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

Emma Demarecaux (s176437)

15/03/2018

The study has been elaborated in colaboration with Adélie Marie Kookai Barre (s170075).

Question 2.1: Stability

2.1.1

Let the process X_t be given by

$$X_t = \epsilon_t + \epsilon_{t-1} + \epsilon_{t-2} + \epsilon_{t-3}$$

where ϵ_t is a white noise process with $\sigma = 0.1$.

Let's investigate analytically the second order moment representation of the process:

The mean value is $E(X_t) = E(\epsilon_t) + E(\epsilon_{t-1}) + E(\epsilon_{t-2}) + E(\epsilon_{t-3})$ hence:

$$E(X_t) = 0$$

Let γ be the autocovariance function of ϵ_t. We can then calculate the autocovariance function of X_t:

$$\gamma_{XX}(h) = Cov[X_t, X_{t+h}]$$

$$\gamma_{XX}(h) = Cov[\epsilon_t + \epsilon_{t-1} + \epsilon_{t-2} + \epsilon_{t-3}, \epsilon_{t+h} + \epsilon_{t+h-1} + \epsilon_{t+h-2} + \epsilon_{t+h-3}]$$

$$\gamma_{XX}(h) = 4\gamma(h) + 3[\gamma(h-1) + \gamma(h+1)] + 2[\gamma(h-2) + \gamma(h+2)] + \gamma(h-3) + \gamma(h+3)$$

Therefore:

$$\gamma_{XX}(h) = \begin{cases} 4\sigma^2 \text{ if } h = 0 \\ 3\sigma^2 \text{ if } h = 1 \text{ or } h = -1 \\ 2\sigma^2 \text{ if } h = 2 \text{ or } h = -2 \\ \sigma^2 \text{ if } h = 3 \text{ or } h = -3 \\ 0 \text{ if } |h| \geq 4 \end{cases}$$

$$\gamma_{XX}(h) = \begin{cases} 0.04 \text{ if } h = 0 \\ 0.03 \text{ if } h = 1 \text{ or } h = -1 \\ 0.02 \text{ if } h = 2 \text{ or } h = -2 \\ 0.01 \text{ if } h = 3 \text{ or } h = -3 \\ 0 \text{ if } |h| \geq 4 \end{cases}$$

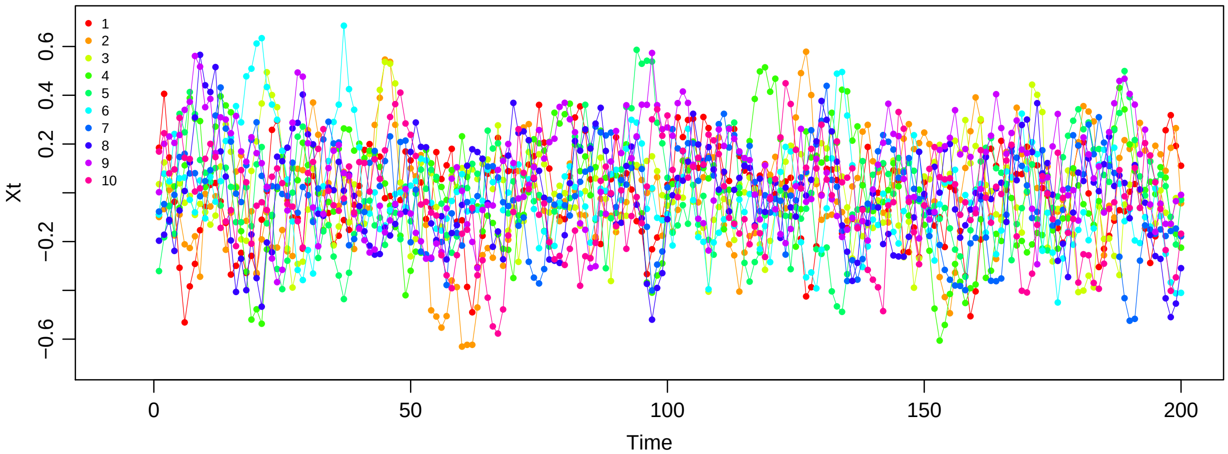
2.1.2

This is a MA(3) process therefore it is stationnary.

Here $\Theta(B) = 1 + B + B^2 + B^3$ and the roots of Θ are -1, i and $-i$. We notice that $|\frac{1}{-1}| = |\frac{1}{i}| = |\frac{1}{-i}| = 1$, therefore these numbers are not within the unit circle and so this MA(3) process is not invertible.

2.1.3

As we have a MA(3) process, we can simply use the function *arima.sim* from R to simulate 10 realisations with 200 observations from the process:



Displaying in the same plot makes things a bit messy.

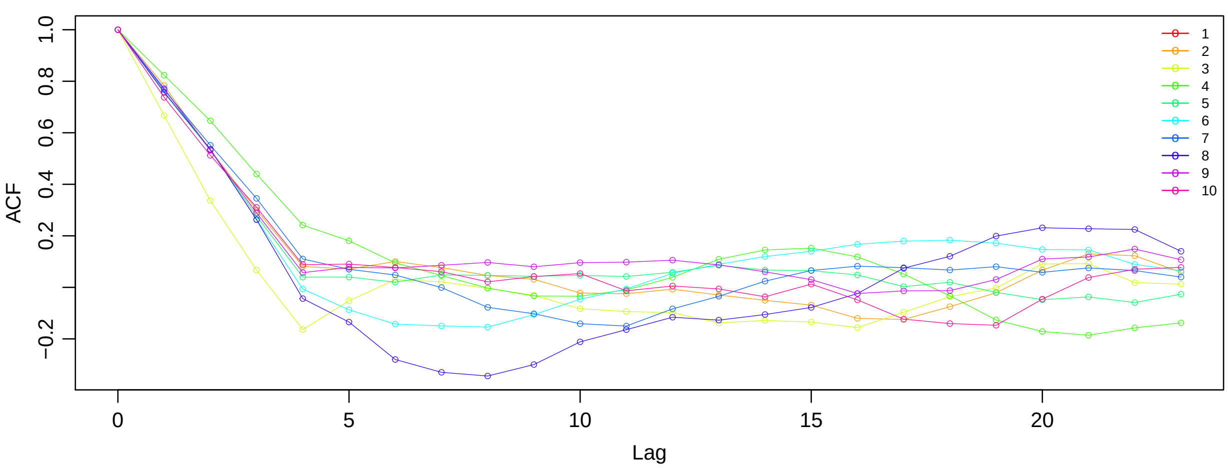
2.1.4

The autocorrelation function is calculated with the following line:

$$\rho_{XX}(h) = \frac{\gamma_{XX}(h)}{\gamma_{XX}(0)}$$

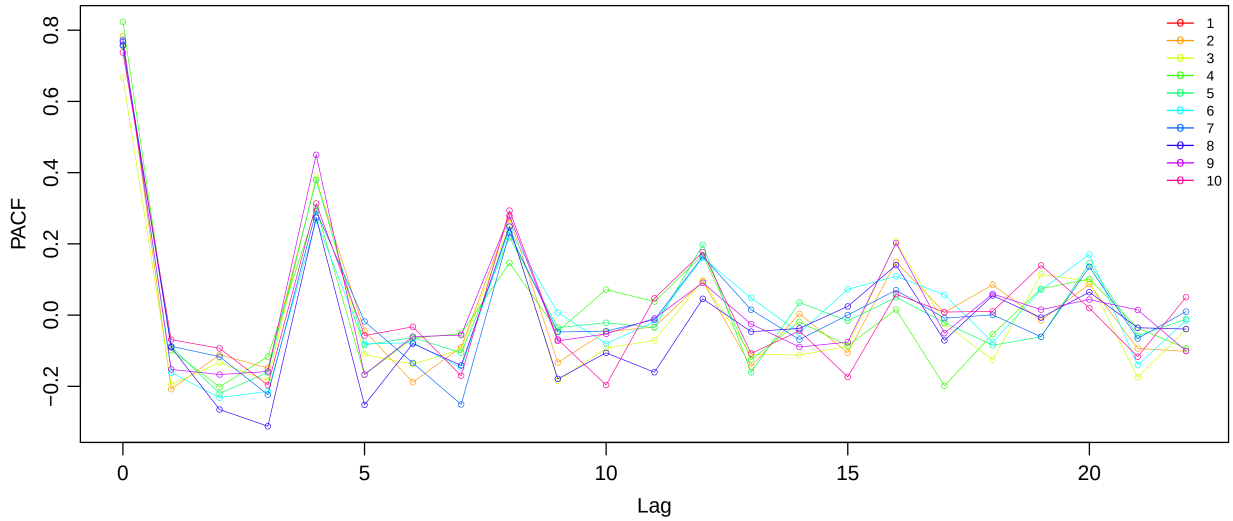
Therefore, we should have:

$$\rho_{XX}(h) = \begin{cases} 1 \text{ if } h = 0 \\ \frac{3}{4} \text{ if } h = 1 \text{ or } h = -1 \\ \frac{1}{2} \text{ if } h = 2 \text{ or } h = -2 \\ \frac{1}{4} \text{ if } h = 3 \text{ or } h = -3 \\ 0 \text{ if } |h| \geq 4 \end{cases}$$



As we have a MA(3) process we can verify that the autocorrelation is close to 0 after lag 3.

2.1.5

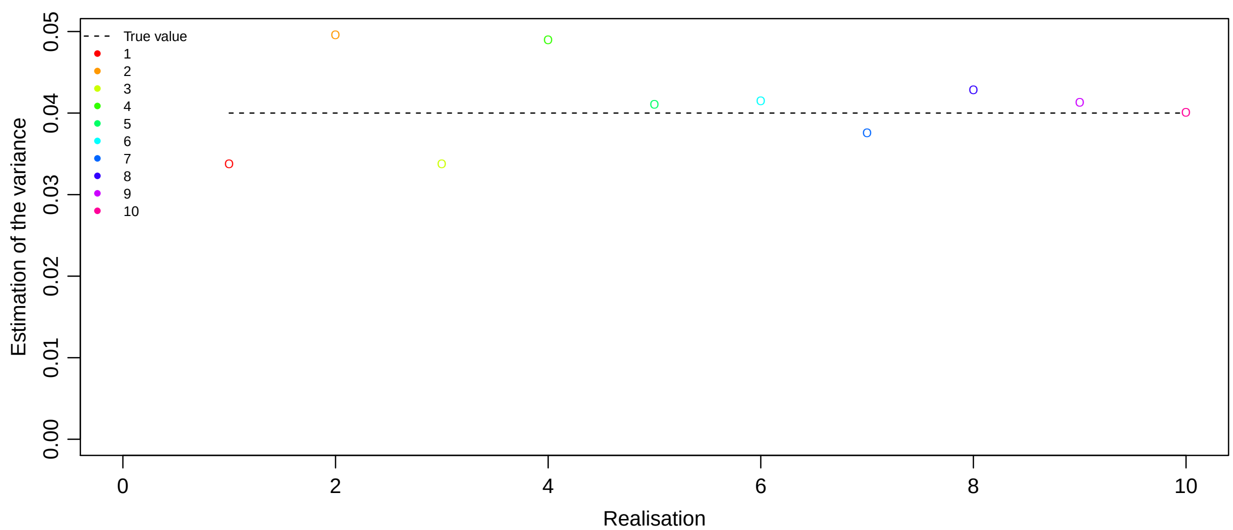


We can verify that the partial autocorrelation is dominated by a damped exponential and sine function.

2.1.6

We can calculate the estimate of $\gamma(0)$:

$$C_{XX}(0) = \hat{\gamma}(0) = \frac{1}{200}\sum_{t=1}^{200}(X_t - \bar{X})^2$$



The estimate of the variance is quite close to the analytical result:0.4. As we said before, we observe that the numerical results are compatible with the fact that we have simulated ten times a MA(3) process.

Question 2.2: Predicting temperature in district heating network

Based on historical data for the forward temperature to a house the following model of the needed temperature has been identified:

$$(1-0.5B+0.3B^2)(1-0.9B^{12})(Y_t-\mu)=\epsilon_t$$

where ϵ_t is a white-noise process with variance $\sigma_\epsilon^2=0.5^2$ and where $\mu=55$. It is an ARMA(2,0)x(1,0)$_{s=12}$

It can be decomposed in:

$$Y_t-\mu=0.5(Y_{t-1}-\mu)-0.3(Y_{t-2}-\mu)+0.9(Y_{t-12}-\mu)-0.45(Y_{t-13}-\mu)+0.27(Y_{t-14}-\mu)+\epsilon_t$$

We want to predict the values of Y_t corresponding to $t=$ Y2017M12 and $t=$ Y2018M1, together with 95% prediction intervals for the predictions. If we want to minimize the expected squared error the optimal prediction is the conditional expectation:

$$\hat{Y}_{t+k|t}=E[Y_{t+k}|Y_t,Y_{t-1},Y_{t-2},...]$$

1-step prediction (Y2017M12): $\hat{Y}_{t+1|t}=E[Y_{t+1}|Y_t,Y_{t-1},Y_{t-2},...]$

2-step prediction (Y2018M01): $\hat{Y}_{t+2|t}=E[Y_{t+2}|Y_t,Y_{t-1},Y_{t-2},...]$

Hence we have:

$$\begin{cases}\hat{Y}_{t+1|t}=\mu+0.5(Y_t-\mu)-0.3(Y_{t-1}-\mu)+0.9(Y_{t-11}-\mu)-0.45(Y_{t-12}-\mu)+0.27(Y_{t-13}-\mu)\\ \hat{Y}_{t+2|t}=\mu+0.5(\hat{Y}_{t+1|t}-\mu)-0.3(Y_t-\mu)+0.9(Y_{t-10}-\mu)-0.45(Y_{t-11}-\mu)+0.27(Y_{t-12}-\mu)\end{cases}$$

The prediction error is:

$$e_{t+k|t}=Y_{t+k|t}-\hat{Y}_{t+k|t}$$

Hence:

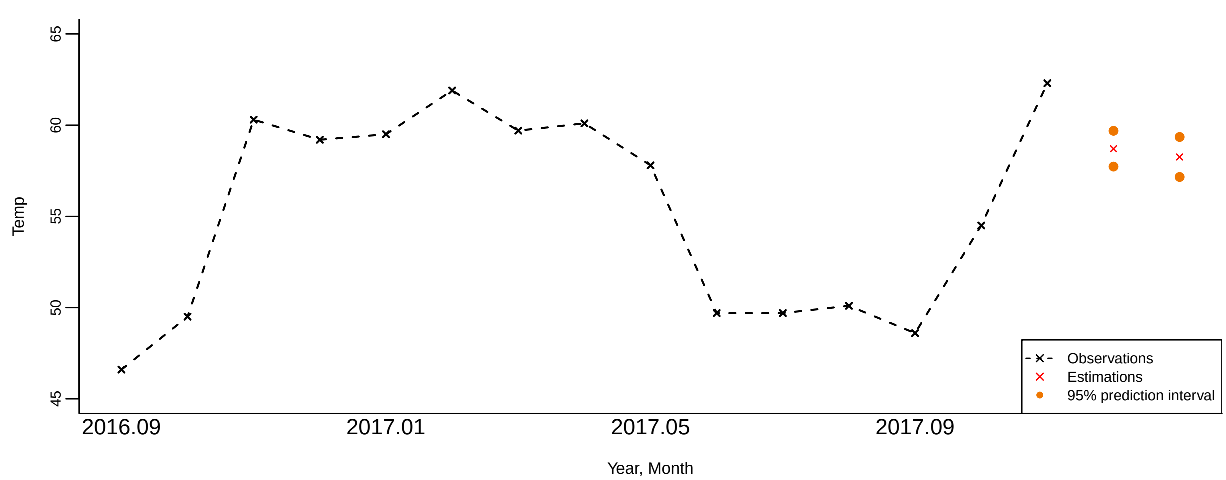
$$\begin{cases}e_{t+1|t}=\epsilon_{t+1}\\ e_{t+2|t}=\epsilon_{t+2}+0.5\epsilon_{t+1}\end{cases}$$

We can now calculate the variance of the prediction error:

$$\begin{cases}\text{Var}[e_{t+1|t}]=\sigma_\epsilon^2=0.5^2\\ \text{Var}[e_{t+2|t}]=\sigma_\epsilon^2(1+0.5^2)=0.5^2(1+0.5^2)\end{cases}$$

The 95% prediction intervals are then (using the quantile of the normal distribution):

$$\begin{cases}\hat{Y}_{t+1|t}\pm u_{0.975}*0.5\\ \hat{Y}_{t+2|t}\pm u_{0.975}*0.5\sqrt{1+0.5^2}\end{cases}$$



All the results are gathered in the following table:

```
Results
```

```
##   Observation Prediction        Sd Lower_bound Upper_bound
## 1    Y2017M12     58.710 0.500000    57.73002    59.68998
## 2    Y2018M01     58.256 0.559017    57.16035    59.35165
```

Qestion 2.3: Simulating seasonal processes

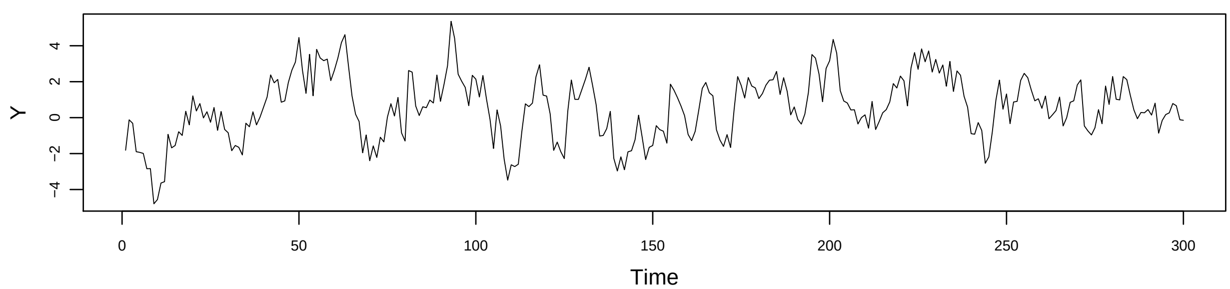
A process Y_t is said to follow a multiplicative $(p, d, q) * (P, D, Q)_s$ seasonal model if:

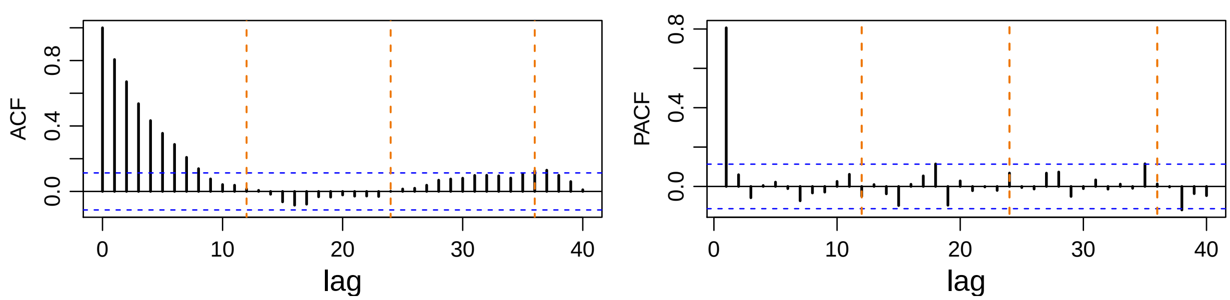
$$\phi(B)\Phi(B^s)\nabla^d\nabla_s^D Y_t = \theta(B)\Theta(B^s)\epsilon_t$$

All the following models have been created manually with a white noise ϵ_t such that $E[\epsilon_t] = 0$ and $\sigma = 1$. We used 300 time steps. For each models the simulation is displayed along with the associated autocorrelation functions (ACF and PACF). 40 lags are represented to observe the seasonal pics at lags 12, 24 and 36 (represented by the vertical orange lines on the plots).

AR(1)

$$Y_t = 0.85Y_{t-1} + \epsilon_t$$

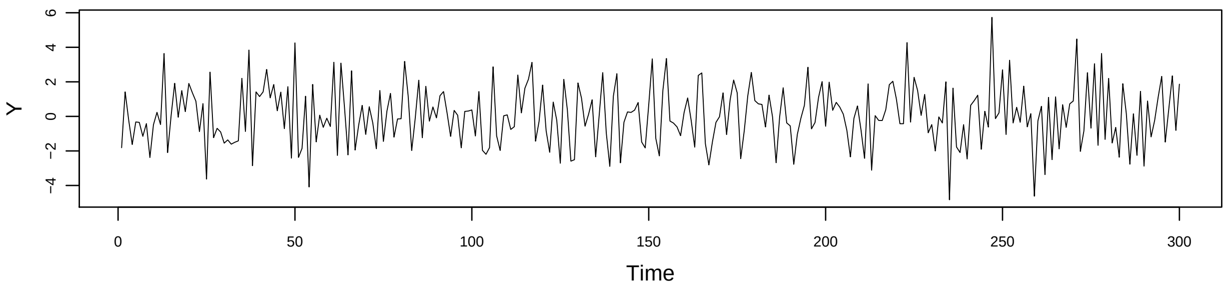


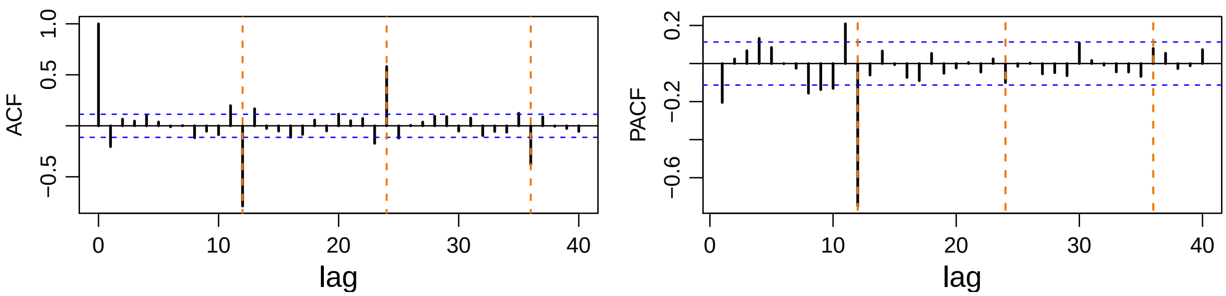


On the ACF, we can observe an exponential decaying. On the PACF, we only observe one significative peak at lag 1 showing that it is a non seasonal model.

AR(1)$_{s=12}$

$$Y_t = -0.85Y_{t-12} + \epsilon_t$$

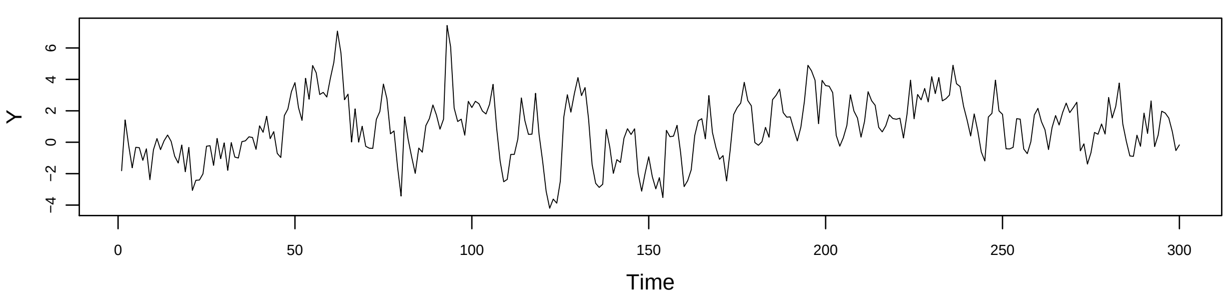


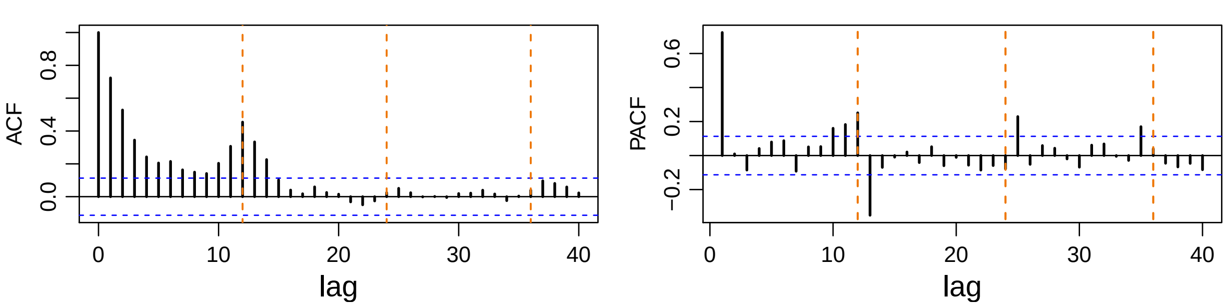


On the ACF one can observe alternative sign clear peaks for all the seasonal lags (12, 24 and 36) with a decreasing amplitude. On the PACF there is one big negative peak at the first seasonal lag (12). Hence, the information is only in the seasonal lags.

ARMA(1,0)x(0,1)$_{s=12}$

$$Y_t = 0.8Y_{t-1} + 0.9\epsilon_{t-12} + \epsilon_t$$

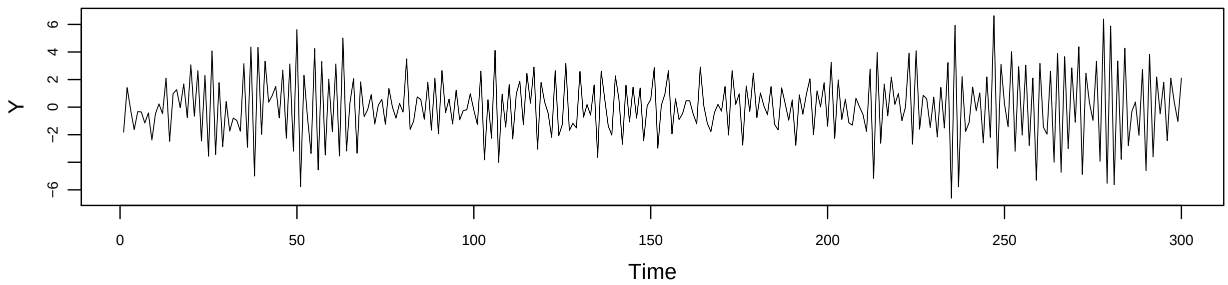


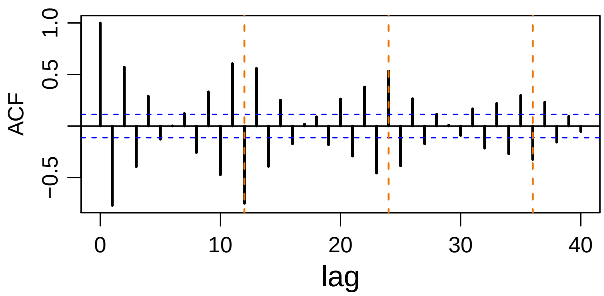


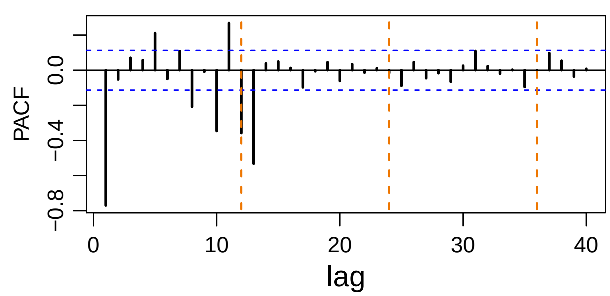
One can observe on the ACF an exponential decay and a significative peak at lag 12. The PACF shows a clear peak at lag 1 and medium peaks around lag 12.

ARMA(1,0)x(1,0)$_{s=12}$

$$Y_t = -0.7Y_{t-1} - 0.8Y_{t-12} - 0.56Y_{t-13} + \epsilon_t$$



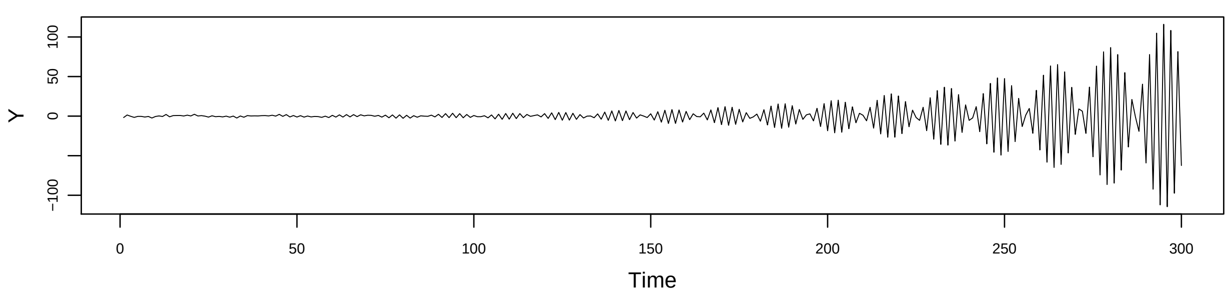


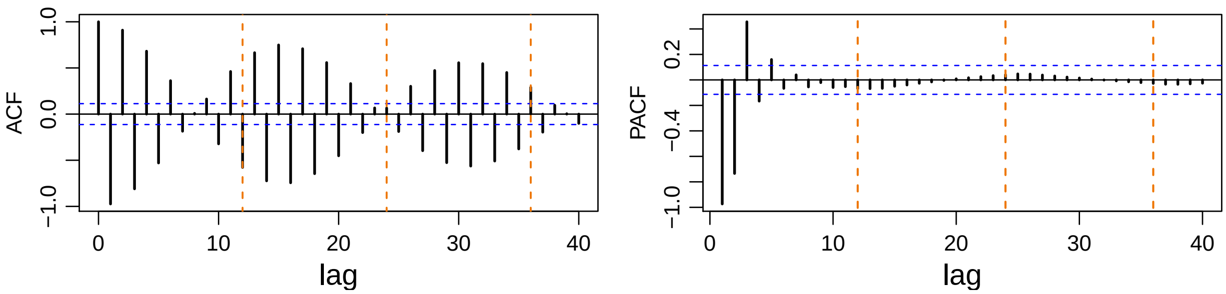


On the ACF there is an alternative sign exponential decay with medium peaks around the seasonal lags (12,24,36). The PACF shows important negative peaks at lags 1,12,13 which is consistent with the above expression of the model.

ARMA(2,0)x(1,0)$_{s=12}$

$$Y_t = -0.6Y_{t-1} + 0.3Y_{t-2} - 0.8Y_{t-12} - 0.48Y_{t-13} + 0.24Y_{t-14} + \epsilon_t$$

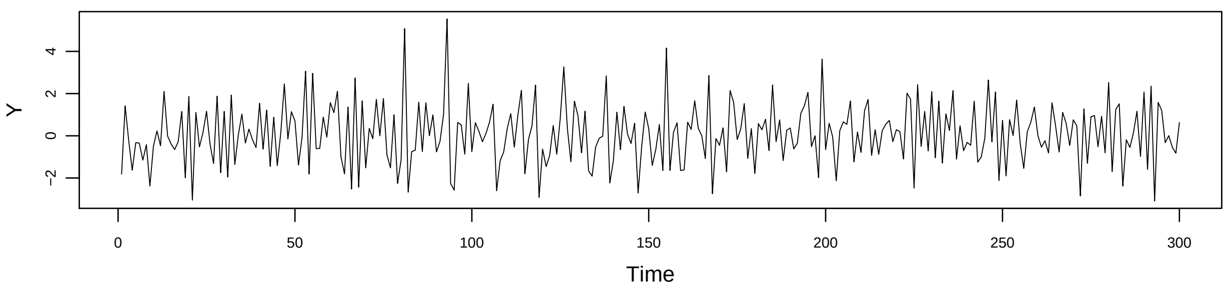


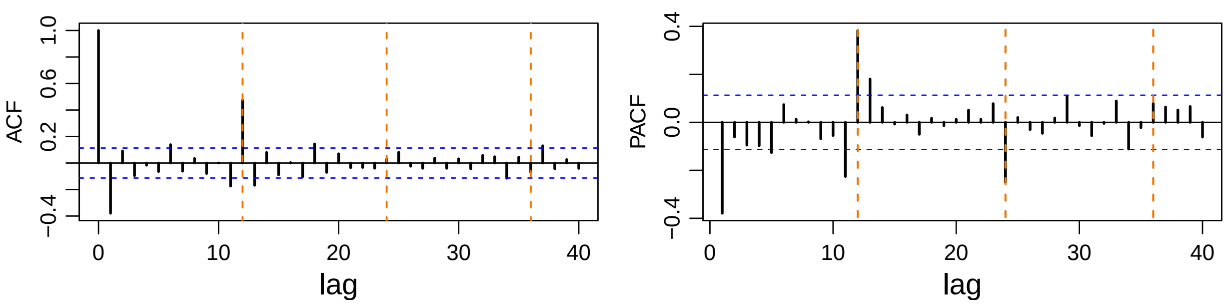


On the ACF there is an alternative sign exponential decay and the trend corresponds to a sine function. The PACF shows important negative peaks at lags 1, 2 and 3, and smaller ones at lages 12, 13 and 14 which is consistent with the above expression of the model.

ARMA(0,1)x(0,1)$_{s=12}$

$$Y_t = -0.32\epsilon_{t-13} + 0.8\epsilon_{t-12} - 0.4\epsilon_{t-1} + \epsilon_t$$





On the ACF there are significative peaks at lags 1 and 12 which are a good indicators of the MA(1) and MA(1)$_{12}$ behavior. On the PACF one can observe clear peaks at lags 1, 11 and 12.

Summary

One can observe that all the models are stationnary and that except model 1, they are all seasonal. We choose to calculate the ACF and PACF to lag 40 in order to cover three seasons and obserse the decay of ACF and PACF. For models with one parameter (1 and 2) the identification is pretty simple using the ACF and the PACF. However for all the other ones that have two parameters it could sometimes be difficult to identify the different behaviors as one is masking the other one the ACF or the PACF.

Appendix

```
################Exercise 1
# Simulate 10 realisations with 200 observations from the process and plot then
N<-10
n<-200
sd<-0.1
col<-rainbow(N)

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(1, type="n", ylab='Xt',xlab='Time', xlim=c(-7, 200), ylim=c(-0.71, 0.75))

Y<-c()
for(i in 1:N) {
  Y<-rbind(Y,arima.sim(model = list(ma=c(1,1,1), order=c(0,0,3)), n = n, sd=sd))
  lines(1:n,Y[i,1:n],type = 'o', pch = 16, col=col[i], lwd=0.5,cex=0.7)
}
legend('topleft',legend = c("1","2","3","4","5","6","7","8","9","10"),
       col = col, pch = c(16,16,16,16,16,16,16,16,16,16),
       cex=0.7, bty = 'n')


# Estimate the ACF for each realisation and plot those
acf <- c()

for(i in 1:N) {
  acf <- rbind(acf,acf(Y[i,1:n],plot = FALSE)$acf)
}

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(1, type="n", ylab='ACF',xlab='Lag', xlim=c(0, 23), ylim=c(min(acf),max(acf)))

for (i in 2:N) {
  lines(0:23,acf[i,1:24],type='o',cex=0.6, bty='l',
        col=col[i],lwd=0.5)
}
legend('topright',legend = c("1","2","3","4","5","6","7","8","9","10"),
       col = col, pch = c('o','o','o','o','o','o','o','o','o','o'),
       lty = c(1,1,1,1,1,1,1,1,1,1), cex=0.7, bty = 'n')

# Estimate the PACF for each realisation and plot those
pacf <- c()

for(i in 1:N) {
  pacf <- rbind(pacf,pacf(Y[i,1:n],plot = FALSE)$acf)
}

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(1, type="n", ylab='PACF',xlab='Lag', xlim=c(0, 22), ylim=c(min(pacf),max(pacf)))
```

```
for (i in 2:N) {
  lines(0:22,pacf[i,1:23],type ='o',cex=0.6, bty='l', col=col[i],lwd=0.5)
}
legend('topright',legend = c("1","2","3","4","5","6","7","8","9","10"),
       col = col, pch = c('o','o','o','o','o','o','o','o','o','o'),
       lty = c(1,1,1,1,1,1,1,1,1,1), cex=0.7, bty = 'n')

#Estimate the variance of the realisations
var_hat <- c()

for(i in 1:N) {
  var_hat <- c(var_hat,acf(Y[i,1:n], type='covariance', plot = FALSE)$acf[1])
}

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(1, type="n", xlab="Realisation", ylab="Estimation of the variance",
     xlim=c(0, 10), ylim=c(0.00, max(var_hat)))
for (i in 1:N) {
  points(i,var_hat[i],type = 'p',cex=0.6,pch='O', bty='l', col=col[i],lwd=0.5)
}
lines(1:10,rep(0.04,10),lty=2,col='black')
legend('topleft',legend = c('True value',"1","2","3","4","5","6","7","8","9","10"),
       col = c('black',col), pch = c(NA,16,16,16,16,16,16,16,16,16,16),
       cex=0.7, lty = c(2,NA,NA,NA,NA,NA,NA,NA,NA,NA,NA), bty = 'n')

#################Exercice 2
#Study of the model
Time<-c(1:15)
Temp<-c(46.6,49.5,60.3,59.2,59.5,61.9,59.7,60.1,57.8,49.7,49.7,50.1,48.6,54.5,62.3)

data<-rbind(Time,Temp)

mean<-55
sd<- 0.5
a1<- 0.5
a2<- -0.3
a4<- 0.9
a5<- -0.45
a6<- 0.27

Model <- function(Y1,Y2,Y12,Y13,Y15,et){
  mean+a1*(Y1-mean)+a2*(Y2-mean)+a4*(Y12-mean)+a5*(Y13-mean)+a6*(Y14-mean)+et
}

#Predictions
Y<-rbind(c(Time,c(16,17)),c(Temp,NA,NA))

for(t in 16:17)
{
  Y1 <- Y[2,Y[1,]==(t-1)]
  Y2 <- Y[2,Y[1,]==(t-2)]
  Y12 <- Y[2,Y[1,]==(t-12)]
```

```
  Y13 <- Y[2,Y[1,]==(t-13)]
  Y14 <- Y[2,Y[1,]==(t-14)]
  et <- 0
  Y[2,Y[1,]==t]<- Model(Y1,Y2,Y12,Y13,Y14,et)
}

sd16=sd
sd17=sqrt((1+0.5^2))*sd

t95_16<-qnorm(c(0.025,0.975))*sd16
t95_17<-qnorm(c(0.025,0.975))*sd17

Yint_95<-rbind(Y[2,16]+t95_16,Y[2,17]+t95_17)

#plot of the results
par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(Time,Temp, xlim=c(1,17),ylim=c(45,65), xaxt="n", type = 'o',lwd=1.5,lty=2,ann=FALSE,bty='l',pch=4,
     cex.axis = 0.7, mgp=c(3,0.5,0))
# Modification de l'axe des x
xtick<-c(2016.09,'','','',2017.01,'','','',2017.05,'','','',2017.09,'','')
axis(side=1, at=xtick)
mtext(xtick,side=1,at=Time,col='black')
mtext(x=xtick, labels = xtick)
mtext(side = 1,text = 'Year, Month', line = 1.5, cex = 0.78)
mtext(side = 2,text = 'Temp', line = 1.9, cex = 0.78)
points(c(16,17),Y[2,16:17],pch=4,cex=0.6,col='red')
points(c(16,17),Yint_95[,1],pch=16,col='darkorange2')
points(c(16,17),Yint_95[,2],pch=16,col='darkorange2')
legend('bottomright',legend = c("Observations", "Estimations", '95% prediction interval'),
       lty = c(2,NA,NA),col = c("black","red",'darkorange2'), pch = c(4,4,16,16),cex=0.7)

Results=data.frame(Observation = c('Y2017M12','Y2018M01'),
                   Prediction=c(Y[2,16],Y[2,17]),Sd = c(sd16,sd17),
                   Lower_bound = c(Yint_95[1],Yint_95[2]),
                   Upper_bound = c(Yint_95[3],Yint_95[4]))
#################Exercice 3

#Defintion of the white noise
mean<-0
sd<-1
N<-300
eps<-rnorm(N, mean = mean, sd=sd)
lag=40
s<-12

#AR(1)
Y<-vector(length = N)
Y[1]<-eps[1]
phi1<--0.85
for (t in 2:N) {
  Y[t]<-eps[t]-phi1*Y[t-1]
}
```

```
par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(x = 1:N, y = Y, type = 'l', cex.axis = 0.7, lwd = 0.7, xlab='Time', ylab='Y')
lines(X,col='red')
par(mfrow=c(1,2))
acf(x = Y, lag = lag, type = 'correlation',ann=FALSE, lwd=2, main ='', xlab='lag'
    , ylab='ACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'ACF', line = 2, cex = 1)

pacf(x = Y, lag = lag, main = '', ann=FALSE, lwd=2,xlab='lag',ylab='PACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'PACF', line = 2, cex = 1)

#AR(1)s=12
Y<-vector(length = N)
Y[1:s]<-eps[1:s]
psi1<-0.85

for (t in (s+1):N)
{
  Y[t]<-eps[t]-psi1*Y[t-s]
}

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(x = 1:N, y = Y, type = 'l', cex.axis = 0.7, lwd = 0.7, xlab='Time', ylab='Y')

par(mfrow=c(1,2))
acf(x = Y, lag = lag, type = 'correlation',ann=FALSE, lwd=2, main ='', xlab='lag'
    , ylab='ACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'ACF', line = 2, cex = 1)

pacf(x = Y, lag = lag, main = '', ann=FALSE, lwd=2,xlab='lag',ylab='PACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'PACF', line = 2, cex = 1)

#ARMA(1,0)x(0,1)s=12
Y<-vector(length = N)
Y[1:s]<-eps[1:s]
```

```r
phi1<- -0.8
Theta1<-0.9

for (t in (s+1):N)
{
  Y[t]<- -phi1*Y[t-1] + eps[t] + Theta1*eps[t-s]
}

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(x = 1:N, y = Y, type = 'l', cex.axis = 0.7, lwd = 0.7, xlab='Time', ylab='Y')

par(mfrow=c(1,2))
acf(x = Y, lag = lag, type = 'correlation',ann=FALSE, lwd=2, main ='', xlab='lag'
    , ylab='ACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'ACF', line = 2, cex = 1)

pacf(x = Y, lag = lag, main = '', ann=FALSE, lwd=2,xlab='lag',ylab='PACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'PACF', line = 2, cex = 1)

#ARMA(1,0)x(1,0)s=12
Y<-vector(length = N)
Y[1:(s+1)]<-eps[1:(s+1)]
phi1<-0.7
psi1<-0.8

for (t in 14:N)
{
  Y[t]<- -phi1*Y[t-1]-psi1*Y[t-s]-phi1*psi1*Y[t-s-1]+eps[t]
}

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(x = 1:N, y = Y, type = 'l', cex.axis = 0.7, lwd = 0.7, xlab='Time', ylab='Y')

par(mfrow=c(1,2))
acf(x = Y, lag = lag, type = 'correlation',ann=FALSE, lwd=2, main ='', xlab='lag'
    , ylab='ACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'ACF', line = 2, cex = 1)

pacf(x = Y, lag = lag, main = '', ann=FALSE, lwd=2,xlab='lag',ylab='PACF')
```

```
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'PACF', line = 2, cex = 1)

#ARMA(2,0)x(1,0)s=12
Y<-vector(length = N)
Y[1:(s+2)]<-eps[1:(s+2)]
phi1<-0.6
phi2<- -0.3
psi1<-0.8

for (t in 14:N)
{
  Y[t]<- -phi1*Y[t-1]-psi1*Y[t-s]-phi1*psi1*Y[t-s-1]-phi2*Y[t-2]
  -phi2*psi1*Y[t-s-2]+eps[t]
}

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(x = 1:N, y = Y, type = 'l', cex.axis = 0.7, lwd = 0.7, xlab='Time', ylab='Y')

par(mfrow=c(1,2))
acf(x = Y, lag = lag, type = 'correlation',ann=FALSE, lwd=2, main ='', xlab='lag'
    , ylab='ACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'ACF', line = 2, cex = 1)

pacf(x = Y, lag = lag, main = '', ann=FALSE, lwd=2,xlab='lag',ylab='PACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'PACF', line = 2, cex = 1)

#ARMA(0,1)x(0,1)s=12
Y<-vector(length = N)
Y[1:13]<-eps[1:13]
theta1<- -0.4
Theta1<-0.8

for (t in 14:N)
{
  Y[t]<- eps[t] + theta1*eps[t-1]+Theta1*eps[t-12]+theta1*Theta1*eps[t-13]
}

par(mfrow=c(1,1))
par(mgp=c(2, 0.7,0), mar=c(3,3,2,1))
plot(x = 1:N, y = Y, type = 'l', cex.axis = 0.7, lwd = 0.7, xlab='Time', ylab='Y')
```

```r
par(mfrow=c(1,2))
acf(x = Y, lag = lag, type = 'correlation',ann=FALSE, lwd=2, main ='', xlab='lag'
    , ylab='ACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.4, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'ACF', line = 2, cex = 1)

pacf(x = Y, lag = lag, main = '', ann=FALSE, lwd=2,xlab='lag',ylab='PACF')
lines(x = c(12,12), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(24,24), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
lines(x = c(36,36), y = c(-10,10), type = 'l', lty = 2,lwd=1.5, col='darkorange2')
mtext(side = 1,text = 'lag', line = 2, cex = 1.3)
mtext(side = 2,text = 'PACF', line = 2, cex = 1)
```