# ALGORITHMS FOR ESTIMATION OF POSSIBLY NONSTATIONARY VECTOR TIME SERIES

GUOFU ZHOU

*Washington University*

*First version received January 1991*

**Abstract**. This paper presents efficient algorithms for evaluating the likelihood function and its gradient of possibly nonstationary vector autoregressive moving-average (VARMA) processes.

**Keywords**. Forecasting; likelihood function; innovation algorithm; Whittle algorithm; Bayesian analysis.

## 1. INTRODUCTION

Time series models are powerful tools in the study and understanding of economic dynamics, and yet there are still many gaps between theory and practice, one of which is the lack of efficient algorithms for implementing the necessary computations in the estimation of the parameters in a vector time series model. This is particularly true for possibly nonstationary models which have been of enormous interest in recent economic studies for at least two reasons. First, it is believed that the nonstationarity (unit-root) question is closely linked to important questions in economic theory. Second, econometric inferences could be misleading if this question is ignored (e.g. Nelson and Kang, 1981, 1984). New approaches and tests for nonstationary models are developed by Dickey and Fuller (1979), Chan (1988), Phillips (1988) and Priestley (1988), among others.

Since no analytical formulae are available for the maximum likelihood estimators in a possibly nonstationary vector autoregressive moving-average (VARMA) model, an iterative procedure has to be used in practice. This inevitably requires many computations of the likelihood function and its gradient. Therefore it is of importance to develop efficient algorithms to solve this problem. In addition, such algorithms are useful in Bayesian analysis. For example, as shown by Zhou (1990), the Monte Carlo integration approach is a solution to the computational problem of posterior densities which are analytically intractable, but to use this approach it is necessary to evaluate the likelihood function thousands of times.

The paper is organized as follows. In Section 2 we synthesize the prediction problem of a stochastic process because forecasts of time series have wide applications. Both the innovation algorithm of Brockwell and Davis (1987, 1988) and Whittle's algorithm (Whittle, 1963; Morf *et al.*, 1978) are

0143-9782/92/02 171–18 $02.50/0
JOURNAL OF TIME SERIES ANALYSIS Vol. 13, No. 2

suggested for evaluating the best linear predictors. In Section 3, these algorithms are developed to evaluate the likelihood function and its gradient efficiently. In Section 4 a few remarks conclude the paper.

## 2. PREDICTIONS

Since forecasts of time series are of general interest, we discuss in some detail the 'forward' and 'backward' best linear predictors. Both the algorithm of Brockwell and Davis (1987, pp. 412–13) and the Whittle algorithm are provided to compute them efficiently. Since these two algorithms will be the foundations of the results in the next section, a comparison of their merits is made. However, it should be pointed out at the outset that there are important alternative algorithms for the computation of predictions, e.g. the lattice algorithms (Friedlander, 1982) and the methods of Kailath based on approximation by combinations of Toeplitz forms. These algorithms are not reviewed here because it seems that further studies are required to apply them to our present case of evaluation of the likelihood function and its gradient.

Let $\{\boldsymbol{Y}_t : t = 0, \pm 1, \pm 2, \ldots\}$ be an $m$-dimensional random process defined on some probability space $(\Omega, \mathcal{F}, P)$. Without loss of generality, we assume that it has zero mean. We further assume that it has finite variance and covariances. Notice that, unless explicitly stated otherwise, the process $\boldsymbol{Y}_t$ is neither necessarily Gaussian nor stationary.

Consider the problem of predicting $\boldsymbol{Y}_{n+1}$ conditional on $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$. There are, presumably, many ways to construct various type of predictors based on different criteria. However, the best linear predictor is the most popular one. This may be due, at least theoretically, to its tractability. It is defined as a linear function of $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$,

$$\hat{\boldsymbol{Y}}_{n+1} = \boldsymbol{\Phi}_{n,1}\boldsymbol{Y}_n + \ldots + \boldsymbol{\Phi}_{n,n}\boldsymbol{Y}_1, \tag{1}$$

where the coefficient matrices $\boldsymbol{\Phi}_{i,j}$ are chosen to minimize the mean-squared error:

$$\min_{\boldsymbol{\Phi}_{n,1},\ldots,\boldsymbol{\Phi}_{n,n}} E(\boldsymbol{Y}_{n+1} - \hat{\boldsymbol{Y}}_{n+1})'\boldsymbol{Q}(\boldsymbol{Y}_{n+1} - \hat{\boldsymbol{Y}}_{n+1}) \tag{2}$$

where $E$ is the expectation operator and $\boldsymbol{Q}$ is a given $m \times m$ positive definite matrix. Thus, $\hat{\boldsymbol{Y}}_{n+1}$ is best in the sense that it has the least mean-squared error. Notice that this error is weighted by the matrix $\boldsymbol{Q}$ which reflects the importance of forecasting error. For example, if $\boldsymbol{Q}$ is chosen to be diagonal with the first diagonal element 10 and the rest 1, then the forecast of the first element of $\boldsymbol{Y}_{n+1}$ is measured as 10 times more important than the rest in minimizing the mean-squared error. It is clear that the prediction with a different $\boldsymbol{Q}$ will be different. Following the usual practice, we assume throughout that $\boldsymbol{Q} = \boldsymbol{I}$, the identity matrix. This will not lose any generality

of the theory, because the coefficient matrices for forecasting a series $\{\boldsymbol{Z}_t\}$ with a general positive definite matrix $\boldsymbol{Q}$ can be obtained by a transformation of the $\boldsymbol{\Phi}_{i,j}$ with $\boldsymbol{Q} = \boldsymbol{I}$. To see this, let $\boldsymbol{Y}_t = \boldsymbol{R}\boldsymbol{Z}_t$ where $\boldsymbol{R}$ is a nonsingular matrix such that $\boldsymbol{Q} = \boldsymbol{R}'\boldsymbol{R}$. The minimization problem

$$\min_{\boldsymbol{\Phi}_{n,1},\ldots,\boldsymbol{\Phi}_{n,n}} E(\boldsymbol{Z}_{n+1} - \hat{\boldsymbol{Z}}_{n+1})'\boldsymbol{Q}(\boldsymbol{Z}_{n+1} - \hat{\boldsymbol{Z}}_{n+1})$$

is then

$$\min_{\boldsymbol{\Phi}_{n,1},\ldots,\boldsymbol{\Phi}_{n,n}} E(\boldsymbol{Y}_{n+1} - \hat{\boldsymbol{Y}}_{n+1})'(\boldsymbol{Y}_{n+1} - \hat{\boldsymbol{Y}}_{n+1}),$$

which is the same as (2) with $\boldsymbol{Q} = \boldsymbol{I}$. It follows from (1) that $\boldsymbol{R}^{-1}\boldsymbol{\Phi}_{n,1}\boldsymbol{R}, \ldots, \boldsymbol{R}^{-1}\boldsymbol{\Phi}_{n,n}\boldsymbol{R}$ are the coefficient matrices for obtaining the best linear forecast of $\boldsymbol{Z}_{n+1}$.

An alternative representation of the best linear predictor is the innovation form

$$\hat{\boldsymbol{Y}}_{n+1} = \boldsymbol{\Theta}_{n,1}(\boldsymbol{Y}_n - \hat{\boldsymbol{Y}}_n) + \quad + \boldsymbol{\Theta}_{n,n}(\boldsymbol{Y}_1 - \hat{\boldsymbol{Y}}_1), \tag{3}$$

which expresses $\hat{\boldsymbol{Y}}_{n+1}$ as a linear function of the innovations $\boldsymbol{Y}_k - \hat{\boldsymbol{Y}}_k$ $(k = 1, \ldots, n)$. There are at least two advantages of considering predictors of this form. One is that the innovation algorithm (discussed later) is suitable for computing $\hat{\boldsymbol{Y}}_{n+1}$ in this way. The second advantage, as we shall see in Section 3, is that it has a close relationship to the fact that a density function can be written as a product of conditional densities. It is this role that makes the innovation algorithm very useful.

The predictor $\hat{\boldsymbol{Y}}_{n+1}$ is a forecast of the future obtained by using the information of the past. Thus it may be called a 'forward' predictor. In contrast, the backward predictor is predicting 'backwards' in the sense that it predicts the value of $\boldsymbol{Y}_1$ conditional on the observations of $\boldsymbol{Y}_{n+1}, \ldots, \boldsymbol{Y}_2$. Notice that no special properties of $\{\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n\}$ have been used in the definition of $\hat{\boldsymbol{Y}}_{n+1}$ other than the fact are random variables. Therefore, by replacing $\{\boldsymbol{Y}$ the backward predictor can be similarly defined and written as

$$\tilde{\boldsymbol{Y}}_1 = \tilde{\boldsymbol{\Phi}}_{n,1}\boldsymbol{Y}_2 + \cdots \quad \tilde{\boldsymbol{\Phi}}_{n,n}\boldsymbol{Y}_{n+1}. \tag{4}$$

The usefulness of this type of predictor will be clarified later.

From the perspective of projections in the Hilbert space $\mathcal{L}^2(\Omega, \mathcal{F}, P)$, the $i$th element of $\hat{\boldsymbol{Y}}_{n+1}$ is exactly the usual projection of $\boldsymbol{Y}^i_{n+1}$ on the closed subspace generated by the elements of $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$. Hence, the existence of the forward predictors or the existence of the coefficient matrices is guaranteed by the standard prediction theorems in a Hilbert space, but the uniqueness of the coefficient matrices can only be assured by further assuming the nonsingularity of the $nm \times nm$ covariance matrix $\boldsymbol{\Gamma} \equiv (K(i, j))$, where $K(i, j) \equiv E\boldsymbol{Y}_i\boldsymbol{Y}_j'$, $m \times m$. This follows from Brockwell and Davis (1987), especially Chapters 2, 5 and 11. Similar assertions are also valid for the backward predictors. The nonsingularity assumption is equivalent to the

requirement that none of the elements of $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_{n+1}$ is an exact linear combination of the rest. It thus excludes redundancy of the information in the forecast. Since it is trivially satisfied in applications, we henceforth make this assumption throughout so that there will be no confusion about obtaining possibly different coefficient matrices. Therefore the remaining key issue is to present computationally efficient ways to obtain these unique matrices.

Before taking up the computation problem, we provide an example of the predictors which are familiar to econometricians. Let the process be Gaussian and put $n = t$. The forward prediction given by (1) is then the familiar conditional expectation, $\hat{\boldsymbol{Y}}_{t+1} = E(\boldsymbol{Y}_{t+1} | \boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_t)$. In a univariate case, it is the well-known best mean-square linear estimation of $\boldsymbol{Y}_{t+1}$ conditional on the available information up to time $t$. In a rational expectation economy, the coefficient matrices in (1) reflect the weights of past information on agents' expectations, while those in (3) catch the effects of 'surprises' on the expectations. This example also illustrates that both the forward and backward predictions can be of interest to econometricians.

Of much practical value is the innovation algorithm that evaluates the $\boldsymbol{\Theta}_{ij}$s in (3). Let $\boldsymbol{V}_{n-1}$ be the covariance matrix of the prediction errors, i.e.

$$\boldsymbol{V}_{n-1} \equiv E(\boldsymbol{Y}_n - \hat{\boldsymbol{Y}}_n)(\boldsymbol{Y}_n - \hat{\boldsymbol{Y}}_n)' \tag{5}$$

Then we have the following algorithm.

INNOVATION ALGORITHM. If $\{\boldsymbol{Y}_t\}$ has zero mean and finite covariances $K(i, j) = E\boldsymbol{Y}_i\boldsymbol{Y}_j'$, then the forward predictors are given by

$$\hat{\boldsymbol{Y}}_{n+1} = \begin{cases} 0 & \text{if } n = 0, \\ \sum_{k=1}^{n} \boldsymbol{\Theta}_{n,k}(\boldsymbol{Y}_{n+1-k} - \hat{\boldsymbol{Y}}_{n+1-k}) & \text{otherwise} \end{cases} \tag{6}$$

where the coefficients and error covariance matrices are recursively determined from

$$\boldsymbol{V}_0 = K(1, 1),$$

$$\boldsymbol{\Theta}_{n,n-k} \left[ K(n+1, k+1) - \sum_{j=0}^{k-1} \boldsymbol{\Theta}_{n,n-j}\boldsymbol{V}_j\boldsymbol{\Theta}_{k,k-j}' \right] \boldsymbol{V}_k^{-1}$$

$$k = 0, 1, \ldots, n-1, \tag{7}$$

$$\boldsymbol{V}_n = K(n+1, n+1) - \sum_{j=0}^{n-1} \boldsymbol{\Theta}_{n,n-j}\boldsymbol{V}_j\boldsymbol{\Theta}_{n,n-j}'.$$

PROOF. See Brockwell and Davis (1987, pp. 412–13).

In a univariate case, the innovation algorithm is a refinement of the square-root-free Cholesky decomposition procedure. Similar ideas can be traced back to Kailath (1968, 1970), Rissanen and Barbosa (1969) and Ansley

(1979), to name just a few. The univariate innovation algorithm is also analyzed by Wincek and Reinsel (1986). However, it appears that it is Brockwell and Davis (1987) (see also Brockwell and Davis, 1988) who first give a thorough treatment and generalize it to evaluate predictors and likelihood functions of multivariate Gaussian stationary processes.

It is interesting to know how the innovation algorithm works in terms of matrices. Recall that every positive definite matrix can be uniquely decomposed as $\boldsymbol{LDL}'$, a product of three matrices, where $\boldsymbol{D}$ is a diagonal matrix and $\boldsymbol{L}$ is lower triangular with identity diagonal elements. It is easy to show that, in a similar fashion, the innovation algorithm decomposes the covariance matrix $\boldsymbol{\Gamma}$ into

$$\boldsymbol{\Gamma} = \boldsymbol{\Theta V \Theta}'$$

where

$$\boldsymbol{\Theta} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{\Theta}_{1,1} & \boldsymbol{I} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{\Theta}_{n,n} & \boldsymbol{\Theta}_{n,n-1} & \boldsymbol{\Theta}_{n,n-2} & & \boldsymbol{I} \end{bmatrix}$$

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{V}_0 & \boldsymbol{0} & \boldsymbol{0} & & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{V}_1 & \boldsymbol{0} & & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & & \boldsymbol{V}_n \end{bmatrix}.$$

Notice that $\boldsymbol{V}$ is *not* a diagonal matrix, but a matrix having $m \times m$ matrices as its diagonal elements and zeros elsewhere.

The efficiency of an algorithm is often judged by how many multiplications it takes. Given the coefficient matrices $\boldsymbol{\Theta}_{i,j}$, the predictors $\hat{\boldsymbol{Y}}_1$ to $\hat{\boldsymbol{Y}}_{n+1}$ can be computed recursively from (6), where the number of multiplications is easily seen to be

$$\mu_1 = m^2 + m^2 2 + \cdots m^2 n = \frac{n(n+1)m^2}{2}$$

This is the straightforward step. The key difficulty is to evaluate the $\boldsymbol{\Theta}_{i,j}$, and this takes

$$\rho_1 = \frac{n(n+1)(2n+1)}{6} m^* + n(n+2)m^*$$

where $m^*$ is the number of multiplications for inverting an $m \times m$ matrix. Notice that $\boldsymbol{\Gamma}$ is an $nm \times nm$ matrix and therefore, if one uses the standard method of linear algebra to find the decomposition (8), it is known to take $O\{(mn)^3\}$ multiplications. The contribution of the innovation algorithm is that it reduces the number of multiplications to $\rho_1 = O\{(mn)^3/3\}$, only about a third of what the traditional algorithm needs for large $n$. This fact is important in practice because $n + 1$ is often the sample size, which can be very large, and thousands of computations are required. Moreover, the

innovation algorithm becomes far better than any available approach (to the author's knowledge) when the positive matrix $\mathbf{\Gamma} = K(i, j)$ is in a band form, i.e. $K(i, j) = \mathbf{0}$ whenever $|i - j| \geq q$ for some positive integer $q$, $q < n$. In this case, it is easy to check that $\mathbf{\Theta}_{n,k} = \mathbf{0}$ for $k > q$, and as a result there is a drastic reduction in both the storage and the amount of computation time. In fact, the number of multiplications is now only

$$\rho_2 = (n - q)q(q + 2)m^* + nm^* + \frac{q(q + 1)(2q + 2)}{6} m^* + q(q + 1)m^*. \tag{10}$$

For a given $q$, Equation (10) suggests that the innovation algorithm in the current case takes a multiplication of only first order in $n$ (if $n$ is large), whereas $\rho_1$ and the multiplications in the standard matrix inverting method are of third order.

Perhaps the best way to see how the recursions work is to examine one example. Consider the VARMA(0, 1) or VMA(1) process

$$\boldsymbol{Y}_t = \mathscr{E}_t + \mathbf{\Theta}_1 \mathscr{E}_{t-1}$$

where $\mathscr{E}_t \sim \text{IIN}(0, \mathbf{\Sigma})$. It is easy to check that

$$K(i, i) = E\boldsymbol{Y}_i\boldsymbol{Y}_i = \mathbf{\Sigma} + \mathbf{\Theta}_1\mathbf{\Sigma}\mathbf{\Theta}_1' \qquad K(i, i + 1) = E\boldsymbol{Y}_i\boldsymbol{Y}_{i+1} = \mathbf{\Sigma}\mathbf{\Theta}_1'$$

and

$$K(i, j) = E\boldsymbol{Y}_i\boldsymbol{Y}_j' = 0 \qquad \text{for } |i - j| \geq 2.$$

An application of the innovation algorithm thus gives, after the first recursion,

$$\boldsymbol{V}_0 = \mathbf{\Sigma} + \mathbf{\Theta}_1\mathbf{\Sigma}\mathbf{\Theta}_1' \qquad \mathbf{\Theta}_{1,1} = \mathbf{\Sigma}\mathbf{\Theta}_1'\boldsymbol{V}_0^{-1} \qquad \boldsymbol{V}_1 = \boldsymbol{V}_0 - \mathbf{\Theta}_{1,1}\boldsymbol{V}_0\mathbf{\Theta}_{1,1}'.$$

Generally,

$$\mathbf{\Theta}_{n,1} = \mathbf{\Sigma}\mathbf{\Theta}_1'\boldsymbol{V}_{n-1}^{-1} \qquad \mathbf{\Theta}_{n,j} = 0 \qquad \text{for } 2 \leq j \leq n$$

and

$$\boldsymbol{V}_n = \boldsymbol{V}_0 - \mathbf{\Theta}_{n,1}\boldsymbol{V}_{n-1}\mathbf{\Theta}_{n,1}'.$$

With the computed $\{\mathbf{\Theta}_{i,1},\ i = 1, \ldots, n\}$, the prediction is straightforward to compute:

$$\hat{\boldsymbol{Y}}_0 = 0 \qquad \hat{\boldsymbol{Y}}_{i+1} = \mathbf{\Theta}_{i,1}(\boldsymbol{Y}_i - \hat{\boldsymbol{Y}}_i) \qquad \text{for } i = 1, \ldots, n.$$

Now consider how to evaluate the coefficient matrices $\mathbf{\Phi}_{ij}$ in (1). It seems that they can be retrieved by the innovation algorithm. However, direct calculation is much simpler by well-known Whittle algorithm (Whittle, 1963; 1983), which is a generalization of the well-known Durbin–Levinson algorithm. However, in his important paper, Whittle failed to observe that the evaluation of the prediction error matrices ($\boldsymbol{V}_n$ and $\tilde{\boldsymbol{V}}_n$) can be computed more efficiently. This is the version presented below.

A process $\{\boldsymbol{Y}_t\}$ is said to be stationary (weakly stationary in time series literature) if $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_k$ have the same covariance matrices as $\boldsymbol{Y}_{t+1}, \ldots, \boldsymbol{Y}_{t+k}$ for any $t$ and $k$, and thus we can denote $K(t+k, t)$ by $\boldsymbol{\Gamma}(k)$. Let $\tilde{\boldsymbol{V}}_{n-1}$ be the error covariance matrix of the backward predictor. It has exactly the same form as (5) with $\boldsymbol{Y}_n$ and $\hat{\boldsymbol{Y}}_n$ replaced by $\boldsymbol{Y}_1$ and $\tilde{\boldsymbol{Y}}_1$. Since the following algorithm computes both the forward and the backward predictor, we name it the Whittle algorithm.

WHITTLE ALGORITHM. If $\{\boldsymbol{Y}_t\}$ is zero mean stationary, then the forward and backward predictors are given by (1) and (3) respectively with the coefficient matrices recursively determined by

$$\boldsymbol{V}_0 = \tilde{\boldsymbol{V}}_0 = \boldsymbol{\Gamma}(0),$$

$$\boldsymbol{\Phi}_{n,n} = \left[\boldsymbol{\Gamma}(n) - \sum_{j=1}^{n-1} \boldsymbol{\Gamma}(n-j)\tilde{\boldsymbol{\Phi}}'_{n-1,j}\right] \tilde{\boldsymbol{V}}_{n-1}^{-1},$$

$$\tilde{\boldsymbol{\Phi}}_{n,n} = \left[\boldsymbol{\Gamma}(n)' - \sum_{j=1}^{n-1} \boldsymbol{\Gamma}(n-j)'\boldsymbol{\Phi}'_{n-1,j}\right] \boldsymbol{V}_{n-1}^{-1},$$

$$\begin{aligned} \boldsymbol{\Phi}_{n,k} &= \boldsymbol{\Phi}_{n-1,k} - \boldsymbol{\Phi}_{n,n}\tilde{\boldsymbol{\Phi}}_{n-1,n-k} \qquad k = 1, \ldots, n-1, \\ \tilde{\boldsymbol{\Phi}}_{n,k} &= \tilde{\boldsymbol{\Phi}}_{n-1,k} - \tilde{\boldsymbol{\Phi}}_{n,n}\boldsymbol{\Phi}_{n-1,n-k} \qquad k = 1, \ldots, n-1, \end{aligned} \tag{11}$$

$$\begin{aligned} \boldsymbol{V}_n &= (\boldsymbol{I}_m - \boldsymbol{\Phi}_{n,n}\tilde{\boldsymbol{\Phi}}_{n,n})\boldsymbol{V}_{n-1}, \\ \tilde{\boldsymbol{V}}_n &= (\boldsymbol{I}_m - \tilde{\boldsymbol{\Phi}}_{n,n}\boldsymbol{\Phi}_{n,n})\,\tilde{\boldsymbol{V}}_{n-1}. \end{aligned} \tag{12}$$

PROOF. See Morf *et al.* (1978) or Zhou (1990).

It is easy to show that the number of multiplications in the Whittle algorithm is

$$\rho_3 = (2n^2 + 6n - 1)m^*. \tag{14}$$

The striking difference between the innovation algorithm and the Whittle algorithm occurs when both of them are considered from a matrix perspective. As shown in (8), the innovation algorithm essentially produces the block $\boldsymbol{LDL}'$ decomposition of the covariance matrix $\boldsymbol{\Gamma}$. In contrast, what the Whittle algorithm does is to decompose the inverse of this covariance matrix into a similar form:

$$\boldsymbol{\Gamma}^{-1} = \boldsymbol{\Phi}'\boldsymbol{V}^{-1}\boldsymbol{\Phi},$$

where

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & & \boldsymbol{0} \\ -\boldsymbol{\Phi}_{1,1} & \boldsymbol{I} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \vdots & & \vdots \\ -\boldsymbol{\Phi}_{n,1} & -\boldsymbol{\Phi}_{n,n-1} & -\boldsymbol{\Phi}_{n,n-2} & & \boldsymbol{I} \end{bmatrix}$$

$$\boldsymbol{V}^{-1} = \begin{bmatrix} \boldsymbol{V}_0^{-1} & \mathbf{0} & \mathbf{0} & & \mathbf{0} \\ \mathbf{0} & \boldsymbol{V}_1^{-1} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{V}_n^{-1} \end{bmatrix}$$

which is well known and easy to derive. The representation (15) is a block form of the usual $\boldsymbol{UDU}'$ decomposition of matrix theory. Since (15) implies $\boldsymbol{\Phi\Gamma\Phi}' = \boldsymbol{V}$, recalling the elementary transformations in matrix theory, we find that $\boldsymbol{\Phi}$ captures all the elementary transformations that are necessary to make the covariance $\boldsymbol{\Gamma}$ block diagonal.

In addition, although the innovation algorithm and the Whittle algorithm can both be used to evaluate the forward predictions, there are important differences. First, the Whittle algorithm requires the stationarity of the process, whereas the innovation algorithm does not. Second, the Whittle algorithm is uniquely suited for computing the backward predictions. Third, the innovation algorithm is far better than direct inversion of the $\boldsymbol{\Gamma}$ matrix, which taks $O(m^3T^3)$ multiplications, but it still consumes $\rho_1 = O(m^3T^3/3)$ multiplications and hence is also a third-order algorithm, where $T = n + 1$ is the sample size. In contrast, the Whittle algorithm is a $\rho_3 = O(3m^3T^2)$ procedure. Fourth, when applied to pure VMA($q$) processes, the innovation algorithm becomes very attractive and takes only $\rho_2 = O\{q(q+2)m^3T\}$ multiplications. However, the Whittle algorithm is better for fitting pure VAR models.

Since the determination of the covariance structure of a pure stationary VAR process (without specifying the initial conditions) is rather tedious, we do not give an example to show that the Whittle algorithm performs better in a pure VAR model (this is because $\rho_1 < \rho_3$ and $\rho_1/\rho_3 \to \infty$ as $n \to \infty$). Instead, we give an example that shows the comparative advantage of the innovation algorithm. Consider the following numerical experiment on a bivariate ARMA(2, 2) model with $T = 50$:

$$\boldsymbol{X}_t = \boldsymbol{\Phi}_1\boldsymbol{X}_{t-1} + \boldsymbol{\Phi}_2\boldsymbol{X}_{t-2} + \boldsymbol{\mathcal{E}}_t + \boldsymbol{\Theta}_1\boldsymbol{\mathcal{E}}_{t-1} + \boldsymbol{\Phi}_2\boldsymbol{\mathcal{E}}_{t-2}.$$

Then predictions of $\boldsymbol{X}_t$ are easily obtained from the predictions of $\boldsymbol{Y}_t$, where $\boldsymbol{Y}_t = \boldsymbol{X}_t - \boldsymbol{\Phi}_1\boldsymbol{X}_{t-1} - \boldsymbol{\Phi}_2\boldsymbol{X}_{t-2}$ is a pure VMA(2) process. On a Sun 3/50, the innovation algorithm takes 0.03 min for all the predictions. Although all the results agree up to 8 digits in double precision computations,[1] the Whittle algorithm and the direct inversion take relatively more time: 0.10 min and 1.46 min respectively. This is what one would expect from the foregoing theoretical analysis. To illustrate this, notice that $m = q = 2$ and $n = 49$ in this example. Thus, by (10) and (14), the innovation algorithm and the Whittle algorithm would require $\rho_2 = 3496$ and $\rho_3 = 58\,800$ multiplications respectively, whereas the standard direct inversion approach takes $\rho = (2 \times 49)^3 = 941\,192$. The actual computation time spent agrees with the relationship $\rho > \rho_3 > \rho_2$. But the proportion of $0.03/0.10 = 30\%$, which measures how fast the innovation algorithm is relative to the Whittle

algorithm, is different from $\rho_2/\rho_3 = 6\%$ (this is the corresponding theoretical measure). This may be explained by the fact that some fixed amount of time has to be used in any of the algorithms. So, if an algorithm takes longer, the fixed amount could be relatively small and less important. As a result, the proportions are likely to be more equal. Indeed, $\rho_2/\rho = 0.037\%$ is closer to $0.03/1.46 = 0.02\%$ than is any other proportion to the theoretical estimation.

## 3. LIKELIHOOD FUNCTION

The prediction algorithms discussed in the previous section are useful not only for prediction and simulation but also for likelihood function evaluatations. Armed with them, we are able in this section to derive algorithms that compute the exact likelihood function and its first-order derivatives for a Gaussian VARMA($p, q$) process:

$$\boldsymbol{X}_t = \boldsymbol{\Phi}_1 \boldsymbol{X}_{t-1} + \ldots + \boldsymbol{\Phi}_p \boldsymbol{X}_{t-p} + \mathscr{E}_t + \boldsymbol{\Theta}_1 \mathscr{E}_{t-1} + \ldots + \boldsymbol{\Theta}_q \mathscr{E}_{t-q}, \quad (16)$$

where the initial values $\boldsymbol{X}_0, \ldots, \boldsymbol{X}_{1-p}$ are given constants, $\mathscr{E}_t$ are normal and independent identically distributed disturbances ($\mathscr{E}_t \sim \mathrm{IIN}(0, \boldsymbol{\Sigma})$, $t = 1, \ldots, T$) and $\boldsymbol{\Sigma}$ is the $m \times m$ variance–covariance matrix of the disturbances that is positively definite. Unlike some other studies (Anderson, 1980; Reinsel, 1979a,b), we do not require the predetermination of the initial $\mathscr{E}$s, i.e. $\mathscr{E}_0, \ldots, \mathscr{E}_{1-q}$. However, since we allow the possibility of nonstationarity, it is necessary to specify the initial motion of the $\boldsymbol{X}$. This is done here, for simplicity, by letting the $p$ initial values be given.

We provide two methods for obtaining the maximum likelihood estimator (the set of parameter values which maximize the likelihood function). The first is a one-step minimization problem which has all the parameters of the model as choice variables. The second, however, is a two-step procedure which is simpler in terms of the computation task and storage. This is because in the first step the minimization problem to be solved involves a much smaller number of parameters, i.e. only those that appear in the moving average and disturbances, and in the second step the computation is totally analytical—the maximum likelihood estimator of the autoregressive parameters is obtained as an explicit function of the known estimator obtained from the first step. All the algorithms are computationally efficient and easily implemented.

To apply the prediction algorithms to the VARMA model (16), we let $\boldsymbol{Y}_t$ be a linear filter of the $\boldsymbol{X}_t$:

$$\boldsymbol{Y}_t = \boldsymbol{X}_t - \boldsymbol{\Phi}_1 \boldsymbol{X}_{t-1} - \qquad - \boldsymbol{\Phi}_p \boldsymbol{X}_{t-p}. \quad (17)$$

The model implies that the filter should admit a multivariate moving-average representation, and hence $\{\boldsymbol{Y}_t\}$ is stationary, although $\{\boldsymbol{X}_t\}$ may not be.

Consistent with Section 2, denote by $\mathbf{\Gamma}(k)$ the covariance matrix of $\boldsymbol{Y}_{t+k}$ with $\boldsymbol{Y}_t$. It follows that

$$\mathbf{\Gamma}(k) = E\boldsymbol{Y}_{t+k}\boldsymbol{Y}_t' = \begin{cases} \sum_{j=0}^{q-k} \mathbf{\Theta}_{j+k}\mathbf{\Sigma}\mathbf{\Theta}_j' & \text{if } k \leq q \\ 0 & \text{otherwise} \end{cases}$$

with $\mathbf{\Theta}_0 \equiv \boldsymbol{I}$, the identity matrix. By the normality assumption on the model errors, $\boldsymbol{Y}_{n+1}$ given $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$ must be normal. Furthermore, this normal distribution has the forward predictor as its mean and the error covariance matrix as its covariance matrix. Notice that a density can be written as a recursive product of a series of conditional densities:

$$P(\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_T) = P(\boldsymbol{Y}_1)P(\boldsymbol{Y}_2|\boldsymbol{Y}_1) \ldots P(\boldsymbol{Y}_T|\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_{T-1}).$$

We thus derive the log likelihood (up to a constant):

$$\log \mathcal{L}(\boldsymbol{\theta}) = -\frac{1}{2}\sum_{j=1}^{T} \log|\boldsymbol{V}_{j-1}| - \frac{1}{2}\sum_{j=1}^{T} (\boldsymbol{Y}_j - \hat{\boldsymbol{Y}}_j)'\boldsymbol{V}_{j-1}^{-1}(\boldsymbol{Y}_j - \hat{\boldsymbol{Y}}_j),$$

where $\boldsymbol{\theta}$ stands for all the parameters. This formula expresses the likelihood explicitly as a function of the filters $\{\boldsymbol{Y}_j\}$ and their forward predictions. It is straightforward to obtain $\boldsymbol{Y}_j$ as a function of the observations on $\boldsymbol{X}_t$ from (17). $\hat{\boldsymbol{Y}}_j$ is easily computed using either the innovation algorithm or the Whittle algorithm of the previous section. In our present case, the innovation algorithm is faster than the Whittle algorithm because $\boldsymbol{Y}_t$ has the special property that $\mathbf{\Gamma}(k) = \mathbf{0}$ whenever $k > q$. Nevertheless, the Whittle algorithm is useful for checking the code and for other purposes, some of which we have already mentioned in the previous section and more will be discussed later. In short, there are two efficient methods for the likelihood function evaluation. The first method uses the innovation algorithm, and the second is based on the Whittle algorithm.

With either of the algorithms, the maximum likelihood estimator can be obtained by applying a derivative-free method to maximize the likelihood function. Equivalently, one solves a minimization problem with the objective function

$$\mathbf{\Pi}(\boldsymbol{\theta}) = \sum_{j=1}^{T} \log|\boldsymbol{V}_{j-1}| + \sum_{j=1}^{T} (\boldsymbol{Y}_j - \hat{\boldsymbol{Y}}_j)'\boldsymbol{V}_{j-1}^{-1}(\boldsymbol{Y}_j - \hat{\boldsymbol{Y}}_j)$$

over all the parameters. It should be noticed that the minimization problem (20) is not unconstrained because the choice of the elements of $\mathbf{\Sigma}$ must be made in a subspace of all possible values such that $\mathbf{\Sigma}$ is a positive definite matrix. Since unconstrained problems are much easier to solve, we eliminate the constraint by making a parameter transformation: $\mathbf{\Sigma} = \boldsymbol{L}\boldsymbol{L}'$, where $\boldsymbol{L}$ is lower triangular with positive diagonal elements. This is the Cholesky

decomposition of the positive definite $\boldsymbol{\Sigma}$ matrix and $\boldsymbol{L}$ exists uniquely. Then an unconstrained minimization[2] of (20) over the parameters

$$\boldsymbol{\Phi}_i,\ \boldsymbol{\Theta}_j,\ \boldsymbol{L} \qquad (i = 1, \ldots, p;\ j = 1, \ldots, q) \tag{21}$$

will produce the maximum likelihood estimator with $\boldsymbol{\Sigma}$ being replaced by the decomposition wherever it appears in the evaluation of $\boldsymbol{\Pi}(\boldsymbol{\theta})$.

In order to facilitate solving the minimization problem and to obtain a better estimation of the Hessian matrix, it is of practical importance to be able to evaluate not only the log likelihood function but also its gradient. As in the case of log likelihood evaluation, two algorithms, which are based upon either the innovation algorithm or the Whittle algorithm, can be used to this end. Because the derivations are similar, and it may be sufficient in practice to use just the first, we derive only those formulae that depend solely on the innovation algorithm.

The basic idea is to differentiate the recursive relationships and write the results in as simple a form as possible. It is along these lines that Wincek and Reinsel (1986) derive the gradient formula for a univariate time series. Fortunately, the procedure can be generalized to the multivariate model. Recalling the standard results in matrix theory, we have the differential of the objective function (20):

$$\begin{aligned} d\boldsymbol{\Pi}(\boldsymbol{\theta}) = {} & \sum_{j=1}^{T} \operatorname{tr} \boldsymbol{V}_{j-1}^{-1} d\boldsymbol{V}_{j-1} - \sum_{j=1}^{T} \boldsymbol{Y}_j^{*\prime} \boldsymbol{V}_{j-1}^{-1} d\boldsymbol{V}_{j-1} \boldsymbol{V}_{j-1}^{-1} \boldsymbol{Y}_j^{*} \\ & + 2 \sum_{j=1}^{T} \boldsymbol{Y}_j^{*\prime} \boldsymbol{V}_{j-1}^{-1} d\boldsymbol{Y}_j^{*} \qquad \boldsymbol{Y}_j^{*} \equiv \boldsymbol{Y}_j - \hat{\boldsymbol{Y}}_j, \end{aligned} \tag{22}$$

where tr is the trace operator. Starting from here, we obtain the derivatives in the following three steps with respect to (w.r.t.) the three different sets of parameters in (21).

### 3.1. *Derivatives with respect to* $\boldsymbol{\Phi}_l$ $(l = 1, \ldots, p)$

Let $\theta_{i,j}^{l}$ be the $(i, j)$th element of the matrix $\boldsymbol{\Phi}_l$. Holding all other parameters constant, then the derivative w.r.t. $\theta_{i,j}^{l}$ is just the differential (divided by $d\theta_{i,j}^{l}$), which is seen from (22) to be

$$d\boldsymbol{\Pi}(\boldsymbol{\theta}) = 2 \sum_{j=1}^{T} \boldsymbol{Y}_j^{*\prime} \boldsymbol{V}_{j-1}^{-1} d\boldsymbol{Y}_j^{*},$$

where the $\boldsymbol{V}$ have already been obtained by using the innovation algorithm. The $d\boldsymbol{Y}^{*}$ can be calculated recursively from

$$d\boldsymbol{Y}_{n+1}^{*} = d\boldsymbol{Y}_{n+1} - \sum_{k=1}^{\min(n,q)} \boldsymbol{\Theta}_{n,k} d\boldsymbol{Y}_{n+1-k}^{*},$$

where $d\boldsymbol{Y}_{n+1}$ is given by the $m \times 1$ vector

$$d\boldsymbol{Y}_{n+1} = -\begin{bmatrix} 0 \\ \vdots \\ 0 \\ x_j^{n+1-p} \\ 0 \\ \vdots \\ 0 \end{bmatrix} d\theta_{i,j}^{l} \qquad (i\text{th row})$$

Both of the last two formulae are easily derived from their definitions ((22), (3) and (17)).

### 3.2. *Derivatives with respect to* $\boldsymbol{L}$

In order to obtain the partials, we need to compute all variables in the differential (22). This can be accomplished by evaluating the differentials of the $\boldsymbol{\Sigma}$ first, the $\Gamma$s second, the prediction coefficient matrices third and the $\boldsymbol{Y}$s and $\boldsymbol{Y}^*$s last. The partial derivative of the objective function w.r.t. $l_{ij}$ $(j \leqslant i)$ is then readily given by (22).

Notice that all other parameters are held constant. It is straightforward to see that the differential of the $\boldsymbol{\Sigma}$ is given by the $m \times m$ matrix

$$d\boldsymbol{\Sigma} = \begin{bmatrix} & l_{jj} & \\ & \vdots & \\ l_{jj} \ldots & 2l_{ij} & l_{mj} \\ & \vdots & \\ & l_{mj} & \end{bmatrix} dl_{ij} \qquad (i\text{th row}).$$

($i$th column)

The differential of the covariance matrix is then computed from

$$d\boldsymbol{\Gamma}(k) \quad \begin{cases} \sum_{j=1}^{q-k} \boldsymbol{\Theta}_{j+k} d\boldsymbol{\Sigma} \boldsymbol{\Theta}_j' & \text{if } k \leqslant 0, \\ 0 & \text{otherwise.} \end{cases}$$

Now we can compute the differentials of the prediction coefficient matrices. Similar to the likelihood evaluations, the differentials are recursively determined from the differential form of the innovation algorithm:

$$d\boldsymbol{V}_0 = d\boldsymbol{\Gamma}(0),$$

$$d\boldsymbol{\Theta}_{n,\min(n,q)} = [d\boldsymbol{\Gamma}\{\min(n, q)\} \quad \boldsymbol{\Theta}_{n,\min(n,q)}]\boldsymbol{V}^{-1}_{\max(0,n-q)},$$

$$d\boldsymbol{\Theta}_{n,n-k} = [\{d\boldsymbol{\Gamma}(n-k) - \boldsymbol{\Theta}_{n,n-k} d\boldsymbol{V}_k\} \sum_{j=\max(0,n-q)}^{k-1} (d\boldsymbol{\Theta}_{n,n-j} \boldsymbol{V}_j \boldsymbol{\Theta}'_{k,k-j}$$

$$+ \boldsymbol{\Theta}_{n,n-j} d\boldsymbol{V}_j \boldsymbol{\Theta}'_{k,k-j} + \boldsymbol{\Theta}_{n,n-j} \boldsymbol{V}_j d\boldsymbol{\Theta}_{k,k-j})]\boldsymbol{V}_k^{-1}$$

$$k = \max(1, n-q), \max(1, n-q)+1, \ldots, n-$$

$$d\mathbf{V}_n = d\boldsymbol{\Gamma}(0) -$$

$$\sum_{j=\max(0,n-q)}^{n-1} (d\boldsymbol{\Theta}_{n,n-j}\mathbf{V}_j\boldsymbol{\Theta}'_{n,n-j} + \boldsymbol{\Theta}_{n,n-j}d\mathbf{V}_j\boldsymbol{\Theta}'_{n,n-j} + \boldsymbol{\Theta}_{n,n-j}\mathbf{V}_j d\boldsymbol{\Theta}'_{n,n-j}),$$

$$n = 1, \ldots, T-1.$$

The remaining calculations for obtaining the partial w.r.t. $l_{ij}$ $(j \leq i)$ follow easily.

3.3. *Derivatives with respect to* $\boldsymbol{\Theta}_l$ $(l = 1, \ldots, q)$

Again, it is sufficient to find the differential $d\boldsymbol{\Pi}(\boldsymbol{\theta})$ when all parameters are held constant except $\theta^l_{i,j}$, the $(i,j)$th element of the matrix $\boldsymbol{\Theta}_l$. Now the differentials of the $\boldsymbol{\Gamma}$ should be computed from

$$d\boldsymbol{\Gamma}(k) = \begin{cases} \boldsymbol{\Theta}_{l+k}\boldsymbol{\Sigma}d\boldsymbol{\Theta}'_l & \text{if } l < k,\ l \leq q-k, \\ \mathbf{0} & \text{if } l < k,\ l > q-k, \\ \boldsymbol{\Theta}_{l+k}\boldsymbol{\Sigma}d\boldsymbol{\Theta}'_l + d\boldsymbol{\Theta}_l\boldsymbol{\Sigma} & \text{if } l = k,\ l \leq q-k, \\ d\boldsymbol{\Theta}_l\boldsymbol{\Sigma} & \text{if } l = k,\ l > q-k, \\ \boldsymbol{\Theta}_{l+k}\boldsymbol{\Sigma}d\boldsymbol{\Theta}'_l + d\boldsymbol{\Theta}_l\boldsymbol{\Sigma}\boldsymbol{\Theta}'_{l-k} & \text{if } l > k,\ l \leq q-k, \\ d\boldsymbol{\Theta}_l\boldsymbol{\Sigma}\boldsymbol{\Theta}'_{l-k} & \text{if } l > k,\ l > q-k. \end{cases}$$

The remaining procedures are exactly the same as in step 2.

The above procedures are the first method mentioned for obtaining the maximum likelihood estimators. The second method is motivated from the idea of generalized least squares (GLS). It turns out that the maximum likelihood estimator of the autoregressive parameters, conditional on $\boldsymbol{\Sigma}$ (or $\boldsymbol{L}$) and the moving-average parameters, can be obtained analytically as a function of other parameters of the model. Therefore the maximum likelihood estimator can be obtained by a two-step procedure. In the first step, the conditional likelihood function is maximized to get the maximum likelihood estimator of the moving-average component and covariance parameters. Then, in the second step, the autoregressive parameters are obtained in closed form by using GLS.

Taking a transpose on both sides of the model (16) and stacking together, we obtain its matrix form:

$$\begin{bmatrix} \boldsymbol{X}'_1 \\ \vdots \\ \boldsymbol{X}'_T \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}'_0 & & \boldsymbol{X}'_{1-p} \\ \vdots & & \vdots \\ \boldsymbol{X}'_{T-1} & \cdots & \boldsymbol{X}'_{T-p} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}'_1 \\ \vdots \\ \boldsymbol{\Phi}'_p \end{bmatrix} + \begin{bmatrix} \mathcal{E}'_1 \\ \vdots \\ \mathcal{E}'_T \end{bmatrix}$$

$$+\begin{bmatrix} \mathscr{E}_0' & \cdots & \mathscr{E}_{1-p}' \\ \vdots & & \vdots \\ \mathscr{E}_{T-1}' & \cdots & \mathscr{E}_{T-p}' \end{bmatrix} \begin{bmatrix} \boldsymbol{\Theta}_1' \\ \vdots \\ \boldsymbol{\Theta}_p' \end{bmatrix}$$

To apply the standard GLS results, we need to use the vector form of the model. In fact, it is enough to do this for the dependent vector. By definition (17) we have

$$\begin{bmatrix} \boldsymbol{Y}_1' \\ \vdots \\ \boldsymbol{Y}_T' \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}_1' \\ \vdots \\ \boldsymbol{X}_T' \end{bmatrix} - \begin{bmatrix} \boldsymbol{X}_0' & \cdots & \boldsymbol{X}_{1-p}' \\ \vdots & & \vdots \\ \boldsymbol{X}_{T-1}' & & \boldsymbol{X}_{T-p}' \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}_1' \\ \vdots \\ \boldsymbol{\Phi}_p' \end{bmatrix}.$$

Vectorizing this expression, we write the result

$$\mathscr{Y} = \boldsymbol{x} - \bar{\mathscr{X}}\boldsymbol{\phi},$$

where $\boldsymbol{\phi}$ is the vector form of the stacked parameters in the autoregressive component and $\bar{\mathscr{X}}$ is the $mT \times m^2 p$ matrix given by

$$\bar{\mathscr{X}} = \boldsymbol{I} \otimes \begin{bmatrix} \boldsymbol{X}_0' & & \boldsymbol{X}_{1-p}' \\ \vdots & & \vdots \\ \boldsymbol{X}_{T-1}' & & \boldsymbol{X}_{T-p}' \end{bmatrix},$$

where $\boldsymbol{I}$ is the $m \times m$ identity matrix and $\otimes$ is the Kronecker operator.

For brevity, we introduce the notation

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{Y}_1 \\ \vdots \\ \boldsymbol{Y}_T \end{bmatrix} \qquad = \begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_T \end{bmatrix}$$

and denote by $\boldsymbol{\Gamma}$ the covariance matrix of the random vector $\boldsymbol{Y}$. Decompose this matrix as $\boldsymbol{\Gamma}^{-1} = \boldsymbol{F}'\boldsymbol{F}$ (this matrix, as a tool in the proof, does not actually need to be computed), and let $\boldsymbol{G}$ be an arrangement of the $\boldsymbol{F}$ matrix,

$$\boldsymbol{G} = (F_1, F_{m+1}, \ldots, F_{(T-1)m+1}; \ldots; F_m, \ldots, F_{(T-1)m+m})$$

where $F_i$ is the $i$th column of $\boldsymbol{F}$. Then the log conditional likelihood function is (up to a conditioning constant)

$$\boldsymbol{Y}'\boldsymbol{\Gamma}^{-1}\boldsymbol{Y} = (\boldsymbol{Y}^* - \mathscr{X}^*\boldsymbol{\phi})'(\boldsymbol{Y}^* - \mathscr{X}^*\boldsymbol{\phi}),$$

where

$$\boldsymbol{Y}^* = \boldsymbol{G}\boldsymbol{x} \qquad \mathscr{X}^* = \boldsymbol{G}\bar{\mathscr{X}}.$$

Therefore, conditioning on $\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_q$ and $\boldsymbol{\Sigma}$, the maximum likelihood estimator of the vector $\boldsymbol{\phi}$ which contains all parameters of $\boldsymbol{\Phi}_1, \ldots, \boldsymbol{\Phi}_p$ as its elements is

$$\boldsymbol{\phi}_{\mathrm{ML}} = (\mathscr{X}^{*\prime}\mathscr{X}^*)^{-1}\mathscr{X}^{*\prime}\boldsymbol{Y}^* \tag{23}$$

and the conditional covariance matrix is $\mathrm{var}(\boldsymbol{\phi}_{\mathrm{ML}}) = (\mathscr{X}^{*\prime}\mathscr{X}^*)^{-1}$. This matrix is useful for generating the conditional samples from the posterior density in Bayesian analysis.

It is obvious that $\boldsymbol{Y}^* = F\boldsymbol{X}$ and $\mathscr{Z}^* = F\mathscr{Z}$, where $\mathscr{Z}$ is an appropriate arrangement of $\bar{\mathscr{Z}}$. Let $\boldsymbol{D}$ be the diagonal matrix $\boldsymbol{D} = \operatorname{diag}[\boldsymbol{V}_0, \boldsymbol{V}_1, \ldots, \boldsymbol{V}_{T-1}]$. Replacing $\boldsymbol{\phi}$ by its conditional estimation (23) in (20), we obtain the reduced objective function

$$\boldsymbol{\Pi}(\boldsymbol{\theta}) = \sum_{j=1}^{T} \log|\boldsymbol{V}_{j-1}| + \tilde{\boldsymbol{Y}}'\tilde{\boldsymbol{M}}\tilde{\boldsymbol{Y}}, \tag{24}$$

where $\tilde{\boldsymbol{Y}} = \boldsymbol{X} - \hat{\boldsymbol{X}}$ and

$$\tilde{\boldsymbol{M}} = \boldsymbol{D}^{-1} - \boldsymbol{D}^{-1}(\mathscr{Z} - \hat{\mathscr{Z}})[(\mathscr{Z} - \hat{\mathscr{Z}})'\boldsymbol{D}^{-1}(\mathscr{Z} - \hat{\mathscr{Z}})]^{-1}(\mathscr{Z} - \hat{\mathscr{Z}})'\boldsymbol{D}^{-1}.$$

The vector $\hat{\boldsymbol{X}}$ denotes the projection of $\boldsymbol{X}$, whose definition and computation are discussed in Section 2. Denoted analogously, the matrix $\hat{\mathscr{Z}}$ is composed of the projections of the corresponding columns.

The reduced objective function depends only upon part of the parameters $\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_q$ and $\boldsymbol{\Sigma}$. Therefore the $\boldsymbol{\theta}$ in (24) should be understood to represent these parameters. The maximum likelihood estimator can be obtained by minimizing this objective function. For practical purposes, it may be easier if it is computed from the following compact form:

$$\boldsymbol{\Pi}(\boldsymbol{\theta}) = \sum_{j=1}^{T} [\log|\boldsymbol{V}_{j-1}| + \tilde{\boldsymbol{Y}}_j'\boldsymbol{V}_j^{-1}\tilde{\boldsymbol{Y}}_j] - \boldsymbol{U}'\boldsymbol{W}^{-1}\boldsymbol{U} \tag{25}$$

where the $\boldsymbol{U}$ and $\boldsymbol{W}$ are an $m^2p$ vector and an $m^2p \times m^2p$ matrix, with elements defined respectively by

$$U_i = \sum_{k=1}^{T} Z_{i,k}\boldsymbol{V}_k^{-1}\tilde{\boldsymbol{Y}}_k \qquad i = 1, \ldots, m^2p$$

and

$$W_{i,j} = \sum_{k=1}^{T} Z_{i,k}\boldsymbol{V}_k^{-1}Z_{j,k} \qquad i, j = 1, \ldots, m^2p.$$

The $Z_{i,k}$ $(1 \leqslant i \leqslant T,\ 1 \leqslant k \leqslant m^2p)$ are vectors of $m$ elements. All $Z_{i,k}$ $(1 \leqslant k \leqslant m^2p)$ stacked one on top of the other form the $i$th column of the $\mathscr{Z}$ matrix. Even though the Whittle algorithm can be used to evaluate the

$- Y = CY \qquad E(Y - Y)'(Y - Y) =$

$$\boldsymbol{Y}'\boldsymbol{\Gamma}^{-1}\boldsymbol{Y} = (\boldsymbol{CY})'\boldsymbol{D}^{-1}(\boldsymbol{CY}) = (\boldsymbol{K}\mathcal{Y})'\boldsymbol{D}^{-1}(\boldsymbol{K}\mathcal{Y}),$$

where $\boldsymbol{K}$ is an $mT \times mT$ matrix, obtained as a column arrangement of $\boldsymbol{C}$ in the same way as for $\boldsymbol{G}$. Let $\boldsymbol{K} = [\boldsymbol{K}_1, \ldots, \boldsymbol{K}_m]$, with the $\boldsymbol{K}_i$ $(1 \leq i \leq m)$ $mT \times T$ matrices. Thus, we get exactly the same formula for evaluating the reduced objective function as given by (25) except for the obvious differences that

$$U_i = (\boldsymbol{K}_i\boldsymbol{Z})'\boldsymbol{D}^{-1}\tilde{\boldsymbol{Y}} \qquad i = 1, \ldots, m^2p$$

$$W_{i,j} = (\boldsymbol{K}_i\boldsymbol{Z})'\boldsymbol{D}^{-1}(\boldsymbol{K}_j\boldsymbol{Z}) \qquad i, j = 1, \ldots, m^2p,$$

where the $T \times mp$ matrix $\boldsymbol{Z}$ is defined by

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{X}_0' & & \boldsymbol{X}_{1-p}' \\ \vdots & & \vdots \\ \boldsymbol{X}_{T-1}' & & \boldsymbol{X}_{T-p}' \end{bmatrix}.$$

Finally, we give the formulae for evaluation of the gradient of the reduced objective function. It is sufficient to provide only

$$d\boldsymbol{\Pi}(\boldsymbol{\theta}) = \sum_{j=1}^{T} \operatorname{tr} \boldsymbol{V}_{j-1}^{-1} d\boldsymbol{V}_{j-1} - (\tilde{\boldsymbol{Y}} - \tilde{\mathcal{X}}\boldsymbol{\phi})'\boldsymbol{D}^{-1}d\boldsymbol{D}\boldsymbol{D}^{-1}(\tilde{\boldsymbol{Y}} - \tilde{\mathcal{X}}\boldsymbol{\phi}) + 2(\tilde{\boldsymbol{Y}} - \tilde{\mathcal{X}}\boldsymbol{\phi})'\boldsymbol{D}^{-1}(d\tilde{\boldsymbol{Y}} - d\tilde{\mathcal{X}}\boldsymbol{\phi} - \tilde{\mathcal{X}}d\boldsymbol{\phi}), \tag{26}$$

where $d\boldsymbol{\phi} = \boldsymbol{W}^{-1}[d\boldsymbol{U} - d\boldsymbol{W}\boldsymbol{\phi}]$, with $d\boldsymbol{W}$ and $d\boldsymbol{U}$ given by

$$d\boldsymbol{W} = d\tilde{\mathcal{X}}'\boldsymbol{D}^{-1}\tilde{\mathcal{X}} + \tilde{\mathcal{X}}'\boldsymbol{D}^{-1}[d\tilde{\mathcal{X}} - d\boldsymbol{D}\boldsymbol{D}^{-1}\tilde{\mathcal{X}}],$$

$$d\boldsymbol{U} = d\tilde{\mathcal{X}}'\boldsymbol{D}^{-1}\tilde{\mathcal{Y}} + \tilde{\mathcal{X}}'\boldsymbol{D}^{-1}[d\tilde{\mathcal{Y}} - d\boldsymbol{D}\boldsymbol{D}^{-1}\tilde{\mathcal{Y}}]$$

respectively. The remaining differentials can be found in the same way as in the first method.

## 4. CONCLUDING REMARKS

We have provided computationally efficient methods for evaluating the likelihood function and its gradient and methods for obtaining the maximum likelihood estimator in a possibly nonstationary vector autoregressive moving-average (VARMA) model. These methods are clearly also applicable for VARMA models with regressors and regression models with VARMA errors. However, only the estimation problem of time series analysis has been dealt with here. For the distribution theory and hypothesis testing, the readers are referred to Dickey and Fuller (1979), Chan (1988), Phillips (1988) and Priestley (1988) and references therein. Further work on the algorithms includes generalizing them to other processes, for example, fractional VARMA and non-Gaussian processes.

## NOTES

This paper is part of my Ph.D. thesis at Duke University. I am grateful to Jean-Francois Richard, George Tauchen and Mike West for serving on my thesis committee and offering helpful comments, and especially to John Geweke for his guidance and to an anonymous referee for his insightful comments on an earlier version of this paper. Naturally, the responsibility of any errors or difficulties is solely my own. Financial support from NSF Grant SES-8908365 and Fossett Foundation is also gratefully acknowledged.

[1] A FORTRAN code that implements all the algorithms is available from the author upon request.

[2] Strictly speaking, it still has the trivial constraint that $L$ must have nonzero diagonal elements.

## REFERENCES

ANDERSON, T. W. (1980) Maximum likelihood estimation for vector autoregressive moving average models. In *Directions in Time Series* (eds D. R. Brillinger and G. C. Tiao). Institute of Mathematical Statistics, 49–59.

ANSLEY, C. F. (1979) An algorithm for the exact likelihood of a mixed autoregressive–moving average process. *Biometrika* 66, 59–65.

BOX, G. E. P. and JENKINS, G. M. (1976) *Time Series Analysis: Forecasting and Control*, revised edn. San Francisco, CA: Holden-Day.

BROCKWELL, P. J. and DAVIS, R. A. (1987) *Time Series: Theory and Methods*. New York: Springer-Verlag.

—and — (1988) Applications of innovation representations in time series analysis. In *Probability and Statistics* (ed. J. N. Srivastara). Amsterdam: North-Holland.

CHAN, N. H. (1988) Limiting distributions of least squares estimates of unstable autoregressive processes. *Ann. Statist.* 16, 367–401.

DICKEY, D. A. and FULLER, W. A. (1979) Distribution of the estimators for autoregressive time series with a unit root. *J. Am. Statist. Assoc.* 74, 427–31.

FRIEDLANDER, B. (1982) *Proc. IEEE* 70, 829–67.

GEWEKE, J. (1984) Measures of conditional linear dependence and feedback. *J. Am. Statist. Assoc.* 79, 907–15.

— (1988) Antithetic acceleration of Monte Carlo integration in Bayesian inference. *J. Econometrics* 38, 73–90.

— (1989) The posterior distribution of roots in multivariate autoregressions. *1989 Proceedings of the Business and Economic Statistics Section—American Statistical Association*, forthcoming.

HANNAN, E. J. (1970) *Multiple Time Series*. New York: Wiley.

HILLMER, S. C. and TAIO, G. C. (1979) Likelihood function of stationary multiple autoregressive moving average models. *J. Am. Statist. Assoc.* 74, 652–60.

KAILATH, T. (1968) An innovation approach to least square estimation—Part I: Linear filtering in additive noise. *IEEE Trans. Autom. Control* 13, 646–54.

— (1970) The innovation approach to detection and estimation theory. *Proc. IEEE* 58, 680–95.

KOHN, R. and ANSLEY, C. F. (1985) Computing the likelihood of and its derivatives for a Gaussian ARMA model. *J. Statist. Comput. Simulation* 22, 229–63.

MELARD, G. (1984) A fast algorithm for the exact likelihood of autoregressive moving average models. *Appl. Statist.* 33, 104–14.

MORF, M., VIEIRA, A. and KAILATH, T. (1978) Covariance charaterization by partial autocorrelation matrices. *Ann. Statist.* 6, 643–8.

NELSON, C. R. and KANG, H. (1981) Spurious periodicity in appropriately detrended time series. *Econometrica* 49, 741–51.

—— and PLOSSER, C. I. (1982) Trends and random walks in macroeconomic time series. *J. Monet. Econ.* 8, 129–62.

PHADKE, M. S. and KEDEM, G. (1978) Computation of the exact likelihood function of multivariate moving average models. *Biometrika* 65, 511–19.

PHILLIP, P. C. B. (1987) Time series regression with unit roots. *Econometrica* 55, 277–302.

PRIESTLEY, M. B. (1988) *Non-linear and Non-stationary Time Series Analysis*. New York: Academic Press.

REINSEL, G. C. (1979) Maximum likelihood estimation of stochastic linear difference equations with autoregressive moving average errors. *Econometrica* 47, 129–51.

RISSANEN, J. and BARBOSA, L. (1973) A fast algorithm for optimum linear predictors. *IEEE Trans. Autom. Control* 18, 555.

TIAO, G. C. and BOX, G. E. P. (1981) Modelling multiple time series with applications. *J. Am. Statist. Assoc.* 76, 802–16.

WHITTLE, P. (1963) On the fitting of multivariate autoregressions, and the approximate canonical factorization of a spectral density matrix. *Biometrika* 50, 129–34.

—— (1983) *Prediction and Regulation by Linear Least-square Methods*, 2nd edn. Minneapolis, MN: University of Minnesota.

WILSON, T. G. (1973) The estimation of parameters in multivariate time series models. *J. R. Statist. Soc. Ser. B* 35, 76–85.

—— (1979) Some efficient computational procedures for high order ARMA models. *J. Statist. Comput. Simulation* 8, 301–9.

WINCEK, M. A. and REINSEL, G. C. (1986) An exact maximum likelihood estimation procedure for regression-ARMA time series models with possibly nonconsecutive data. *J. R. Statist. Soc., B Ser.* 48, 303–13.

ZHOU, G. (1990) A Bayesian analysis of time series with applications to stationarity and causality. Ph. D. Dissertation, Duke University, Durham, NC.