Recursive estimation of autoregressive models with Markov-switching

Joseph Rynkiewicz
SAMOS, Université Paris I - Panthéon Sorbonne
Paris, France
rynkiewi@univ-paris1.fr

October 3, 2001

Abstract

Autoregressive models with Markov switching are useful to model piecewise stationary time series, in the way that a hidden Markov chain governs the time dependent distribution of an observed stochastic process. Since this chain is hidden, common approach to the recursive estimation problem is to use algorithms suitable for missing data like recursive E.M. algorithm or suboptimal Kalman filtering techniques. However, recent maximum likelihood method based on the gradient have shown better properties of convergence in the framework of standard hidden Markov model. In this paper, we generalise and improve this approach for Markov switching autoregressive models.

Keywords. Switching autoregression, Markov regime, maximum likelihood estimator; recursive algorithm.

1 Introduction

Dynamical systems which alternate between different dynamics are useful description of time-varying situations. Hidden Markov Model (HMM) have been widely applied for modelling such processes. Although original Hidden Markov Models use mean instead of regression for the structure of the time series, the generalisation is straightforward and these models where, for example, used in econometry by Hamilton ([5]). The main method for estimate the parameter of such model are based on the maximum likelihood estimator (MLE), the non-recursive estimator have been widely studied in pratice or in theory, see Rabiner [14] for a tutorial, Leroux [10] , Bickel, Ritov and Rydén [1] for statistical properties of the MLE in the case of original HMM, Krishnamurthy and Rydén [8], Douc, Moulines and Rydén [3] for the statistic properties of MLE in the case of autoregressive models with Markov switching. In this paper we deal with recursive estimation of stationary switching autoregressions. There are two major methods for recursive estimation of a HMM model. The first one, is the recursive E.M. algorithm as in Krishnamurthy and Moore [7] or in Holst et al. [6], the second method is to use a modification of kalman filtering technique as in chapter 6 of Elliott et al. [4] or Millnert [12]. However, recently, Collings and Rydén [2] have studied maximum likelihood method based on gradient algorithm technique. This method has shown better properties of convergences and seems to be near to the optimal asymptotic properties , indeed the author shows that their algorithm is better than the kalman technique especially in low noise conditions. In this paper we generalise their algorithm

in the framework of autoregressive models. Moreover we improve their implementation in the following ways, first we use a parametrisation which avoid to use constrained optimisation for the transition matrix of the hidden Markov chain; second we simplify greatly the calculus of the derivative of the log-likelihood thanks the use of the predictive filter of the hidden state ; and finally we explicit entirely the calculus in the fundamental case of multivariate regression models with Gaussian noise. The paper is organised as follow. Switching autoregressive models are introduced in Section 2. In Section 3, we show how to compute the log-likelihood and it's derivative in a general framework then in the case of multivariate Gaussian regression models. In Section 4 we deduce from the previous section the recursive algorithm. The section 5 is devoted to numerical examples and comparison with the recursive E.M. scheme.

2 Switching autoregressive models

Consider the process $(X_t, Y_t)_{t\in\mathbb{Z}}$, such that

1. $(X_t)_{t\in\mathbb{Z}}$ is a Markov chain in a finite state space $\mathbb{E} = \{e_1, ..., e_N\}$, which can be identified without loss of generality with the simplex of $\mathbb{R}^N$, where e_i is the unit vector in $\mathbb{R}^N$ with unity as the ith element and zeros elsewhere.

2. Given $(X_t)_{t\in\mathbb{Z}}$, the process $(Y_t)_{t\in\mathbb{Z}}$ is a sequence of autoregressive models of known order p in $\mathbb{R}^d$, so the distribution of Y_n depends only on X_n and $Y_{n-1}, \cdots, Y_{n-p}$.

For a fixed t , the dynamic of the model is :

$$Y_{t+1} = F_{X_{t+1}}(Y_{t-p+1}, \cdots, Y_t) + \varepsilon_{t+1}^{X_{t+1}}$$

with $F_{X_{t+1}} \in \{F_{e_1}, ..., F_{e_N}\}$ continuous derivable functions and for each $e_i \in \mathbb{E}$, $(\varepsilon_t^{e_i})_{t\in\mathbb{N}^*}$ is a i.i.d sequence of random variables of $\mathbb{R}^d$, note that the two sequence $(\varepsilon_t^{e_i})_{t\in\mathbb{N}^*}$ and $\left(\varepsilon_t^{e_j}\right)_{t\in\mathbb{N}^*}$ are independent if $i \neq j$.

Now we will introduce an state space model notation as in Elliott et al. [4].

If we write $\mathcal{F}_t = \sigma\{X_0, \cdots, X_t\}$, for the σ-field generated by $X_0, \cdots, X_t$., the Markov property implies that

$$P(X_{t+1} = e_i \,|\mathcal{F}_t) = P(X_{t+1} = e_i \,|X_t).$$

Write

$$a_{ij} = P(X_{t+1} = e_i \,|X_t = e_j) \text{ and } A = (a_{ij}) \in \mathbb{R}^{N\times N}$$

and define the martingale increment :

$$V_{t+1} := X_{t+1} - E[X_{t+1} \,|\mathcal{F}_t] = X_{t+1} - AX_t.$$

With the previous notations, we obtain the general equation of the model, for $t \in \mathbb{N}$:

$$\begin{cases} X_{t+1} = AX_t + V_{t+1} \\ Y_{t+1} = F_{X_{t+1}}(Y_{t-p+1}, \cdots, Y_t) + \varepsilon_{X_{t+1}} \end{cases} \tag{1}$$

Note that it is possible to have different known orders for the individual autoregressive processes. Then p should be interpreted as the maximum of the individual orders, but for simplicity of notation we have used a common order p.

Sufficient conditions for stationary solution of such model can be found in Yao and Attali [16]. It is worth to note that some autoregressive process might result individually in an unstable system but the whole switching process may be stabilised when allowed to switch with a Markovian regime.

3 The log-likelihood function and it's derivative

3.1 Model parameters

The parameters to be estimated in the model are :

- The coefficients (a_{ij}) of the transition matrix A
- The parameter vectors $(\Sigma_{e_i})_{1\leq i\leq N}$ of the noise
- The parameter vectors $(\omega_{e_i})_{1\leq i\leq N}$ of regression functions $(F_{e_i})_{1\leq i\leq N}$.

The parameter vector $\theta \in \mathbb{R}^D$, where D is the dimension of θ, denote the concatenation of all parameter vectors, i.e. the complete vector to be estimated.

The properties of a recursive estimator can be highly dependent on the parametrisation of the model. In our model the transition matrix A is stochastic, the sum of a any column of A is 1, so we have $N-1$ free parameters for each column. To deal with this constraint, we write $\upsilon_{ij} = \ln \frac{a_{ij}}{a_{Nj}}$, note that $\upsilon_{Nj} = 0$, and $(\upsilon_{1j}, \cdots, \upsilon_{N-1,j}) \in \mathbb{R}^{N-1}$. This parametrisation yields us to optimise the matrix A without constrained optimisation. If $a_{ij} = 1$ or $a_{ij} = 0$ then $\nu_{ij} = \pm\infty$ and a consistent estimator should tend to $\pm\infty$ respectively.

hence, if we note A_j the jth column of A, we have :

$$A_j = \left(\frac{e^{\upsilon_{ij}}}{1 + e^{\upsilon_{1j}} + \cdots + e^{\upsilon_{N-1j}}}\right)_{1\leq i\leq N}$$

Thus, we deduce the calculus the partial derivatives of A with respect of the parameters υ_{ij}:

$$\frac{\partial a_{ij}}{\partial \upsilon_{ij}} = \frac{\partial}{\partial \upsilon_{ij}} \frac{e^{\upsilon_{ij}}}{1 + e^{\upsilon_{1j}} + \cdots + e^{\upsilon_{N-1j}}} = \frac{e^{\upsilon_{ij}}}{1 + e^{\upsilon_{1j}} + ... + e^{\upsilon_{N-1j}}} \left(1 - \frac{e^{\upsilon_{ij}}}{1 + e^{\upsilon_{1j}} + \cdots + e^{\upsilon_{N-1j}}}\right)$$

then

$$\frac{\partial a_{ij}}{\partial \upsilon_{ij}} = a_{ij}(1 - a_{ij}) \tag{2}$$

and for $l \neq i$

$$\frac{\partial a_{ij}}{\partial \upsilon_{lj}} = \frac{e^{\upsilon_{ij}}}{1 + \cdots + e^{\upsilon_{N-1j}}} \times \left(-\frac{e^{\upsilon_{lj}}}{1 + \cdots + e^{\upsilon_{N-1j}}}\right) == -a_{ij}a_{lj} \tag{3}$$

Moreover, if $k \neq j$:

$$\frac{\partial a_{ij}}{\partial \upsilon_{lk}} = 0$$

3.2 The log-likelihood function

In order to build our recursive algorithm we begin to calculate the log-likelihood for constant parameter θ et observation $(y_{-1}, \cdots, y_n)$. Let $L_\theta(y_1, \cdots, y_n)$ be the log-likelihood conditionally to the first p observation $y_{-p+1}, \cdots, y_0$ and the initial state X_1. We have

$$L_\theta(y_1, ..., y_n) = L_\theta(y_n \,|y_1, \cdots, y_{n-1}) \times \prod_{t=1}^{n-1} L_\theta(y_t \,|y_1, \cdots, y_{t-1})$$

$$= \sum_{i=1}^{N} L_\theta(y_n \,|X_n = e_i, y_1, \cdots, y_{n-1}) P_\theta(X_n = e_i \,|y_1, \cdots, y_{n-1})$$

$$\times \prod_{t=1}^{n-1} L_\theta(y_t \,|y_1, \cdots, y_{t-1})$$

Note, in the sequel

- p_t^θ the vector with i-th components : $p_t^\theta(i) = P_\theta(X_t = e_i \,|y_1, \cdots, y_{t-1},)$, p_t^θ is known as the predictive filter of X_n.
- b_t^θ the vector with i-th components : $b_t^\theta(i) = L_\theta(y_t \,|X_t = e_i, y_1, \cdots, y_{t-1})$, the conditional density of y_t knowing $X_t = e_i$ and $(y_1, \cdots, y_{t-1})$.
- $B_t^\theta = diag\left(b_t^\theta\right)$ the matrix with b_t^θ for diagonal and zeros elsewhere.

The log-likelihood is then

$$\ln(L_\theta(y_1, \cdots, y_n)) = \sum_{t=1}^{n} \ln({b_t^\theta}^T p_t^\theta) \tag{4}$$

Where the upperscript T denote the transposition.

Note the aditive form of this formule. Moreover, the predictive filter p_t^θ verifies a "Baum-like" recursion, since a straightforward adaptation of Legland and Mevel [9] shows that it verifies the recurrence :

$$p_{t+1}^\theta = \frac{A B_t^\theta p_t^\theta}{{b_t^\theta}^T p_t^\theta} \tag{5}$$

Moreover these author show that the choice of the initial condition p_1 don't really influence the value of $L_\theta(y_1, \cdots, y_n)$ because of the exponential forgetting properties. So, we will suppose that p_1^θ is the uniform distribution on E and we can recursively calculate p_t^θ, $t = 1, \cdots n$.

3.3 Derivative of the log-likelihood

Let θ_j be the j-th component of θ, we have :

$$\frac{\partial \ln(L_\theta(y_1, \cdots, y_n))}{\partial \theta_j} = \sum_{t=1}^{n} \frac{\frac{\partial {b_t^\theta}^T p_t^\theta}{\partial \theta_j}}{{b_t^\theta}^T p_t^\theta}$$

with

$$\frac{\partial b_t^{\theta T} p_t^\theta}{\partial \theta_j} = \frac{\partial b_t^{\theta\, T}}{\partial \theta_j}\, p_t^\theta + b_t^{\theta T} \frac{\partial p_t^\theta}{\partial \theta_j} \tag{6}$$

The calculus of $\frac{\partial b_t^\theta}{\partial \theta_j}$ depend of the model and is generally easy if we know the derivative of the regression function and the density of the noise with respect to their parameters. We will explicit this derivative for the Gaussian case in the next section.

The calculus of $\frac{\partial p_t^\theta}{\partial \theta_j}$ with respect to θ_j can be obtained by derivating the relation (5) with respect to parameter θ_j :

$$\frac{\partial p_{t+1}^\theta}{\partial \theta_j} = \frac{\partial A B_t^\theta p_t^\theta}{\partial \theta_j} \times \frac{1}{b_t^{\theta T} p_t} + A B_t^\theta p_t^\theta \times \frac{\partial b_t^{\theta T} p_t^\theta}{\partial \theta_j} \times \left(-\frac{1}{(b_t^{\theta T} p_t^\theta)^2} \right)$$

so

$$\frac{\partial p_{t+1}^\theta}{\partial \theta_j} = \left(\frac{\partial A B_t^\theta}{\partial \theta_j} p_t^\theta + A B_t^\theta \frac{\partial p_t^\theta}{\partial \theta_j} \right) \times \frac{1}{b_t^{\theta T} p_t^\theta} + A B_t^\theta p_t^\theta \times \left(\frac{\partial b_t^{\theta T}}{\partial \theta_j} p_t^\theta + b_t^{\theta T} \frac{\partial p_t^\theta}{\partial \theta_j} \right) \times \left(-\frac{1}{(b_t^{\theta T} p_t^\theta)^2} \right).$$

Then, we have :

$$\frac{\partial p_{t+1}^\theta}{\partial \theta_j} = \frac{A B_t^\theta}{b_t^\theta p_t^\theta} \left[I - \frac{p_t^\theta b_t^{\theta T}}{b_t^{\theta T} p_t^\theta} \right] \frac{\partial p_t^\theta}{\partial \theta_j} + \left(\frac{\partial A B_t^\theta}{\partial \theta_j} \right) \frac{p_t^\theta}{b_t^{\theta T} p_t^\theta} - \frac{A B_t^\theta p_t^\theta}{(b_t^{\theta T} p_t^\theta)^2} \left(\frac{\partial b_t^{\theta T}}{\partial \theta_j} p_t^\theta \right)$$

and

$$\frac{\partial p_{t+1}^\theta}{\partial \theta_j} = \frac{A B_t^\theta}{b_t^{\theta T} p_t^\theta} \left[I - \frac{p_t^\theta b_t^{\theta T}}{b_t^{\theta T} p_t^\theta} \right] \frac{\partial p_t^\theta}{\partial \theta_j} + \left(\frac{\partial A}{\partial \theta_j} B_t^\theta + A \frac{\partial B_t^\theta}{\partial \theta_j} \right) \frac{p_t^\theta}{b_t^\theta p_t^\theta} - \frac{A B_t^\theta p_t^\theta}{(b_t^{\theta T} p_t^\theta)^2} \left(\frac{\partial b_t^{\theta T}}{\partial \theta_j} p_t^\theta \right) \tag{7}$$

with $\frac{\partial p_1}{\partial \theta_j} = 0$ for all j.

The calculus of $\frac{\partial A}{\partial \theta_j}$ is easy since if θ_j is an element belonging to $(\nu_{ij})_{1 \le i,j \le N-1}$ say ν_{lm}, we have thanks equation (2) and (3) :

$$\frac{\partial A}{\partial \theta_j} = C(v_{lm})$$

with $C(v_{lm})$ a matrix with only the mth column C_m non null and with :

$$\begin{cases} C_m(i) = -a_{im} a_{lm} & \text{if } \ i \neq l \\ C_m(i) = a_{lm}(1 - a_{lm}) & \text{if } \ i = l \end{cases} \tag{8}$$

3.4 The multivariate Gaussian case

In this section we explicit the calculus of the derivative $\frac{\partial b_t^\theta}{\partial \theta_j}$ for the multidimensional Gaussian case, we introduce first the parametrisation of the parameter of the noise (the covariance matrix) and the regression functions.

3.4.1 Parametrisation in the Gaussian case

Without loss of generality, for simplify the notation, we will suppose in the sequel that $p = 1$, the generalisation to $p > 1$ is straightforward.

Parametrisation of the covariances matrix To be sure that the likelihood is well defined we have to suppose that the covariances matrices Σ_{e_i} are positive definite, so we can consider the coefficients of it inverse $\Sigma_{e_i}^{-1}$ as parameters, moreover we considere only the coefficient upper of the diagonal (diagonal include) because the matrix is symmetric.

Parametrisation of the regressions functions We suppose only that the functions F_{e_i}, $1 \leq i \leq N$ are continuesly derivable with respect of is parameter vector ω_{e_i} . So F_{e_i} can be classical linear function but also more complicated function like multilayer perceptrons.

the calculus of b_t^θ is then easy since

$$b_t^\theta(i) = L_\theta(y_t \,|X_t = e_i, y_{t-1}, \cdots, y_1)$$

$$= \frac{1}{\sqrt{2\pi^d \det(\Sigma_{e_i})}} \exp\left(-\frac{1}{2}\left((y_t - F_{e_i}(y_{t-1}))^T \Sigma_{e_i}^{-1}(y_t - F_{e_i}(y_{t-1}))\right)\right)$$

is the conditional density of y_t knowing $X_t = e_i$ and $(y_1, \cdots, y_{n-1})$.

3.4.2 Calculus of $\frac{\partial b_t^\theta}{\partial \theta_j}$

Since $b_t^\theta(i)$ is never null, we can use the formulas :

$$\frac{\partial b_t^\theta(i)}{\partial \theta_j} = b_t^\theta(i) \times \frac{\partial \ln\left(b_t^\theta(i)\right)}{\partial \theta_j}$$

because the derivative of the logarithm of $b_t^\theta(i)$ is easy to calculate. Now, we recall some classic formulas which can be found in Magnus and Neudecker [11]:

- If A (with coefficients a_{ij}) is a constant matrix and X a matrix with coefficients x_{ij} :

$$\frac{\partial}{\partial x_{ij}} Tr(AX) = a_{ji} \tag{9}$$

- Let $X(\theta)$ be a paramtrized invertible matrix, note X^{-1} it's inverse, if θ_j is one component of θ, we have:

$$\frac{\partial}{\partial \theta_j} \ln(\det(X)) = tr(X^{-1} \frac{\partial}{\partial \theta_j} X) \tag{10}$$

- If A, B, C are three matrix with convenient size, the trace of their product is invariant by circular permutation :

$$Tr(ABC) = Tr(BCA) = Tr(CAB) \tag{11}$$

apply the formulas (11) and (9) give :

$$\frac{\partial(Tr((y_t - F_{e_i}(y_{t-1}))^T \Sigma_{e_i}^{-1}(y_t - F_{e_i}(y_{t-1}))))}{\partial\theta_j} = \left((y_t - F_{e_i}(y_{t-1}))(y_t - F_{e_i}(y_{t-1}))^T\right)_{kl}$$

and the formula (10) give (remember that we take only the coefficient upper of the diagonal) :

$$\frac{\partial \ln(\det(\Sigma_{e_i}^{-1}))}{\partial\theta_j} = (\Sigma_{e_i})_{kl},\ if\ k = l$$
$$\frac{\partial \ln(\det(\Sigma_{e_i}^{-1}))}{\partial\theta_j} = 2 \times (\Sigma_{e_i})_{kl},\ if\ k \neq l$$

Finally the ith element of u_i is:

$$u_i(i) = b_t^\theta(i) \times \frac{1}{2}\left((\Sigma_{e_i})_{kl} - \left((y_t - F_{e_i}(y_{t-1}))(y_t - F_{e_i}(y_{t-1}))^T\right)_{kl}\right) \tag{12}$$

if $k = l$, and

$$u_i(i) = b_t^\theta(i) \times \left((\Sigma_{e_i})_{kl} - \left((y_t - F_{e_i}(y_{t-1}))(y_t - F_{e_i}(y_{t-1}))^T\right)_{kl}\right) \tag{13}$$

if $k \neq l$.

If θ_j is a coefficient of the matrix $\Sigma_{e_i}^{-1}$: $\theta_j = (\Sigma_{e_i}^{-1})_{kl}$

$$\frac{\partial b_t^\theta}{\partial\theta_j} = u_i$$

where u_i is a vector of $\mathbb{R}^N$ with all coordinate null, except the ith which is :

$$u_i(i) = b_t^\theta(i) \times \frac{\partial[-\frac{1}{2}(d\ln(2\pi) - \ln(\det(\Sigma_{e_i}^{-1})) + Tr((y_t - F_{e_i}(y_{t-1}))^T \Sigma_{e_i}^{-1}(y_t - F_{e_i}(y_{t-1}))]}{\partial\theta_j} \tag{14}$$

If θ_j is a parameter of the regression function F_{e_i} Applying formula (10), give us :

$$\frac{\partial b_t^\theta(i)}{\partial\theta_j}$$

$$= b_t^\theta(i) \times \sum_{1 \leq m,l \leq d} (\Sigma_{e_i}^{-1})_{lm} \left((F_{e_i}(y_{t-1}) - y_t)(l)\frac{\partial F_{e_i}(y_{t-1})(m)}{\partial\theta_j} + (F_{e_i}(y_{t-1}) - y_t)(m)\frac{\partial F_{e_i}(y_{t-1})(l)}{\partial\theta_j}\right)$$

Now by collecting these derivatives with the formulas (7), (8), we obtain the expression of the derivative of the log-likelihood. The application to the recursive estimation is treated in the next section.

4 Recursive estimation

4.1 Recursive maximum likelihood estimation

A recursive estimator θ_{n+1} of the parameter θ based on the first $n+1$ observations of $(y_t)_{t\in\mathbb{N}^*}$ is of the form :

$$\theta_{n+1} = \theta_n + \gamma_n H_n h(y_{n+1}, \theta_n)$$

where h(y,θ) is a score vector function, the matrix H_n is an adaptative matrix and γ_n is a gain sequence satisfying

$$\gamma_n \leq 0, \sum_{n=1}^{\infty} \gamma_n = \infty, \sum_{n=1}^{\infty} \gamma_n^2 < \infty \tag{15}$$

For independent observations with density $f(y,\theta)$, the score function is

$$h(y,\theta) = \left\{ \frac{\partial \ln\left(f(y,\theta)\right)}{\partial \theta_i}, 1 \leq i \leq D \right\}$$

and an optimal choice of H_n is the inverse of the information matrix, i.e. $H_n^{-1} = I(\theta_n)$, where

$$I\left(\theta\right) = E\left[h(y,\theta)h(y,\theta)^T\right]$$

Computation of this information matrix often requires numerical integration and it is thus cumbersome. So we shall instead use the inverse of the observed information matrix, i. e.

$$H_n^{-1} = \frac{1}{n}\sum_{k=1}^{n} h(y_k, \theta_{k-1})h(y_k, \theta_{k-1})^T$$

The matrix H_n can be computed recursively by means of the matrix inversion lemma, writing $h_n = h(y_n, \theta_{n-1})$, we have

$$H_n = \frac{1}{1-\gamma_n}\left(H_{n-1} - \frac{\gamma_n H_{n-1} h_n h_n^T H_{n-1}}{(1-\gamma_n) + \gamma_n h_n^T H_n h_n}\right)$$

4.2 Recursive maximum likelihood estimation in switching autoregression with Markov regime

In our case the observation are not i.i.d., however the log-likelihood has an additive form similar to log-likelihood for the i.i.d. case. So, we have

$$\begin{cases} \theta_{n+1} = \theta_n + \gamma_n H_n h_{n+1} \\ H_n = \frac{1}{1-\gamma_n}\left(H_{n-1} - \frac{\gamma_n H_{n-1} h_n h_n^T H_{n-1}}{(1-\gamma_n)+\gamma_n h_n^T H_n h_n}\right) \end{cases}$$

with h_n is the gradient vector so that the jth coordinate is :

$$h_{n+1}(j) = \left(\frac{\partial b_{n+1}^{\theta_n}}{\partial \theta_n^j}\right)^T p_{n+1}^{\theta_n} + \left(b_{n+1}^{\theta_n}\right)^T \frac{\partial p_{n+1}^{\theta_n}}{\partial \theta_n^j}$$

where

$$p_{n+1}^{\theta_n} = \frac{A_n B_n^{\theta_n} p_n^{\theta_{n-1}}}{{b_n^{\theta_n}}^T p_n^{\theta_{n-1}}}$$

and

$$\frac{\partial p_{n+1}^{\theta_n}}{\partial \theta_n^j}$$

$$= \frac{A_n B_n^{\theta_n}}{{b_n^{\theta_n}}^T p_n^{\theta_{n-1}}} \left[I - \frac{p_n^{\theta_{n-1}} {b_n^{\theta_n}}^T}{{b_n^{\theta_n}}^T p_n^{\theta_n}} \right] \frac{\partial p_n^{\theta_{n-1}}}{\partial \theta_n^j} + \left(\frac{\partial A_n}{\partial \theta_n^j} B_n^{\theta_n} + A_n \frac{\partial B_n^{\theta_n}}{\partial \theta_n^j} \right) \frac{p_n^{\theta_{n-1}}}{b_n^{\theta_n} p_n^{\theta_{n-1}}} - \frac{A_n B_n^{\theta_n} p_n^{\theta_{n-1}}}{({b_n^{\theta_n}}^T p_n^{\theta_{n-1}})^2} \left(\frac{\partial b_n^{\theta_n}}{\partial \theta_n^j} p_n^{\theta_{n-1}} \right)$$

The conditions for consistency and asymptotic normality of these involved procedure are in general still open questions even for independently and identically distributed (i.i.d.) observations. Note that we are not in a Robbins-Monroe framework mainly because the vector $p_{n+1}^{\theta_n}$ is not the update of $p_n^{\theta_n}$ but the update of $p_n^{\theta_{n-1}}$, so the parameter vary at each update (we can do the same remark for the derivative $\frac{\partial p_{n+1}^{\theta_n}}{\partial \theta_n^j}$). However we will see that this algorithm seems to works very well on simulated data.

5 Simulations

In this section we compare the performance of our algorithm on the examples proposed by Holst [6] to study a recursive E.M. algorithm. The model are very simple it's consist only on a two regime switching Markov regression, with AR1 regressive models. Simulation on multivariate models with multilayer perceptron as autoregressive function can be found in Rynkiewicz [15].

5.1 The models

The test examples are the following :

Example 1

$$\left\{ \begin{array}{c} F_{e_1}(y) = 1.5 - 0.70y \\ F_{e_2}(y) = 0 \end{array} \right\}$$

Example 2

$$\left\{ \begin{array}{c} F_{e_1}(y) = 1.5 - 0.70y \\ F_{e_2}(y) = 1.7 - 0.72y \end{array} \right\}$$

In both cases the variances $\sigma_{e_1} = \sigma_{e_2} = 1$ and the transitions matrix is

$$A = \left(\begin{array}{cc} 0.9 & 0.1 \\ 0.1 & 0.9 \end{array} \right)$$

The first example is considered as an easy problem and the second one as a difficult one since the information matrix of the model is nearly singular. The problem with the second model is the proximity of the parameters of the two regression which implies great difficulties in the estimation of coefficients in the transition matrix. To keep the same conditions than Holst et al. we have simulated series of 10000 data. The number of replicates for the two experiences was 1000.

5.2 The estimation

The following precautions were used for the estimation task.

The choice of initial values Contrary to Holst et al. [6], we don't use a priori knowledge about the true value of the parameters., since this knowledge is almost always impossible to have in real situations. So, we choose randomly (with an uniform law between -1 and 1) the initial value of the regression function.

The initial value for the transition matrix was in all examples the neutral matrix

$$A_0 = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}$$

To avoid to introduce a priori knowledge in our initial parameter we chose to initialise the variances with $\sigma_{e_1} = \sigma_{e_2} = 2$.

The initial value of the information matrix H_0 is the identity. In order to avoid inconsistency the estimate of σ_{e_i} were supervised and projected such than $\sigma_{e_i} > 0$. Moreover, the random initial values lead us to do two pass on the data for estimating the parameter. Indeed 10000 observations isn't a very long series for recursive estimation which is used in general for very long time series.

Note that the second example is a more difficult task, so we do 10 random initialization of the parameter for each estimation and we keep the estimation with the best likelihood.

Increased step size and averaging The gain sequence γ_n scales the update of both H_n and θ_n. Apart from satisfying the restriction (15), it can be any function. Generally it has the form $\gamma_n = \frac{\gamma_0}{n^l}$, $l \in \mathbb{R}$. With the choice $\gamma_n = \frac{\gamma_0}{n}$, γ_n tends to become too small too quickly and does not allow fast convergence for initial estimates chosen far from the minimum error point. To overcome this problem Polyak and Juditsky [13] suggest a method for applying a larger step size and then averaging the estimate. Averaging is used to get a smoother estimate as the larger step will mean higher sensitivity to noise, and also to ensure that the requirement 15 remains satisfied. In our simulations we chose $l = 0.5$.

5.3 Result

Holst et al. use a very expensive initialisation of the Hessian matrix thanks the estimation of the information matrix on 100000 simulated data. But, this method seems to be contradictory with the use of only 10000 data for the estimation task. Moreover we don't know what will be the behaviour of their algorithm if the preliminary initialisation (and so the initialisation of the Hessian) is poor, in contrast we always use the identity matrix to initialise the Hessian.

The results are presented in Tables I-II which give observed means and standard deviations of the final estimates.

The main remark about these results is that our seems to be a little bit better than those of Holst et al. Indeed, in the first example the bias and the variances of is better for all parameter. In the second example the situation is the same except for the a_{12} parameter (the last line).

Hence, the RMLE algorithm gives us a method to achieve very good results without using unrealistic a priori knowledge about the true parameter of the model.

Table 1: Parameter estimates in Example 1

Param.	Start (R.E.M.)	Mean (R.E.M.)	Std (R.E.M.)	Start (R.M.L.E.)	Mean (R.M.L.E.)	Std (R.M.L.E.)
$F^1_{e_1} = 1.5$	1.0	1.497	0.025	random	1.499	0.024
$F^2_{e_1} = -0.7$	-0.5	-0.697	0.022	random	-0.699	0.012
$\sigma_{e_1} = 1$	2	1.006	0.017	2	1.003	0.020
$F^1_{e_2} = 0$	0.2	-0.000	0.016	random	-0.000	0.016
$F^2_{e_2} = 0$	0.2	0.002	0.032	random	-0.000	0.018
$\sigma_{e_2} = 1$	2	1.010	0.175	2	1.003	0.024
$a_{11} = 0.9$	0.5	0.898	0.008	0.5	0.899	0.007
$a_{12} = 0.1$	0.5	0.101	0.015	0.5	0.101	0.007

Table 2: Parameter estimates in Example 2

Param.	Start (R.E.M.)	Mean (R.E.M.)	Std (R.E.M.)	Start (R.M.L.E.)	Mean (R.M.L.E.)	Std (R.M.L.E.)
$F^1_{e_1} = 1.5$	1.0	1.457	0.070	random	1.472	0.065
$F^2_{e_1} = -0.7$	-0.5	-0.679	0.033	random	-0.667	0.020
$\sigma_{e_1} = 1$	2	0.984	0.114	2	0.987	0.091
$F^1_{e_2} = 1.7$	1.5	1.675	0.067	random	1.700	0.065
$F^2_{e_2} = -0.72$	-0.56	-0.720	0.039	random	-0.730	0.035
$\sigma_{e_2} = 1$	2	1.077	0.620	2	0.997	0.240
$a_{11} = 0.9$	0.5	0.819	0.056	0.5	0.860	0.046
$a_{12} = 0.1$	0.5	0.104	0.065	0.5	0.105	0.072

Finally, the RMLE algorithm seems to be the state of art algorithm in recursive estimation of switching Markov autoregressions. This results is unsurprising since most of consistent recursive likelihood algorithms with suitable scaling Hessian matrix are asymptotically efficient.

Acknowledgements

The author is grateful to Jian-Feng Yao for helpful comment on this work.

References

[1] P. J. Bickel, Y. Ritov, and T. Rydén. Asymptotic normality of the maximum-likelihood estimator for general hidden Markov models. *The Annals of Statistics*, 26:4:1614–1635, 1998.

[2] I.B. Collings and T. Rydén. A new maximum likelihood gradient algorithm for on-line hidden Markov model identification. In *Proc. Int. Conf. on Acoustics, Speech and Sig. Proc.*, volume IV, pages 2261–2264, 1998.

[3] R. Douc and E.and Rydén T. Moulines. Asymptotic properties of the maximum likelihood estimator in autoregressive models with Markov regimes. Technical reports 9, University of Lund, 2001.

[4] R. J. Elliott, L. Aggoun, and J. B. Moore. *Hidden Markov models : estimation and control.* Springer, 1997.

[5] J. D. Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57:357–384, 1989.

[6] U. Holst, G. Lindgren, J. Holst, and M. Thuvesholmen. Recursive estimation in Switching autoregressions with a Markov regime . *Journal of time series analysis*, 77:257–287, 1994.

[7] V Krishnamurthy and John B. Moore. On-line estimation of hidden Markov model parameters based on the Kullback-Leibler information mesure. *IEEE transaction on signal processing*, 41:8:2557–2573, 1993.

[8] V. Krishnamurthy and T. Rydén. Consistent estimation of linear and non-linear autoregressive models with Markov regime. *Journal of time series analysis*, 19:3:291–307, 1998.

[9] F. LeGLand and L. Mevel. Exponential forgetting and geometric ergodicity in Hidden Markov Models. *Mathematics of control, Signal, and Systems*, 13:63–93, 2000.

[10] B. G. Leroux. Maximum-likelihood estimation for hidden Markov models. *Stochastic Process. Appl.*, 40:127–143, 1992.

[11] J.R. Magnus and H. Neudecker. *Matrix Differential Calculus with Applications in Statistics and Econometrics.* Wiley Series in Probability and Statistics. John Wiley and sons, 1999.

[12] M. Millnert. Identification of ARX models with Markovian parameters. *Int. J. Control*, 45:2045–2058, 1987.

[13] B.T. Polyak and A.B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on control and optimization*, 30(4):838–855, 1992.

[14] L.R. Rabiner. A tutorial on hidden Markov models and selected application in speech application. *Proceedings of the IEEE*, 77:257–287, 1993.

[15] J. Rynkiewicz. Estimation of Hybrid HMM/MLP models. In *ESANN'2001*, 2001.

[16] J. Yao and J.G. Attali. On stability of nonlinear AR processes with Markov switching. *Adv. Applied Probab.*, 32:394–407, 2000.