IMA Journal of Management Mathematics (2008) **19**, 379−401
doi:10.1093/imaman/dpm026
Advance Access publication on October 8, 2007

Hidden Markov models for scenario generation

ENZA MESSINA†
DISCO (Dipartimento di Informatica Sistemistica e COmunicazione), University of Milano Bicocca, Via Bicocca degli Arcimboldi 8, 20126 Milano, Italy

AND

DANIELE TOSCANI‡
DISCO (Dipartimento di Informatica Sistemistica e COmunicazione), University of Milano Bicocca, Via Bicocca degli Arcimboldi 8, 20126 Milano, Italy and Consorzio Milano Ricerche, Via Cicognara 7, 20129 Milano, Italy

[Accepted on 2 May 2007]

We consider the problem of modelling processes sequentially changing behaviour and unexpected changes that can hinder finding the best approximation function. These dynamics cannot be observed directly either because they are masked by observational noise or because the process generating them is too complex and involves too many variables. In this paper, the problem of modelling financial time series has been approached using hidden Markov models (HMMs), which have been shown to be suitable for sequential data analysis and in particular for financial time series modelling and forecasting. HMMs are essentially data-driven models that allow us to focus attention on the observation generation process, which is indeed final objective. The goal of our time series analysis model is the generation of scenarios to be included in decision models. Therefore, our focus will not be on determining the best forecast but in capturing the generation process behaviour in order to characterize its possible evolutions.

Keywords: scenario generation; simulation; hidden Markov models.

1. Introduction

Many traditional time series models are based on local time information and assume stationarity, or weak stationarity, of the time series. However, time series, and in particular financial time series, are certainly non-stationary. Moreover, they show asymmetries between growth and decline periods and it is often difficult to distinguish between long-term trends and movements due to noise. Empirically, we can indeed observe that in many cases series are characterized by heteroskedasticity of the variance, i.e. variance is not constant, and leptokurtosis (fat tails) of the distribution, i.e. values far from the centre of the distribution receive a relatively high probability. It is therefore important to employ models that are able to capture the temporal structure in time series in order to characterize them dynamically.

On the other end, in presence of fast changes in the market, a single model is valid only for short periods. Rapid and unexpected changes in data trends can hinder finding the best approximation function. In this case, a single model is not enough to explain all the different patterns.

A new perspective to deal with non-stationary processes is introduced by 'switching models'. Instead of considering a single global model, switching models are concerned with adjusting several local

†Email: messina@disco.unimib.it
‡Email: toscani@milanoricerche.it

models for the different time series regimes. The first introduction of the ‘regime-switching’ concept is due to Hamilton and Lam (Hamilton, 1989); they proposed state transition among autoregressive regimes driven by a variable that follows a Markov process. In Hamilton & Susmel (1994), several regime-switching models are analysed, varying the number of regimes and the model underlying them. Their objective is to model various econometric series; for this purpose, the more complicated autoregressive conditionally heteroskedastic-type models within regimes seem to be necessary. In Hardy (2001), the author applies regime-switching log-normal models for pricing options and deriving risk measures for equity-linked insurance.

The difficulty in financial time series modelling is to distinguish between short-term movements and long-term trends affected by contingent behavioural noise. This noise can pose a severe problem if not treated properly, because it masks the true dynamic of the system, leading to models that underestimate the functional relation between past and future values. Unfortunately, a direct mapping from the observed data and the system state does not exist. This requires the notion of hidden state.

We propose an alternative to classical approaches that is based on hidden Markov models (HMM) and captures the underlying dynamics through a state-space model. The key of the state-space modelling is to split the noise into two parts: dynamic noise, which drives the evolution of the hidden state, and observational noise, which is a non-explainable additive contribution to the observed value. Generally, it is recognized that HMMs are well-suited for sequential data analysis and have been successfully applied in financial time series modelling and forecasting. We can think of data as generated by a process that we cannot directly observe or that is too complex and involves too many variables. By using HMM, we can ignore, or partially simplify, its nature while focusing our attention on the observation generation process, which is indeed the final objective of our analysis.

The proposed approach for estimating model parameters depends entirely on the observed data (i.e. it is data driven); no identification of the underlying process’ state is required.

Zhang (2004) proposes a HMM-based financial time series model, whose observations are generated by a mixture of Gaussian functions. In his model, each observed value depends only on the emitting state.

In this paper, we propose a richer model that more realistically links subsequent observations through an autoregressive process. In our model, we assume that the probability law governing the transitions among states is time invariant and not conditioned by exogenous variables. A version in which transition probabilities are time variant is presented in González *et al.* (2005) for the Spanish wholesale electricity market. Another formulation with variant state transition probabilities can be found in Shi & Weigend (1997); this state model, based on ‘Input–Output HMM’ (Bengio *et al.*, 2001; Bengio & Frasconi, 1996) generalizes data emission functions, called ‘experts’, which can be any linear or non-linear model (e.g. a neural network). All these models are, however, focused on point prediction instead of probabilistic density function estimation, whereas the goal of our time series analysis model is the generation of scenarios that describe the possible evolution of a stochastic process with the aim of integrating them in decision models. In particular, we refer to the definition of stochastic programming models, an extension of mathematical programming models where the value of some parameters can be uncertain and are represented through stochastic variables (stochastic processes in the multi-period case) over some canonical probability space. In order to provide decisions which hedge against future uncertainty, stochastic programming models combine the paradigm of dynamic programming with modelling of random parameters (Di Domenica *et al.*, 2007). Additionally, scenario generation is important as a financial risk management tool (Alexander, 2001). Therefore, our focus is not on determining better forecasts but in capturing the behaviour of the uncertain parameters in order to characterize the span of their possible evolution.

Although scenario generation is not necessarily equivalent to finding a statistical approximation, as clearly stated in Kaut & Wallace (2007), as a first step to show that HMM-based models could be used as scenario generator models, we show that the autoregressive hidden Markov model (ARHMM)-based model can replicate with a good approximation the empirical distribution of the observed data.

The proposed model has been validated using daily data from various stock indices for fitting the model parameters. The fit of the model has been compared with generalized autoregressive conditional heteroskedasticity (GARCH) and a mixture of Gaussian HMM models.

The rest of this paper is organized as follows: Section 2 introduces the general definition of HMMs and their application to financial time series analysis. Moreover, inference and learning algorithms are described for both discrete and continuous observation HMM and particular attention is devoted to the proposed ARHMM. In Section 3, the scenario generation procedure is considered and proposed along with the results obtained by modelling four different time series of stock index data.

2. Hidden Markov models

HMMs were first introduced by Baum & Petrie (1966) in the late 1960s as a tool for probabilistic sequence modelling, but the interest in this area developed particularly in the 1980s, within the speech recognition research community. During recent years, a large number of variants and improvements over the standard HMM have been proposed and applied; a more recent view of the HMM as a particular case of Bayesian networks (Smyth *et al.*, 1997) has helped their theoretical understanding and the ability to conceive extensions to the standard model in a sound and formally elegant framework.

There is a vast literature about HMM applications. Due to their versatility, they have been applied in speech recognition (Rabiner, 1989; Charniak, 1993), in bioinformatics for protein classification and sequence alignment (Hunkapiller *et al.*, 1994) and in different time series analysis, such as weather data, semiconductor malfunction (Ge & Smyth, 2000) and financial time series (Shi & Weigend, 1997).

A HMM can be considered as a stochastic process whose evolution is governed by an underlying discrete Markov process (Markov chain) with a finite number of states $s_i \in S$, $i = 1, \dots, N$, which are hidden, i.e. not directly observable. A Markov chain describes one of the mutually exclusive states that may characterize a system at any time step. Let $q_1, q_2, \dots, q_t$ be a sequence of random variables taking values from a finite set. The first-order Markovian property implies

$$P(q_t|q_{t-1}, q_{t-2}, \dots, q_1) = p(q_t|q_{t-1}), \tag{1}$$

i.e. given the present state, the future probabilistic behaviour is independent of its history except from the previous state.

At each hidden state of the underlying Markov chain, a unique probability density function (pdf) governs the emission of observations so that at each time t, the observation o_t is generated by the state $q_t = s_j$ according to the probability function

$$b_j(o_k) = P(o_t = k|q_t = s_j), \tag{1a}$$

where the observation o_t can be multidimensional (Hannaford & Lee, 1990).

A discrete HMM λ is therefore characterized by five elements:

1. A set of states $s_i \in S$, $i = 1, \dots, N$.
2. A set of observable symbols $o_m \in M$, $m = 1, \dots, K$.
3. A probability distribution for the initial state $\pi(q_1 = s_i)$, $s_i \in S$, $i = 1, \dots, N$.

4. A set of state transition probabilities that can be represented by a transition matrix A, where $a_{ij} = P(q_{t+1} = s_j \,|q_t = s_i)$ with $\sum_j a_{ij} = 1$ for each $i = 1, \ldots, N$.
5. A set of 'pdfs' B_i whose elements $b_{ik} = P(k|q_t = s_i)$ give the probability of observing in state s_i the emission of the symbol $k \in M$.

Therefore, a HMM is fully specified once we know the initial state distribution $\pi(s_i), i = 1, \ldots, N$, the state transition probabilities in A and the output probabilities $B = [B_1, \ldots, B_N]$. The number of states N and observation K is implicit in the set of vectors B; thus we will refer to the HMM through the simplified notation $\lambda = (\pi, A, B)$.

Let us use a simple example for clarifying these concepts. Suppose we want to model the evolution of an asset price; we have its daily values and observe the price variation between two consecutive days: price increase (i), price decrease (d) and unmodified (u) price. If the objective is to infer market trust assigned to the asset, we can model the process with a three-state HMM, where each state corresponds to a different trend: 'positive (p)' if market traders think that the price is undervalued, 'negative (n)' if the price is assumed to be overvalued or 'stationary (s)'. We cannot explicitly know market actors opinion but we can observe price fluctuations; therefore we assume our model has $K = 3$ observations: i, d, u.

The resulting model is depicted in Fig. 1.

Each state can emit the same set of symbols but with different probability distribution, e.g. price increases are more common during positive trend periods rather than during negative trends; therefore $b_{\mathrm{p}i} > b_{\mathrm{n}i}$ and vice versa $b_{\mathrm{n}d} > b_{\mathrm{p}d}$.

HMM applications are supported by efficient algorithms for parameter estimation and inference; the expectation–maximization (EM) algorithm (Baum & Petrie, 1966) and in particular its simplified version due to Baum and Welch (Baum *et al.*, 1970), are a well-established method for HMM parameter estimation.

In particular, three basic problems have to be addressed for employing HMM to model real-world applications:

(P1) Given an observation sequence $O = (o_1, \ldots, o_T)$ with $o_t \in \{1, \ldots, k\},\ t = 1, \ldots, T$, and an HMM $\lambda = \{\pi, A, B\}$, how can we determine the probability $P(O|\lambda)$? Referring to the above example, this corresponds to determine how well the model explains a sequence of prices.

(P2) Given $O = (o_1, \ldots, o_T)$ and the model λ, how can we find the sequence $Q = \{q_1, \ldots, q_T\}$ of consecutive states $q_t = s_i \in \{1, \ldots, |S|\},\ t = 1, \ldots, T$, which maximizes $P(Q|O, \lambda)$? This

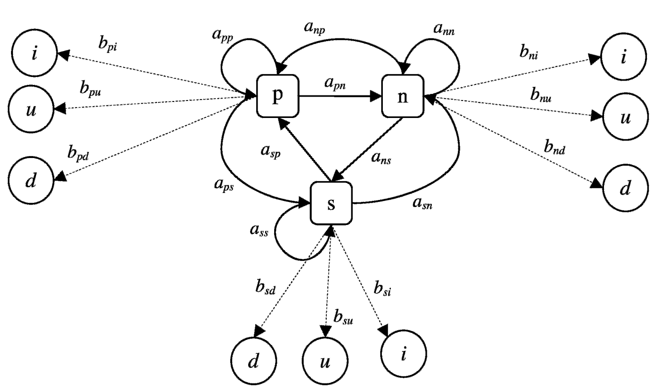

FIG. 1. HMM with three states and three discrete observations.

corresponds to determine the sequence of market states (trends) that most likely produced the observed price sequence.

(P3) Given a set of observations, how can we determine the HMM parameters that best fit them? This corresponds to the learning problem aimed at finding the model parameters that explain the given time series.

Once the inference and learning problems described above have been solved, HMM can be used for time series prediction, classification, outlier detection and simulation, allowing us to reply to a series of questions like: Will the price go up or down? What type of stock is this? Is the observed behaviour abnormal? In this paper, we use HMM for simulating sample paths which can be used for generating those scenarios able to characterize the possible evolution of a stochastic process.

2.1 *Inference and learning algorithms for discrete observation HMMs*

2.1.1 *Inferring the probability of an observation sequence.* The simplest way to compute the probability $P(O|\lambda)$ of an observation sequence $O = (o_1, o_2, \ldots, o_T)$ given the model λ is to sum the joint probabilities for all the possible state sequences of length T:

$$P(O|\lambda) = \sum_{\text{all } Q} P(O|Q,\lambda) \cdot P(Q|\lambda). \tag{2}$$

Since the probability $P(O|Q,\lambda)$ that the observation sequence O is generated by a state sequence Q is

$$P(O|Q,\lambda) = b_{q1}(o_1) \cdot b_{q2}(o_2) \cdots b_{qT}(o_T) = \prod_{i=1}^{T} b_{qi}(o_i) \tag{3}$$

and the probability of the state sequence Q is

$$P(Q|\lambda) = \pi_{q1} \cdot a_{q1q2} \cdot a_{q2q3} \cdots a_{q(T-1)qT} = \pi_{q1} \cdot \prod_{i=1}^{T-1} a_{qi,i+1}, \tag{4}$$

we obtain

$$P(O|\lambda) = \sum_{\text{all } Q} P(O|Q,\lambda) \cdot P(Q|\lambda) = \sum_{q1,\ldots,qT} \pi_{q1} b_{q1}(o_1) \cdot \prod_{t=2}^{T} (a_{q(t-1)qt} \cdot b_{qt}(o_t)). \tag{5}$$

The drawback of this intuitive formulation is its high computational cost since it requires executing $(2T-1)N^T$ multiplications and N^{T-1} sums. A more efficient procedure is the so-called 'forward–backward', which can operate in linear time.

Let us introduce the 'forward variable' $\alpha_t(i)$ which measures the probability of observing the sequence $o_1, o_2, \ldots, o_t$ given the current state s_i at time t:

$$\alpha_t(i) = P(o_1, o_2, \ldots, o_t, q_t = s_i|\lambda). \tag{6}$$

Then, we have

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i). \tag{7}$$

The value of $\alpha_T(i)$ can be computed inductively as follows:

$$\alpha_1(i) = \pi_i b_i(o_1), \quad 1 \leqslant i \leqslant N, \tag{8}$$

$$\alpha_{t+1}(j) = b_j(o_{t+1}) \sum_{i=1}^{N} \alpha_t(i) a_{ij}, \quad 1 \leqslant j \leqslant N, \ 1 \leqslant t \leqslant T-1. \tag{9}$$

This algorithm requires only $N(N+1)(T-1)+N$ multiplications and $N(N-1)(T-1)$ additions, obtaining a substantial computational improvement.

2.1.2 *Inferring the probability of a state sequence.* In this section, we are interested in finding the state sequence Q which is most likely to explain a given observation sequence O, i.e. which maximizes $P(Q|O,\lambda)$. This problem can be solved by using the well-known 'Viterbi algorithm' (Forney, 1973; Viterbi, 1967) based on dynamic programming.

This is done through the definition of a variable $\delta_t(i)$ that represents the maximum probability of observing $O = o_1, \ldots, o_t$ through the state sequence that terminates in state s_i at time t, given the model λ:

$$\delta_t(i) = \max_{q1,q2,\ldots,qt-1} P(q_1, \ldots, q_t = s_i, o_1, \ldots, o_t|\lambda). \tag{10}$$

Since $\delta_1(j) = b_j(o_1)\pi_j$, we can obtain it recursively from

$$\delta_{t+1}(j) = b_j(o_{t+1}) \max_{1\leqslant i\leqslant N} [\delta_t(i) a_{ij}]. \tag{11}$$

In order to discover the best state sequence, the algorithm needs to keep track, for each time t, of the argument that maximizes (11), i.e.

$$\psi_t(j) = \arg\max_{1\leqslant i\leqslant N} [\delta_{t-1}(i) a_{ij}], \quad 1 \leqslant j \leqslant N. \tag{11a}$$

At time T, the final state is chosen in order to maximize the whole sequence's probability:

$$q_T^* = \arg\max_{1\leqslant i\leqslant N} [\delta_T(i)]. \tag{12b}$$

It is now possible to obtain the most likely state sequence through a backtracking process. At every time t, the best state q_t^* is given by

$$q_t^* = \psi_{t+1}(q_{t+1}^*), \quad t = T-1, T-2, \ldots, 1. \tag{12c}$$

For a more detailed description of the backtracking algorithm, refer to Rabiner (1989).

2.1.3 *The learning problem.* Given a sequence O, the learning process iteratively adjusts HMM parameters to optimize model fitting w.r.t. some criterion (Baldi & Chauvin, 1994). Usually the objective is to find the model $\lambda*$ that maximizes the 'likelihood function' $P(O|\lambda)$.

Learning from examples in HMMs is typically accomplished using a particular adaptation of the EM algorithm known as the Baum–Welch algorithm (Baum *et al.*, 1970), which updates transition and emission parameters a_{ij} and $b_i(O)$ at every iteration, by computing their expected frequencies, given the current model and the observed sequence (training data).

As reported in Rabiner (1989), we define

$$\xi_t(i, j) = P(q_t = s_i, q_{t+1} = s_j | O, \lambda) \tag{13}$$

as the probability of being in state s_i at time t and s_j at $t + 1$ given O and λ and

$$\gamma_t(i) = \sum_{j=i}^{N} \xi_y(i, j) \tag{14}$$

as the probability of being in state s_i at time t. These quantities are used to estimate the model parameters iteratively as follows:

$$\pi_i = \gamma_1(i) = \text{occurrencies of state } s_i \text{ at time } t = 1, \tag{15}$$

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} = \frac{\text{expected number of transitions from state } s_i \text{ to state } s_j}{\text{expected number of transitions from state } s_i}, \tag{16}$$

$$b_j(k) = \frac{\sum_{\substack{t=1 \\ \text{s.t. } Ot=k}}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} = \frac{\text{expected number of observations of } o_k \text{ in state } s_j}{\text{expected number of occurrencies of state } s_j}. \tag{17}$$

In order to compute $\xi_t(i, j)$ and $\gamma_t(i)$, let us define

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \ldots, o_T | q_t = S_i, \lambda) \tag{18}$$

also called 'backward variable', which measures the probability of observing a sequence from $t + 1$ to T, given the current state i. $\beta_t(i)$ is computed recursively as

$$\beta_T(i) = 1, \quad 1 \leqslant i \leqslant N, \tag{19}$$

$$\beta_t(i) = \sum_{j=1}^{N} \beta_{t+1}(j) a_{ij} b_j(o_{t+1}), \quad 1 \leqslant i \leqslant N. \tag{20}$$

Since $\alpha_t(i)\beta_t(i)$ represents the probability of observing the sequence O and to be in the state S_i at the time t, we can write

$$P(O|\lambda) = \sum_{i=1}^{N} P(q_t = i, O|\lambda) = \sum_{i=1}^{N} \alpha_t(i)\beta_t(i), \quad 1 \leqslant t \leqslant T, \tag{21}$$

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{P(O|\lambda)}, \tag{22}$$

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i, j). \tag{23}$$

Note that Baum–Welch algorithm can lead to local optima; therefore, starting with a (any) set of parameters and iteratively re-estimating the parameters, one can improve the likelihood function until some limiting point is reached. For this reason, the choice of the starting set of parameters can be of crucial importance (Liu *et al.*, 2004).

Furthermore, the number of states should be specified in advance. In fact, learning the model topology is a difficult task (Kwong *et al.*, 2001; Abou-Moustafa *et al.*, 2004; Psaradakis & Spagnolo, 2006). In our application, we assume that state number, which represents the number of different time series behaviours, is fixed and that the underlying Markov chain is ergodic. This is because in principle it is always possible for the market to migrate in another state when the economic situation changes.

2.2 *Continuous observations HMM*

In the previous example, we considered cases where observations are discrete symbols chosen from a finite alphabet. This is, by all means, restrictive for many applications. In particular, in financial time series analysis if we use logarithmic returns as the data source, then the observed values are 'continuous'. Although it is possible to discretize these values, this might lead to serious degradation in model reliability.

In continuous observations HMM, $b_j(O)$ are generally represented by a finite mixture of the form

$$b_j(O) = \sum_{m=1}^{M} c_{jm} F[O, \mu_{jm} U_{jm}], \quad 1 \leqslant j \leqslant N, \tag{24}$$

where O is the observation vector, c_{jm} is the mixture coefficient, m identifies the mixture component and F is any log-concave or elliptically symmetric density, usually Gaussian, with mean vector μ and covariance matrix U (Rabiner, 1989). Mixture coefficients satisfy the property

$$\sum_{m=1}^{M} c_{jm} = 1, \quad c_{jm} \geqslant 0, \ \ 1 \leqslant j \leqslant N. \tag{25}$$

It is well-known that returns in finance are not always normally distributed (Bollerslev, 1987) and characteristics, like high kurtosis and fat-tail phenomena, make it inappropriate to approximate the distribution with a single Gaussian function. A multiple-component Gaussian mixture can be a suitable solution for representing this situation.

We need to redefine the re-estimation formulae for the emission probabilities that are now described for each state j by three M-sized vectors containing mixture coefficients $\bar{c}_{jm}$, means $\bar{\mu}_{jm}$ and variances $\bar{U}_{jm}$ of each mixture component m.

Let $\gamma_t(j, m)$ be the probability of being in state S_j at time t with respect to the mth mixture component and observation O_t:

$$\gamma_t(j, m) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}\right] \left[\frac{c_{jm} F[O_t, \mu_{jm} U_{jm}]}{\sum_{k=1}^{M} F[O_t, \mu_{jk} U_{jk}]}\right]. \tag{26}$$

It was proved (Rabiner, 1989) that the reestimation formulas for the unknown parameters are defined as

$$\bar{c}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j, m)}{\sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t(j, m)}, \tag{27}$$

$$\bar{\mu}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j,m) \cdot O_t}{\sum_{t=1}^{T} \gamma_t(j,m)}, \tag{28}$$

$$\bar{U}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j,m) \cdot (O_t - \mu_{jm})(O_t - \mu_{jm})'}{\sum_{t=1}^{T} \gamma_t(j,m)}. \tag{29}$$

Unlike in the discrete observation case, $b_j(O)$ can assume arbitrary high values when the variance of samples in O is near to zero; the likelihood $P(O|\lambda)$ can grow indefinitely and the process never converges. To avoid this problem, in our implementation we followed the approach presented in Ridolfi & Idier (2002) that introduces a 'penalization' function for limiting the likelihood value. The formula becomes

$$P(O|\lambda) = \rho(\sigma) \cdot \sum_{i=1}^{N} \alpha_t(i)\beta_t(i), \quad 1 \leqslant t \leqslant T, \tag{30}$$

where ρ is based on the gamma-inverted distribution g:

$$\rho(\sigma) = \prod_{i,m} g(\sigma_{im}), \tag{31}$$

$$g(\sigma) = \frac{\theta^{\varepsilon}}{\Gamma(\alpha)} \frac{1}{\sigma^{2\varepsilon}} \exp\left(-\frac{\theta}{\sigma^2}\right). \tag{32}$$

Here, $\Gamma(\alpha)$ is the Euler gamma function.

The reestimation formula for the covariance matrix becomes

$$\bar{U}_{jm} = \frac{2\varepsilon + \sum_{t=1}^{T} \gamma_t(j,m) \cdot (O_t - \mu_{jm})(O_t - \mu_{jm})'}{2\theta + \sum_{t=1}^{T} \gamma_t(j,m)}. \tag{33}$$

2.3 *Autoregressive HMM*

A limitation of continuous HMM is that it assumes that the next observation depends on the current state while it is independent from the previous observations. Nevertheless, the local behaviour of a time series, given a state of the system, can often be described by autoregressive models where predictions are linear combination of the previous ones. For this reason, we extended our analysis to ARHMM first introduced by Poritz (1982) and subsequently recalled by Juang & Rabiner (1985). Here subsequent observations are grouped in vectors, which define autoregressive processes given a time series of length K and following the definition of autoregressive process, observations are defined as

$$O_k = \sum_{i=1}^{p} r_i O_{k-i} + e_k, \tag{34}$$

where:

- e_k ($k = 1, 2, \ldots, K - 1$) is a Gaussian i.i.d. (independent identically distributed) random variables with zero mean and variance σ^2, representing the noise component,

- r_i $(i = 1, 2, \ldots, p)$ are the autoregressive coefficients,
- p is the autoregressive process order

This formulation requires a specific definition of the observation pdfs:

$$b_j(O) = \sum_{m=1}^{M} c_{jm} b_{jm}(O), \tag{35}$$

$b_{jm}(O)$ is the observation density for the state j and the mth component, defined by (35), with autoregressive vectors r_{jm} (or equivalently by autocorrelation vector $R_{r_{jm}}$), i.e.

$$b_{jm}(O) = \left(\frac{2\pi}{K}\right)^{-K/2} \exp\left(-\frac{K}{2}\delta(O, r_{jm})\right), \tag{36}$$

where:

- $\delta(O, r_{jm}) = R_{r_{jm}}(0)R(0) + 2\sum_{i=1}^{p} R_{r_{jm}}(i)R(i)$,
- $R_{r_{jm}}(i) = \sum_{n=0}^{p-1} r_{jmn} r_{jm,n+1}$ $(r_{jm0} = 1)$, $1 \leqslant i \leqslant p$, is the autocorrelation of the autoregressive coefficients,
- $R(i) = \sum_{n=0}^{K-i-1} o_n o_{n+1}$, $0 \leqslant i \leqslant p$, is the autocorrelation of the observation samples.

$R_{jm}(i)$ can be estimated recursively by

$$\bar{R}_{jm}(i) = \frac{\sum_{t=1}^{K} \gamma_t(j,m) \cdot R_{jmt}(i)}{\sum_{t=1}^{K} \gamma_t(j,m)} = \frac{\sum_{t=1}^{K} \gamma_t(j,m) \cdot \sum_{n=0}^{t-i-1} o_n o_{n+1}}{\sum_{t=1}^{K} \gamma_t(j,m)}. \tag{37}$$

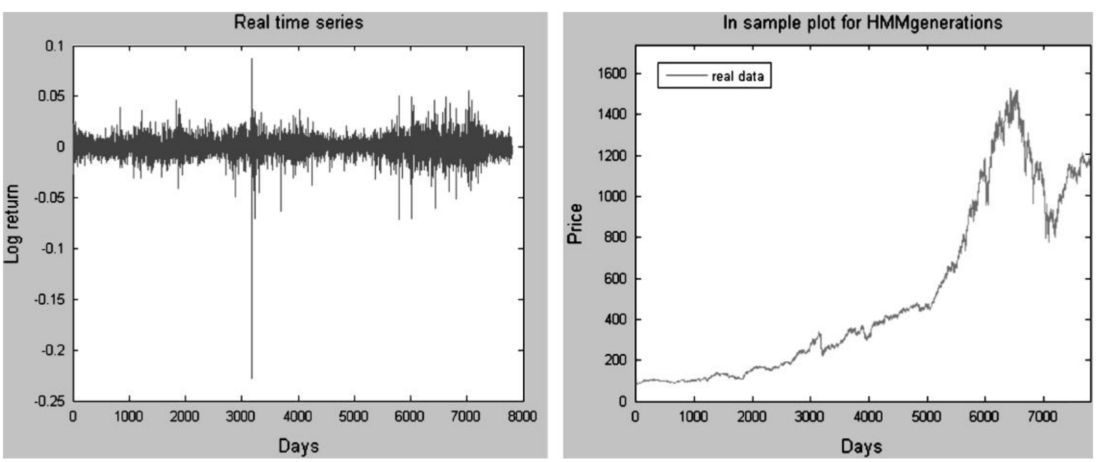

FIG. 2. Plot of S&P 500 log-returns (left) and price values (right). Please note that a colour version of this figure is available as supplementary data at www.imaman.oxfordjournals.org.

TABLE 1 *Moment values for S&P 500*

	DATA	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
Mean	0.006813	0.009961	0.008017	0.007892	0.007906	0.008514	0.008286	0.008421
Variance	0.000099	0.000156	0.000099	0.000099	0.000102	0.000113	0.000106	0.00011
Skewness	−0.157321	0.04997	−0.013809	0.038849	0.078481	−0.098566	−0.130171	0.061771
Kurtosis	1.546801	−0.268547	−0.236051	−0.134649	0.121125	0.006754	−0.02566	−0.203949

TABLE 2 *Percentage errors on mean and variance for S&P 500*

	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
Mean	46.20%	17.68%	15.84%	16.04%	24.96%	21.62%	23.60%
Variance	56.89%	0.48%	0.36%	3.08%	14.11%	6.79%	10.62%
Skewness	131.76%	91.22%	124.69%	149.89%	37.35%	17.26%	139.26%
Kurtosis	117.36%	115.26%	108.71%	92.17%	99.56%	101.66%	113.19%

TABLE 3 *Log-return values corresponding to different quantiles for S&P 500*

Quantile	DATA	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
0.05	−0.015085	−0.019826	−0.015807	−0.016525	−0.01637	−0.016795	−0.016761	−0.016319
0.1	−0.01037	−0.015183	−0.012319	−0.013502	−0.012935	−0.012885	−0.013059	−0.012797
0.25	−0.004574	−0.007964	−0.006443	−0.006749	−0.006672	−0.006149	−0.006701	−0.006681
0.5	0.000114	0.000575	0.000519	0.000023	0.00002	0.000441	0.000184	0.000426
0.75	0.00536	0.008662	0.007609	0.006498	0.006458	0.007781	0.007651	0.008022

TABLE 4 *Percentage errors on quantile for S&P 500*

Quantile	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
0.05	31.43%	4.79%	9.54%	8.52%	11.34%	11.11%	8.18%
0.1	46.41%	18.79%	30.20%	24.73%	24.24%	25.93%	23.40%
0.25	74.09%	40.86%	47.54%	45.85%	34.42%	46.49%	46.05%
0.5	406.17%	356.65%	79.46%	82.22%	288.26%	62.07%	275.19%
0.75	61.61%	41.97%	21.23%	20.49%	45.17%	42.76%	49.67%

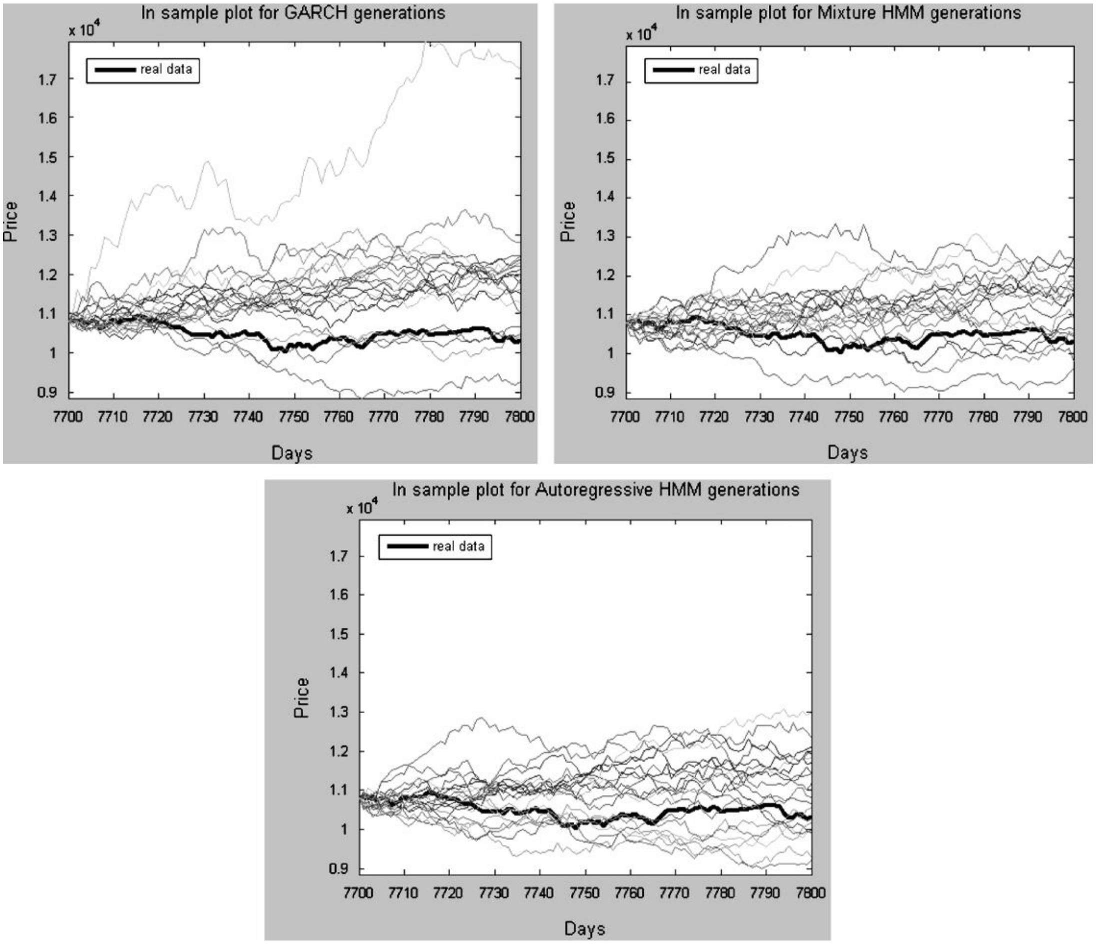

FIG. 3. In-sample plot of paths generated by GARCH, three-state Gaussian mixture HMM and ARHMM3 for S&P 500. Please note that a colour version of this figure is available as supplementary data at www.imaman.oxfordjournals.org.

3. Using HMM for scenario generation

In this section, we investigate the use of HMMs as Monte Carlo scenario simulation models. A scenario generation procedure usually involves three main steps:

- Given the historical observation choose the model assumption that explains their behaviour (for instance, econometric models for interest rates and price indices).
- Estimate the parameters for the chosen model by training it with historical data and eventually subjective judgement.
- Generate data trajectories (paths) according to the chosen distributions discretization model using approximation of statistical properties.

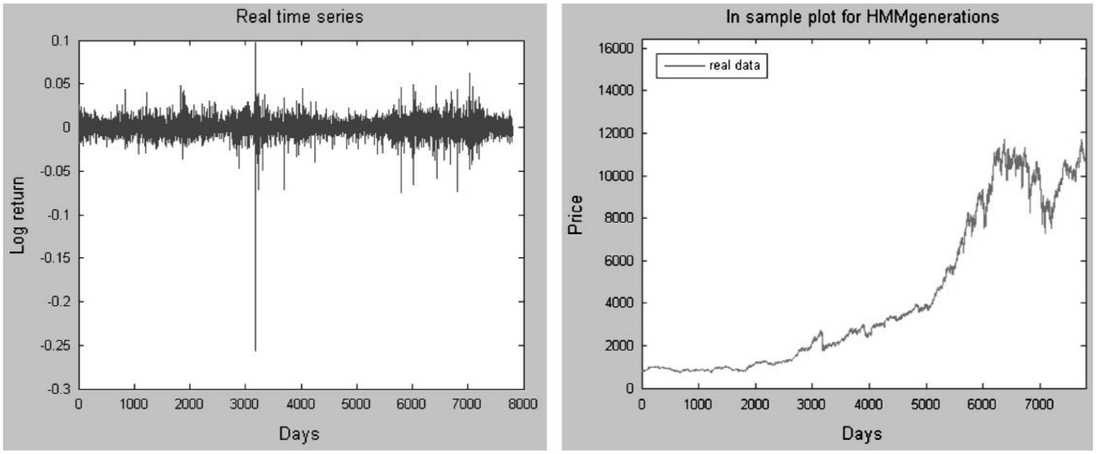

FIG. 4. Plot of DJ log-returns (left) and price values (right). Please note that a colour version of this figure is available as supplementary data at www.imaman.oxfordjournals.org.

As historical data, we used the time series of four different financial indices, as described in Section 3.1. We had to make assumptions on the number of states and the number of mixture components. This choice has been done through empirical evaluation. Of models having different number N of states, as reported in Section 3.1.

The parameter estimation phase has been conducted by performing the learning algorithms described in Section 2. Note that any subjective judgement could be introduced at the end of the training phase by manually setting the parameter values to take into consideration the knowledge about the process.

For data paths generation, we used Monte Carlo simulation: the resulting models have been used to generate out-of-sample paths (scenarios) of length T. At each time step t of the simulation process, with $1 \leqslant t \leqslant T$ and given the current state $q_t = s_i$, we choose the next state $q_{t+1} = s_j$ according to the state transition probability described by matrix A; we then generate the corresponding observation following the distribution b_j, driven by a mixture of Gaussian densities. The initial state for the sample path simulation corresponds to the most likely state at the end of the training process.

In order to integrate randomness in a stochastic programming model and keep it computationally tractable, the distributions of the stochastic parameters have to be approximated by discrete distributions, with a limited number of outcomes (scenario tree) (Kaut & Wallace, 2007; Di Domenica *et al.*, 2007; Mitra, 2006). In Pflug (2001), it is described how it is possible to construct an optimal scenario on the basis of a simulation model of the underlying stochastic process.

In this paper, as a first step towards the validation of HMM-based models for generating scenarios, we show that their Monte Carlo-sampled distributions can replicate with good approximation the empirical distribution of the observed data.

We consider both continuous HMM and ARHMM and we compare the results obtained with the well-known GARCH model. GARCH assumes that the series are heteroskedastic, i.e. variances change with time; these are included in the explanation of future variances (Bollerslev, 1986). It is particularly suitable for financial time series modelling because it takes into account two important characteristics: excess kurtosis (i.e. fat-tail behaviour) and volatility clustering. GARCH models have been applied to many finance-related fields (Bollerslev *et al.*, 1992). A limitation of these models is that they

TABLE 5 *Moment values for DJ*

	DATA	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
Mean	0.006946	0.009152	0.008276	0.00817	0.008296	0.008352	0.008241	0.008648
Variance	0.000104	0.00014	0.000105	0.000107	0.000106	0.000109	0.000107	0.000114
Skewness	−0.145129	−0.032635	0.002253	0.139468	0.040867	0.101024	−0.018989	0.032927
Kurtosis	1.704199	0.091014	−0.20012	0.085751	−0.167557	−0.173627	−0.19416	−0.296651

TABLE 6 *Percentage errors for DJ*

	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
Mean	31.77%	19.16%	17.63%	19.45%	20.25%	18.65%	24.52%
Variance	35.06%	1.50%	2.80%	1.81%	4.61%	2.99%	9.75%
Skewness	77.51%	101.55%	196.10%	128.16%	169.61%	86.92%	122.69%
Kurtosis	94.66%	111.74%	94.97%	109.83%	110.19%	111.39%	117.41%

TABLE 7 *Log-return values corresponding to different quantiles for DJ*

Quantile	DATA	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
0.05	−0.014843	−0.019005	−0.017153	−0.016332	−0.017316	−0.016081	−0.016013	−0.017263
0.1	−0.010435	−0.014445	−0.013547	−0.012344	−0.013399	−0.01169	−0.012546	−0.013189
0.25	−0.004742	−0.007547	−0.007176	−0.006338	−0.006946	−0.00628	−0.006804	−0.006744
0.5	0.000084	−0.000234	−0.000811	0.000348	−0.000281	0.000489	0.000876	0.000863
0.75	0.005499	0.0075	0.007048	0.007127	0.007508	0.008122	0.007468	0.007741

TABLE 8 *Percentage errors on quantiles for DJ*

Quantile	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
0.05	28.04%	15.57%	10.04%	16.66%	8.34%	7.88%	16.31%
0.1	38.42%	29.81%	18.29%	28.40%	12.02%	20.23%	26.39%
0.25	59.15%	51.33%	33.65%	46.47%	32.44%	43.49%	42.21%
0.5	376.80%	1060.49%	311.52%	432.22%	478.80%	937.42%	922.32%
0.75	36.39%	28.16%	29.60%	36.54%	47.70%	35.80%	40.76%

TABLE 9 *Moment values for FTSE100*

	DATA	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
Mean	0.006148	0.008532	0.008347	0.008198	0.008279	0.009412	0.008674	0.008699
Variance	0.000108	0.000114	0.000107	0.000108	0.000106	0.000589	0.000118	0.000119
Skewness	0.169075	−0.008094	−0.097206	0.169294	0.056433	0.282966	−0.018664	0.17802
Kurtosis	8.298811	−0.090937	−0.218095	0.091203	−0.219846	16.359754	−0.191362	0.180095

TABLE 10 *Percentage errors for FTSE100*

	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
Mean	38.77%	35.75%	33.34%	34.65%	53.08%	41.07%	41.49%
Variance	5.05%	0.86%	0.25%	2.27%	445.40%	8.88%	10.40%
Skewness	104.79%	157.49%	0.13%	66.62%	67.36%	111.04%	5.29%
Kurtosis	101.10%	102.63%	98.90%	102.65%	97.13%	102.31%	97.83%

TABLE 11 *Log-return values corresponding to different quantiles for FTSE100*

Quantile	DATA	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
0.05	−0.014322	−0.017373	−0.017361	−0.017211	−0.018269	−0.017674	−0.017468	−0.016626
0.1	−0.009696	−0.01363	−0.013701	−0.013518	−0.01438	−0.013962	−0.013233	−0.012893
0.25	−0.003496	−0.006858	−0.007082	−0.007229	−0.007432	−0.007396	−0.007434	−0.007066
0.5	0.00001	0.001112	0.000265	−0.000327	−0.000019	−0.000159	−0.000926	0.000343
0.75	0.004522	0.007681	0.007515	0.00616	0.007474	0.006848	0.006674	0.008217

TABLE 12 *Percentage errors for FTSE100*

Quantile	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
0.05	21.30%	21.21%	20.17%	27.56%	23.40%	21.96%	16.08%
0.1	40.58%	41.31%	39.42%	48.31%	44.00%	36.48%	32.98%
0.25	96.15%	102.53%	106.75%	112.55%	111.53%	112.63%	102.09%
0.5	11020.00%	2550.00%	3370.00%	290.00%	1690.00%	9360.00%	3330.00%
0.75	69.86%	66.20%	36.24%	65.29%	51.45%	47.60%	81.72%

TABLE 13 *Moment values for FTSE350*

	DATA	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
Mean	0.006789	0.006422	0.007883	0.007499	0.007649	0.008139	0.008426	0.008344
Variance	0.000093	0.000069	0.000096	0.000091	0.000094	0.000101	0.00011	0.000108
Skewness	−0.173328	0.046324	0.082706	0.161758	0.021836	0.108989	0.073746	0.056099
Kurtosis	1.057306	0.407916	−0.201965	0.115746	−0.041733	−0.302194	−0.108362	−0.228675

TABLE 14 *Percentage errors for FTSE350*

	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
Mean	5.41%	16.11%	10.45%	12.67%	19.88%	24.11%	22.90%
Variance	26.01%	2.48%	2.75%	0.16%	8.51%	17.71%	15.66%
Skewness	126.73%	147.72%	193.32%	112.60%	162.88%	142.55%	132.37%
Kurtosis	61.42%	119.10%	89.05%	103.95%	128.58%	110.25%	121.63%

TABLE 15 *Log-return values corresponding to different quantiles for FTSE350*

Quantile	DATA	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
0.05	−0.014362	−0.01318	−0.015774	−0.01558	−0.016141	−0.016064	−0.016201	−0.015667
0.1	−0.010357	−0.009715	−0.01301	−0.012093	−0.012595	−0.012461	−0.012593	−0.013022
0.25	−0.004625	−0.004602	−0.006728	−0.006163	−0.006176	−0.006931	−0.006316	−0.006856
0.5	0.000321	0.000803	−0.000214	0.000074	0.000003	−0.000289	0.000428	0.000568
0.75	0.005491	0.005595	0.006613	0.006048	0.00665	0.007235	0.008393	0.007789

TABLE 16 *Percentage errors on quantiles for FTSE350*

Quantile	GARCH	ARHMM2	ARHMM3	ARHMM4	MIXT_HMM2	MIXT_HMM3	MIXT_HMM4
0.05	8.23%	9.84%	8.49%	12.39%	11.86%	12.81%	9.09%
0.1	6.20%	25.61%	16.76%	21.60%	20.30%	21.58%	25.72%
0.25	0.50%	45.46%	33.24%	33.51%	49.86%	36.56%	48.23%
0.5	150.10%	166.63%	77.05%	99.09%	190.01%	33.20%	76.79%
0.75	1.90%	20.43%	10.15%	21.10%	31.77%	52.85%	41.85%

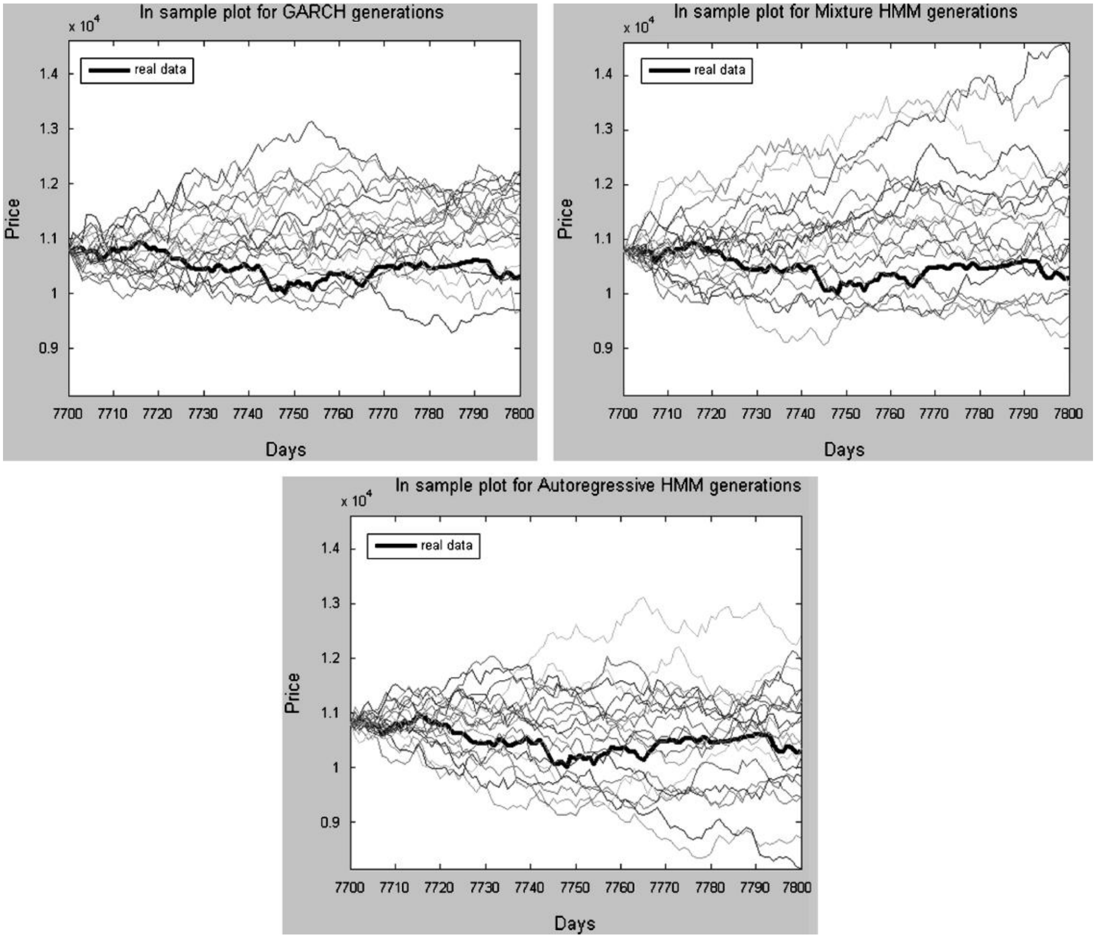

FIG. 5. In-sample plot of paths generated by GARCH, three-state Gaussian mixture HMM and ARHMM3 for DJ. Please note that a colour version of this figure is available as supplementary data at www.imaman.oxfordjournals.org.

are not able to capture highly irregular phenomena, including wild market fluctuations (e.g. crashes and subsequent rebounds) and other highly unanticipated events that can lead to have significant structural change (Gourieroux, 1997).

3.1 *Experimental results*

In order to validate our approach based on HMM, we considered four different time series related to the following financial indexes:

- Standard & Poor's 500 (S&P 500): from 11 August 1975 to 11 August 2005 (7829 days).
- Dow Jones (DJ): from 11 August 1975 to 11 August 2005 (7829 days).
- FTSE350: from 31 December 1985 to 11 August 2005 (5118 days).
- FTSE100: from 31 January 1978 to 11 August 2005 (7444 days).

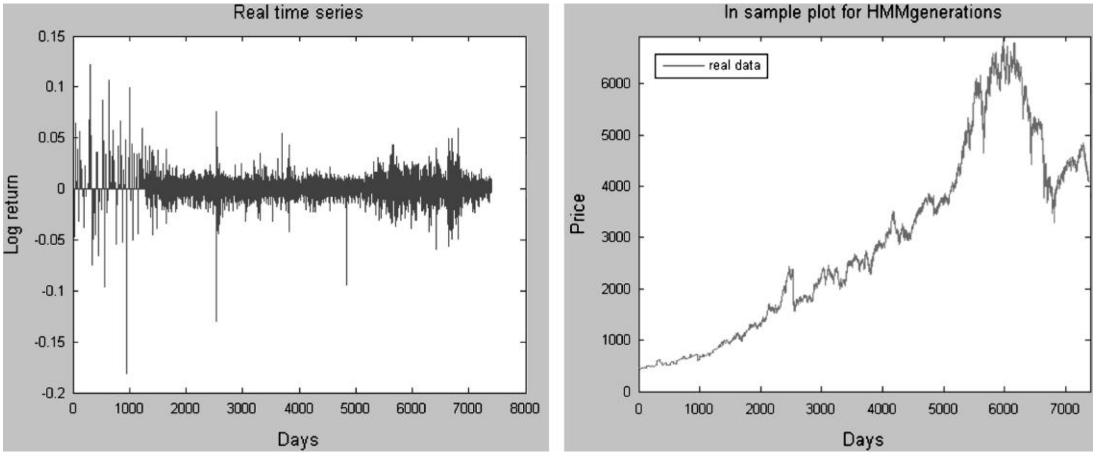

FIG. 6. Plot of FTSE100 log-returns (left) and price values (right). Please note that a colour version of this figure is available as supplementary data at www.imaman.oxfordjournals.org.

We used these data for learning the parameters of the following models:

- GARCH: a GARCH(1, 1) model.
- MIXT_HMM2, MIXT_HMM3, MIXT_HMM4: respectively, two-, three- and four-state HMMs whose continuous observations are modelled by a four-component mixture of Gaussian functions.
- ARHMM2, ARHMM3 and ARHMM4: ARHMMs with, respectively, two, three and four states and continuous observations modelled by a four-component mixture of Gaussians, in which it is defined as an autoregressive process of size three.

In order to compute model performances, we considered the fit of the sampled distribution, obtained through the Monte Carlo out-of-sample simulation of the learned models, with the empirical distribution of the historical data. In particular, we considered the following statistical indicators: mean, variance, skewness, kurtosis and the 0.05, 0.1, 0.25, 0.5 and 0.75 quantiles.

Since Baum–Welch algorithm can find only local maxima, as mentioned in Section 2, we repeated the training process 10 times for each series, starting from different initial values, and we considered the estimated set of parameters having the highest likelihood function value.

For the out-of-sample simulation, we generated 200 log-return sequences, each of them constituted by 100 observed values.

Moreover, we performed in-sample simulation by excluding the last set of data from the training set and comparing the behaviour predicted by the model learned on the reduced training set with the real one.

We now describe the results obtained for each single time series.

3.1.1 *Standard & Poor's 500.* Figure 2 shows log-returns plot (left) and price values (right) for the 'S&P 500' index . We see (on the left) that returns initially have a medium volatility but after two extreme high and low peaks, they fall in a low-volatility period. After day 5500, they traverse a period of high fluctuations. This behaviour seems to suggest that a three-state HMM, characterized by low,

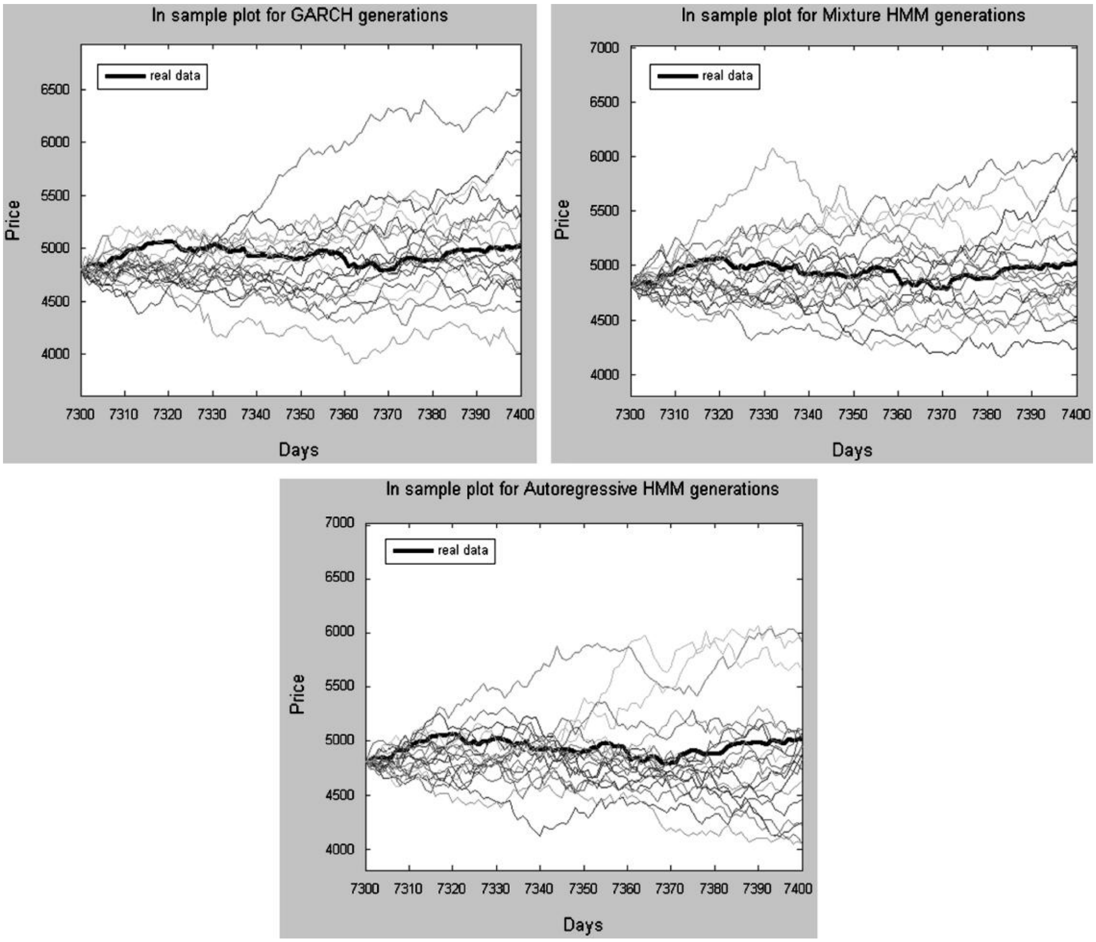

FIG. 7. In-sample plot of paths generated by GARCH, three-state Gaussian mixture HMM and ARHMM3 for FTSE100. Please note that a colour version of this figure is available as supplementary data at www.imaman.oxfordjournals.org.

medium and high volatility, could be suitable for describing this process. In order to validate this belief, we performed experiments with two, three and four states.

Table 1 shows the first four moments of the sampled data obtained by varying the number of HMM states; percentage errors are reported in Table 2. Table 3 shows quantiles of the same data, whose percentage errors are found in Table 4. ARHMM3 (with 3 states) (Table 2) has low error on all moments, except for skewness coefficient for which the Gaussian mixture HMM has a lower value. In Table 4, we see that ARHMM3 better approximates higher distribution values, while ARHMM4 obtains lower errors on 0.05, 0.1 and 0.25 quantiles.

Figure 3 shows the price plots obtained by in-sample simulation with GARCH, MIXT_HMM3 and ARHMM3. We generated log-returns and, starting from the asset price at the last day of the training period, we calculated the corresponding prices. Sampled price paths are plotted together with the last 100 values of the index prices (the darkest line), which have of course been excluded from the training set of the learning process. For better readability, in each plot we reported only 20 randomly generated paths.

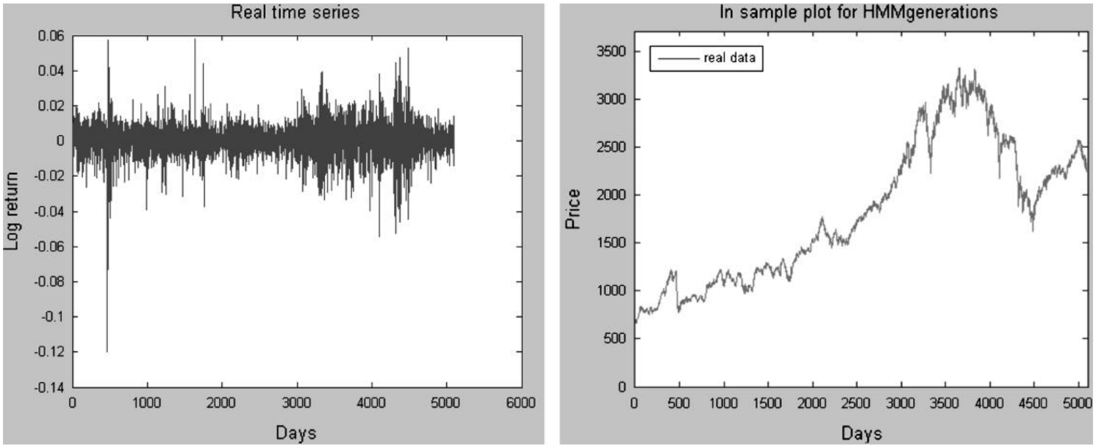

FIG. 8. Plot of FTSE350 log-returns (left) and price values (right). Please note that a colour version of this figure is available as supplementary data at www.imaman.oxfordjournals.org.

3.1.2 *Dow Jones.* In Fig. 4, we see that the 'DJ' index has a behaviour that is very similar to S&P 500: an initial medium-volatility period is followed by few days of extreme values and then a relatively stable period. From day 5000, prices have greater volatility with long positive and negative trend periods.

As reported in Tables 5 and 6, ARHMM3 shows lower error on the first two moments and a good approximation of the kurtosis. Skewness is better captured by GARCH. As seen for S&P 500, if compared with other models ARHMM3 has relatively low error on all quantiles (see Tables 7 and 8), showing good capabilities of modelling the true data distribution.

In Fig. 5, we show price plots obtained by performing in-sample simulation. On the left, GARCH shows the tendency to overestimate prices, generating paths that are distributed mainly above the real values. This characteristic is observed also for the Gaussian mixture HMM paths. ARHMM3 better distributes values around real paths, considering also the possibility of value decrease.

3.1.3 *FTSE100.* Figure 6 depicts log-returns and the price of 'FTSE100' index. Once more, the three-state model seems to be a semantically reasonable choice: a first high fluctuation period is followed by a relatively smooth period and by a final volatility increase. On the right, the visual analysis of rough price values shows a constant, small positive trend followed by alternation of high value increase and decrease.

ARHMM3 has almost null error for variance and skewness, it also gives the lowest error on the mean and a good approximation of kurtosis (see Tables 9 and 10). The quantile analysis (see Tables 11 and 12) shows that ARHMM3 produces generally the lowest error with the only exception of quantile 0.5.

The graphical presentation of price forecasts shows a quite similar behaviour for the three models (see Figure 7).

3.1.4 *FTSE350.* In Fig. 8, we can observe logarithmic returns and prices of the FTSE350 index. In price values, we find a rapid increase with a final long phase of high fluctuation.

Here, the Gaussian mixture HMM obtains the worst results in moment forecasting (see Tables 13 and 14), while ARHMM3 and ARHMM4 compete for the lowest error: the first has good performances for mean and kurtosis and the latter produces a more realistic skewness, with almost null error on variance.

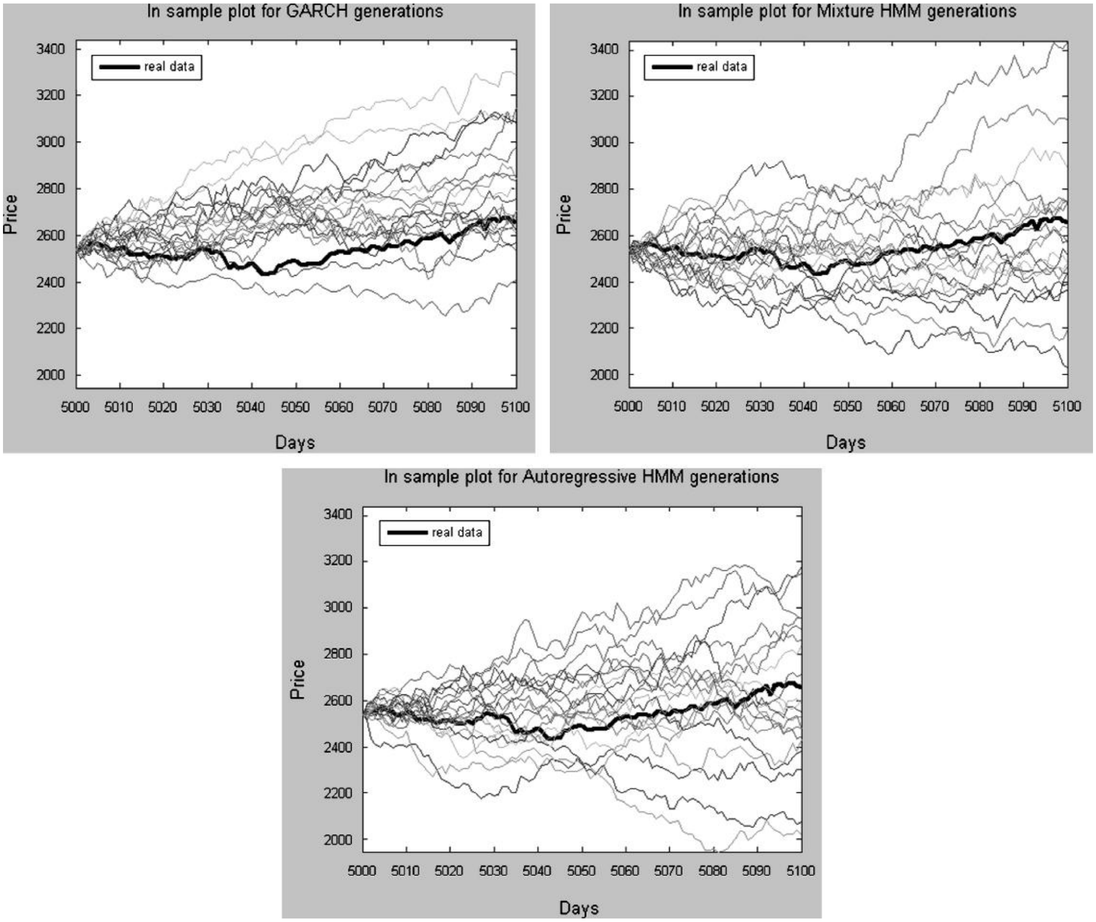

FIG. 9. In-sample plot of paths generated by GARCH, three-state Gaussian mixture HMM and ARHMM3 for FTSE350. Please note that a colour version of this figure is available as supplementary data at www.imaman.oxfordjournals.org.

In quantile analysis (see Tables 15 and 16), GARCH is undoubtedly the best model, even if the Gaussian mixture HMM has a very low error on 0.5 quantile forecast.

Looking at the in-sample simulation paths in Fig. 9, we see that GARCH produces high price values that overestimate real data. Three-state Gaussian mixture HMM better centres paths around real data but with some excessively low or high forecasts. ARHMM3, again, shows to be able to generate smoother and well-distributed paths.

4. Conclusions and future work

In this paper, we proposed an alternative to classical approaches of time series modelling for scenario generation, based on HMM, which are able to capture the dynamic of the underlying time series using a state-space model. The advantage is that, since data are generated by a process that we cannot directly observe or that it is too complex and involves too many variables, during the generation process we focus our attention only on the observation values, which is our final objective. The proposed approach is entirely data driven and it does not make any assumption on the models underlying the system states.

The proposed model is based on ARHMM in order to take into account the inter-time dependencies between subsequent data. The preliminary results obtained on single-valued time series seem to be promising compared with both GARCH and simple continuous HMM-based models although more investigation needs to be carried out in order to determine the advantages of HMM-based scenario generation models when included in the decision process.

Future work should consider efficient procedures for automatically determining the optimal number of different states to be included in the model and the extension of the model to the case of time-varying transition probabilities between states.

References

Abou-Moustafa, K. T., Cheriet, M. & Suen, C. Y. (2004) On the structure of hidden Markov models. *Pattern Recognit. Lett.*, **25**, 923–931.

Alexander, C. (2001) *Market Models*. John Wiley & Sons, New York.

Baldi, P. & Chauvin, Y. (1994) Smooth on-line learning algorithms for hidden Markov models. *Neural Comput.*, **6**, 305–316.

Baum, L. E. & Petrie, T. (1966) Statistical inference for probabilistic functions of finite state Markov chains. *Ann. Math. Stat.*, **37**, 1559–1563.

Baum, L. E., Petrie, T., Soules, G. & Weiss, N. (1970) A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Stat.*, **41**, 164–171.

Bengio, Y. & Frasconi, P. (1996) Input-output HMM's for sequence processing. *IEEE Trans. Neural Netw.*, **7**, 1231–1249.

Bengio, Y., Lauzon, V. P. & Ducharme, R. (2001) Experiments on the application of IOHMMs to model financial returns series. *IEEE Trans. Neural Netw.*, **12**, 113–123.

Bollerslev, T. (1986) Generalized autoregressive conditional heteroskedasticity. *J. Econom.*, **31**, 307–327.

Bollerslev, T. (1987) A conditionally heteroskedastic time series model for speculative prices and rates of return. *Review Econ. Stat.*, **69**, 542–547.

Bollerslev, T., Chou, R. Y. & Kroner, K. F. (1992) ARCH modeling in finance: a review of the theory and empirical evidence. *J. Econom.*, **52**, 5–59.

Charniak, E. (1993) *Statistical Language Learning*. Cambridge, MA: MIT Press.

Di Domenica, N., Birbilis, G., Mitra, G. & Valente, P. (2007) Stochastic programming and scenario generation within a simulation framework: an information systems perspective. *Decis. Support Syst.*, **42**, 2197–2218.

Forney, G. D. (1973) The Viterbi algorithm. *Proc. IEEE*, **61**, 268–278.

Ge, X. & Smyth, P. (2000) Deformable Markov model templates for time-series pattern matching. *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (W. Kim & I. Parsa eds). New York, NY: ACM Press, pp. 81–90.

González, A. M., Muñoz San Roque, A. & García-González, J. (2005) Modeling and forecasting electricity prices with input/output hidden Markov models. *IEEE Trans. Power Syst.*, **20**, 13–24.

Gourieroux, C. (1997) *ARCH Models and Financial Applications*. Berlin: Springer.

Hamilton, J. D. (1989) A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, **57**, 357.

Hamilton, J. D. & Susmel, R. (1994) Autoregressive conditional heteroskedasticity and changes in regime. *J. Econom.*, **64**, 307–333.

HANNAFORD, B. & LEE, P. (1990) Multi-dimensional hidden Markov model of telemanipulation tasks with varying outcomes. *Proceedings of the IEEE International Conference on Systems Man and Cybernetics, 4–7 November 1990*. Los Angeles, CA: IEEE, pp. 127–133.

HARDY, M. R. (2001) A regime-switching model of long-term stock returns. *North Am. Actuar. J.*, **5**, 41.

HUNKAPILLER, T., BALDI, P., CHAUVIN, Y. & MCCLURE, M. A. (1994) Hidden Markov models of biological primary sequence information. *Proc. Natl. Acad. Sci. USA*, **91**, 1059–1063.

JUANG, B. H. & RABINER, L. R. (1985) Mixture autoregressive hidden Markov models for speech signals. *IEEE Trans. Acoust. Speech Signal Process*, **33**, 1404.

KAUT, M. & WALLACE, S. W. (2007) Evaluation of scenario generation methods for stochastic programming. *Pac. J. Optim.*, **3**, 257–271.

KWONG, S., CHAU, C. W., MAN, K. F. & TANG, K. S. (2001) Optimization of HMM topology and its model parameters by genetic algorithms. *Pattern Recognit.*, **34**, 509–522.

LIU, N., DAVIS, R. I. A., LOVELL, B. C. & KOOTSOOKOS, P. J. (2004) Effect of initial HMM choices in multiple sequence training for gesture recognition. *Int. Conf. Inf. Technol.*, **1**, 608–613.

MITRA, S. (2006) *Scenario Generation for Stochastic Programming*. Optyrisk white paper series, OPT004. London, UK: OPTIRISK Systems.

PFLUG, G. CH. (2001) Scenario tree generation for multiperiod financial optimization by optimal discretization. *Math. Program. Ser. B*, **89**, 251–257.

PORITZ, A. B. (1982) Linear predictive hidden Markov models and the speech signal. *Proc. ICASSP*, **1**, 1291–1294.

PSARADAKIS, Z. & SPAGNOLO, N. (2006) Joint determination of the state dimension and autoregressive order for models with Markov regime switching. *J. Time Ser. Anal.*, **27**, 753–766.

RABINER, L. R. (1989) A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, **77**, 257–286.

RIDOLFI, A. & IDIER, J. (2002) Penalized maximum likelihood estimation for normal mixture distributions. *School of Computer and Information Sciences Technical Report 2002 85*. Lausanne, Switzerland: École Polytechnique Fédérale de Lausanne, EPFL, School of Computer and Information Sciences.

SHI, S. & WEIGEND, A. S. (1997) Taking time seriously: hidden Markov experts applied to financial engineering. *Proceedings of the IEEE/IAFE 1997 Conference on Computational Intelligence for Financial Engineering (CIFEr'97, New York, March 1997)*. New York, NY: IEEE, pp. 244–252.

SMYTH, P., HACKERMAN, D. & JORDAN, M. I. (1997) Probabilistic independence networks for hidden Markov probability models. *Neural Comput.*, **9**, 227–269.

VITERBI, A. J. (1967) Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Trans. Inf. Theory*, **13**, 260–269.

ZHANG, Y. (2004) Prediction of financial time series with hidden markov models. *M.Sc. Thesis*, Simon Fraser University, China.