STAT 804: Notes on Lecture 6

The Multivariate Normal Distribution

Definition: $Z \in R^1 \sim N(0,1)$ iff

$$f_Z(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2}$$

Definition: $Z \in R^p \sim MVN_p(0,I)$ iff $Z = (Z_1, \ldots, Z_p)'$ (a column vector for later use) with the Z_i independent and each $Z_i \sim N(0,1)$.

In this case,

$$\begin{aligned} f_Z(z_1, \ldots, z_p) &= \prod \frac{1}{\sqrt{2\pi}} e^{-z_i^2/2} \\ &= (2\pi)^{-p/2} \exp\{-z'z/2\} \end{aligned}$$

where the superscript $'$ denotes matrix transpose.

Definition: $X \in R^p$ has a multivariate normal distribution if it has the same distribution as $AZ + \mu$ for some $\mu \in R^p$, some $p \times q$ matrix of constants A and $Z \sim MVN_q(0,I)$.

Lemma: The matrix A can be taken to be square with no loss of generality.

Proof: The simplest proof involves multivariate characteristic functions:

$$\phi_X(t) = \mathrm{E}(e^{it'X})$$

You check that for $Z \sim MVN_p(0,I)$ we have

$$\phi_Z(t) = e^{-t't/2}$$

Then

$$\begin{aligned} \phi_{AZ+\mu}(u) &= \mathrm{E}(e^{iu'(AZ+\mu)}) \\ &= e^{iu'\mu} \phi_Z(A'u) \\ &= e^{iu'\mu - u'AA'u/2} \end{aligned}$$

Since the result depends only on μ and $\Sigma = AA'$ the distribution is the same for any two A giving the same AA'. In particular given an A which is $p \times q$ we need only use ideas of Cholesky decomposition to find a square B with $BB' = AA'$. I omit a proof.

If the matrix A is singular then X will not have a density. If A is invertible then we can derive the multivariate normal density by the change of variables formula:

$$X = AZ + \mu \Leftrightarrow Z = A^{-1}(X - \mu)$$

We get

$$f_X(x) = (2\pi)^{-p/2} \det(\Sigma)^{-1/2} \exp\{-\frac{1}{2}(x-\mu)'\Sigma^{-1}(x-\mu)\}$$

where $\Sigma = AA'$.

Properties of the MVN distribution

1. If $X \sim MVN(\mu, \Sigma)$ then $MX + b \sim MVN(A\mu + b, A\Sigma A')$ follows easily from our definition. (You work with the real definition not densities, etc.)

2. All margins are multivariate normal: if

$$X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$$

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}$$

   and

$$\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}$$

   then $X \sim MVN(\mu, \Sigma)$ implies that $X_1 \sim MVN(\mu_1, \Sigma_{11})$. This is really a special case of the previous property.

3. All conditionals are normal: the conditional distribution of X_1 given $X_2 = x_2$ is $MVN(\mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})$ To prove this you have to do algebra with the definition of the conditional density. It is possible to give a sensible formula even if Σ_{22} is singular.

Partial Correlations

If X is an $AR(p)$ process then

$$X_t = \sum_{j=1}^{p} a_j X_{t-j} + \epsilon_t$$

If you hold $X_{t-1}, \ldots, X_{t-p}$ fixed then X_t is ϵ_t plus a fixed constant. Since ϵ_t is independent of X_s for all $s < t$ we see that given $X_{t-1}, \ldots, X_{t-p}$ the variables X_t and X_{t-r} for $r > p$ are independent. This would imply

$$\text{Cov}(X_t, X_{t-q} | X_{t-1}, \ldots, X_{t-p}) = 0$$

for all $q > p$. When the process X is Gaussian our results above on conditional distributions guarantee that we can calculate this **partial autocovariance** from the variance covariance matrix of the vector $(X_t, X_{t-q}, X_{t-1}, \ldots, X_{t-p})$. If we partition this vector into a first piece with 2 entries and a second piece with p entries we find that the conditional variance covariance matrix of (X_t, X_{t-q}) given the others is

$$\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$$

In this formula Σ_{11} is the 2 by 2 unconditional covariance matrix of (X_t, X_{t-q}), namely,

$$\begin{bmatrix} C(0) & C(q) \\ C(q) & C(0) \end{bmatrix}$$

The matrix Σ_{22} is the $p \times p$ Toeplitz matrix with $C(0)$ down the diagonal, $C(1)$ down the first sub and super diagonal, and so on. The matrix $\Sigma_{12} = \Sigma_{21}'$ is the $2 \times p$ matrix

$$\left[\begin{array}{ccc} C(1) & \cdots & C(p) \\ C(q-1) & \cdots & C(q-p) \end{array}\right]$$

Once you have then calculated the resulting 2×2 matrix you pick out the off diagonal element; this is the partial autocovariance. *Even when the data are not Gaussian we will be doing this piece of arithmetic with the autocovariance to define the partial autocovariance.*

To use this idea we define the partial autocorrelation function to be

$$PACF(h) = \text{Corr}(X_t, X_{t-h} | X_{t-1}, \ldots, X_{t-h+1})$$

If $h = 1$ then there are no Xs between X_t and X_{t-h} so there is nothing to condition on. This makes $PACF(1) = ACF(1)$.

Qualitative idea: A plot of the sample partial autocorrelation function (derived by replacing C_X by $\hat{C}_X$ in the definition of $PACF$) is examined. For an $AR(p)$ process this sample plot ought to drop suddenly to 0 for $h > p$.

$AR(1)$ example

Now I calculate $PACF(2)$ for a mean 0 $AR(1)$ process. We have $C_X(h) = C_X(0)\rho^{|h|}$. Let

$$\left[\begin{array}{c} Y \\ Z \end{array}\right] = \left[\begin{array}{c} X_t \\ X_{t-2} \\ X_{t-1} \end{array}\right]$$

with $Y = (X_t, X_{t-2})'$ and $Z = X_{t-1}$. Then $(Y, Z)'$ has a $MVN(0, \Sigma)$ distribution with

$$\Sigma = C_X(0) \left[\begin{array}{ccc} 1 & \rho^2 & \rho \\ \rho^2 & 1 & \rho \\ \rho & \rho & 1 \end{array}\right]$$

Following the partitioning into Y and Z we find

$$\Sigma_{11} = \Sigma = C_X(0) \left[\begin{array}{cc} 1 & \rho^2 \\ \rho^2 & 1 \end{array}\right]$$

$$\Sigma_{12} = C_X(0) \left[\begin{array}{c} \rho \\ \rho \end{array}\right]$$

and

$$\Sigma_{22} = C_X(0)\,[1]$$

We find

$$\begin{aligned} \text{Var}(X_t, X_{t-2} | X_{t-1}) &= \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} \\ &= C_X(0) \left[\begin{array}{cc} 1 & \rho^2 \\ \rho^2 & 1 \end{array}\right] - C_X(0) \left[\begin{array}{cc} \rho^2 & \rho^2 \\ \rho^2 & \rho^2 \end{array}\right] \\ &= C_X(0) \left[\begin{array}{cc} 1-\rho^2 & 0 \\ 0 & 1-\rho^2 \end{array}\right] \end{aligned}$$

Now to compute the correlation we divide the conditional covariance by the product of the two conditional SDs, that is:

$$\begin{aligned} PACF(2) &= \frac{\mathrm{Cov}(X_t, X_{t-2}|X_{t-1})}{\sqrt{\mathrm{Var}(X_t|X_{t-1})\mathrm{Var}(X_{t-2}|X_{t-1})}} \\ &= \frac{0}{\sqrt{(1-\rho^2)(1-\rho^2)}} \\ &= 0 \end{aligned}$$

Consider the problem of prediction X_t knowing $X_{t-1}, \ldots, X_{t-K}$. In this course we choose a predictor $\hat{X}_t = f(X_{t-1}, \ldots, X_{t-K})$ to minimize the mean squared prediction error:

$$\mathrm{E}[(X_t - \hat{X}_t)^2]$$

The solution is to take

$$\hat{X}_t = \mathrm{E}(X_t|X_{t-1}, \ldots, X_{t-K})$$

For multivariate normal data this predictor is the usual regression solution:

$$\mathrm{E}(X_t|X_{t-1}, \ldots, X_{t-K}) = \mu + A \begin{bmatrix} X_{t-1} - \mu \\ \vdots \\ X_{t-k} - \mu \end{bmatrix}$$

where the matrix A is $\Sigma_{12}\Sigma_{22}^{-1}$ in the notation of the earlier notes. This can be written in the form

$$\hat{X}_t = \mu + \sum_1^k \alpha_i (X_{t-i} - \mu)$$

for suitable constants α. Let $Y_t = X_t - \mu$. The α_i minimize

$$\begin{aligned} \mathrm{E}[(Y_t - \sum \alpha_i Y_{t-i})^2] &= \mathrm{E}(Y_t^2) - 2\sum \alpha_i \mathrm{E}(Y_t Y_{t-i}) + \sum_{ij} \alpha_i \alpha_j \mathrm{E}(Y_{t-i} Y_{t-j}) \\ &= C(0) - 2\sum \alpha_i C(i) + \sum_{ij} \alpha_i \alpha_j C(i-j) \end{aligned}$$

Take the derivative with respect to α_m and set the resulting k quantities equal to 0 giving

$$C(m) = \sum_j \alpha_j C(m-j)$$

or, dividing by $C(0)$

$$\rho(m) = \sum_j \alpha_j \rho(m-j)$$

(These are the Yule Walker equations again!) It is a fact that the solution α_m is

$$\mathrm{Corr}(X_t, X_{t-m}|X_{t-1}, \ldots, X_{t-k} \text{ but not } X_{t-m})$$

so that α_k (the last α) is $PACF(k)$.
Remark: This makes it look as if you would compute the first 40 values of the PACF by solving 40 different problems and picking out the last α in each one but in fact there is an explicit recursive algorithm to compute these things.

Estimation of PACF

To estimate the PACF you can either estimate the ACF and do the arithmetic above with estimates instead of theoretical values or minimize a sample version of the mean squared prediction error:

The Mean Squared Prediction Error is

$$\mathrm{E}\left\{[(X_t - \mu - \sum_{j=1}^{k} \alpha_j (X_{t-j} - \mu)]^2\right\}$$

We estimate μ in this formula with $\bar{X}$ and then replace the expected value with an average over the data so that we minimize

$$\sum_{t=k}^{T-1} \left\{[(X_t - \bar{X} - \sum_{j=1}^{k} \alpha_j (X_{t-j} - \bar{X})]^2\right\} / T$$

This is just a least squares problem regressing the vector

$$Y = \begin{bmatrix} X_k - \bar{X} \\ \vdots \\ X_{T-1} - \bar{X} \end{bmatrix}$$

on the design matrix

$$Z = \begin{bmatrix} X_{t-1} - \bar{X} & \cdots & X_0 - \bar{X} \\ \vdots & \vdots & \vdots \\ X_{T-2} - \bar{X} & \cdots & X_{T-k-1} - \bar{X} \end{bmatrix}$$

The estimate of $PACF(k)$ is the kth entry in

$$\hat{\alpha} = (Z'Z)^{-1} Z'Y \,.$$

Some example plots

What follows are plots of 5 series, their fitted ACFs and their fitted PACFs. The series are:

1. A generated $MA(2)$ series: $X_t = \epsilon_t + 0.8\epsilon_{t-1} - 0.9\epsilon_{t-2}$
2. A generated $AR(2)$ series $X_t = \epsilon_t + X_{t-1} - 0.99X_{t-2}$
3. A generated $AR(3)$ series: $X_t = \epsilon_t + 0.8X_{t-1} - X_{t-2}/3 + .8X_{t-3}/\sqrt{3}$

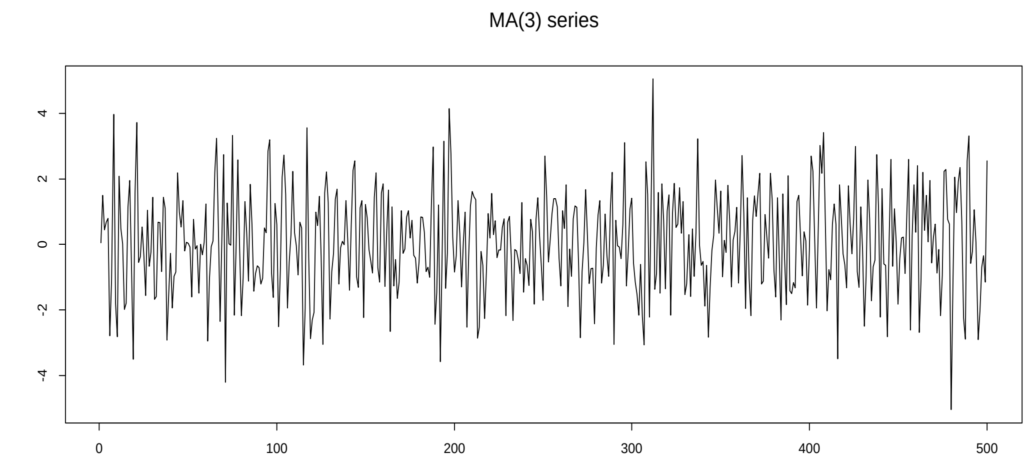
4. A series of monthly sunspot counts over a 200 year period.

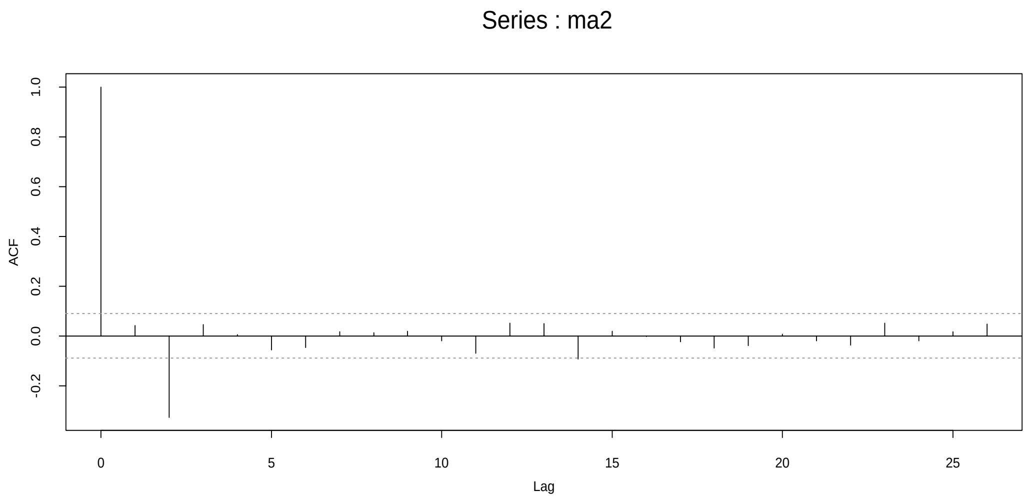
5. A series of annual rain fall measurements in New York City.

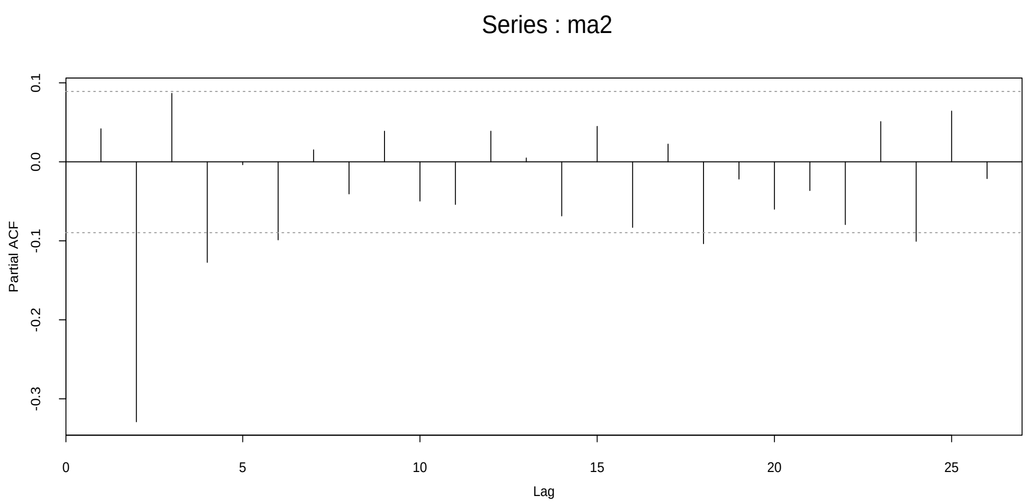
Here is the SPlus code I used to make the following plots. You should note the use of the function `acf` which calculates and plots ACF and PACF.

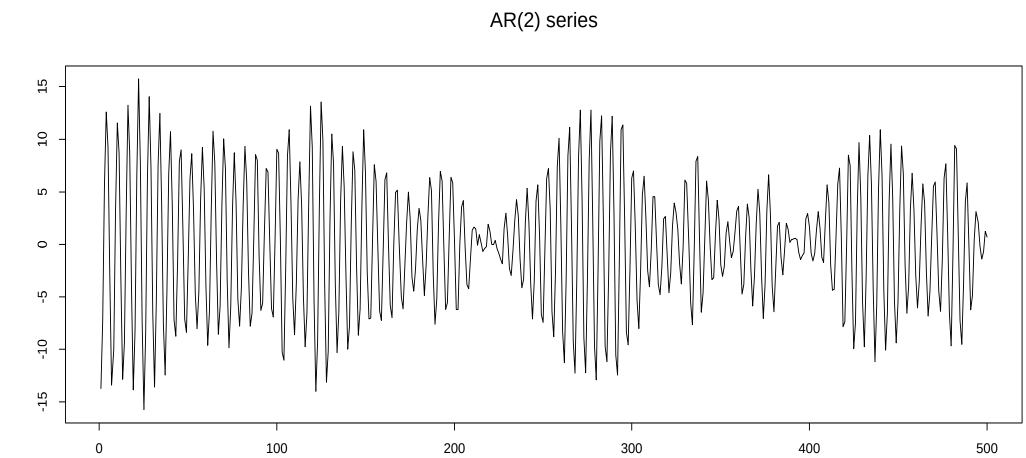
```
#
# Comments begin with #
#
# Begin by generating the series.  The AR series generated are NOT
# stationary.  But I generate 10000 values from the recurrence relation
# and use the last 500. Asymptotic stationarity should guarantee that
# these last 500 are pretty stationary.
#
n<- 10000
ep <- rnorm(n)
ma2 <- ep[3:502]+0.8*ep[2:501]-0.9*ep[1:500]
ar2 <- rep(0,n)
ar2[1:2] <- ep[1:2]
for(i in 3:n) {ar2[i] <- ep[i]
              + ar2[i-1] -0.99*ar2[i-2]}
ar2 <- ar2[(n-499):n]
ar2 <- ts(ar2)
ar3 <- rep(0,n)
ar3[1:3] <- ep[1:3]
for(i in 4:n) {ar3[i] <- ep[i] + 0.8*ar3[i-1]
           -ar3[i-2]/3 +(0.8/1.712)*ar3[i-3]}
ar3 <- ar3[(n-499):n]
ar3 <- ts(ar3)
#
# The next line turns on a graphics device -- in this case the
# graph will be made in postscript in a file called ma2.ps. It
# will come out in portrait, not landscape, format.
#
postscript(file="ma2.ps",horizontal=F)
#
# The next line says to put 3 pictures
# in a single column on the plot
#
par(mfcol=c(3,1))
tsplot(ma2,main="MA(2) series")
acf(ma2)
acf(ma2,type="partial")
#
```

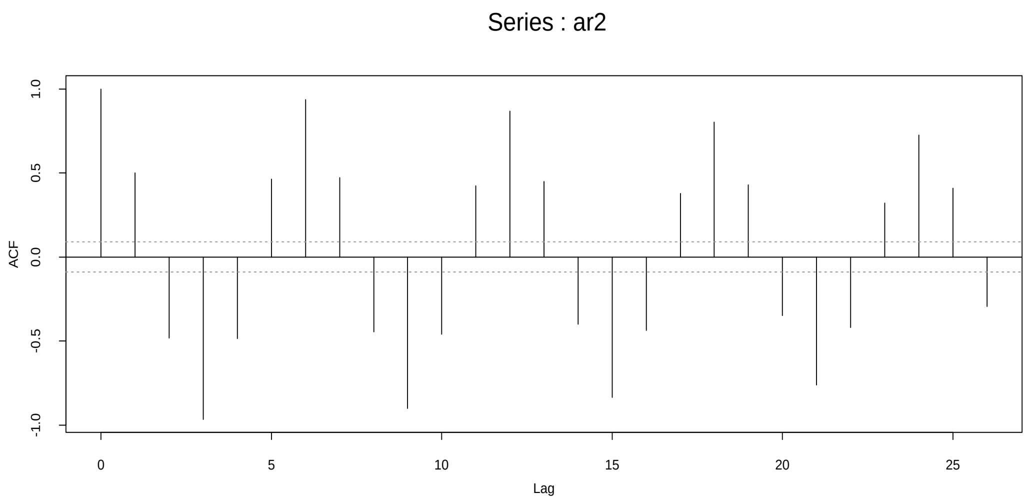
```
# When you finish a picture you turn
# off the graphics device
# with the next line.
#
dev.off()
#
#
#
postscript(file="ar2.ps",horizontal=F)
par(mfcol=c(3,1))
tsplot(ar2,main="AR(2) series")
acf(ar2)
acf(ar2,type="partial")
dev.off()
#
#
postscript(file="ar3.ps",horizontal=F)
par(mfcol=c(3,1))
tsplot(ar3,main="AR(3) series")
acf(ar3)
acf(ar3,type="partial")
dev.off()
#
#
postscript(file="sunspots.ps",horizontal=F)
par(mfcol=c(3,1))
tsplot(sunspots,main="Sunspots series")
acf(sunspots,lag.max=480)
acf(sunspots,lag.max=480,type="partial")
dev.off()
#
#
postscript(file="rain.nyc1.ps",horizontal=F)
par(mfcol=c(3,1))
tsplot(rain.nyc1,main="New York Rain Series")
acf(rain.nyc1)
acf(rain.nyc1,type="partial")
dev.off()
```

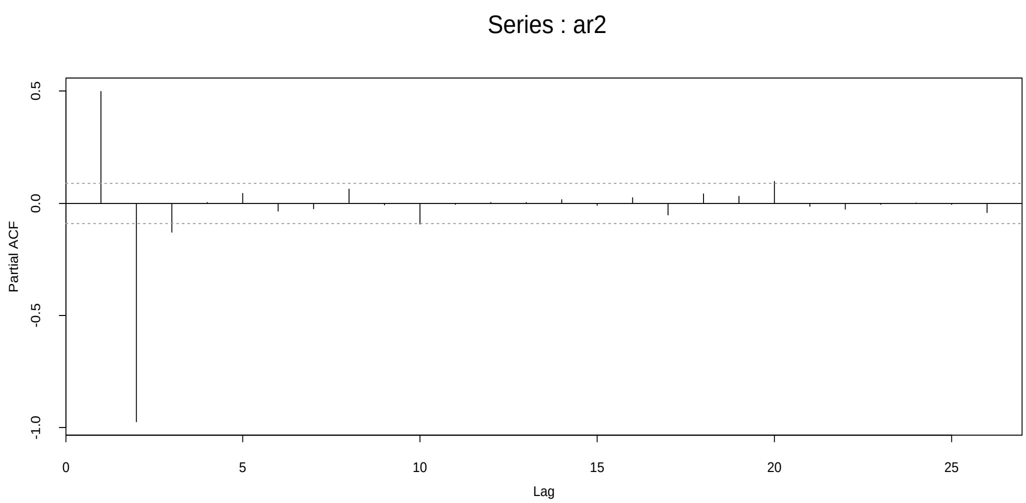

MA(3) series
4
2
0
-2
-4
0
100
200
300
400
500

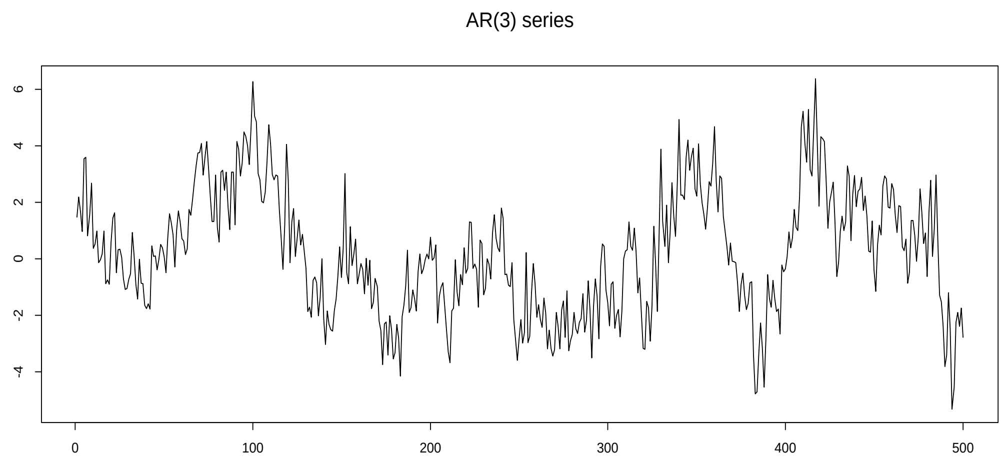

Series : ma2
ACF
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
0
5
10
15
20
25
Lag

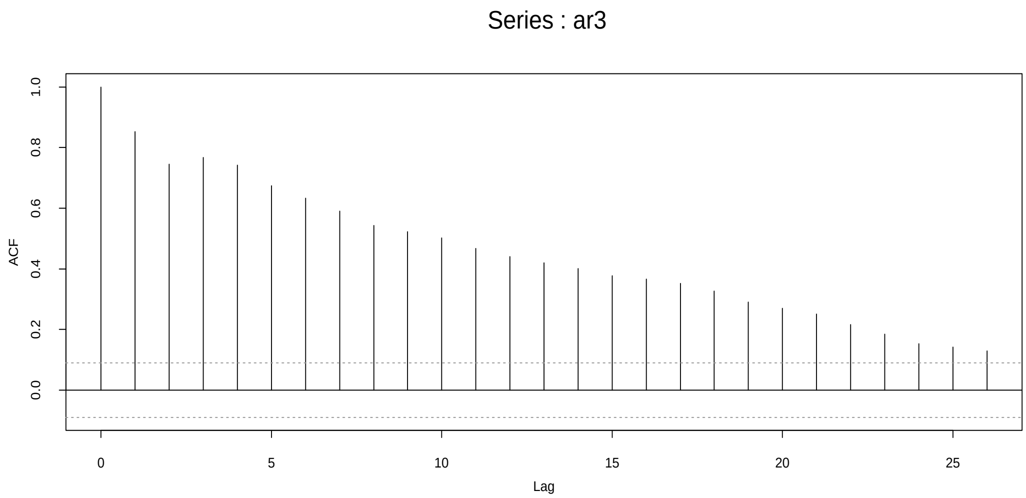

Series : ma2
Partial ACF
0.1
0.0
-0.1
-0.2
-0.3
0
5
10
15
20
25
Lag

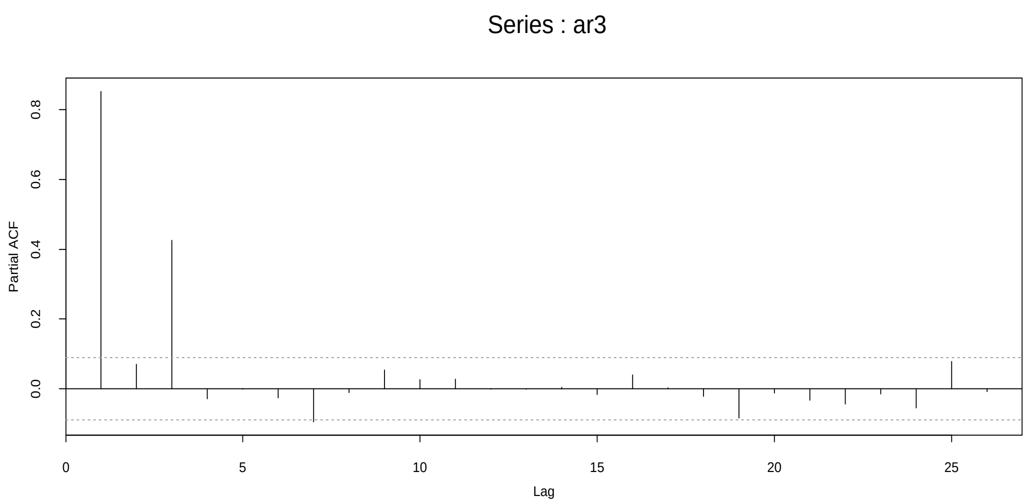

AR(2) series
15
10
5
0
-5
-10
-15
0
100
200
300
400
500


Series : ar2
ACF
1.0
0.5
0.0
-0.5
-1.0
0
5
10
15
20
25
Lag

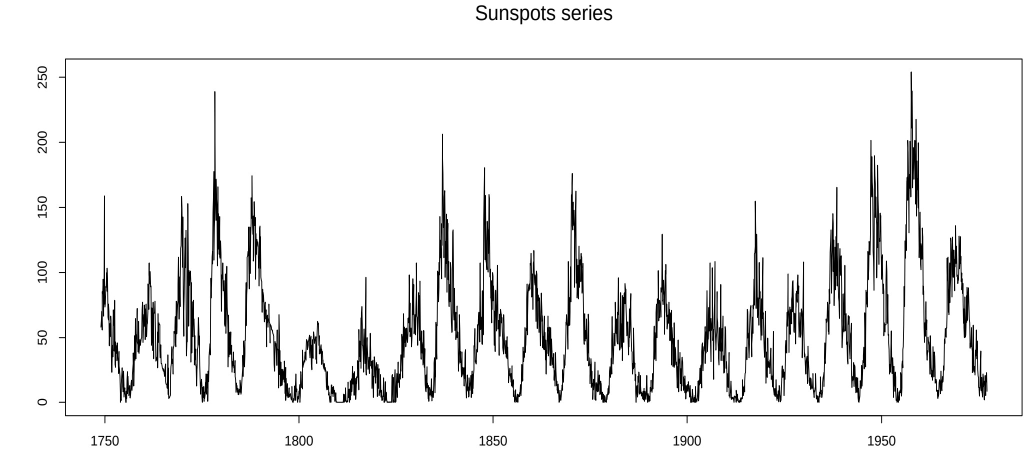

Series : ar2
Partial ACF
0.5
0.0
-0.5
-1.0
0
5
10
15
20
25
Lag

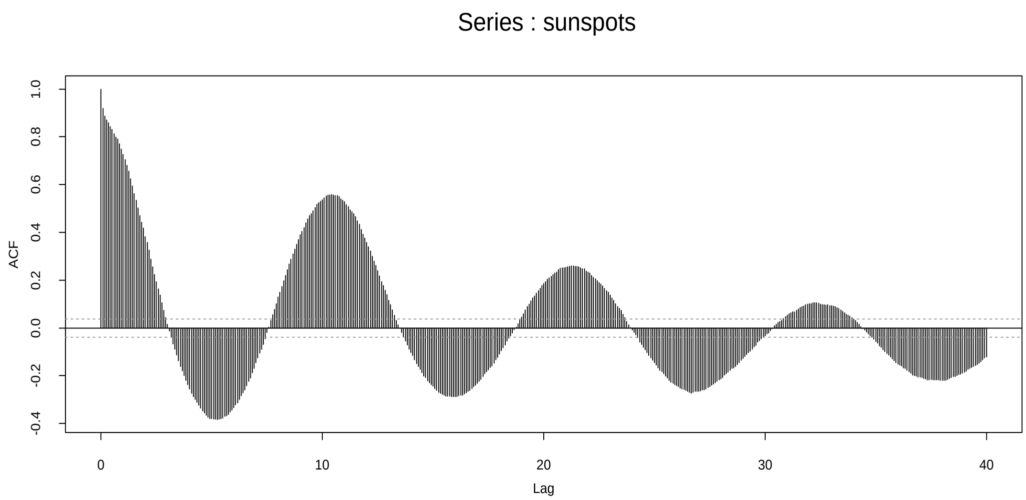

AR(3) series
6
4
2
0
-2
-4
0
100
200
300
400
500

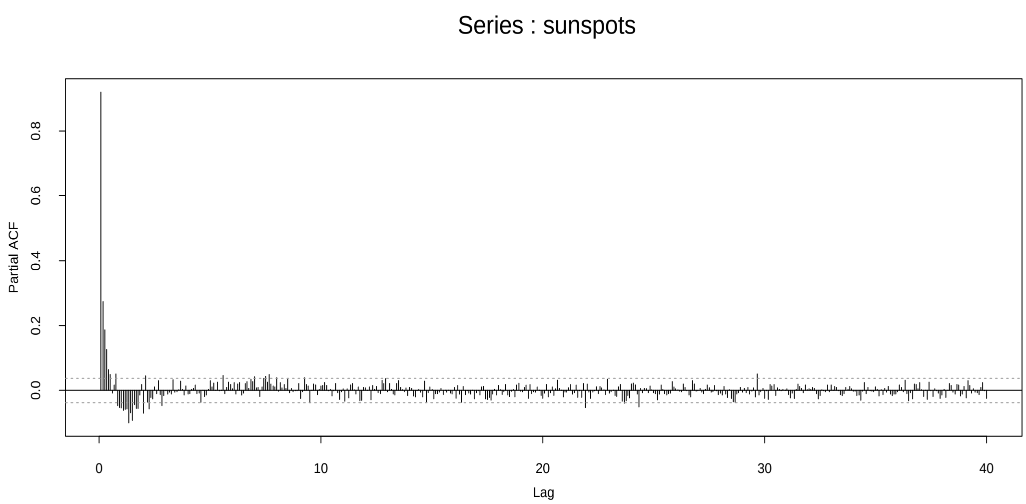

Series : ar3
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Lag


Series : ar3
Partial ACF
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Lag

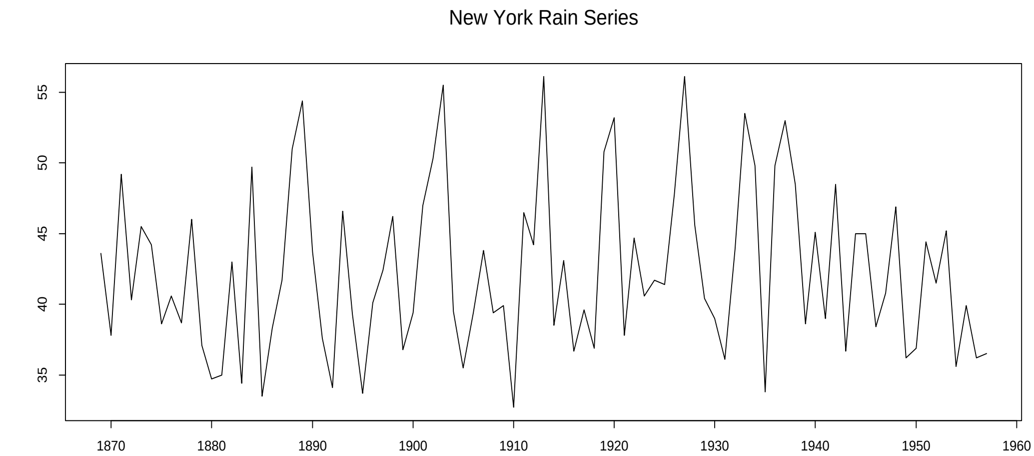
Sunspot data

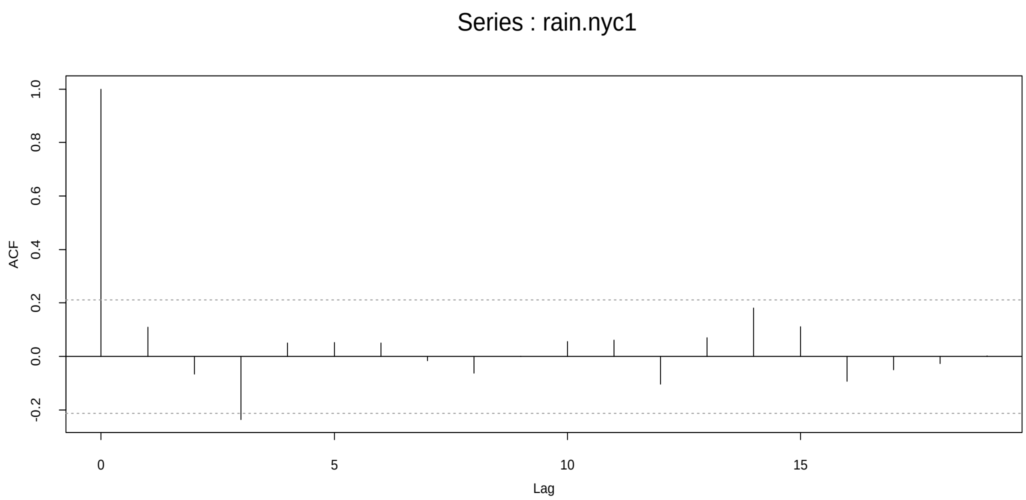


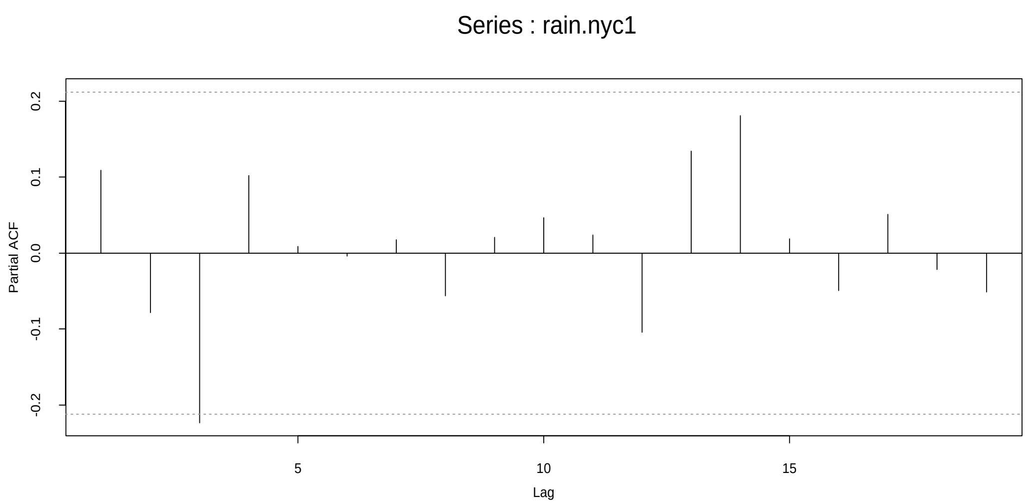




New York City Rainfall







You should notice:

- The MA series has an ACF which is near 0 for lags over 2 as predicted theoretically.
- The $AR(p)$ series have complicated ACF's but the PACFs vanish at lags more than p.
- The sunspot data has significant autocorrelation over many years but the PACF is pretty small after say 2 years or so. Notice that SPlus has labelled the lags in years so that the number 2 on the x axis corresponds to about 24 months. An $AR(p)$ model with p around 24 would then be a possibility.
- The New York rainfall series looks quite a bit like white noise.