ENV 797 - Time Series Analysis for Energy and Environment Applications | Spring 2024

Assignment 6 - Due date 02/28/24

Student Name

Directions

You should open the .rmd file corresponding to this assignment on RStudio. The file is available on our class repository on Github.

Once you have the file open on your local machine the first thing you will do is rename the file such that it includes your first and last name (e.g., "LuanaLima_TSA_A06_Sp24.Rmd"). Then change "Student Name" on line 4 with your name.

Then you will start working through the assignment by **creating code and output** that answer each question. Be sure to use this assignment document. Your report should contain the answer to each question and any plots/tables you obtained (when applicable).

When you have completed the assignment, **Knit** the text and code into a single PDF file. Submit this pdf using Sakai.

R packages needed for this assignment: "ggplot2", "forecast", "tseries" and "sarima". Install these packages, if you haven't done yet. Do not forget to load them before running your script, since they are NOT default packages.

```
#Load/install required package here
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(tseries)
library(sarima)
```

```
## Loading required package: stats4
```

```
##
## Attaching package: 'sarima'
```

```
## The following object is masked from 'package:stats':
##
##     spectrum
```

```r
library(ggplot2)
library(cowplot)
```

This assignment has general questions about ARIMA Models.

Q1

Describe the important characteristics of the sample autocorrelation function (ACF) plot and the partial sample autocorrelation function (PACF) plot for the following models:

- AR(2)

  Answer: p=2. The ACF plot of an AR(2) model will have a exponential/slow decay and the PACF plot will have a clear cut off at lag 2.

- MA(1)

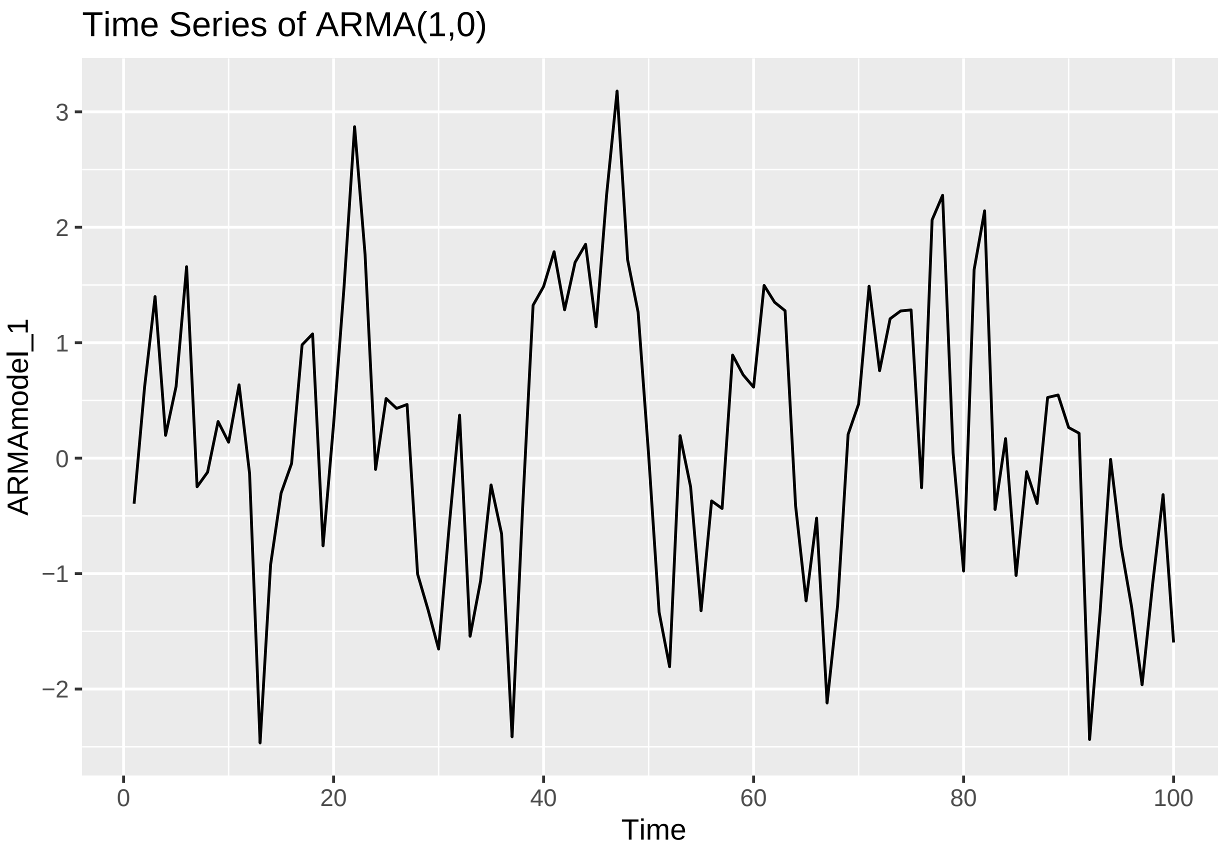
  Answer: q=1. The PACF plot of a MA(1) model will have a exponential/slow decay and the ACF plot will have a clear cut off at lag 1.

Q2

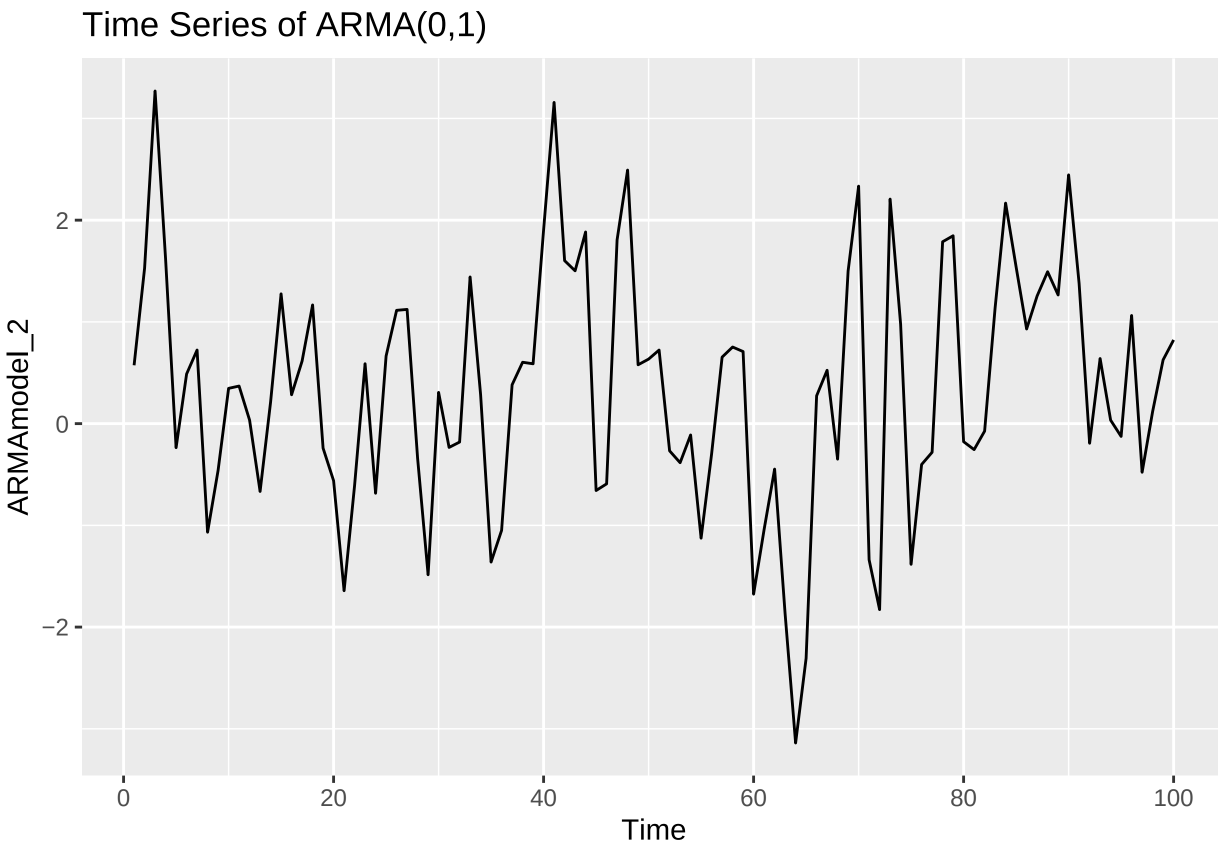
Recall that the non-seasonal ARIMA is described by three parameters ARIMA(p, d, q) where p is the order of the autoregressive component, d is the number of times the series need to be differenced to obtain stationarity and q is the order of the moving average component. If we don't need to difference the series, we don't need to specify the "I" part and we can use the short version, i.e., the ARMA(p, q).

(a) Consider three models: ARMA(1,0), ARMA(0,1) and ARMA(1,1) with parameters $\phi = 0.6$ and $\theta = 0.9$. The ϕ refers to the AR coefficient and the θ refers to the MA coefficient. Use the `arima.sim()` function in R to generate $n = 100$ observations from each of these three models. Then, using `autoplot()` plot the generated series in three separate graphs.

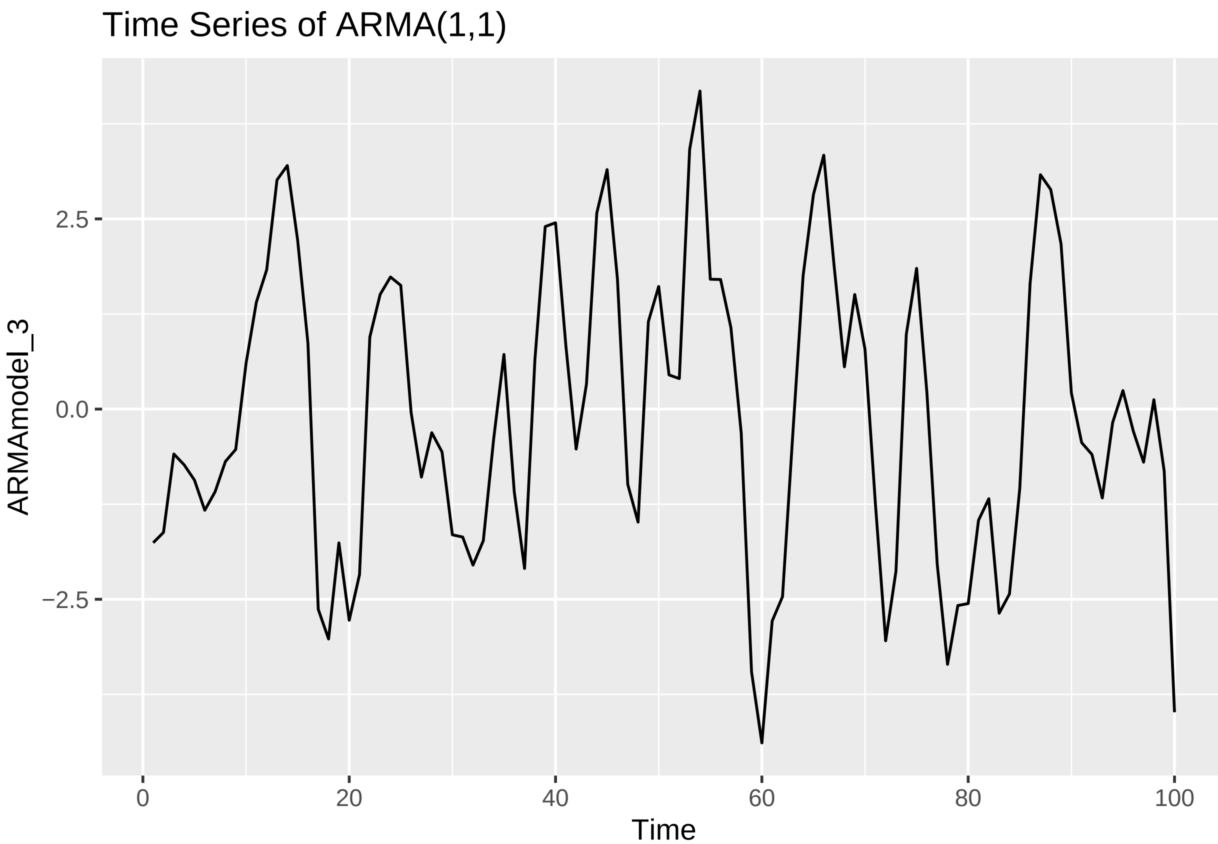
```r
#ARMA(1,0)
ARMAmodel_1<- arima.sim(model=list(ar=0.6), n=100) #the AR coefficient is 0.6
#ARMA(0,1)
ARMAmodel_2<- arima.sim(model=list(ma=0.9), n=100) #the MA coefficient is 0.9
#ARMA(1,1)
ARMAmodel_3<- arima.sim(model=list(ar=0.6,ma=0.9), n=100) #the AR coefficient is 0.6 and the MA coeffic

#plot the time series, optional
autoplot(ARMAmodel_1, main="Time Series of ARMA(1,0)")
```




```r
autoplot(ARMAmodel_2, main="Time Series of ARMA(0,1)")
```



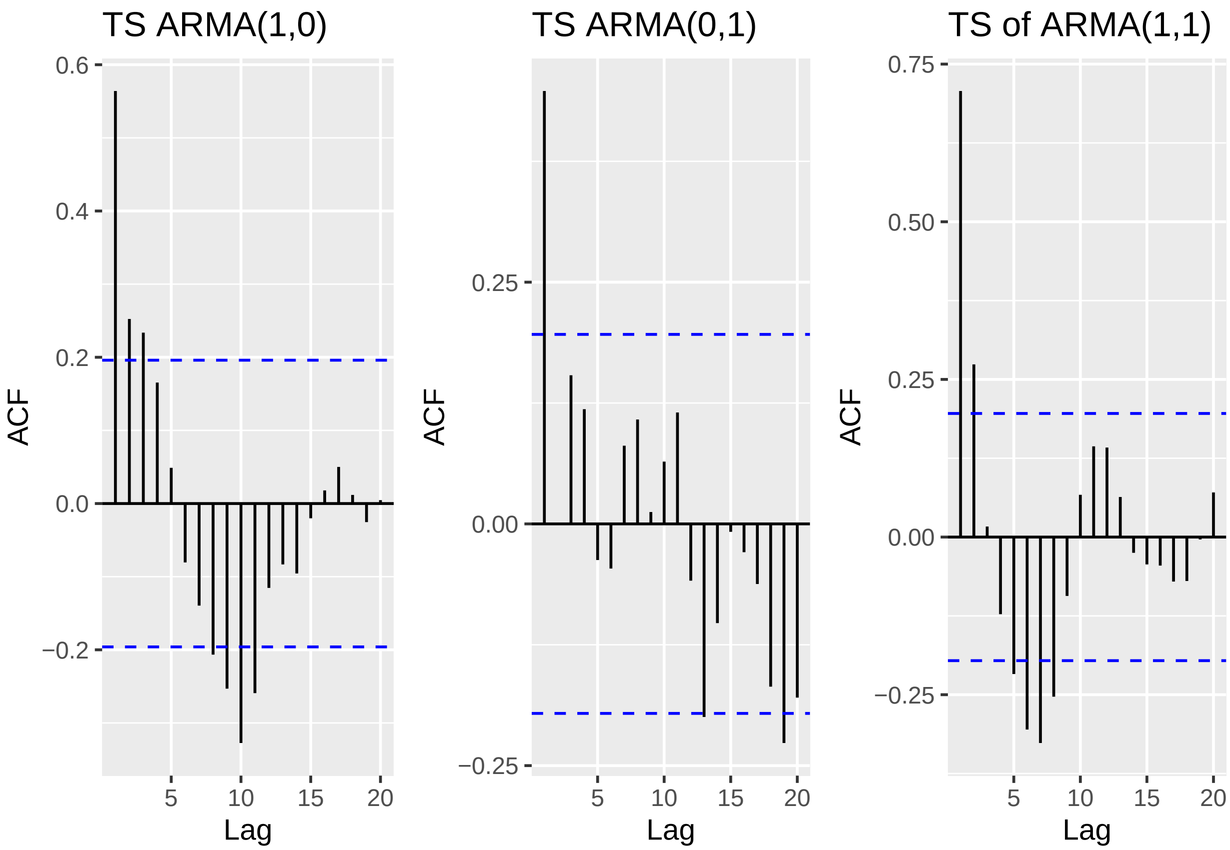

```r
autoplot(ARMAmodel_3, main="Time Series of ARMA(1,1)")
```



(b) Plot the sample ACF for each of these models in one window to facilitate comparison (Hint: use `cowplot::plot_grid()`).

```
plot_grid(
  autoplot(Acf(ARMAmodel_1, plot=F), main="TS ARMA(1,0)"),
  autoplot(Acf(ARMAmodel_2, plot=F), main="TS ARMA(0,1)"),
  autoplot(Acf(ARMAmodel_3, plot=F), main="TS of ARMA(1,1)"),
  nrow=1
)
```

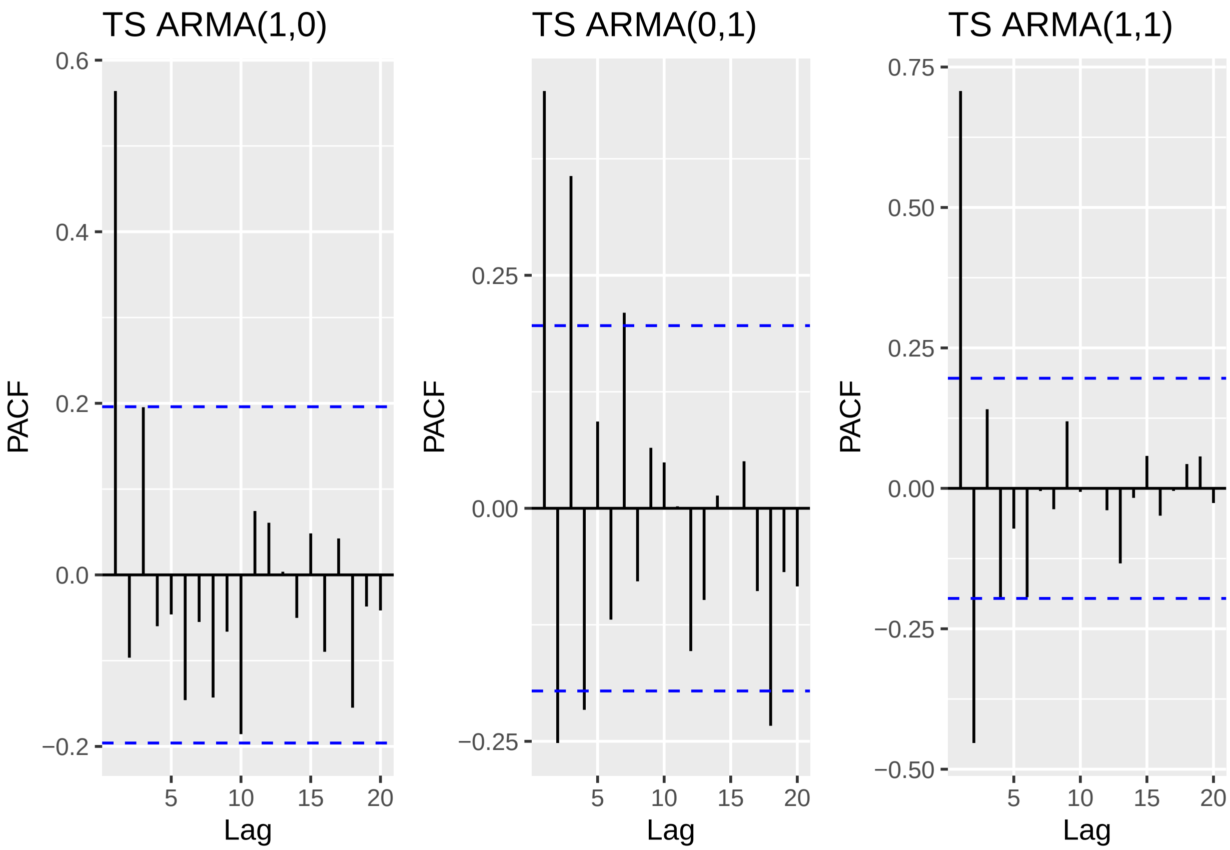
```
## Warning in ggplot2::geom_segment(lineend = "butt", ...): Ignoring unknown parameters: ‘main‘
## Ignoring unknown parameters: ‘main‘
## Ignoring unknown parameters: ‘main‘
```



(c) Plot the sample PACF for each of these models in one window to facilitate comparison.

```
plot_grid(
  autoplot(Pacf(ARMAmodel_1, plot=F), main="TS ARMA(1,0)"),
  autoplot(Pacf(ARMAmodel_2, plot=F), main="TS ARMA(0,1)"),
  autoplot(Pacf(ARMAmodel_3, plot=F), main="TS ARMA(1,1)"),
  nrow=1
)
```

```
## Warning in ggplot2::geom_segment(lineend = "butt", ...): Ignoring unknown parameters: ‘main‘
## Ignoring unknown parameters: ‘main‘
## Ignoring unknown parameters: ‘main‘
```



(d) Look at the ACFs and PACFs. Imagine you had these plots for a data set and you were asked to identify the model, i.e., is it AR, MA or ARMA and the order of each component. Would you be able identify them correctly? Explain your answer.

Answer: For model 1 you can tell it is an autoregressive model with p=1 because of a slow decay in the ACF plot and a clear cut off at lag 1 in the PACF plot (p=1). For model 2, you can tell it's a moving average model because the PACF plot have a slow decay and the ACF plot has a cut off at lag 1 (q=1). For model 3, it's hard to tell because we are superimposing AR and MA properties.

(e) Compare the PACF values R computed with the values you provided for the lag 1 correlation coefficient, i.e., does $\phi = 0.6$ match what you see on PACF for ARMA(1,0), and ARMA(1,1)? Should they match?

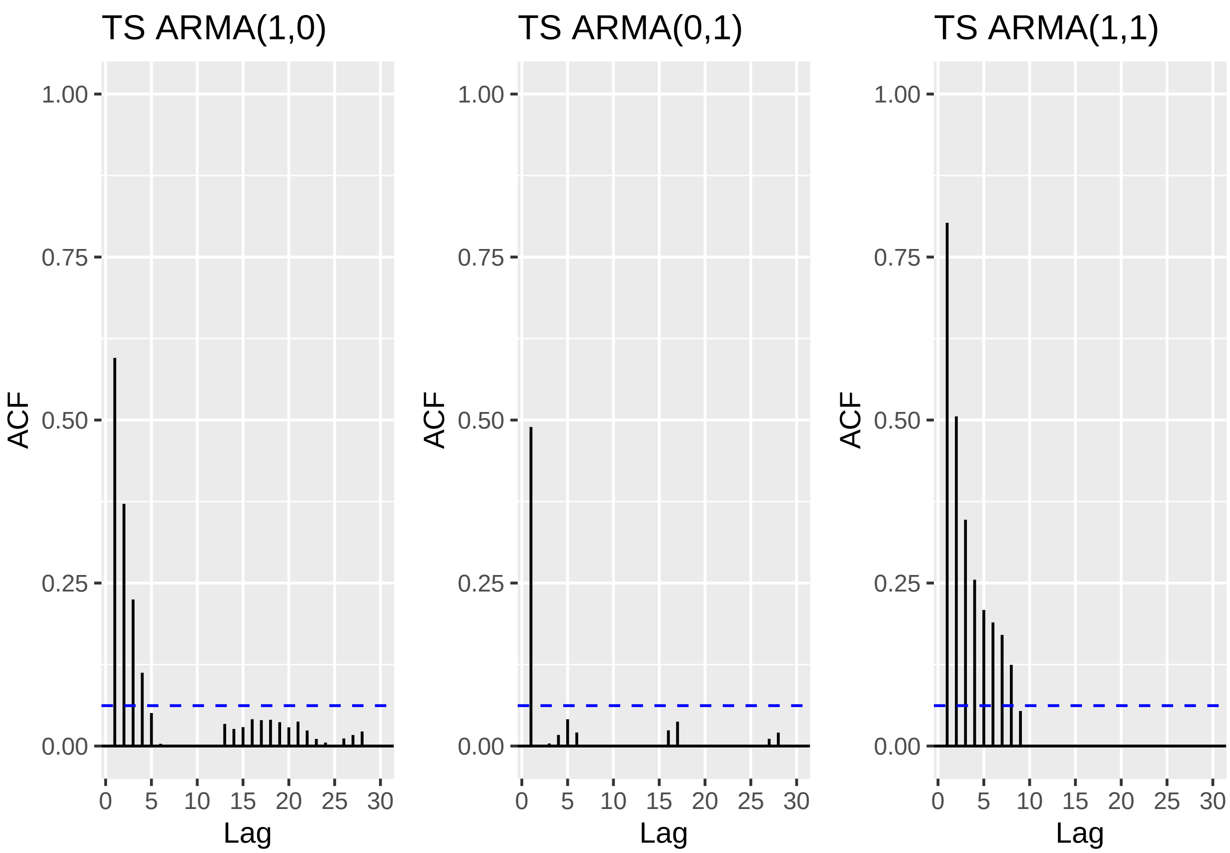
Answer: For model 1 and 3, since the order of AR component is 1, the value of PACF at lag 1 is close to the value we specify. For model 2 with only MA process, you can't read the coefficient value on ACF or PACF.

(f) Increase number of observations to $n = 1000$ and repeat parts (b)-(e).

```
#ARMA(1,0)
ARMAmodel_1_f<- arima.sim(model=list(ar=0.6), n=1000) #the AR coefficient is 0.6
#ARMA(0,1)
ARMAmodel_2_f<- arima.sim(model=list(ma=0.9), n=1000) #the MA coefficient is 0.9
#ARMA(1,1)
ARMAmodel_3_f<- arima.sim(model=list(ar=0.6,ma=0.9), n=1000) #the AR coefficient is 0.6 and the MA coef
```

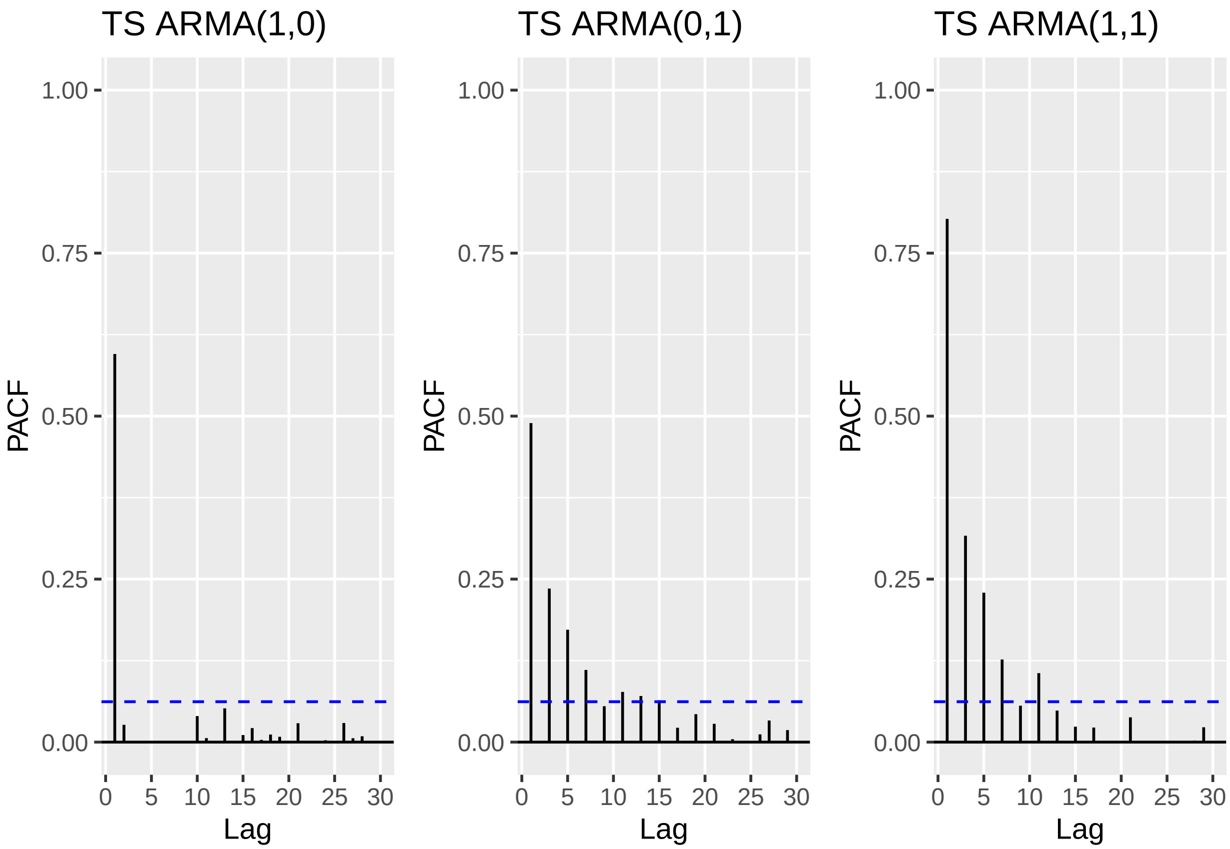
```r
#ACF plots
plot_grid(
  autoplot(Acf(ARMAmodel_1_f, plot=F), main="TS ARMA(1,0)",ylim=c(0,1)),
  autoplot(Acf(ARMAmodel_2_f, plot=F), main="TS ARMA(0,1)",ylim=c(0,1)),
  autoplot(Acf(ARMAmodel_3_f, plot=F), main="TS ARMA(1,1)",ylim=c(0,1)),
  nrow=1
)
```

```
## Warning in ggplot2::geom_segment(lineend = "butt", ...): Ignoring unknown parameters: ‘main‘ and ‘yli
## Ignoring unknown parameters: ‘main‘ and ‘ylim‘
## Ignoring unknown parameters: ‘main‘ and ‘ylim‘

## Warning: Removed 8 rows containing missing values (‘geom_segment()‘).

## Warning: Removed 1 rows containing missing values (‘geom_hline()‘).

## Warning: Removed 20 rows containing missing values (‘geom_segment()‘).

## Warning: Removed 1 rows containing missing values (‘geom_hline()‘).

## Warning: Removed 21 rows containing missing values (‘geom_segment()‘).

## Warning: Removed 1 rows containing missing values (‘geom_hline()‘).
```



```r
#PACF plots

plot_grid(
  autoplot(Pacf(ARMAmodel_1_f, plot=F), main="TS ARMA(1,0)",ylim=c(0,1)),
  autoplot(Pacf(ARMAmodel_2_f, plot=F), main="TS ARMA(0,1)",ylim=c(0,1)),
  autoplot(Pacf(ARMAmodel_3_f, plot=F), main="TS ARMA(1,1)",ylim=c(0,1)),
  nrow=1
)
```

```
## Warning in ggplot2::geom_segment(lineend = "butt", ...): Ignoring unknown
## parameters: ‘main‘ and ‘ylim‘

## Warning in ggplot2::geom_segment(lineend = "butt", ...): Ignoring unknown parameters: ‘main‘ and ‘yli
## Ignoring unknown parameters: ‘main‘ and ‘ylim‘

## Warning: Removed 14 rows containing missing values (‘geom_segment()‘).

## Warning: Removed 1 rows containing missing values (‘geom_hline()‘).

## Warning: Removed 15 rows containing missing values (‘geom_segment()‘).

## Warning: Removed 1 rows containing missing values (‘geom_hline()‘).

## Warning: Removed 16 rows containing missing values (‘geom_segment()‘).

## Warning: Removed 1 rows containing missing values (‘geom_hline()‘).
```



Answer: Now for the ARMA(1,0) we were able to match the number since we are generating more observations. But for the ARMA(1,1) the coefficient is higher than 0.6.

Q3

Consider the ARIMA model $y_t = 0.7 * y_{t-1} - 0.25 * y_{t-12} + a_t - 0.1 * a_{t-1}$

(a) Identify the model using the notation ARIMA$(p, d, q)(P, D, Q)_s$, i.e., identify the integers p, d, q, P, D, Q, s (if possible) from the equation.

Answer: From the model equation, y_t depends on two previous observations, y_{t-1} and y_{t-12}, since 12 is our seasonal lag the order of the AR component will be p=1 and SAR component will be P=1. The value of y_t also depends on the previous residual a_{t-1} meaning we have a MA component of order q=1. There is no a_{t-12} so SMA component order will be Q=0.

With respect to differencing, i.e., d and D, it's hard to tell from the equation but there are two things you could argue. The first is there is no constant term, therefore we are either working with a zero-mean process or this series has already been differenced, leading to the conclusion that maybe d=1 or D=1 or both d=1 and D=1. No points will be deducted if you did not say anything about d or D.

Thus, this equation represents ARIMA$(1, d, 1)(1, D, 0)_{12}$.

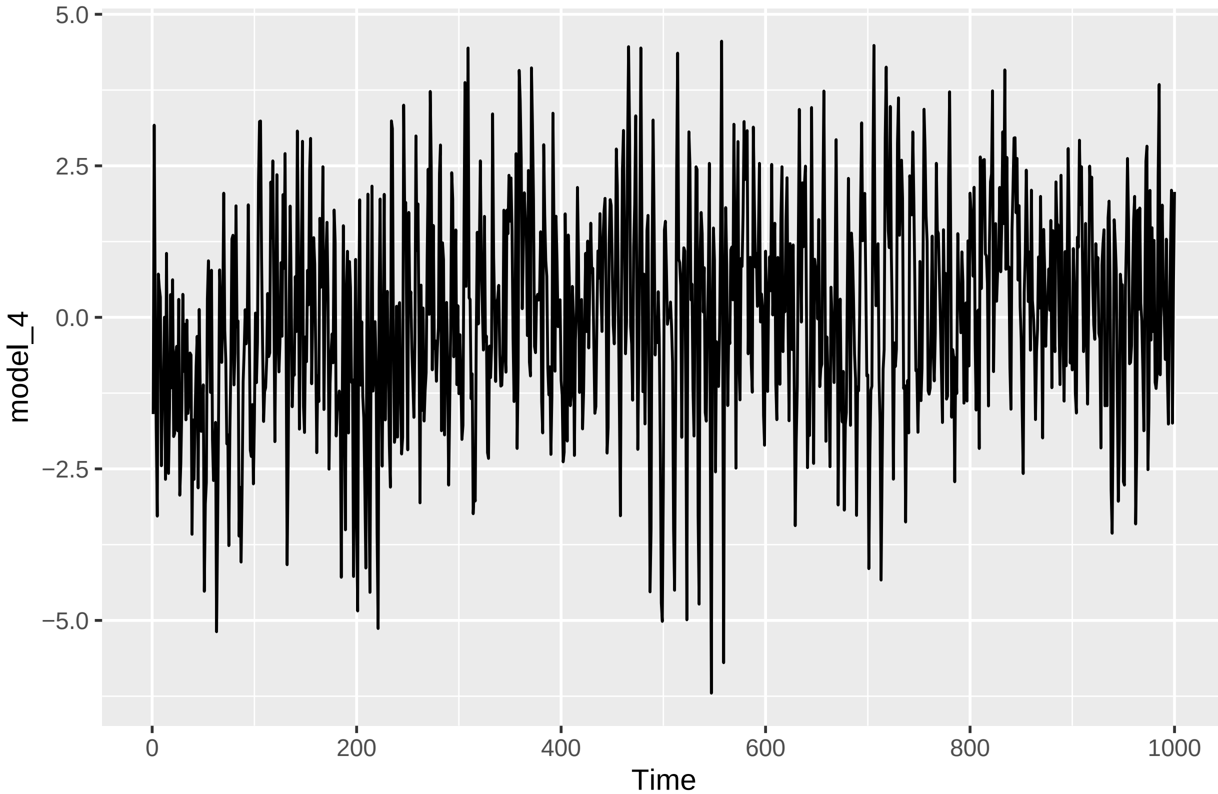
(b) Also from the equation what are the values of the parameters, i.e., model coefficients.

Answer: $\phi_1 = 0.7$, $\phi_12 = -0.25$, $\theta = 0.1$. No points will be deducted if you did not get the sign right.

Q4

Simulate a seasonal ARIMA$(0, 1)(1, 0)_{12}$ model with $\phi = 0.8$ and $\theta = 0.5$ using the `sim_sarima()` function from package `sarima`. The 12 after the bracket tells you that $s = 12$, i.e., the seasonal lag is 12, suggesting monthly data whose behavior is repeated every 12 months. You can generate as many observations as you like. Note the Integrated part was omitted. It means the series do not need differencing, therefore $d = D = 0$. Plot the generated series using `autoplot()`. You will need to convert the series into a `ts()` first. Does it look seasonal?

```
model_4 <- sim_sarima(model=list(ma=0.5,sar=0.8, nseasons=12), n=1000)
#moving average coefficient for nonseasonal component is 0.5; the AR coefficient for seasonal component
#could also use sarima.sim() from package "astsa"
model_4 <- ts(model_4)
autoplot(model_4)
```

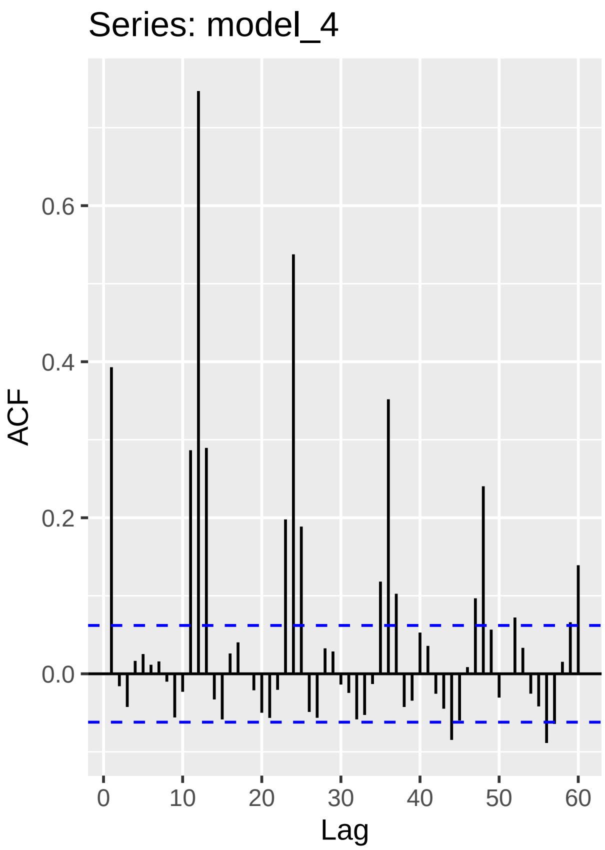


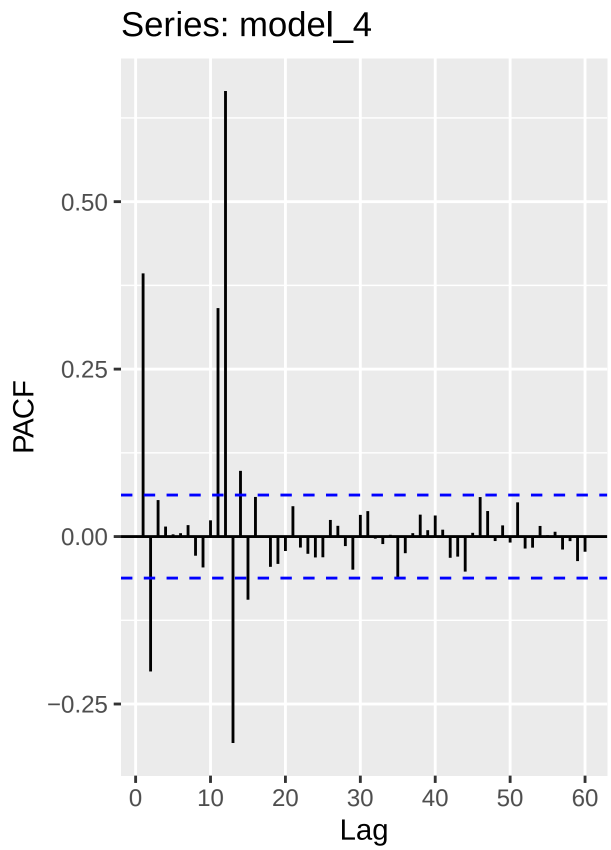
Answer: Yes, the series looks seasonal.

Q5

Plot ACF and PACF of the simulated series in Q4. Comment if the plots are well representing the model you simulated, i.e., would you be able to identify the order of both non-seasonal and seasonal components from the plots? Explain.

```
#ACF and PACF
plot_grid(
  autoplot(Acf(model_4, plot=F,lag.max=60)),
  autoplot(Pacf(model_4, plot=F,lag.max=60)),
  nrow=1
)
```






```r
#coefficients
coeffi_model_4=Arima(model_4, order=c(0,0,1), seasonal=list(order=c(1,0,0), period=12))
print(coeffi_model_4)
```

```
## Series: model_4
## ARIMA(0,0,1)(1,0,0)[12] with non-zero mean
##
## Coefficients:
##          ma1    sar1    mean
##       0.4987  0.7627  0.0279
## s.e.  0.0266  0.0203  0.1987
##
## sigma^2 = 1.071:  log likelihood = -1456.94
## AIC=2921.87   AICc=2921.91   BIC=2941.5
```

Answer: For the order of the non-seasonal component, we focus on the early lags (ex.before lag=12). We can see that PACF does not show a clear cut off, we could argue we have a slow decay and ACF has a cut off at lag 1, note that correlation goes to zero at lag 2, which indicates that the non-seasonal component has moving average process, q=1.

As for the seasonal component, we are only intrested at seasonal lags 12, 24, 36 and so forth. We can see that the ACF has multiple spikes at the seasonal lags and PACF has only one spike at lag 12. It shows that we should include a SAR process, P=1.

The plot result is consistent with the parameters specified in the simulation. Note that the PACF at lag 12 is 0.7, which is close to the AR coefficient $phi_1 2 = 0.8$.