Modern Applied Science
www.ccsenet.org/journal.html
Vol. 3, No. 5
May 2009

One New Method on ARMA Model Parameters Estimation

Xiaoqin Cao (Corresponding author)
Department of Science, Yanshan University
Qin Huangdao 066004, China
E-mail:fanjing850328@163.com
Rui Shan
Department of Science, Yanshan University
Qin Huangdao 066004, China
E-mail: srlhy@hotmail.com
Jing Fan & Peiliang Li
Department of Science, Yanshan University
Qin Huangdao 066004, China

Abstract

The estimation of ARMA model parameters really belongs to the least-square problem,in ARMA model because the residual are calculated by given time series,the time series and parameter are nonlinear.However it is difficult to calculate the derivative of objective function.This paper substitutes derivative with difference,then calculate the first derivative and the second derivative of objective function.Finally we prove that, under suitable hypotheses, the proposed algorithm converges globally.

Keywords: ARMA(n,m), Trust-region, BFGS

1. Introduction

For nonlinear least-squares problem,we minimize ε_t in the sense of the sum of squares.We can take full advantage of their special structure to design a more effective method.For example,Gauss-Newton method,damping Gauss-Newton method & LM method and so on. However Gauss-Newton method just use the information of the first derivative (V_k) of the function to get the approximation of the Hesse matrix,however neglect the nonlinear items of the $\nabla^2 S(\beta)$,therefore the performance and convergence of the Gauss-Newton method will certainly be affected.because of these problems,we present a new method.

In ARMA model,consider ε_{t-1} ε_{t-2} ε_{t-m} ... are calculated by x_{t-1} x_{t-2} ..., therefore X_t and β are nonlinear.However when we only know the form of ARMA model,we are unable to directly express ε_{t-1} $\varepsilon_{t-2} \ldots \varepsilon_{t-m}$ with x_{t-1} x_{t-2} ... Therefore we substitute derivative with difference,and then calculate the approximation of the gradient and the Hesse matrixof the objective function.

Trust region methods are a class of optimization methods to guarantee the overall convergence of technologies.Although the trust region method can be traced back to Levenberg(1944),Marquardt(1963),Goldfeld,Quandt & Trotter(1966),the modern trust region method is raised by Powell (1970).He clearly posed the trust region subproblem and the convergence theorem.These measures show the trust region has the greater advantages than the linear search method.We make a small improvement of the traditional trust region method,consider the non-linear degree of the nonlinear least-squares problem,and correct the Hesse matrix of the objective function from two respects to make the algorithm has a better nature.

2. The basic theory

2.1 Objective function

For ARMA(n,m) model

$$x_t = \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \ldots + \varphi_n x_{t-n} - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \ldots - \theta_m \varepsilon_{t-m} + \varepsilon_t \tag{1}$$

Substitute (1) with the form of the matrix

$$x_t = f(X_t, \beta) + \varepsilon_t$$

Where

$$X_t = \left[x_{t-1}\ x_{t-2} \cdots x_{t-n}\quad \varepsilon_{t-1}\ \varepsilon_{t-2} \cdots \varepsilon_{t-m}\right]^T$$

$$\beta = \left[\varphi_1\ \varphi_2 \cdots \varphi_n\quad \theta_1\ \theta_2 \cdots \theta_m\right]^T$$

$$f(X_t, \beta) = X_t^T \beta \tag{2}$$

The so-called parameters estimation is to select the appropriate model parameters β ,so that the residual sum of squares of the model is minimal

$$B(\beta) = \nabla^2 S(\beta) = V(\beta)^T V(\beta) + \sum_{t=n+1}^{N} \varepsilon_i(\beta) \nabla^2 \varepsilon_i(\beta) \sum_t [X_t, \beta)]^2 \tag{3}$$

For ARMA(n,m) model,because $f(X_t, \beta)$ is nonlinear function,this problem is called non-linear least-squares problem,we can adopt a variety of iterative methods of optimization theory to calculate it,finally obtain model parameters β which make the objective function $S(\beta)$ minimal.

2.2 Determination of initial value

At the iterative calculations of nonlinear least-squares,we should give the initial value of iterative calculation.

2.2.1 Determination of parameters initial value β^0

The selection of the parameters initial value β^0 is extremely important,which relates to the convergence speed of iterative calculation,this paper uses the long autoregression model of AR(n_0).

The transfer function of the equivalent system described by AR(n_0) model is

$$\frac{1}{\varphi_{n_0}^0(B)} = \frac{1}{1 - \sum_{i=1}^{n_0} I_i B^i} \tag{4}$$

Where, I_i is inverse function, I_i equal AR model parameters φ_i^0

$$I_0 = -1, I_j = \begin{cases} \varphi_j^0 & (1 \le j \le k); \\ 0 & (j > k). \end{cases}$$

the transfer function of the equivalent system described by ARMA(n,m) model is

$$\frac{\theta_n^0(B)}{\varphi_m^0(B)} = \frac{1 - \sum_{j=1}^{n} \theta_j^0 B^j}{1 - \sum_{i=1}^{m} \varphi_i^0 B^i} \tag{5}$$

Because of the transfer function described in the system are equivalent,(4) and (5) should be equal,namely

$$\left(1 - I_1 B - I_2 B^2 - \cdots - I_{n_0} B^{n_0}\right)\left(1 - \theta_1^0 B - \theta_2^0 B^2 - \cdots - \theta_m^0 B^m\right) =$$
$$1 - \varphi_1^0 B - \varphi_2^0 B^2 - \cdots - \varphi_n^0 B^n \tag{6}$$

Compare the same power coefficient of B on both sides of (6), we have

$$\begin{cases} \varphi_1^0 = \theta_1^0 + I_1; \\ \varphi_2^0 = \theta_2^0 - \theta_1^0 I_1 + I_2; \\ \varphi_3^0 = \theta_3^0 - \theta_2^0 I_1 - \theta_1^0 I_2 + I_3; \\ \qquad \vdots \\ \varphi_n^0 = -\theta_m^0 I_{n-m} - \cdots - \theta_2^0 I_{n-2} - \theta_1^0 I_{n-1} + I_n; \\ 0 = -\theta_m^0 I_{k-m} - \cdots - \theta_2^0 I_{k-2} - \theta_1^0 I_{k-1} + I_k; \quad (k > n) \end{cases} \tag{7}$$

For the first n equations of the formula,when θ_j^0 is known,this is the linear equations about φ_i^0, φ_i^0 can be easily solved

$$\begin{bmatrix}\varphi_1^0\\ \varphi_2^0\\ \varphi_3^0\\ \vdots\\ \varphi_n^0\end{bmatrix}=\begin{bmatrix}\theta_1^0\\ \theta_2^0\\ \theta_3^0\\ \vdots\\ \theta_n^0\end{bmatrix}+\begin{bmatrix}1 & 0 & 0 & \cdots & 0\\ -\theta_1^0 & 1 & 0 & \cdots & 0\\ -\ \theta_2^0 & -\theta_1^0 & 1 & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ -\theta_{n-1}^0 & -\theta_{n-2}^0 & -\theta_{n-3}^0 & \cdots & 1\end{bmatrix}\begin{bmatrix}I_1\\ I_2\\ I_3\\ \vdots\\ I_n\end{bmatrix} \tag{8}$$

When $j > m$,we set $\theta_j^0 = 0$ .For the last formula in (7),we separately set $k = n+1, n+2, \cdots, n+m$,and $n+m = p$,written by matrix form

$$\begin{bmatrix}I_{n+1}\\ I_{n+2}\\ \vdots\\ I_{n+m}\end{bmatrix}=\begin{bmatrix}I_n & I_{n-1} & I_{n-2} & \cdots & I_{n+1-m}\\ I_{n+1} & I_n & I_{n-1} & \cdots & I_{n+2-m}\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ I_{n+m-1} & I_{n+m-2} & I_{n+m-3} & \cdots & I_n\end{bmatrix}\begin{bmatrix}\theta_1^0\\ \theta_2^0\\ \vdots\\ \theta_m^0\end{bmatrix} \tag{9}$$

this is the linear equations about θ_j^0 , θ_j^0 can be easily solved.Therefore,we first solve (9) to obtain θ_j^0 ,then solve (8) to obtain φ_i^0 ,this is the calculation principle of long autoregression model.

2.2.2 Determination of residuals initial value ε_t

Residual value can be determined by the follow formula

$$\begin{cases}\varepsilon_1 = x_1\\ \varepsilon_2 = x_2 - \varphi_1 x_1 + \theta_1\varepsilon_1\\ \varepsilon_3 = x_3 - \varphi_1 x_2 - \varphi_2 x_1 + \theta_1\varepsilon_2 + \theta_2\varepsilon_1\\ \qquad\vdots\\ \varepsilon_n = x_n - \varphi_1 x_{n-1} - \cdots - \varphi_{n-1}x_1 +\\ \qquad\theta_1\varepsilon_{n-1} + \cdots + \theta_m\varepsilon_{n-m}\end{cases} \tag{10}$$

2.3 The calculation of the gradient $g(\beta)$ and Hesse matrix $B(\beta)$ of the $S(\beta)$

We conduct Taylor expansion for x_t in ARMA model at $\boldsymbol{\beta}^0$

$$x_t = f\left(X_t,\boldsymbol{\beta}^0\right)+\left(\beta_1-\beta_1^0\right)\frac{\partial f\left(X_t,\boldsymbol{\beta}\right)}{\partial\beta_1}\bigg|_{\boldsymbol{\beta}=\boldsymbol{\beta}^0}+\left(\beta_2-\beta_2{}^0\right)\frac{\partial f\left(X_t,\boldsymbol{\beta}\right)}{\partial\beta_2}+\cdots+$$

$$\left(\beta_p-\beta_p^0\right)\frac{\partial f\left(X_t,\boldsymbol{\beta}\right)}{\partial\beta_p}\bigg|_{\boldsymbol{\beta}=\boldsymbol{\beta}^0}+o\left(\left(\beta_1-\beta_1^0\right)^2+\left(\beta_2-\beta_2^0\right)^2+\cdots+\left(\beta_p-\beta_p^0\right)^2\right) \tag{11}$$

2.3.1 The calculation of the vector ε_t^k

$$\varepsilon_t^0 = x_t - \sum_{i=1}^{n}\varphi_i^0 x_{t-i} + \sum_{j=1}^{m}\theta_j^0\varepsilon_{t-j}^0 \qquad \text{(t=n+1,n+2,}\cdots\text{ N)} \tag{12}$$

Where,the initial value of the residual $\varepsilon_1^0, \varepsilon_2^0, \cdots \varepsilon_{n-1}^0, \varepsilon_n^0$ are calculated by (10),vector $\varepsilon_0 = \left[\varepsilon_{n+1}^0\ \varepsilon_{n+2}^0 \cdots \varepsilon_N^0\right]^T$ .Therefore $\varepsilon_k = \left[\varepsilon_{n+1}^k\ \varepsilon_{n+2}^k \cdots \varepsilon_N^k\right]^T$.

2.3.2 The calculation of the vector V_k

$$V_{it}^k = -\frac{\partial\varepsilon_t}{\partial\beta_i}\bigg|_{\beta=\beta_k} \approx -\frac{\varepsilon_t(\beta^k+\Delta_i)-\varepsilon_t(\beta^k)}{\Delta} \tag{14}$$

Where, Δ_i is p=n+m vector $\Delta_i = \left[0\cdots 0\ \Delta\ 0\cdots 0\right]^T$,combining (12) with (14),we have

$$V_{it}^{k}=\begin{cases}x_{t-i} & (1\le i\le n)\\ -\varepsilon_{t-j}^{k} & (n+1\le i\le n+m=p, j=i-n)\end{cases} \tag{15}$$

We let $t=n+1,n+2,\cdots,N; i=1,2\cdots p=n+m$,

$$V^{k}=\begin{bmatrix} x_{n} & x_{n-1} & \cdots & x_{1} & -\varepsilon_{n}^{k} & -\varepsilon_{n-1}^{k} & \cdots & -\varepsilon_{n-m+1}^{k}\\ x_{n+1} & x_{n} & \cdots & x_{2} & -\varepsilon_{n+1}^{k} & -\varepsilon_{n}^{k} & \cdots & -\varepsilon_{n-m+2}^{k}\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\ x_{N-1} & x_{N-2} & \cdots & x_{N-n} & -\varepsilon_{N-1}^{k} & -\varepsilon_{N-2}^{k} & \cdots & -\varepsilon_{N-m}^{k}\end{bmatrix} \tag{16}$$

Because ε_{t-1} $\varepsilon_{t-2}\ldots\varepsilon_{t-m}$ in x_t are unknown,which are calculated by x_{t-1} x_{t-2} ...,however we are unable to show ε_{t-1} $\varepsilon_{t-2}\ldots\varepsilon_{t-m}$,then the gradient $g(\beta)$ and *Hesse matrix* $B(\beta)$ are unable to be calculated.We substitute derivative with difference to calculate the gradient $g(\beta)$ and *Hesse matrix* $B(\beta)$ *of* $S(\beta)$.

$$g(\beta)=\nabla S(\beta)=V(\beta)^{T}\varepsilon(\beta)$$

$$B(\beta)=\nabla^{2}S(\beta)=V(\beta)^{T}V(\beta)+\sum_{t=n+1}^{N}\varepsilon_{i}(\beta)\nabla^{2}\varepsilon_{i}(\beta) \tag{17}$$

If the non-linear degree of the this problem is relatively high,we approximate Hesse matrix by BFGS formula;If the non-linear degree of the this problem is relatively low,we approximate Hesse matrix by first derivative.

2.4 Trust region method

Trust region method is iterative method,at each iteration,we solve trust region subproblem

$$\begin{cases}\min q^{k}(d)=S\left(\boldsymbol{\beta}^{k}\right)+(g^{k})^{T}d+\frac{1}{2}d^{T}B^{k}d\\ s.t.\|d\|\le\Delta_{k}\end{cases} \tag{18}$$

Where $g(\beta)=\nabla S(\beta)$ or its approximation, Δ_k is trust region radius.

At this iterative point $\boldsymbol{\beta}^{k}$,we solve the trust region subproblem to obtain the trial step d_k ,we consider the actual reduction given by

$$\Delta S_{k}=S_{k}-S_{k+1} \tag{19}$$

and the predictable reduction will be given by

$$\Delta S_{k}^{\ pred}=S_{k}-q^{k}(d_{k}) \tag{20}$$

Define their ratio

$$r^{k}=\frac{\Delta S_{k}}{\Delta S_{k}^{\ pred}} \tag{21}$$

Trust region method adjust the radius of trust region by the information of their ratio .

3. Algorithm

Step 1 given time series { x_t }(t=1,2, $\cdots$ N),model order n,m and the initial value of the model parameters $\boldsymbol{\beta}^{0}=\left[\varphi_{1}^{0},\cdots,\varphi_{n}^{0},\theta_{1}^{0},\cdots,\theta_{m}^{0}\right]^{T}$.

Step 2 calculate { ε_{t}^{k} }(t=n+1,n+2, $\cdots$ N) ,form vector ε^{k} ,calculatev V^{k} by (16).Then calculat $S^{k}=S\left(\boldsymbol{\beta}^{k}\right)=\left(\varepsilon^{k}\right)^{T}\varepsilon^{k}$, $g^{k}=\left(V^{k}\right)^{T}\varepsilon^{k}$ and $B^{k}=\left(V^{k}\right)^{T}V^{k}$ .set $k=0$, $\Delta_{0}=\|g_{0}\|, 0<c_{1}<c_{2}<1$.

Step 3 solve the subproblem (18) and obtain d_k

Step 4 Set $\boldsymbol{\beta}^{k+1}=\boldsymbol{\beta}^{k}+d^{k}$,calculate $S\left(\boldsymbol{\beta}^{k+1}\right)$

if $S\left(\boldsymbol{\beta}^{k+1}\right)>S\left(\boldsymbol{\beta}^{k}\right)$ set $\Delta_{k+1}=\frac{\Delta_{k}}{2}$,go to step 3;

else if $S\left(\boldsymbol{\beta}^{k+1}\right) \ge S\left(\boldsymbol{\beta}^{k}\right) + \rho(g^k)^T(\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^k)$,

set $\Delta_{k+1} = -\Delta_k \dfrac{(g^k)^T d^k}{2(S^{k+1} - S^k - (g^k)^T d)}$,go to step 3;

else go to step 5.

Step 5 if $S^{k+1} - S^k \le \varepsilon \max(1, S^k)$,stop;else go to step 6.

Step 6 calculate V^{k+1} by (16)

Step 7 update trust region radius

$$\Delta_{k+1} = \begin{cases} 2\Delta_k, r_k \ge c_2 \\ \dfrac{\Delta_k}{4}, r_k \le c_1 \\ \Delta_k, c_1 \le r_k \le c_2 \end{cases} \tag{24}$$

Step8 update B^k ,set k=k+1,go to step 3.

$$B^{k+1} = \begin{cases} B_k + \dfrac{y^k (y^k)^T}{\left(d^k\right)^T y^k} - \dfrac{\boldsymbol{B}^k d^k \left(d^k\right)^T \boldsymbol{B}^k}{d^k \boldsymbol{B}^k d^k}, \dfrac{S_k - S_{k+1}}{S_k} < \rho_1, \rho_1 \in (0,1) \\ (V^{k+1})^T V^{k+1}, otherwise \end{cases} \tag{25}$$

4. The analysis of the convergence property

Assumptions:

1) The level set $L(\beta) = \left\{\beta \middle| S(\beta) \le S(\beta_0)\right\}$ is bounded.

2) Hesse approximation matrix B_k is uniformly bounded according to norm,namely $\|B_k\| \le M$, M is a positive constant.

3) The function S is second continuously differentiable and has the lower bound at level set.

Lemma *1 if* d_k is the solution of (18), then it satisfies

$$(g_k)^T d_k \le -\frac{1}{2}\|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}$$

$$(g_k)^T d_k + \frac{1}{2} d_k{}^T B_k d_k \le -\frac{1}{2}\|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}$$

Lemma *2* Assume { β_k } are iterative sequence generated by the algorithm,then $S(\beta_k)$ is monotonous and non-increasing.

Proof:assume $\alpha_k = \dfrac{p}{2L\|d_k\|^2}, (L \ge 1)$,

where $p = \|g_k\| \min\left\{\Delta_k, \dfrac{\|g_k\|}{\|B_k\|}\right\}, \beta_{k+1} = \beta_k + \alpha_k d_k$

By the mean value theorem,there exists

$$S(\beta_{k+1}) - S(\beta_k) = \bar{g}^T(\beta_{k+1} - \beta_k)$$

Where $\bar{g} = g(\bar{x}), \bar{x} \in [x_{k+1}, x_k]$,by Lipschitz conditions ,we have $\|x - x_k\| \le \alpha_k d_k$ and

$$\begin{aligned} \bar{g}^T(\beta_{k+1} - \beta_k) &= \bar{g}_k^T(\beta_{k+1} - \beta_k) + (\bar{g} - g_k)^T(\beta_{k+1} - \beta_k) \\ &\le \alpha_k g_k{}^T d_k + \alpha_k L(\beta_{k+1} - \beta_k)^T d_k \\ &= \alpha_k g_k{}^T d_k + \alpha_k^2 L\|d_k\|^2 \end{aligned}$$

$$= \alpha_k g_k{}^T d_k + \alpha_k \frac{p}{2\|d_k\|^2}\|d_k\|^2$$

By lemma 1,we have $(g_k)^T d_k \le -\frac{p}{2}$, so $\overline{g}^T(\beta_{k+1}-\beta_k) \le \alpha_k(-\frac{p}{2}+\frac{p}{2}) = 0$

Therefore, $S(\beta_k)$ is monotonous and non-increasing.

Next we analyze the global convergence of this algorithm.

Theorem 1 If the assumptions hold,then the sequence generated by the algorithm satisfies $\lim_{k\to\infty}\inf\|g_k\| = 0$ ·

Proof:Assume the conclusion does not hold,that is to say here exists $\varepsilon > 0$,for all k ,we have $\|g_k\| \ge \varepsilon$ and $-q^k(d^k) \ge \frac{1}{2}\|g_k\|\min\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}$,

Because Δ_k has lower bound which is recorded as $\Delta\eta$ and $\|B_k\| \le M$,we have

$$-q^k(d^k) \ge \frac{1}{2}\varepsilon\min\left\{\Delta\eta, \frac{\varepsilon}{M}\right\}$$

This show $-q^k(d^k)$ has lower bound,which is contradictory with $\lim_{k\to\infty} - q^k(d^k) = 0$

This proposition is proved.

References

Y Dai. (2003). Convergence properties of the BFGS a lgorithm. *Journal on Optimization*, 2003, (13): 693~-701.

Yang,, Shuzi & Wu, Ya. (2007). *The engineering applications in time series analysis*. Huazhong University of Science and Technology Publishing House.

Yuan, Gonglin & Wei, Zengxin. (2004). One new BFGS trust region algorithm.Guangxi Science. 2004, 11, 195~196.

Zhang, Changhai, Fan, Ying, & Zhang, Lianbing. (1993). An amendment to the trust region method and its convergence. Heilongjiang Science and Technology Press, 1993, 64~65.

Zhang, Jianahong & Deng,Naiyang. (1999). A new quasi-Newton equation and related methods for unconstrained optimrzation. 1999,102(3); 147~187.