Assignment 2: Time Series Analysis:ARMA Processes and Seasonal Processes

March 17, 2017

1 ARMA Processes and Seasonal Processes

1.1 Stability

Rearranging the equation so we can see what type of ARMA model it is we get:

$$X_t(1 + \Phi_1 B + \Phi_2 B^2 = \epsilon_t, \tag{1}$$

Where $\Phi_1 = \frac{-1}{3}$. From above, we know that this is an AR(2) model and which has no moving average component. Then, using the transfer function $H(z)$:

$$H(z) = \frac{\theta(z^{-1})}{\phi(z^{-1})} = \frac{1}{1 + \Phi_1 z^{-1} + \Phi_2 z^{-2}} = \frac{z^2}{z^2 + \Phi_1 z + \Phi_2} \tag{2}$$

To be stationary, the roots of the polynomial have to lie within the unit circle interval $< -1, 1 >$. Thus, from the poles of the above equation, we get the following:

$$|\frac{1/3 \pm sqrt(1/9 - \Phi_2 4)}{2}| < 1 \tag{3}$$

Solving for Φ_2 above we get:

$$\Phi_2 > \frac{-2}{3} \tag{4}$$

Notice that this is for the positive case (from the $\pm$). Since to be stationary our coefficients have to lie within the unit circle interval, we get:

$$\frac{-2}{3} < \Phi_2 < 1 \tag{5}$$

To exhibit damped harmonic oscillations, the roots have to be complex, which occurs when the term inside the square root in equation 3 is negative. This leads to:

$$\frac{1}{9} - \Phi_2 4 < 0 \tag{6}$$

And solving for Φ_2 we get 1/36. Again, since stationarity imposes the coefficients to lie within the unit circle interval, then damped harmonic oscillations occur when $1/36 < \Phi_2 < 1$.

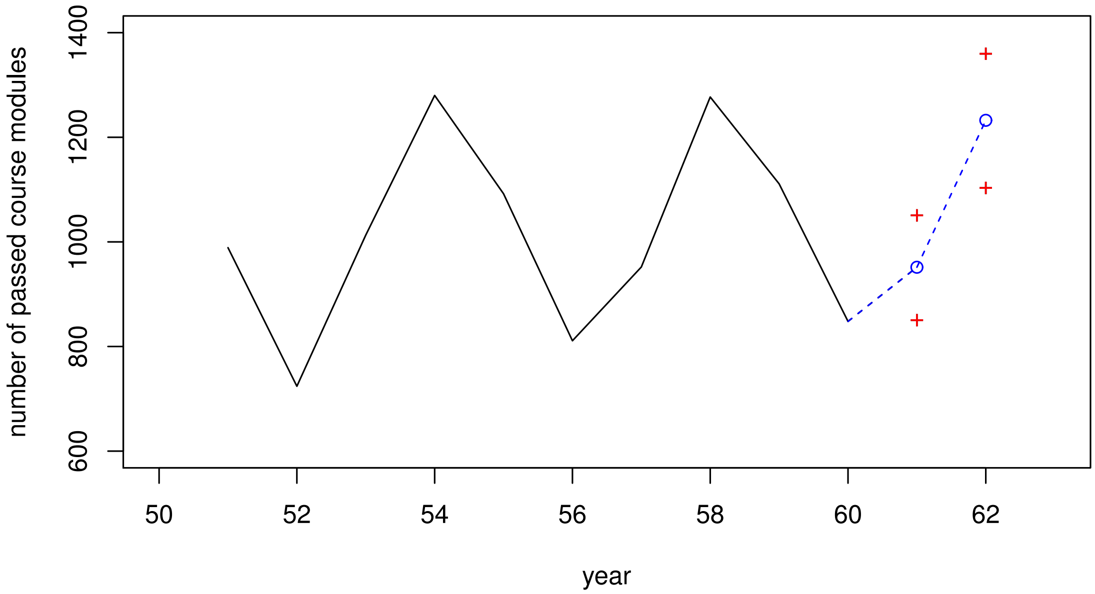

Figure 1: The modules passed at the university over time.Note that the forecast model is able to replicate the see-saw pattern (i.e. seasonality) of the data. The observations are denoted by the black solid line while the predictions are denoted by the blue empty circles. The 95% confidence intervals are denoted by the red crosses. Table 1 shows the exact values.

1.2 University activity: Predict the values of Y_t corresponding to t = 61 and 62, together with 95% prediction intervals for the predictions.

Table1: Predictions of AR(2) with Season(4) Model with 95% CI

Time	Prediction	Lower	Upper
61	951.54	851.38	1051.7
62	1232.42	1104.15	1360.69

1.3 Random walk

1.3.1 Mean value, variance and covariance functions of the process

Mean value: 0.25

$$E[Y_t] = E[0.25 + \sum_{i=1}^{t} \epsilon_i] = 0.25 + E[\sum_{i=1}^{t} \epsilon_i] = 0.25 + 0 \tag{7}$$

The expected value of the sum of the error terms (the 2^{nd} term above) is zero since $\epsilon_t = N(0, \sigma^2)$.

Variance: $t\sigma^2$

$$V[Y_t] = V[0.25 + \sum_{i=1}^{t} \epsilon_i] = V[0.25] + V[\sum_{i=1}^{t} \epsilon_i] = 0 + t\sigma^2 \tag{8}$$

1.3.2 Is the process stationary? If so, in what sense? If not, in what sense?

By definition, a random walk is not stationary. In the equations above we have shown that the first moment (mean value) is 0.25 and does not depend on time. Meaning that (theoretically) the mean will not change over time. The second moment (the variance), however, does depend on time ($t\sigma^2$). In this case we see that the variance will increase linearly with time. Therefore, Y_t is a weakly stationary process of order 1 since only its 1^{st} moment (mean) is time invariant.

1.3.3 Simulate a white noise process of 1000 values. Then figure out a fast and compact way to calculate the process Y_t based on this.

The first line simulates the white noise (see Figure 2 for graphical representation). The rest, encapsulate a fast and compact way to calculate the Y_t random walk process in R:

```
Wnoise <- rnorm(n = 1000, mean = 0, sd = 2^0.25)
Yt <- rep(NA, 1000)
#Initialize 1st value
Yt[1] <- 0.25 + Wnoise[1]
#Then, we let the random walk do the walk
for (i in 2:1000) {
  Yt[i] <- Yt[i-1] + Wnoise[i]
}
```

1.3.4 Simulate 10 realizations of the process Y_t. Plot their autocorrelation functions and comment. Confirm you answers in step 2.

(See Figure 3 for 10 realizations)

(See Figure 4 for their autocorrelation function plot)

Comments:

The $Y_t = 0.25 + \sum_{i=1}^{t} \epsilon_i$ process is a pure random walk ($Y_t = Y_{t-1} + \epsilon_t$). The hint lies in the sum of the epsilon terms ($\sum_{i=1}^{t} \epsilon_i$). If we substitute Y_{t-1} for $Y_{t-2} + \epsilon_{t-2}$; and Y_{t-2} for $Y_{t-3} + \epsilon_{t-3}$, and so on recursively, we see that we can eliminate the Y_{t-1} component in the pure random walk and substitute it with a Y_0 plus the sum of all the previous error terms. Therefore, Y_0 must be 0.25, and we end up with $Y_t = 0.25 + \sum_{i=1}^{t} \epsilon_i$.

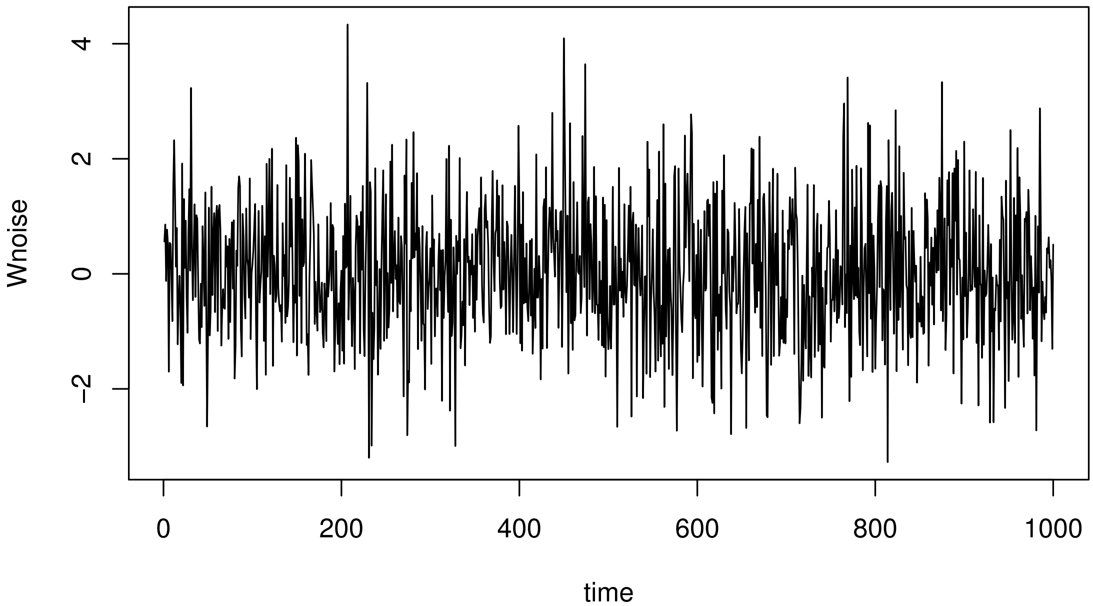

Figure 2: Simulated white noise of 1000 values.

By definition, a pure random walk is not stationary. Simply speaking, a pure random walk is a series of accumulated confined error terms stacked on Y_0 (the initial value) over time. This explains why each of the 10 realizations in Figure 3 takes its own random path yet starting from the same starting point. The autocorrelation plot also tells us the same story. None of the realizations' acf values decay fast enough to zero, and a fast decay to zero is indicative of stationarity.

Confirming answers in Step 2: In Step 2 we found that Y_t is a weak stationary process of order 1; meaning that it's mean is constant over time, but the second moment (variance) changes over time. Figure 3 shows how the variance of each random walk increases over time. From the results in Step 2 we know that it grows linearly ($t\sigma^2$).

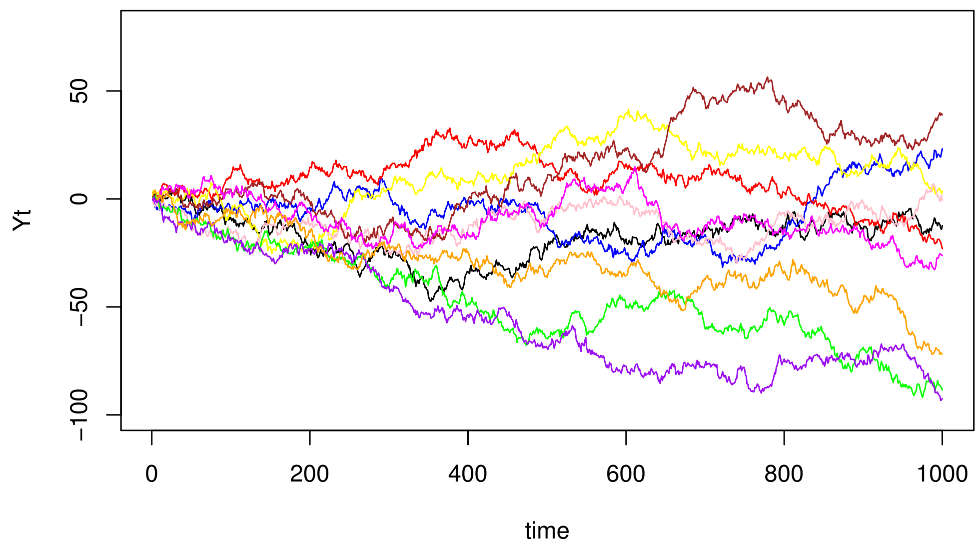

Figure 3: Each color represents 1 of the 10 different random walk realizations.

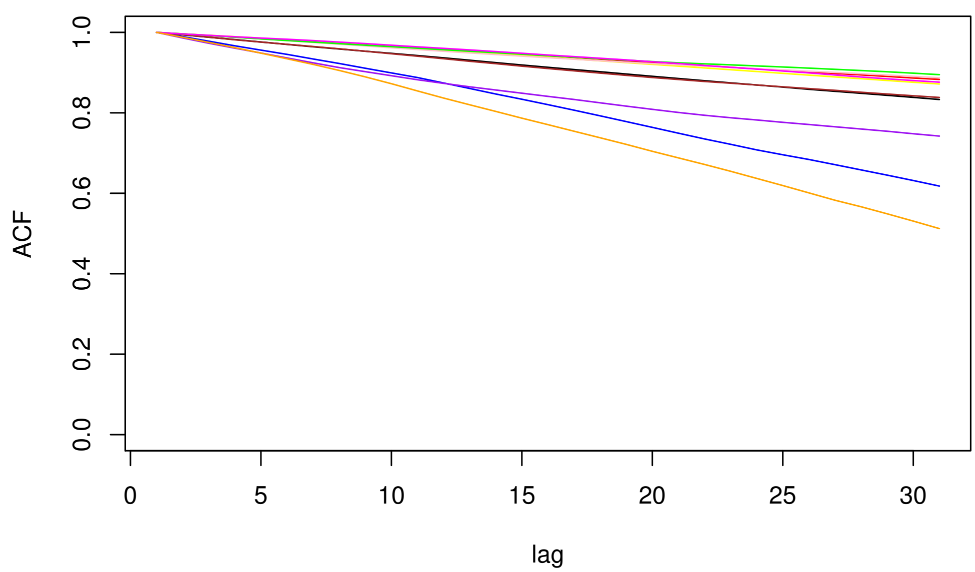

Figure 4: Autocorrelation function of the 10 simulated random walks (Y_t processes) from Figure 3. Notice that none of the realizations decay fast enough to zero. If they were stationary, we would expect them to decay rapidly towards zero.

1.4 Simulating seasonal processes

1.4.1 Are all models seasonal?

All models except Model 2 are seasonal as indicated by their (P, D, Q) component. Model 2 is a non-seasonal AR(1) process.

1.4.2 Which conclusions can you draw on the general behavior of the autocorrelation function for seasonal processes?

As said previously, all models have seasonality except for model 2. Interpreting models 3, 4, 5 and 6 is not as clear cut as models 1 and 2 in regards to their ACF and PACF simply because there is a mix of 2 coefficients, whose signals overlap and can mask each other to some extent. However, if we separate the seasonal models into the camp that has an AR component in (p,d,q) (models 3 and 5) and those that have an MA component in (p,d,q) (models 4 and 6), we can see that those with an AR component have ACF functions with alternating signs. This is due to the fact that when we move the AR component to the other side of the equality (to make it compatible with R's arima.sim function) it becomes negative. In contrast, the seasonal MA models seem to behave differently. As expected (from the "golden table") we do see how the AR models' (models 2,3,5) ACF values drop to 0 in an exponential manner.

1.4.3 Summarize your observations on the processes and the autocorrelation functions.

Model 1: Model 1 has seasonality of 12. We can see that the ACF values drop to zero, but spikes in multiples of 12 thereby highlighting how the correlation is found between lags of 12 (thus seasonality). The PACF confirms this as there is a clear spike in lag 12.

Model 2: Model 2 is an AR(1) process as described by the ACF and PACF plots. We can see that the ACF values drop to zero, which is a sign of an AR process. The ACF tells us that there is no seasonality component either. If there was, we would see how the s lag (12 in this case) and potentially its multiples would spike above the 95% confidence interval or at least show some of the correlation that propagates through. We do not see this behavior. In addition, the PACF tells us that the only strong correlation is in lag 1, highlighting the AR(1) component on this model (since it autoregresses on the previous observation).

Model 3: Model 3 shows ACF values decaying to 0 exponentially, which is most likely due to its AR component. There is also a seasonality component and it can barely be seen that the 12th lag in the ACF is a bit more accentuated than the rest in that area. The PACF confirms the AR(1) and the seasonality as we can see that the lag 1 has a strong correlation as well as lag 13.

Model 4: This MA model has a strong spike in lag 1 as expected. Notice that lag 12 also spikes up highlighting the carry over from seasonality. We can observe similar spikes in lags 1 and 12 in the PACF although neighboring lags also spike to a lesser extent.

Model 5: We see an exponential decay (also alternating signs as discussed previously) as is symptomatic of an AR process. In addition, we see that lag 1 spikes in the PACF to -0.8 as expected as that is the AR coefficient when arranged to fit the definition of R's arima.sim formulation. There are significant spikes in lags 11 and 12 in the PACF as well.

Model 6: This model has a moving average and a seasonality component. We can see in the ACF that it rapidly decays to 0. The PACF shows strong peaks in lags 1 and 12, as well as some of peaks immediately next to them.

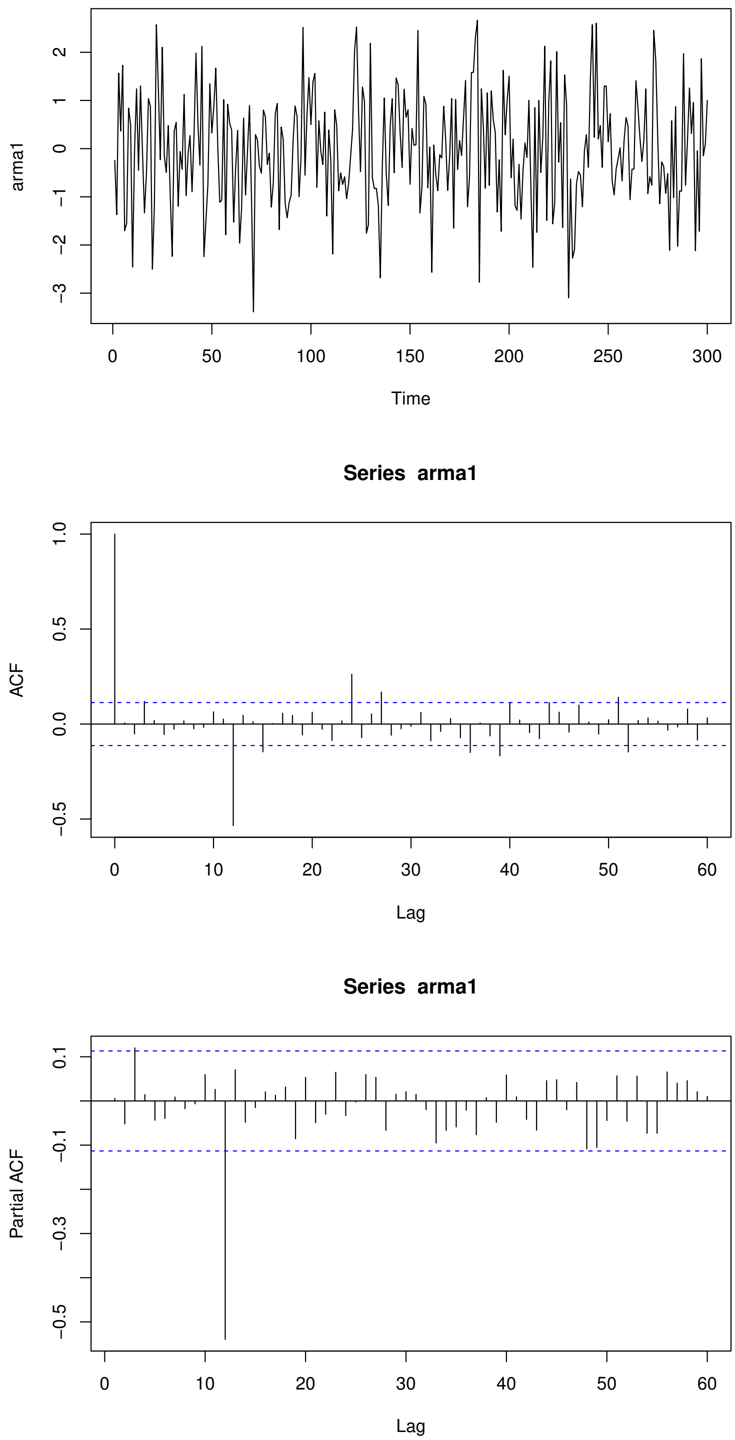

Figure 5: Model 1: A$(0,0,0)$ x $(1,0,0)_{12}$ model with the parameter $\Phi_1 = 0.6$

2 Code

```
# ___________ Script Setup _____________ #

rm(list=ls())
setwd("~/Desktop/TSA/assignment2")
library(expm)
library(timeSeries)
library(tseries)
library(forecast)

# _________ Question 2.2: Univeristy Activity _______________ #

# Variable Setup
t    <- (51:60)
mu <- 1000
Y1    <- c(989,724,1013,1280,1092,811,952,1277,1111,848)
Y1 <- Y1 - mu
psi1 <- 0.8; psi2 <- 0.7; psi3 <- 0.9

# Predictions Y_t+1, Y_t+2
pred1 <- psi1*Y1[10] - psi2*Y1[9] + psi3*Y1[7] - psi1*psi3*Y1[6] + psi2*psi3*Y1[5]
pred2 <- psi1*pred1  - psi2*Y1[10] + psi3*Y1[8] - psi1*psi3*Y1[7] + psi2*psi3*Y1[6]
Y1 <- Y1 + mu
pred1 <- pred1+mu
pred2 <- pred2+mu
# 95% Confidence Intervals
upper1 <- pred1 + qt(0.975, 56)*50
lower1 <- pred1 + qt(0.025, 56)*50
upper2 <- pred2 + qt(0.975, 56)*50*sqrt(1+psi1^2)
lower2 <- pred2 + qt(0.025, 56)*50*sqrt(1+psi1^2)
#Plot
plot(t, Y1, type = "l", xlab="year", ylab="number of passed course modules", ylim=c
    (600,1400),xlim=c(50,63))
lines(61:62, c(pred1,pred2), type="o", lty=2, col="blue" )
lines(60:61, c(Y1[10],pred1), lty=2, col="blue")
points(61:62, c(upper1,upper2), pch="+", col="red")
points(61:62, c(lower1,lower2), pch="+", col="red")




# ______Question 2.3: Random walk ______________ #
# 2.3.3: Simulate a white noise process of 1000 values with mean zero and s^2 = sqrt
    (2).
# Then figure out a fast and compact way to calculate the process Yt based on this.

Wnoise <- rnorm(n = 1000, mean = 0, sd = 2^0.25)
Yt <- rep(NA, 1000)
#Initialize 1st value
Yt[1] <- 0.25 + Wnoise[1]
#Then, we let the random walk do the walk
for (i in 2:1000) {
  Yt[i] <- Yt[i-1] + Wnoise[i]
}
plot(Wnoise, type="l", xlab="time")

# 2.3.4: Simulate 10 realizations of the process Yt: nonstationary as the acf shows
    a slow decay
W <- matrix(NA, nrow = 1000, ncol = 10)
```

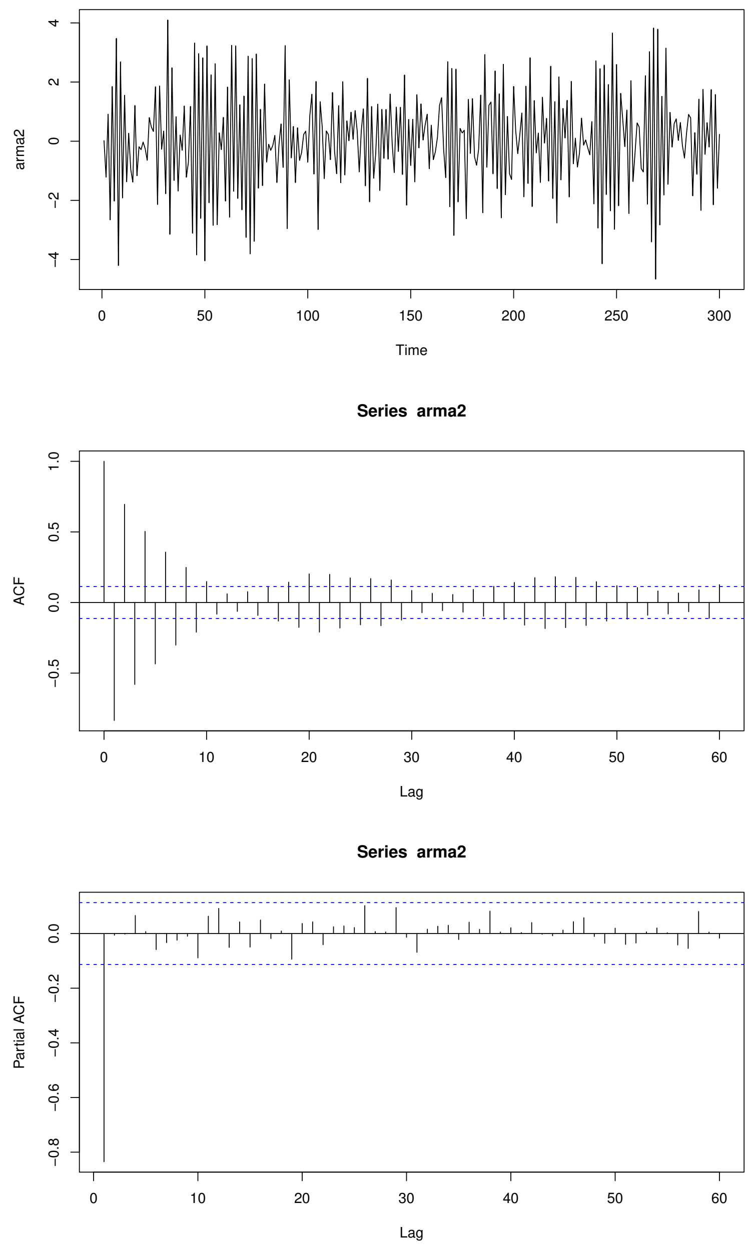

Figure 6: Model 2: A$(1,0,0)$ x $(0,0,0)_{12}$ model with the parameter $\phi_1 = 0.8$

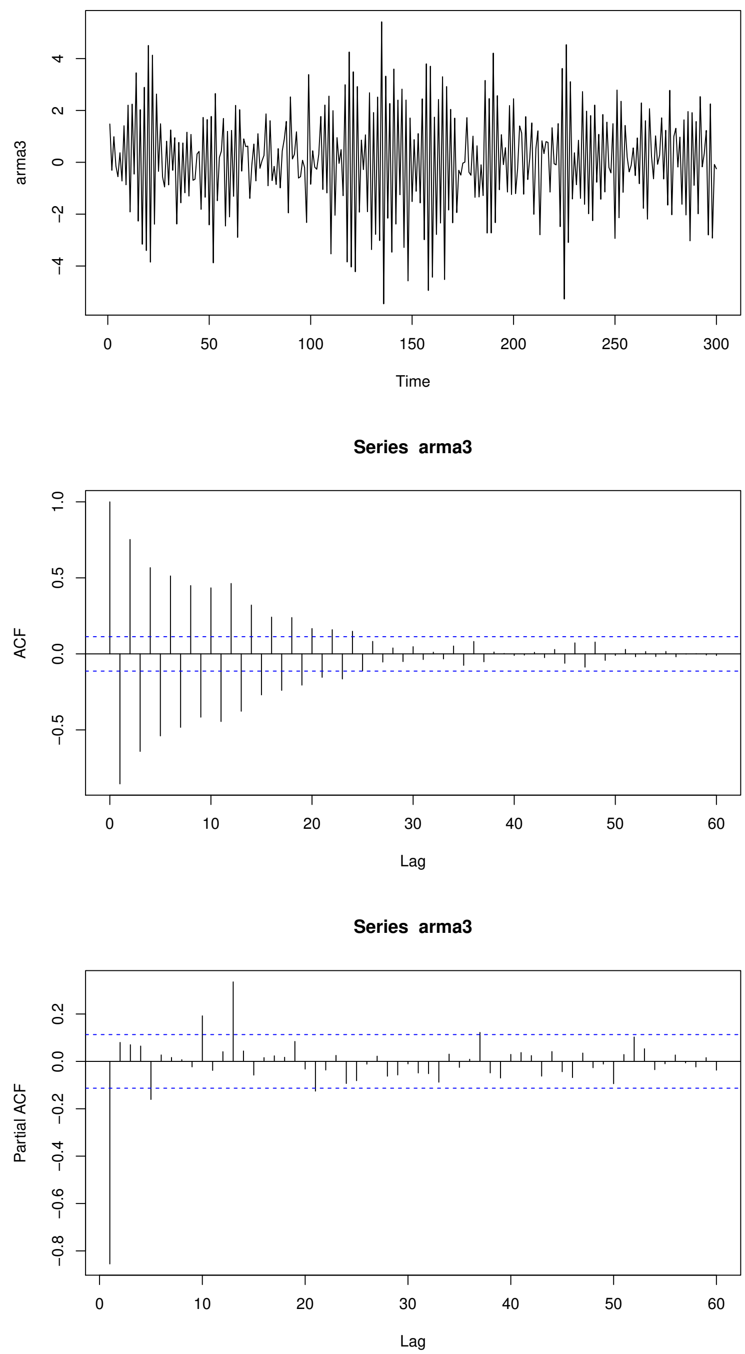

Figure 7: Model 3: A$(1,0,0)$ x $(0,0,1)_{12}$ model with the parameter $\phi_1 = 0.8, \Theta_1 = 0.9$

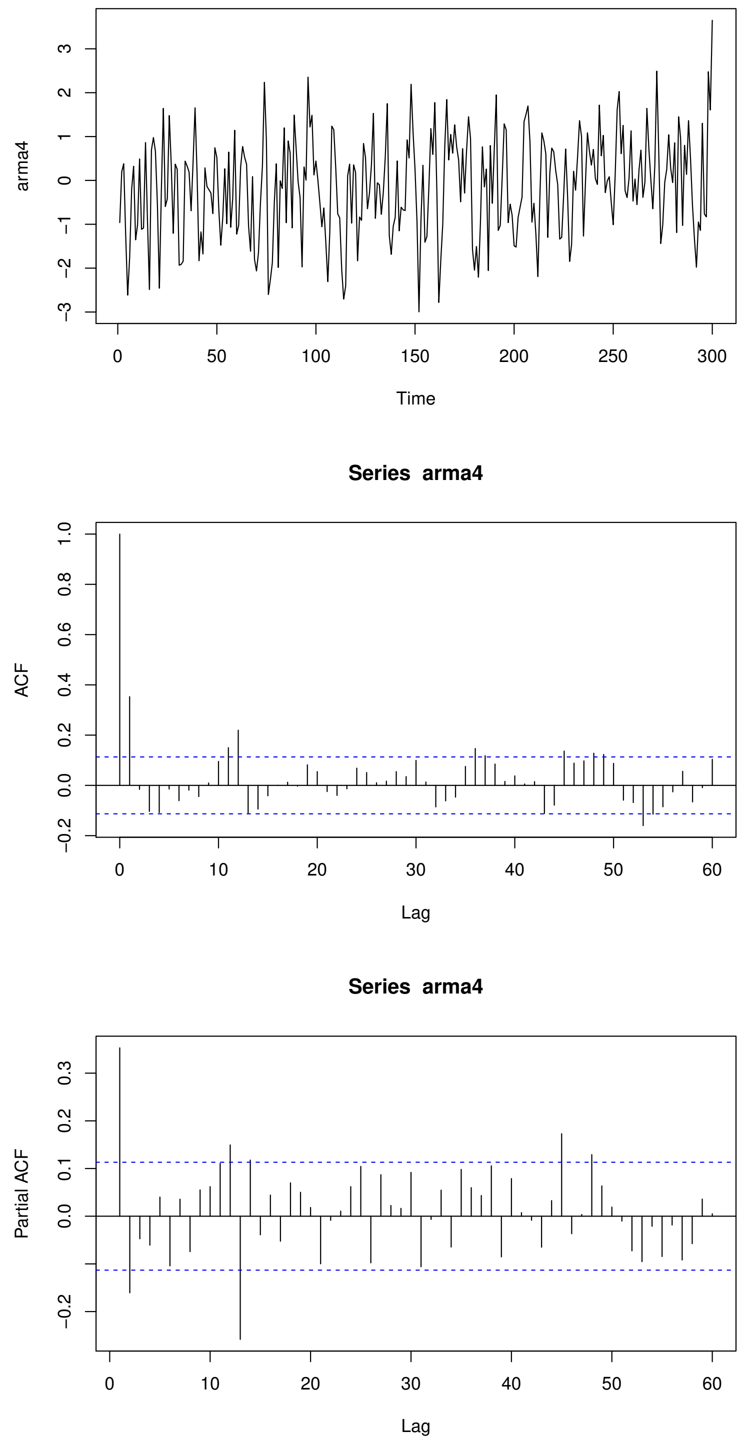

Figure 8: Model 4: A$(0, 0, 1)$ x $(0, 0, 1)_{12}$ model with the parameter $\theta_1 = 0.5, \Theta_1 = 0.4$

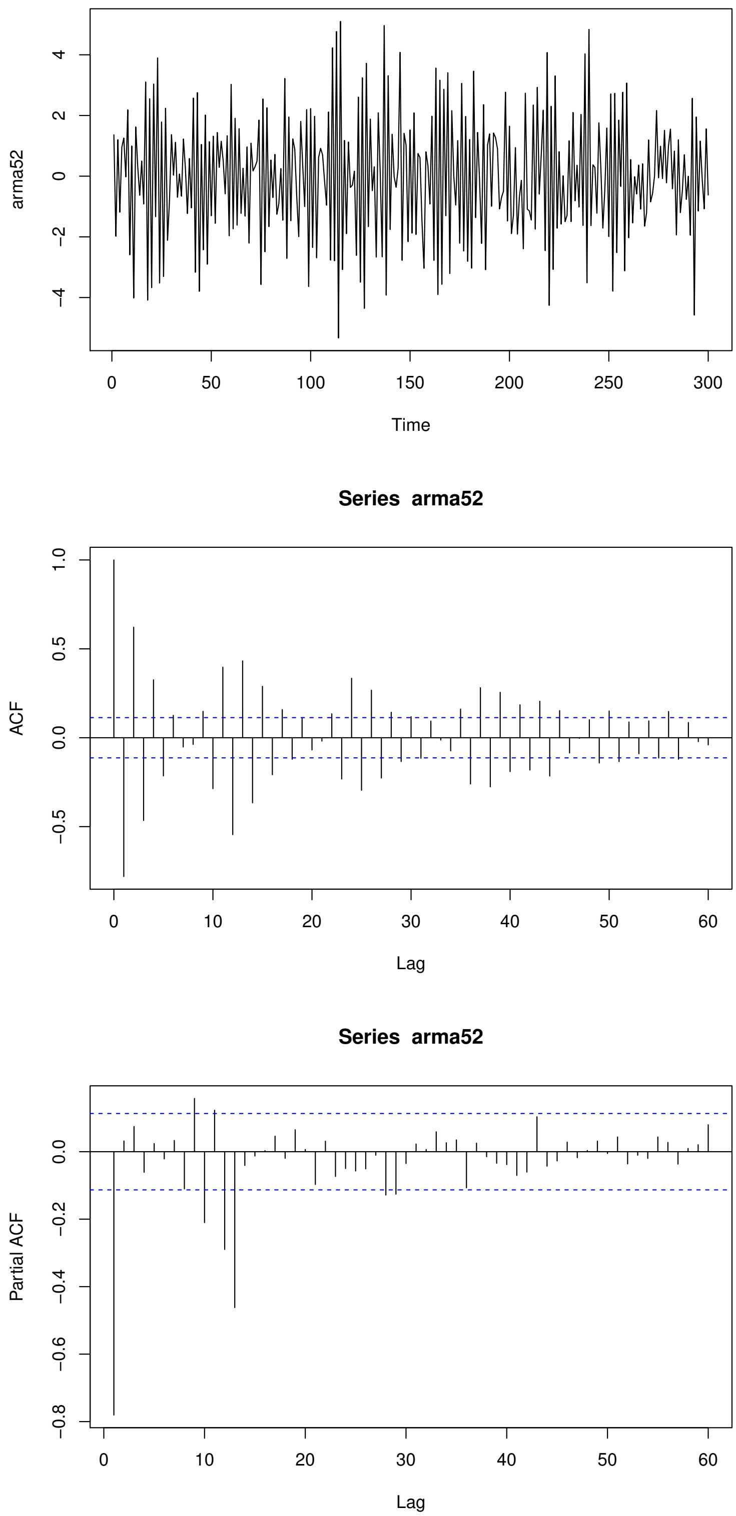

Figure 9: Model 5: A$(1, 0, 0)$ x $(1, 0, 0)_{12}$ model with the parameter $\phi_1 = 0.8, \Phi_1 = 0.6$

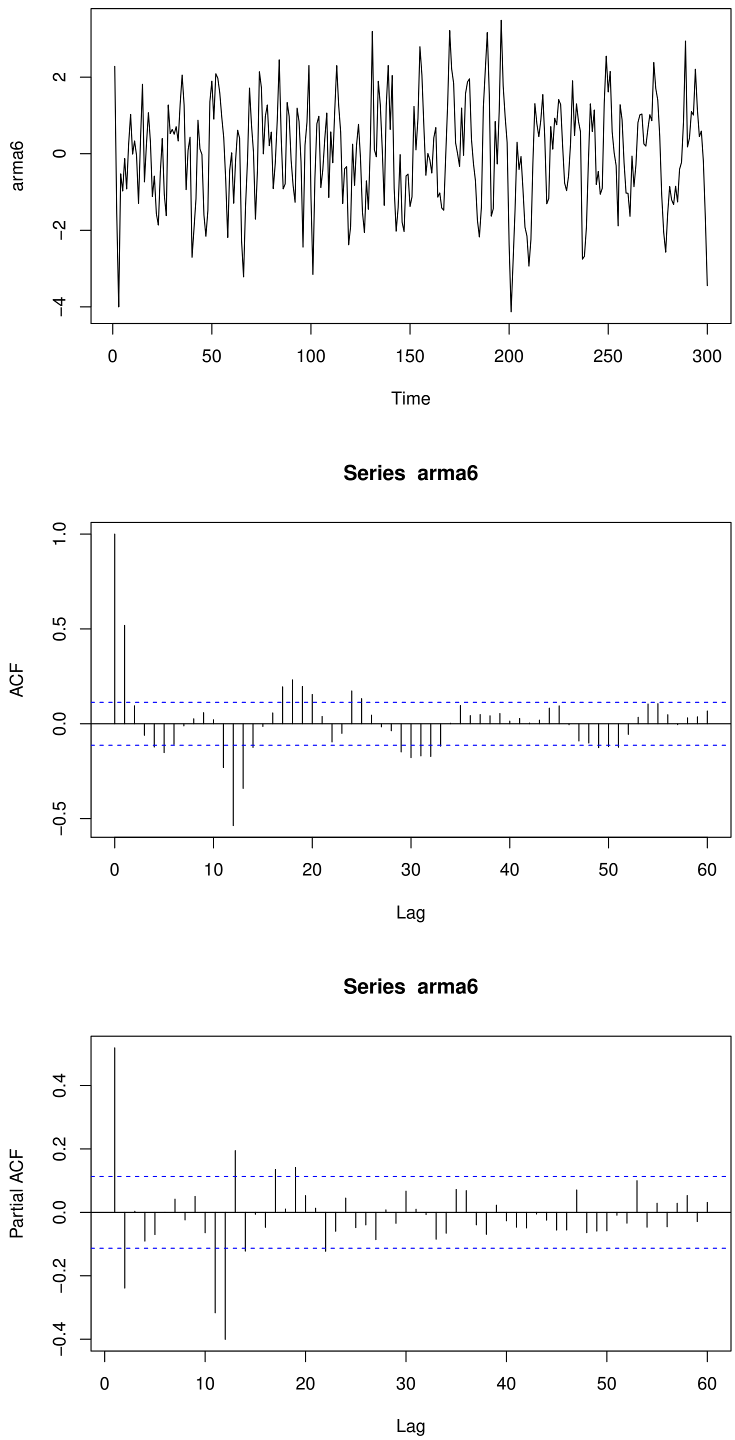

Figure 10: Model 6: A$(0, 0, 1)$ x $(1, 0, 0)_{12}$ model with the parameter $\theta_1 = 0.5, \Phi_1 = 0.6$

```r
Y10 <- matrix(NA, nrow = 1000, ncol = 10)
#create 10 white noises
for (i in 1:10) { W[ ,i] <- rnorm(n=1000, mean=0, sd=2^0.25) }
#initialize 1st value
for (i in 1:10) { Y10[1,i] <- 0.25 + W[1,i] }
#let the random walks, walk
for (j in 1:10) {
  for (i in 2:1000) { Y10[i,j] <- Y10[i-1,j] + W[i,j] }
}
#plot of all 10 random walks
plot(Y10[,1], type="l", ylim=c(-100,80), ylab="Yt", xlab="time")
lines(Y10[,2], col="blue")
lines(Y10[,3], col="red")
lines(Y10[,4], col="yellow")
lines(Y10[,5], col="pink" )
lines(Y10[,6], col="brown")
lines(Y10[,7], col="green")
lines(Y10[,8], col="magenta")
lines(Y10[,9], col="purple")
lines(Y10[,10], col="orange")

#from list to numeric output
ACF <- vector("list", 10)
for (i in 1:10) { ACF[i] <- acf(Y10[,i], plot = FALSE) }
ACFp <- sapply(1:length(ACF), function(i) as.numeric(ACF[[i]]))
#plot their estimated autocorrelation functions
plot(ACFp[,1], ylim=c(0,1), type="l", ylab="ACF", xlab="lag")
lines(ACFp[,2], col="blue")
lines(ACFp[,3], col="red")
lines(ACFp[,4], col="yellow")
lines(ACFp[,5], col="pink" )
lines(ACFp[,6], col="brown")
lines(ACFp[,7], col="green")
lines(ACFp[,8], col="magenta")
lines(ACFp[,9], col="purple")
lines(ACFp[,10], col="orange")

# ______ Question 2.4: Simulating seasonal processes _______________ #

# Simulate the following models (where monthly data are assumed)
# Notice we show 60 lags to see what happens in 5 12month seasons
#1: AR(1) process w/out seasonality
arma1 <- arima.sim(n = 300, list(ar = c(rep(0,11),-0.6)))
plot(arma1)
acf(arma1, lag.max = 60)
pacf(arma1, lag.max = 60)
#2
arma2 <-   arima.sim(n = 300, list(ar = c(-0.8) ))
plot(arma2)
acf(arma2, lag.max = 60)
pacf(arma2, lag.max = 60)
#3
arma3 <- arima.sim(n = 300, list(ar = c(-0.8), ma = c(rep(0,11), 0.5)) )
plot(arma3)
acf(arma3, lag.max = 60)
pacf(arma3, lag.max = 60)
#4
arma4 <- arima.sim(n = 300, list(ma = c(0.5,rep(0,10),0.4,0.2)))
plot(arma4)
acf(arma4, lag.max = 60)
```

```
pacf(arma4, lag.max = 60)
#5
arma5 <- arima.sim(n = 300, list(ar = c(-0.8, rep(0,10),-0.6,-0.48) ) )
plot(arma5)
acf(arma5, lag.max = 60)
pacf(arma5, lag.max = 60)
#6
arma6 <-  arima.sim(n = 300, list(ar = c(rep(0,11),-0.6), ma=c(0.5)))
plot(arma6)
acf(arma6, lag.max = 60)
pacf(arma6, lag.max = 60)
```