linear ARFIMA model.

Acknowledgements

I would like to thank Monica Billio, Davide Raggi and Silvano Bordignon for their valuable comments, which resulted in a great improvement of an earlier version of this paper. Useful comments were also received from seminar participants at the University of Venezia. All errors are my own responsibility.

References


Li, W.K. and McLeod, A.I. (1986). Fractional time series modelling. *Biome-


8 Appendix

Table 1: Simulated Data From MS-LM Model

<table>
<thead>
<tr>
<th>True Parameter</th>
<th>Prior MEAN</th>
<th>SD</th>
<th>Posterior MEAN</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d$ (0.300)</td>
<td>-</td>
<td>-</td>
<td>0.297</td>
<td>0.061</td>
</tr>
<tr>
<td>$\sigma^2$ (0.131)</td>
<td>-</td>
<td>-</td>
<td>0.134</td>
<td>0.022</td>
</tr>
<tr>
<td>$\mu_0$ (-0.411)</td>
<td>0</td>
<td>10</td>
<td>-0.446</td>
<td>0.031</td>
</tr>
<tr>
<td>$\mu_1$ (0.622)</td>
<td>0.3</td>
<td>10</td>
<td>0.671</td>
<td>0.049</td>
</tr>
<tr>
<td>$p$ (0.840)</td>
<td>0.8</td>
<td>0.16</td>
<td>0.832</td>
<td>0.136</td>
</tr>
<tr>
<td>$q$ (0.815)</td>
<td>0.8</td>
<td>0.16</td>
<td>0.827</td>
<td>0.185</td>
</tr>
</tbody>
</table>

The prior distribution of $\sigma^2$ is improper while $d$ is uniformly distributed over the set of real values such that $0 < d < 0.5$ for all $t$. The true parameters are the point estimates of the MS-LM model (see Table 4). The priors are similar to those used by Albert and Chib (1993).
The prior distribution of $\sigma^2$ is improper while $d$ is uniformly distributed over the set of real values such that $0 < d < 0.5$ for all $t$. 

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Prior</th>
<th>Posterior</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MEAN</td>
<td>SD</td>
</tr>
<tr>
<td>$d$</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$\sigma^2$</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$\mu$</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>$AIC$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$BIC$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The prior distribution of $\sigma^2$ is improper while $d$ is uniformly distributed over the set of real values such that $0 < d < 0.5$ for all $t$. The priors are similar to those used by Albert and Chib (1993).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Prior MEAN</th>
<th>Prior SD</th>
<th>Posterior Mean</th>
<th>Posterior SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma^2$</td>
<td>-</td>
<td>-</td>
<td>0.170</td>
<td>0.003</td>
</tr>
<tr>
<td>$\mu_0$</td>
<td>0</td>
<td>10</td>
<td>-0.512</td>
<td>0.012</td>
</tr>
<tr>
<td>$\mu_1$</td>
<td>0.3</td>
<td>10</td>
<td>1.056</td>
<td>0.134</td>
</tr>
<tr>
<td>$p$</td>
<td>0.8</td>
<td>0.16</td>
<td>0.950</td>
<td>0.011</td>
</tr>
<tr>
<td>$q$</td>
<td>0.8</td>
<td>0.16</td>
<td>0.912</td>
<td>0.037</td>
</tr>
</tbody>
</table>

$AIC = -1.166$

$BIC = -1.123$
Table 4: MS-LM model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Prior MEAN</th>
<th>SD</th>
<th>Posterior Mean</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d$</td>
<td>-</td>
<td>-</td>
<td>0.300</td>
<td>0.070</td>
</tr>
<tr>
<td>$\sigma^2$</td>
<td>-</td>
<td>-</td>
<td>0.131</td>
<td>0.042</td>
</tr>
<tr>
<td>$\mu_0$</td>
<td>0</td>
<td>10</td>
<td>-0.411</td>
<td>0.021</td>
</tr>
<tr>
<td>$\mu_1$</td>
<td>0.3</td>
<td>10</td>
<td>0.622</td>
<td>0.133</td>
</tr>
<tr>
<td>$p$</td>
<td>0.8</td>
<td>0.16</td>
<td>0.840</td>
<td>0.035</td>
</tr>
<tr>
<td>$q$</td>
<td>0.8</td>
<td>0.16</td>
<td>0.815</td>
<td>0.101</td>
</tr>
<tr>
<td>$AIC$</td>
<td>-1.301</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$BIC$</td>
<td>-1.159</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The prior distribution of $\sigma^2$ is improper while $d$ is uniformly distributed over the set of real values such that $0 < d < 0.5$ for all $t$. The priors are similar to those used by Albert and Chib (1993).

Table 5: Out-of-sample forecast evaluation, October 2003-February 2005

<table>
<thead>
<tr>
<th>Forecast period</th>
<th>1</th>
<th>5</th>
<th>10</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MSE</td>
<td>MAE</td>
<td>MSE</td>
<td>MAE</td>
</tr>
<tr>
<td>ARFIMA</td>
<td>1.304</td>
<td>0.689</td>
<td>1.329</td>
<td>0.647</td>
</tr>
<tr>
<td>MS</td>
<td>0.922</td>
<td>0.566</td>
<td>0.938</td>
<td>0.529</td>
</tr>
<tr>
<td>MS-LM</td>
<td>0.709</td>
<td>0.248</td>
<td>0.736</td>
<td>0.227</td>
</tr>
<tr>
<td>Loss function</td>
<td>Base model</td>
<td>Alternative models</td>
<td>p-value $T_H$</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>------------</td>
<td>--------------------</td>
<td>---------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ARFIMA</td>
<td>MSE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSE</td>
<td>-</td>
<td>2.42</td>
<td>4.02</td>
<td>0.000</td>
</tr>
<tr>
<td>MS</td>
<td>-2.42</td>
<td>-</td>
<td>3.41</td>
<td>0.003</td>
</tr>
<tr>
<td>MS-LM</td>
<td>-4.02</td>
<td>-3.41</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>MAE</td>
<td>ARFIMA</td>
<td>MSE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>3.57</td>
<td>9.66</td>
<td>0.000</td>
</tr>
<tr>
<td>MS</td>
<td>-3.57</td>
<td>-</td>
<td>5.33</td>
<td>0.015</td>
</tr>
<tr>
<td>MS-LM</td>
<td>-9.66</td>
<td>-5.33</td>
<td>-</td>
<td>1</td>
</tr>
</tbody>
</table>

A positive value in a column of an alternative model $M_k$ indicates that it is superior to the base model according to a specific loss function. Final column has the p-value of the $T_H$ test. It can be interpreted as the intensity of the base model producing superior forecast. A unity value indicates here that the test is not computed since the model outperformed all others.
Figure 1:
Figure 2: Density functions of prior and posterior distributions in a ARFIMA model. The dashed line denoted prior density and solid line denotes the posterior density.
Figure 3: Density functions of prior and posterior distributions in a MS model.

The dashed line denoted prior density and solid line denotes the posterior density.
Figure 4: Density functions of prior and posterior distributions in a MS-LM model. The dashed line denoted prior density and solid line denotes the posterior density.