Econ 211C, Problem Set 1

Swati Sharma

4/23/2018

Question 1

Part (a)

Yes, the process is weakly stationary because the mean and covariances of the process is not time-dependent due the error term being white noise and iid. MA(q) processes are always stationary.

These are the calculated autocovariances:

$$Y_t = (1 + 2.4L + 0.8L^2)\epsilon_t$$

$$\gamma_0 = (1 + 2.4^2 + 0.8^2)\sigma^2$$
$$\gamma_0 = (1 + 5.76 + 0.64)\sigma^2$$
$$\gamma_0 = 7.4\sigma^2$$

$$\gamma_1 = (2.4 + 2.4 * 0.8)\sigma^2$$
$$\gamma_1 = 4.32\sigma^2$$

$$\gamma_2 = 0.8\sigma^2$$

$$\gamma_3 = \gamma_4 = ... = 0$$

Part (b)

$$\theta(L) = (1 + 2.4L + 0.8L^2)$$
$$\theta(L) = (1 + 2L)(1 + 0.4L)$$

Unit roots:

$$\frac{1}{2} = 0.5$$
$$\frac{1}{0.4} = 2.5$$

One of the unit roots does not lie outside the unit circle so it is not invertible.

Finding an invertible representation:

$$\tilde{\theta}(L) = (1 + \frac{1}{2})(1 + 0.4) = 1 + 0.9L = 0.2L^2$$

$\tilde{Y}_t = \tilde{\theta}(L)\tilde{\epsilon}_t$ where $\tilde{\epsilon}_t \sim WN(0, \sigma^2(2^2))$

Part (c)

$$\gamma_0 = (1 + 0.9^2 + 0.2^2)2^2\sigma^2$$
$$\gamma_0 = 1.85 * 4\sigma^2$$
$$\gamma_0 = 7.4\sigma^2$$

$$\gamma_1 = (0.9 + 0.9 * 0.2)2^2\sigma^2$$
$$\gamma_1 = 1.08 * 4\sigma^2$$
$$\gamma_1 = 4.32\sigma^2$$

$$\gamma_2 = 0.2 * 2^2\sigma^2$$
$$\gamma_2 = 0.8\sigma^2$$

$$\gamma_3 = \gamma_4 = ... = 0$$

The autocovariances are the same as the ones we found in part (a). The first two moments of this invertible representation and non invertable representation should be identical.

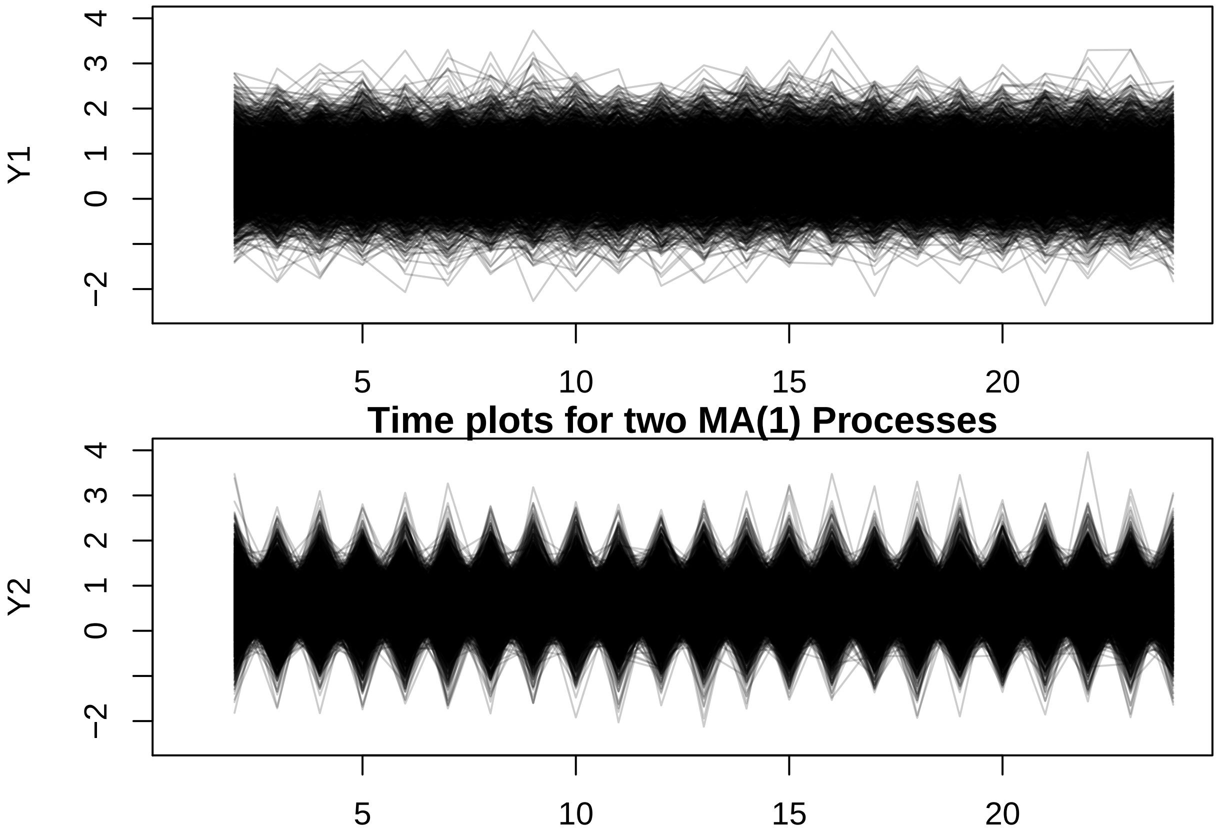
Question 2

Note: I only ran 100 simulations because I put way too many loops in (exactly like you told us not to) and it resulted in a stack overflow. Rendering the markdown file would have been a task...

```r
N = 24;
sigma = 0.5;
mu = 0.61;
theta = 0.95;

df <- data.frame(matrix(nrow = N, ncol = 2300))
indexing <- seq(1,4600,2)
for(i in indexing) {
   eps <-  rnorm(N, 0, sigma);
   df[2:N,i] <-  mu + eps[2:N] + theta*eps[1:(N-1)];
   df[2:N,i+1] <-  mu + eps[2:N] - theta*eps[1:(N-1)];
}

indexing <- seq(3,4600,2)
par(mfrow=c(2,1), mar=c(2,3.9,1,1))
plot(df[,1], ylab="Y1",type="l", col=rgb(0,0,0,0.2), ylim = c(-2.5,4))
for (i in indexing) {
  lines(df[,i],type="l", col=rgb(0,0,0,0.2), ylim = c(-2.5,4))
}
indexing <- seq(2,4600,2)
plot(df[,2],ylab="Y2",type="l", col=rgb(0,0,0,0.2), ylim = c(-2.5,4))
for(i in indexing) {
  lines(df[,i],type="l", col=rgb(0,0,0,0.2), ylim = c(-2.5,4))
}
title("Time plots for two MA(1) Processes")
```



Question 3

Part (a)

$$Y_t = 1.3Y_{t-1} - 0.4Y_{t-2} + \epsilon_t + 0.7\epsilon_{t-1} + 0.1\epsilon_{t-3} - 0.5\epsilon_{t-4} - 0.2\epsilon_{t-5}$$
$$Y_t - 1.3Y_{t-1} + 0.4Y_{t-2} = \epsilon_t + 0.7\epsilon_{t-1} + 0.1\epsilon_{t-3} - 0.5\epsilon_{t-4} - 0.2\epsilon_{t-5}$$

$$\phi(L) = 1 - 1.3L + 0.4L^2$$

Factoring $\phi(L)$:

$$\phi(L) = (1 - 0.8L)(1 - 0.5L)$$

Unit roots of $\phi(L)$:

$$\frac{1}{0.8} = 1.25$$
$$\frac{1}{0.5} = 2$$

The ARMA(p,q) process is stationary if the roots of the $\phi(L)$ polynomial lie outside the unit circle. Both roots lie outside the unit circle so this ARMA process is stationary.

Part (b)

If all of the roots of $\theta(L)$ lie outside the unit circle, the ARMA(p,q) process has a unique invertable representation.

Factoring $\theta(L)$:

$$\phi(L) = -0.2(L - 1.29316)(L + 0.948269)(L + 2.76131)(L^2 + 0.08358L + 1.47663)$$

Unit roots of $\theta(L)$:

$$\frac{1}{2.761} = 0.3621$$
$$\frac{1}{1.293} = 0.7734$$
$$\frac{1}{0.948} = 1.0549$$

Only one of these roots lies outside the unit circle so the process is not invertable.

Part (c)

Solving for values of $\psi(L)$:

$$\psi(L) = \frac{\theta(L)}{\phi(L)}$$
$$\phi(L)\psi(L) = \theta(L)$$
$$(1 - 1.3L + 0.4L^2)(\psi_0 + \psi_1 L + \psi_2 L^2 + \psi_3 L^3 + \psi_4 L^4 + \psi_5 L^5 = 1 + 0.7L + 0.1L^2 - 0.5L^4 - 0.2L^5$$

$$\psi_0 = 1$$
$$\psi_1 = 0.7 + 1.3 = 2$$
$$\psi_2 = 2.6 - 0.4 = 2.2$$
$$\psi_3 = 0.1 + 2.2 * 1.3 = 2.96$$
$$\psi_4 = -0.5 + 2.968 = 2.468$$
$$\psi_5 = -0.2 + (1.03)(2.468) - (0.4)(2.96) = 1.8244$$

$$\gamma_j = \phi_1\gamma_{j-1} + \phi_2\gamma_{j-2} + \sigma^2(\theta_j\psi_0 + \theta_{j+1}\psi_1 + \theta_{j+2}\psi_2 + \theta_{j+3}\psi_3 + \theta_{j+4}\psi_4 + \theta_{j+5}\psi_5$$

$$\gamma_0 = 1.3\gamma_1 - 0.4\gamma_2 + \sigma^2(1 + 2 * 0.7 + 2.96 * 0.1)$$
$$\gamma_0 = 1.3\gamma_1 - 0.4\gamma_2 + \sigma^2(1.09712)$$

Where $\gamma_1 = \gamma_{-1}$

$$\gamma_1 = 1.3\gamma_0 - 0.4\gamma_1 + \sigma^2(0.7 + 0.3 * 2.2 - 0.5 * 2.96 - 0.2 * 2.468)$$
$$\gamma_1 = 1.3\gamma_0 - 0.4\gamma_1 - \sigma^2(0.6136)$$

$$\gamma_2 = 1.3\gamma_1 - 0.4\gamma_0 + \sigma^2(0.1 * 2 - 0.5 * 2.2 - 0.2 * 2.96)$$

$$\gamma_2 = 1.3\gamma_1 - 0.4\gamma_0 - \sigma^2(1.492)$$

$$\gamma_3 = 1.3\gamma_2 - 0.4\gamma_1 + \sigma^2(0.1 - 0.5 * 2 - 0.2 * 2.2)$$
$$\gamma_3 = 1.3\gamma_2 - 0.4\gamma_1 - \sigma^2(0.957)$$

$$\gamma_4 = 1.3\gamma_3 - 0.4\gamma_2 + \sigma^2(-0.5 * 2 - 0.2 * 2)$$
$$\gamma_4 = 1.3\gamma_3 - 0.4\gamma_2 - \sigma^2(0.9)$$

$$\gamma_5 = 1.3\gamma_4 - 0.4\gamma_3 - \sigma^2(0.2)$$

Generalized recursive funtion for subsequent autocovariances:

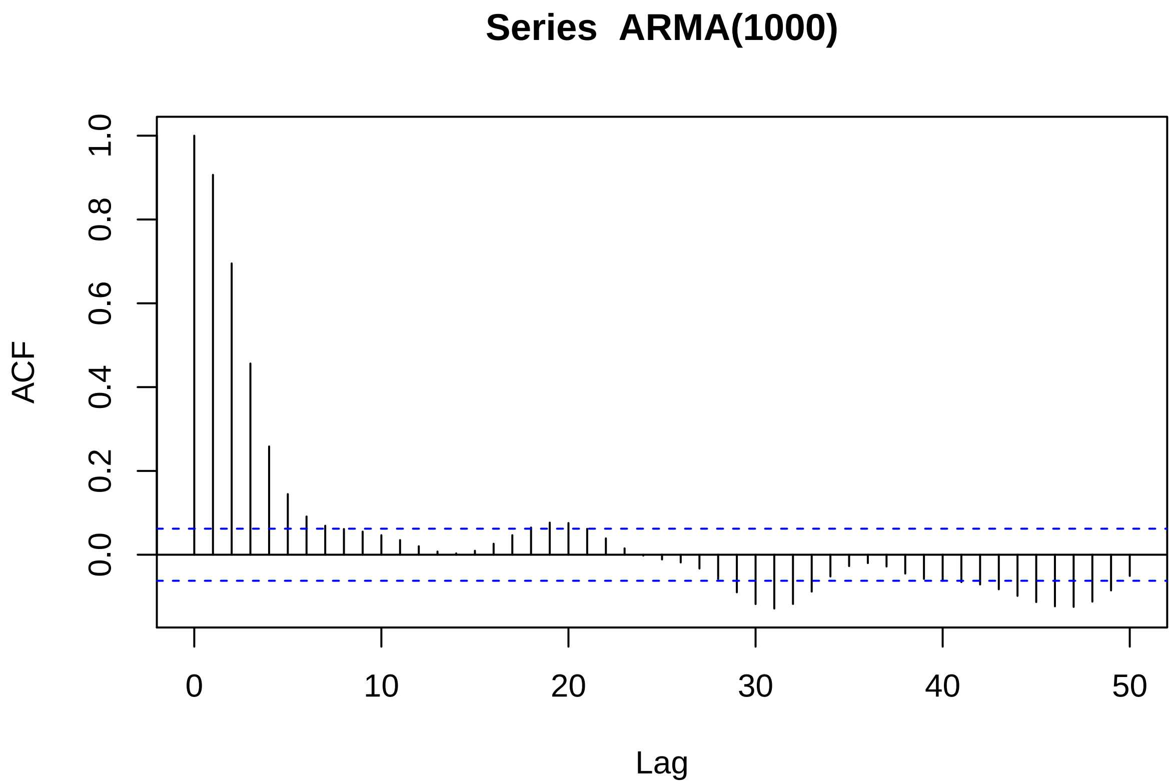
$$\gamma_j = 1.3\gamma_{j-1} - 0.4\gamma_{j-2} + \sigma^2(0)$$
$$\gamma_j = 1.3\gamma_{j-1} - 0.4\gamma_{j-2}$$

Part (d)

```r
phi <- c(1.3,-0.4)
theta <- c(0.7, 0, 0.1, -0.5, -0.2)
p <- length(phi)
q <- length(theta)
nn <- max(p,q)

ARMA <- function(n) {
  eps <- rnorm(n+nn)
  Y<-rep(0,n)
  for(i in (nn+1):length(Y)) {
    Y[i]<-t(phi)%*%Y[(i-1):(i-p)] + t(theta)%*%eps[(i-1):(i-q)] + eps[i]
  }
  return(Y[-(1:nn)])
}

acf(ARMA(1000),type = "correlation",lag.max = 50)
```



Part (e)

```
simulation <- ARMA(1000)
cat("The mean of this simulation is", mean(simulation), "\n")
```

```
## The mean of this simulation is 0.3848307
```

```
cat("The variance of this simulation is", var(simulation), "\n")
```

```
## The variance of this simulation is 14.40856
```

```
acf(simulation,type = "covariance",lag.max = 5,plot = FALSE)
```

```
##
## Autocovariances of series 'simulation', by lag
##
##      0      1      2      3      4      5
## 14.394 12.851  9.392  5.526  2.280  0.273
```

Part (f)

```
for(N in c(10000,100000,1000000)){
  simulation <- ARMA(N)
  cat("For N =", N, "\n")
  cat("The mean of this simulation is", mean(simulation), "\n")
  cat("The variance of this simulation is ", var(simulation), "\n")
  print(acf(simulation, type = "covariance", lag.max = 5, plot = FALSE))
  cat("\n")

}
```

```
## For N = 10000
## The mean of this simulation is 0.08774979
## The variance of this simulation is  17.13532
##
## Autocovariances of series 'simulation', by lag
##
##     0     1     2     3     4     5
## 17.13 15.56 12.01  7.99  4.61  2.54
##
## For N = 1e+05
## The mean of this simulation is 0.009556783
## The variance of this simulation is  17.93415
##
## Autocovariances of series 'simulation', by lag
##
##     0     1     2     3     4     5
## 17.93 16.35 12.77  8.74  5.37  3.29
##
## For N = 1e+06
## The mean of this simulation is 0.001144048
## The variance of this simulation is  17.48942
##
## Autocovariances of series 'simulation', by lag
##
##     0     1     2     3     4     5
## 17.49 15.92 12.37  8.37  5.03  2.99
```

For higher values of n, the mean shrinks towards 0, the variance gets larger (at a decreasing rate), and the autocovariances get larger.