An algorithm for computing estimates for parameters of an ARMA-model from noisy measurements of inputs and outputs
Vregelaar, ten, J.M.

Published: 01/01/1987

Document Version
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 02. Jan. 2019
COSOR-memorandum 87-13

An algorithm for computing estimates for parameters of an ARMA-model from noisy measurements of inputs and outputs

by

J.M. ten Vregelaar

Eindhoven, May 1987
The Netherlands
An algorithm for computing estimates for parameters of an ARMA-model from noisy measurements of inputs and outputs

ABSTRACT

By applying some iterative algorithm a nonlinear minimization problem is solved in order to obtain estimates for the parameters of an ARMA-model. The algorithm contains a special $Q - R$ decomposition using the structure of the problem in an optimal way.

1. Introduction

When one introduces some least squares estimation method for the parameters of an ARMA-model based on a set of noisy measurements of inputs and outputs, the following minimization problem is involved (cf. Ten Vregelaar '87):

$$\min_{\Theta} J_N(\Theta),$$  

where

$$J_N(\Theta) = z^T P_N(\Theta) z.$$  

Here $z$ is the known $(s + r) (N + m)$ (column) vector of measurements,

$\Theta$ is the $s \times (ps + (q + 1)sr)$ matrix of unknown parameters (cf. (3.2)),

and

$$P_N := D_N^+ D_N$$ (1.2a)

with

$$D_N^+ := D_N^T (D_N D_N^T)^{-1}.$$ (1.2b)

The $sN \times (s + r) (N + m)$ matrix $D_N(\Theta)$ has been introduced in order to denote the model equation as

$$D_N(\Theta) \zeta = 0$$ (1.3)

where $\zeta$ is the vector of (unknown) true inputs and outputs.
For the moment we do not specify $D_N$ and $\zeta$ any further, but $D_N$ will be rowregular for all $\Theta$ anyway. Later on, we will choose $D_N$ and $\zeta$ conveniently. The integers $r, s, m$ and $N$ are supposed to be known, actually $r, s$ and $m$ will be small in comparison to $N$.

The solutions of (1.1) represent least squares estimates for the unknown matrix of parameters $\Theta$.

Since (1.1) has no closed-form solution, an iterative algorithm has to be applied to obtain estimates, provided an initial estimate for $\Theta$.

In this paper we will describe an algorithm to compute the object function $J_N$ and its vector of derivatives $J_N'$ for given $\Theta$.

The iteration procedure is supposed to be based on both $J_N$ and $J_N'$.

If $\hat{J}_N$ denotes any component of $J_N'$, $(\cdot$ representing $\frac{\partial}{\partial \Theta_i}$ for any element of the matrix $\Theta)$,

$$\hat{J}_N(\Theta) = 2z^T D_N^+ \hat{D}_N P_N^{-1} z$$

(1.4a) holds, with

$$P_N^{-1} = I - P_N$$

(1.4b) $I$ denoting the $(s + r)(N + m)$ identity matrix.
2. Application of a Q – R decomposition

We are concerned with the computation of \( J_N(\Theta) \) and \( J_N'(\Theta) \) for \( \Theta \) known. Let us omit the subscript \( N \) in \( D_N \) and \( P_N \).

Since \( D \) is rowregular, \( D^T \) has full column rank.

Therefore it admits a \( Q – R \) decomposition

\[
Q^T D^T = \begin{bmatrix} R \\ O \end{bmatrix},
\]

for some orthogonal matrix \( Q \) and \( sN \) square upper triangular regular \( R \) (it will be more convenient here to construct a lower triangular \( R \), as one can see from the next section).

In this section we will deal with the use of this \( Q – R \) decomposition for the computation of \( J_N(\Theta) \) and \( J_N'(\Theta) \). In the next section we will describe how to perform efficiently the \( Q – R \) decomposition for a suitable choice of \( D \).

If

\[
Q = (Q_1 \ Q_2), \text{ with } Q_1 \text{ is } (s + r) (N + m) \times sN
\]

then

\[
D^T = Q_1 R.
\]

Hence

\[
P = Q_1 Q_1^T
\]

using the orthogonality of \( Q \).

Furthermore, from (1.1b)

\[
J_N = \| Q_1^T z \|^2,
\]

\( \| . \| \) denoting the Euclidean norm.

Defining the \( sN \)-vector

\[
\lambda := (D^*)^T z,
\]

implies

\[
D^T \lambda = P z
\]

and from (1.4a)
\[ j_N = 2 \lambda^T \dot{D} (z - D^T \lambda) \]  \hfill (2.7)

From (2.2) and (2.5) we obtain
\[ R \lambda = Q_1^T z \]  \hfill (2.8)

Introducing the \( sN \)-vector
\[ u := Q_1^T z \]  \hfill (2.9)

we summarize.

By adding the column vector \( z \) to \( D \), it follows
\[
Q^T \begin{pmatrix} D^T \ 1 \ z \end{pmatrix} = \begin{bmatrix} R \ u \\ 0 \ v \end{bmatrix}
\]  \hfill (2.10)

with \( v := Q_2^T z \) \((sm + r(N + m) \text{ vector})\) for completeness.

Then \( J_N \) and an arbitrary component \( j_N \) of its derivative vector satisfy
\[
J_N = \| u \|^2 ,
\]
\[
\dot{j}_N = 2 \lambda^T \dot{D} (z - D^T \lambda)
\]  \hfill (2.11a)  \hfill (2.11b)

where
\[ R \lambda = u \]  \hfill (2.11c)

From the \( Q - R \) decomposition we only need \( R \) and \( u \) to compute \( J_N \) and \( J_N' \). The solution of \( \lambda \) from (2.11c) can be done easily, since \( R \) turns out to be lower triangular.
3. Performing the $Q - R$ decomposition

According to relation (2.1) in Ten Vregelaar '87 we can denote the model equation as

$$ D \Theta \zeta = 0 , $$

where

$$ D = \begin{bmatrix} \alpha_0 & \alpha_1 & \ldots & \alpha_m \\ \vdots & \alpha_0 & \alpha_1 & \ldots & \alpha_m \\ \vdots & \vdots & \alpha_0 & \ldots & \alpha_m \\ \beta_0 & \beta_1 & \ldots & \beta_m \\ \vdots & \vdots & \vdots & \beta_0 & \beta_1 & \ldots & \beta_m \end{bmatrix} $$

(3.1b)

and the corresponding column

$$ \zeta = \begin{bmatrix} \eta \\ \xi \end{bmatrix} $$

(3.1c)

with \( \eta = \begin{bmatrix} \eta_{N+m-1} \\ \vdots \\ \eta_1 \\ \eta_0 \end{bmatrix} \) \((N + m) s\)-vector of true outputs

and \( \xi = \begin{bmatrix} \xi_{N+m-1} \\ \vdots \\ \xi_1 \\ \xi_0 \end{bmatrix} \) \((N + m) r\)-vector of true inputs.

The unknown parametermatrices to be estimated are collected in the \( s \times (ps + (q + 1)sr) \) matrix \( \Theta \),

$$ \Theta = (\alpha_1 \alpha_2 \ldots \alpha_p \ \beta_0 \beta_1 \ldots \beta_q) $$

(3.2)

(the \( \alpha_i \) and \( \beta_j \) are resp. \( s \times s \) and \( s \times r \) matrices).

In (3.1b) we have

$$ \alpha_0 := -I_s \ , \ \alpha_K := 0 \text{ for } K > p \text{ and } \beta_K := 0 \text{ for } K > q . $$

(3.3)

It is obvious that the \( sN \times (s + r)(N + m) \) matrix \( D \) is rowregular, so its transposed \( D^T \) has full column rank.

The integers \( p, q, r, s \) are known, \( m := \max (p, q) \) and \( N + m \) represents the number of measurements of inputs and outputs.
Introducing $a_i := \alpha_i^T$ and $b_i := \beta_i^T$ ($i = 0, 1, \ldots, m$), we obtain for $D^T$

$$D^T = \begin{bmatrix} a_0 & \cdots & \cdots & a_0 \\ a_1 & \cdots & \cdots & a_1 \\ \vdots & \ddots & \ddots & \vdots \\ a_m & \cdots & \cdots & a_m \\ b_0 & \cdots & \cdots & b_0 \\ b_1 & \cdots & \cdots & b_1 \\ \vdots & \ddots & \ddots & \vdots \\ b_m & \cdots & \cdots & b_m \end{bmatrix}, \ (s + r) (N + m) \times sN$$

(3.4)

consisting of two block-Toeplitz band matrices.

We split up the process of transforming $D^T$ into two steps:

$$\Omega D^T = \begin{bmatrix} S \\ 0 \end{bmatrix}$$

(3.5a)

and

$$P \begin{bmatrix} S \\ 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix},$$

(3.5b)

where $\Omega$ and $P$ are orthogonal,

$$S = \begin{bmatrix} S_{1,1} & \cdots & \cdots & S_{1,(N+m)} \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ S_{(m+1),1} & \cdots & \cdots & S_{(m+1),(N+m)} \\ \vdots & \cdots & \cdots & \vdots \\ S_{N,1} & \cdots & \cdots & S_{N,(N+m)} \\ \vdots & \cdots & \cdots & \vdots \\ S_{N+N,m,1} & \cdots & \cdots & S_{N+N,m,N} \end{bmatrix}, \ s(N + m) \times sN$$

(3.6a)

and
Defining $Q^T := P \Omega$ (orthogonal), (3.5) implies

$$Q^T D^T = \begin{bmatrix} R \\ 0 \end{bmatrix}.$$  

**Step 1:** $\Omega D^T = \begin{bmatrix} S \\ 0 \end{bmatrix}$.

We choose orthogonal matrices $\Omega_1, \Omega_2, \ldots, \Omega_{N+m}$ acting successively on $D^T$ in the following way

$(\Omega = \Omega_{N+m} \Omega_{N+m-1} \ldots \Omega_2 \Omega_1)$:
Some explanation is needed.

In any step $\Omega_i$, Householder transformations are used in order to transform $\begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$ into $\begin{bmatrix} a_0 \\ 0 \end{bmatrix}$, where the transformed pairs $\begin{bmatrix} a_j \\ b_j \end{bmatrix}$ are called $\begin{bmatrix} a_j \\ b_j-1 \end{bmatrix}$ ($j = 0, 1, \ldots, m$) with $b_{-1} = 0$ and $b_m = 0$ if $i > 1$.

Now, the block Toeplitz structure is maintained by applying each Householder transformation on all pairs $\begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$ along the diagonals, moreover on all corresponding pairs $\begin{bmatrix} a_j \\ b_j \end{bmatrix}$, $j = 0, 1, \ldots, m$.

The result of each $\Omega_i$ is a 0-diagonal in the lower matrix and a block-row of $S$.

By using and maintaining the block Toeplitz structure the effect of any $\Omega_i$ is known completely by just computing the effect of $s$ Householder matrices $(r + 1) \times (r + 1)$ on all pairs $\begin{bmatrix} a_j \\ b_j \end{bmatrix}, j = 0, 1, \ldots, m$. 

\begin{equation}
\begin{bmatrix} a_0 \\ \vdots \\ a_m \\ \vdots \\ a_m \\ a_{m-1} \\ \vdots \\ b_0 \\ \vdots \\ b_m-1 \\ \vdots \\ b_m-1 \end{bmatrix} \rightarrow \begin{bmatrix} \alpha \end{bmatrix} \quad (3.7)
\end{equation}
In $D^T$ we have $a_0 = -I_x$.

In order to keep the blocks $a_0$ and thus $s_{k,k}$ ($k = 1,2,\ldots,N$) lower triangular, column $s$ of $\begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$ is attacked first. The nonzero diagonal element of the $a_0$ block will make the column (size $r$) of $b_0$ zero. After applying this Householder transformation on all pairs $\begin{bmatrix} a_j \\ b_j \end{bmatrix}$, $j = 0,1,\ldots,m$, we proceed with column $s-1$ of $\begin{bmatrix} a_0 \\ b_0 \end{bmatrix}$, etc.

**Step 2:** $P \begin{bmatrix} S \\ 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix}$

There will be orthogonal matrices $\tilde{P}_1, \tilde{P}_2,\ldots,\tilde{P}_N$ acting on $S$ as follows (cf. (3.6)):

$$
\begin{bmatrix}
S_{1,1} & \cdots & S_{m+1,1} \\
\vdots & \ddots & \vdots \\
S_{m+1,m+1} & \cdots & S_{m,m+1} \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\star & \cdots & \star & 0 \\
\vdots & \ddots & \vdots & \vdots \\
\star & \cdots & \star & 0 \\
\end{bmatrix}
$$

(3.8)

Defining $\tilde{P} = \tilde{P}_N \tilde{P}_{N-1} \cdots \tilde{P}_2 \tilde{P}_1$ and $P = \text{diag} (\tilde{P}, I)$ (orthogonal),

we obtain $P \begin{bmatrix} S \\ 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix}$.

Each $\tilde{P}_i$ consists of $(ms + 1) \times (ms + 1)$ Householder transformations. The result is a zero (block) column and a block-row of $R$. The square appearing in the matrices has size $ms \times ms$.

The first step of $\tilde{P}_1$ is making zero the very last column (size $ms$) of $S$ by using the Householder matrix determined by this column and the (nonzero) diagonal element of the $s_{N,N}$-block. Proceed with the last but one column, again using the diagonal element of the $s_{N,N}$-block. After $s$ Householders we continue with $\tilde{P}_2$, now using the $s_{N-1,N-1}$-block etc.

As a consequence, the diagonal blocks of $R$, the $r_{i,i}$ are lower triangular and regular. Obviously the same holds for $R$ itself.

The determination of $R$ is the result of the two steps described above. In order to compute $J_N$ and $J_N'$ we also need the $sN$-vector $u$.

From (2.10) it is clear that $u$ is determined by performing the same transformations on the measurement vector
\[ z = \begin{bmatrix} y \\ x \end{bmatrix} \quad \text{(cf. (3.1c))} \]

as we have applied on the columns of \( D^T \).

Since \( R \) is a lower triangular and band matrix the computation of \( \lambda \) from \( R \lambda = u \) is easy to carry out.

Determination of \( J_N \) and \( J_N' \) is now straightforward (cf. (2.11a,b)).

From (3.1b) the \( N \) nonzero elements of \( \hat{D} \) are equal to 1

\[ \left( \cdot = \frac{\partial}{\partial \Theta_i}, \text{ any } i = 1, 2, ..., ps^2 + (q + 1)sr \right). \]
4. Discussion

An upper bound for the number of operations to obtain \( R \) in (3.8) is

\[
s(m+1)(r+1^2(N+m)+s(m+1)(ms+1)^2N,
\]

so with respect to \( N \) the algorithm is of order \( O(N) \).

A \( Q \)-\( R \) algorithm for a general \( m \times n \) matrix \( (m \geq n) \), based on Householder matrices, is of order \( n^2(m - \frac{N}{3}) \) (cf. Golub and Van Loan '83, p. 148).

From (3.1b) it follows

\[
\begin{bmatrix}
d_0 & d_1^T & \ldots & d_m^T \\
d_1 & \ddots & & \vdots \\
\vdots & & \ddots & \vdots \\
d_m & \ldots & & d_1^T \\
\end{bmatrix}
\]

(4.1)

where \( d_k = \sum_{i=0}^{n-1} (\alpha_i \alpha_{i+k}^T + \beta_i \beta_{i+k}^T) \).

So \( D \) is symmetric, block-Toeplitz and band matrix. In literature some algorithms are known to invert such matrices or to solve linear systems, containing those matrices (cf. Jain '78, Trench '74, Meek '83).

For our purpose, the special \( Q \)-\( R \) decomposition introduced here, will be superior to the existing algorithms if \( r, s, m << N \).

For the special case \( r = s = 1 \), the Householder transformations of step 1 in section 3 reduce to Givens transformations.

The matrix of second derivatives of \( J_N(\Theta) \) is not used in the iterative algorithm to find the minimum of \( J_N \) because of its form (cf. Ten Vregelaar '87)

\[
\frac{\partial^2 J_N}{\partial \Theta_j \partial \Theta_i} = 2\bar{z}^T [P^i(D^j)^T(DD^T)^{-1}D^iP^j - D^+(D^iD^jD^+P^j + D^iD^jD^+)P^i - D^+D^jP^j(D^j)^T(D^+)^T] \bar{z} ,
\]

(4.2)

with \( D^i := \frac{\partial}{\partial \Theta_i} D \).
The minimization problem is solved by using the Broyden-Fletcher-Goldfarb-Shanno formula (cf. Scales '85 p. 89-90).

If some relations exist between the parameters $\alpha_1, ..., \alpha_p, \beta_0, ..., \beta_q$ we have to solve a constrained optimization problem. The described algorithm to compute the object function and its gradient can be applied for this case as well.
References

**Golub and Van Loan '83**
Golub G.H., Van Loan, C.F.,
Matrix Computations, North Oxford Academic,

**Jain '78**
Jain, A.K.,

**Meek, '83**
Meek, D.S.,
The inverses of Toeplitz band matrices,

**Scales '85**
Scales, L.E.,
Introduction to Non-Linear Optimization,

**Ten Vregelaar '87**
Ten Vregelaar, J.M.,

**Trench '74**
Trench, W.F.,
Inversion of Toeplitz band matrices,