Spring 2015 Statistics 153 (Time Series) : Lecture Sixteen

Aditya Guntuboyina

19 March 2015

1 Asymptotic Distribution of the Estimates for AR parameters

The following holds for each of the Yule-Walker, Conditional Least Squares and ML estimates:

For n large, the approximate distribution of $\sqrt{n}\left(\hat{\phi} - \phi\right)$ is normal with mean 0 and variance covariance matrix $\sigma_Z^2 \Gamma_p^{-1}$ where Γ_p is the $p \times p$ matrix whose (i,j)th entry is $\gamma_X(i-j)$.

A rough sketch of the proof of this can be found in the notes for the last lecture.

1.1 Special Instances

In the AR(1) case:

$$\Gamma_p = \Gamma_1 = \gamma_X(0) = \sigma_Z^2/(1-\phi^2).$$

Thus $\hat{\phi}$ is approximately normal with mean ϕ and variance $(1-\phi^2)/n$.

For AR(2), using

$$\gamma_X(0) = \frac{1-\phi_2}{1+\phi_2} \frac{\sigma_Z^2}{(1-\phi_2)^2 - \phi_1^2} \qquad \text{and} \qquad \rho_X(1) = \frac{\phi_1}{1-\phi_2},$$

we can show that $(\hat{\phi}_1, \hat{\phi}_2)$ is approximately normal with mean (ϕ_1, ϕ_2) and variance-covariance matrix is $1/n$ times

$$\begin{pmatrix} 1-\phi_2^2 & -\phi_1(1+\phi_2) \\ -\phi_1(1+\phi_2) & 1-\phi_2^2 \end{pmatrix}$$

Note that the approximate variances of both $\hat{\phi}_1$ and $\hat{\phi}_2$ are the same. Observe that if we fit AR(2) model to a dataset that comes from AR(1), then the estimate of $\hat{\phi}_1$ might not change much but the standard error will be higher. We lose precision. See Example 3.34 in the book.

2 More General ARMA model fitting

2.1 Invertibility

Consider the case of the MA(1) model whose acvf is given by $\gamma_X(0) = \sigma_Z^2(1+\theta^2)$ and $\gamma_X(1) = \theta\sigma_Z^2$ and $\gamma_X(h) = 0$ for all $h \geq 2$. It is easy to see that for $\theta = 5, \sigma_Z^2 = 1$, we get the same acvf as for $\theta = 1/5, \sigma_Z^2 = 25$. In other words, there exist different parameter values that give the same acvf. More generally, the parameter pairs (θ, σ_Z^2) and $(1/\theta, \theta^2\sigma_Z^2)$ correspond to the same acvf.

This implies that one **can not uniquely** estimate the parameters of an MA(1) model from data. A natural fix is to consider only those MA(1) for which $|\theta| < 1$. This condition is called *invertibility.* The condition $|\theta| < 1$ for the MA(1) model is equivalent to stating that the moving average polynomial $\theta(z)$ has all roots of magnitude strictly larger than one. This gives the general definition of invertibility for an ARMA process.

An ARMA model $\phi(B)(X_t - \mu) = \theta(B)Z_t$ is said to be invertible if all roots of the moving average polynomial $\theta(z)$ have magnitude strictly larger than one. It can be shown (in analogy with causality) that this condition is equivalent to Z_t being written as a linear combination of the present and past values of X_t alone.

From now on, we shall only consider stationary, causal and invertible ARMA models i.e., we shall assume that both the polynomials $\phi(z)$ and $\theta(z)$ do not have any roots in the unit disk.

Let us now study the problem of fitting a stationary, causal and invertible ARMA model to data assuming that the orders p and q are known.

Each of the three methods for the AR model fitting carry over (with additional complications) to the general ARMA case. It is easiest to start by learning the relevant R function. The function to use is *arima*(). We will see later that ARIMA is a more general class of models that include the ARMA models as a special case (actually, ARIMA is just differencing + ARMA). This function *arima*() can be used to fit ARMA models to data. It also has a method argument that has three values: CSS-ML, ML and CSS, the default being CSS-ML.

2.2 Yule-Walker or Method of moments

This proceeds, in principle, by solving some subset of the following set of equations for the unknown parameters $\theta_1, \dots, \theta_q, \phi_1, \dots, \phi_p$ and σ_Z^2 (and μ is estimated by the sample mean)

$$\hat{\gamma}(k) - \phi_1\hat{\gamma}(k-1) - \cdots - \phi_p\hat{\gamma}(k-p) = (\psi_0\theta_k + \psi_1\theta_{k+1} + \cdots + \psi_{q-k}\theta_q)\sigma_Z^2$$

for $0 \le k \le q$ and

$$\hat{\gamma}(k) - \phi_1\hat{\gamma}(k-1) - \cdots - \phi_p\hat{\gamma}(k-p) = 0 \qquad \text{for } k > q.$$

Note that ψ_j above are functions of $\theta_1, \dots, \theta_q$ and $\phi_1, \dots, \phi_p$.

This method of estimation has the following problems:

1. It is cumbersome (unless we are in the pure AR case): Solutions might not always exist to these equations (for example, in the MA(1), this method entails solving $r_1 = \theta/(1+\theta^2)$ which of course does not have a solution when $r_1 \notin [-0.5, 0.5]$). The parameters are estimated in an arbitrary fashion when these equations do not have a solution.
2. The estimators obtained are *inefficient.* The other techniques below give much better estimates (smaller standard errors).

Because of these problems, no one uses method of moments for estimating the parameters of a general ARMA model. R does not even have a function for doing this. Note, however, that both of these problems disappear for the case of the pure AR model.

There are basically two methods for fitting general ARMA models (Yule-Walker is not really used unless we are fitting AR models): these are conditional least squares and maximum likelihood.

3 Conditional Least Squares

Let us first consider the special case of the MA(1) model: $X_t - \mu = Z_t + \theta Z_{t-1}$. We want to fit this model to data $x_1, \dots, x_n$. If the data were indeed generated from this model, then

$$Z_1 = x_1 - \mu - \theta Z_0; Z_2 = x_2 - \mu - \theta Z_1; \dots; Z_n = x_n - \mu - \theta Z_{n-1}.$$

If we set Z_0 to its mean 0, then for **every fixed values** of θ and μ, we can recursively calculate $Z_1, \dots, Z_n$. We can then compute the sum of squares $\sum_{i=1}^n Z_i^2$. This value would change for different values of θ. We would then choose the value of θ for which it is small (this is accomplished by an optimization procedure).

This is also called conditional least squares because this minimzation is obtained when one tries to maximize the conditional likelihood of the data conditioning on $z_0 = 0$.

Note that conditional likelihood works differently in the AR(1) case compared to the MA(1) case. It works in a yet another different way in the ARMA(1, 1) case for example. Here the model is $X_t - \mu - \phi(X_{t-1} - \mu) = Z_t + \theta Z_{t-1}$. Here it is convenient to set Z_1 to be zero. Then we can write

$$Z_2 = x_2 - \mu - \phi(x_1 - \mu); Z_3 = x_3 - \mu - \phi(x_2 - \mu) - \theta Z_2; \dots; Z_n = x_n - \mu - \phi(x_{n-1} - \mu) - \theta Z_{n-1}.$$

After this, one forms the sum of squares $\sum_{i=2}^n Z_i^2$ which can be computed for every fixed values of θ, ϕ and μ. One then minimizes these resulting sum of squares over different values of the unknown parameters.

For a general ARMA(p, q) model:

$$X_t - \mu - \phi_1(X_{t-1} - \mu) - \dots - \phi_p(X_{t-p} - \mu) = Z_t + \theta_1 Z_{t-1} + \dots + \theta_q Z_{t-q},$$

we set $Z_t = 0$ for $t \leq p$ and calculate recursively

$$Z_t = X_t - \mu - \phi_1(X_{t-1} - \mu) - \dots - \phi_p(X_{t-p} - \mu) - \theta_1 Z_{t-1} - \dots - \theta_q Z_{t-q}$$

for $t = p+1, \dots, n$. This is equivalent to writing the likelihood coniditioning on $X_1, \dots, X_p$ and $Z_t = 0$ for $t \leq p$. If $q = 0$ (AR models), minimizing the sum of squares is equivalent to linear regression and no iterative technique is needed. If $q > 0$, the problem becomes nonlinear regression and numerical optimization routines need to be used.

In R, this method is performed by calling the function *arima*() with the method argument set to *CSS* (CSS stands for conditional sum of squares).

4 Maximum Likelihood

This method is simple in principle. Assume that the errors $\{Z_t\}$ are gaussian. Write down the likelihood of the observed data $x_1, \dots, x_n$ in terms of the unknown parameter values $\mu, \theta_1, \dots, \theta_q$, $\phi_1, \dots, \phi_p$ and σ_Z^2. Maximize over these unknown parameter values.

It is achieved in R by calling the function *arima*() with the method argument set to *ML* or *CSS-ML*. ML stands of course for Maximum Likelihood. R uses an optimization routine to maximize the likelihood. This routine is iterative and needs suitable initial values of the parameters to start. In CSS-ML, R selects these starting values by CSS. I do not quite know how the starting values are selected in ML. The default method for the arima function in R is CSS-ML. The R output for the methods CSS-ML and ML seems to be identical.

5 Asymptotic Distribution of ARMA ML Estimates

Let $\beta = (\phi_1, \ldots, \phi_p, \theta_1, \ldots, \theta_q)$. The distribution of $\hat{\beta}$ is approximately normal with mean β and variance covariance matrix $\sigma_Z^2 \Gamma_{p,q}^{-1}/n$ where $\Gamma_{p,q}$ is a $(p+q) \times (p+q)$ matrix of the form:

$$\begin{pmatrix} \Gamma_{\phi\phi} & \Gamma_{\phi\theta} \\ \Gamma_{\theta\phi} & \Gamma_{\theta\theta} \end{pmatrix}$$

The (i,j)th entry of the $p \times p$ matrix $\Gamma_{\phi\phi}$ equals $\gamma_A(i-j)$ where $\{A_t\}$ is the AR(p) process: $\phi(B)A_t = Z_t$. Similarly, the $q \times q$ matrix $\Gamma_{\theta\theta}$ has (i,j)th entry equalling $\gamma_B(i-j)$ for the AR(q) process $\theta(B)B_t = Z_t$. The (i,j)th entry of $\Gamma_{\phi\theta}$ equals the covariance between A_i and B_j.

6 ARIMA Models

ARIMA is essentially differencing plus ARMA. We have seen previously that differencing is commonly used on time series data to remove trends and seasonality.

For example, differencing can be used for

1. Removing polynomial trends: Suppose the data come from the model $Y_t = \mu_t + X_t$ where μ_t is a polynomial of order k and X_t is stationary, then differencing of order k: $\nabla^k Y_t = (I-B)^k Y_t$ results in stationary data to which an ARMA model can be fit.
2. Random walk models: Suppose that the data come from the random walk model: $Y_t = Y_{t-1} + X_t$ where X_t is stationary. Then clearly $\nabla Y_t = X_t$ is stationary and an ARMA model can be fit to this difference data.

Such models, which after appropriate differencing, reduce to ARMA models are called ARIMA models.

Definition 6.1 (ARIMA). *A process Y_t is said to be ARIMA(p,d,q) with mean μ if $X_t = (I-B)^d Y_t$ is ARMA(p, q) with mean μ. In other words:*

$$\phi(B)(X_t - \mu) = \theta(B)Z_t,$$

where $\{Z_t\}$ is white noise.