ENV 797 - Time Series Analysis for Energy and Environment Applications | Spring 2025

Assignment 6 - Due date 02/27/25

Jessalyn Chuang

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_Sp25.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(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(ggplot2)
library(forecast)
```

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

```
library(Kendall)
library(tseries)
library(outliers)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr   1.1.4     v stringr 1.5.1
## v forcats 1.0.0     v tibble  3.2.1
## v purrr   1.0.2     v tidyr   1.3.1
## v readr   2.1.5

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(cowplot)
```

```
##
## Attaching package: 'cowplot'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
```

```r
#install.packages('sarima')
library(sarima)
```

```
## Loading required package: stats4
##
## Attaching package: 'sarima'
##
## The following object is masked from 'package:stats':
##
##     spectrum
```

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: For AR models, we are looking for the ACF to decay exponentially with time, which reflects the persistence of autocorrelations within the data over time. The PACF will identify the order of the AR model. In this case, this is a second order model where lag 2 must be included in the model. This is the number of lags that have a direct effect on the current observations. As a result, the current value of this time series is modeled as a linear combination of the two immediately preceding values.

- MA(1)

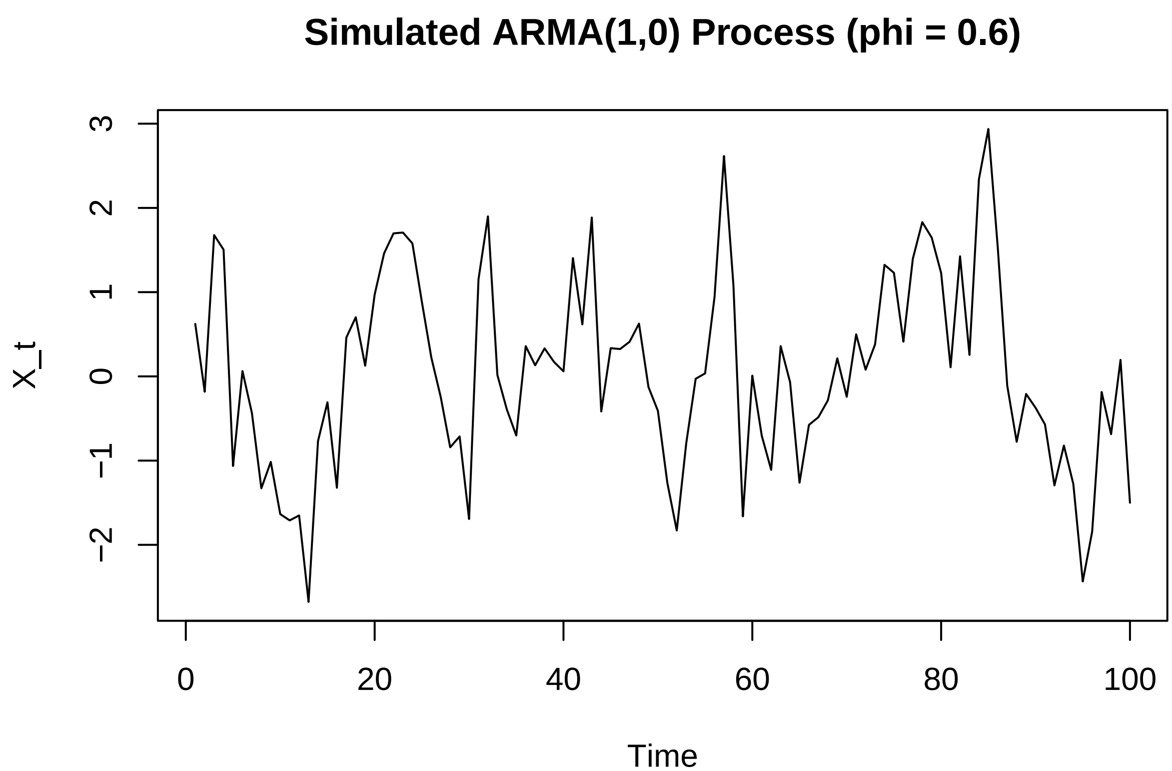
  Answer: In MA models, the order is determined by the ACF. For an MA(1) model, the ACF would show a significant spike at lag 1 and then cuts off abruptly for higher lags. This is because only a limited number of past error terms directly affect the current observations. However, the PACF for an MA(1) model tends to decay gradually because the effects of past error terms tend to be spread out over several lags.

Q2

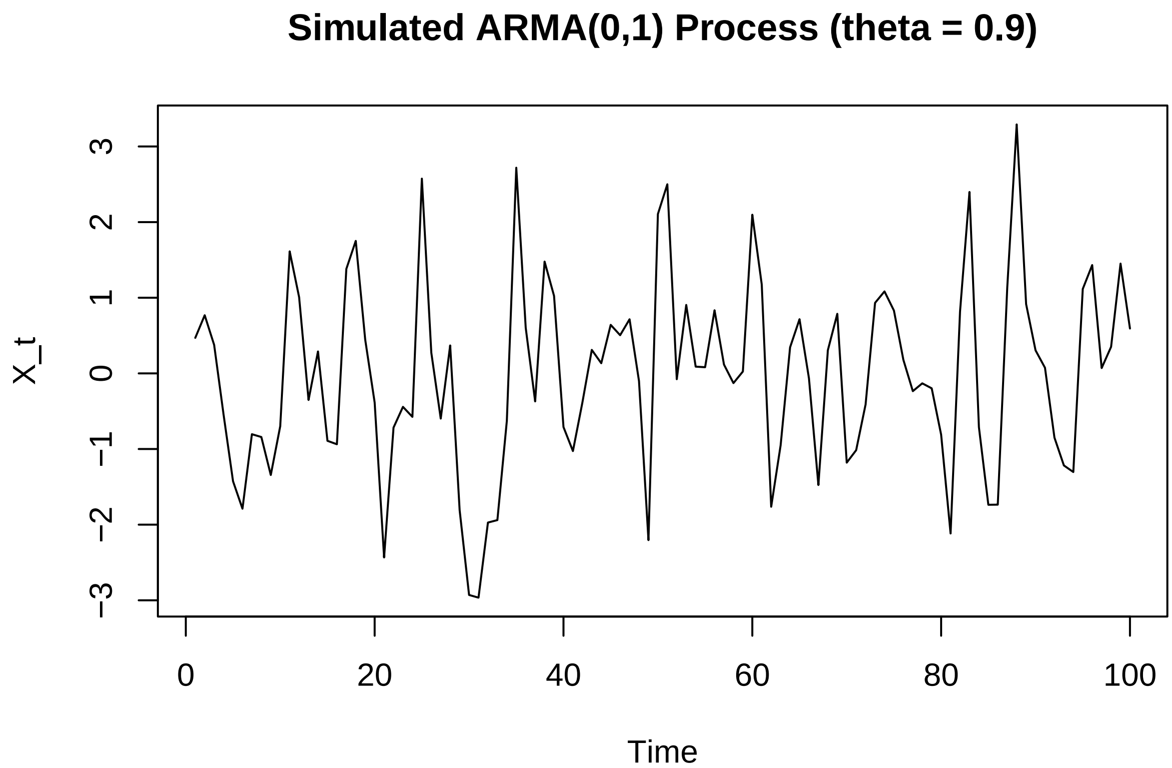
Recall that the non-seasonal ARIMA is described by three parameters ARIMA(p, d, q) where p is the order of the autoregressive component, d is the number of times the series need to be differenced to obtain stationarity and q is the order of the moving average component. If we don't need to difference 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.

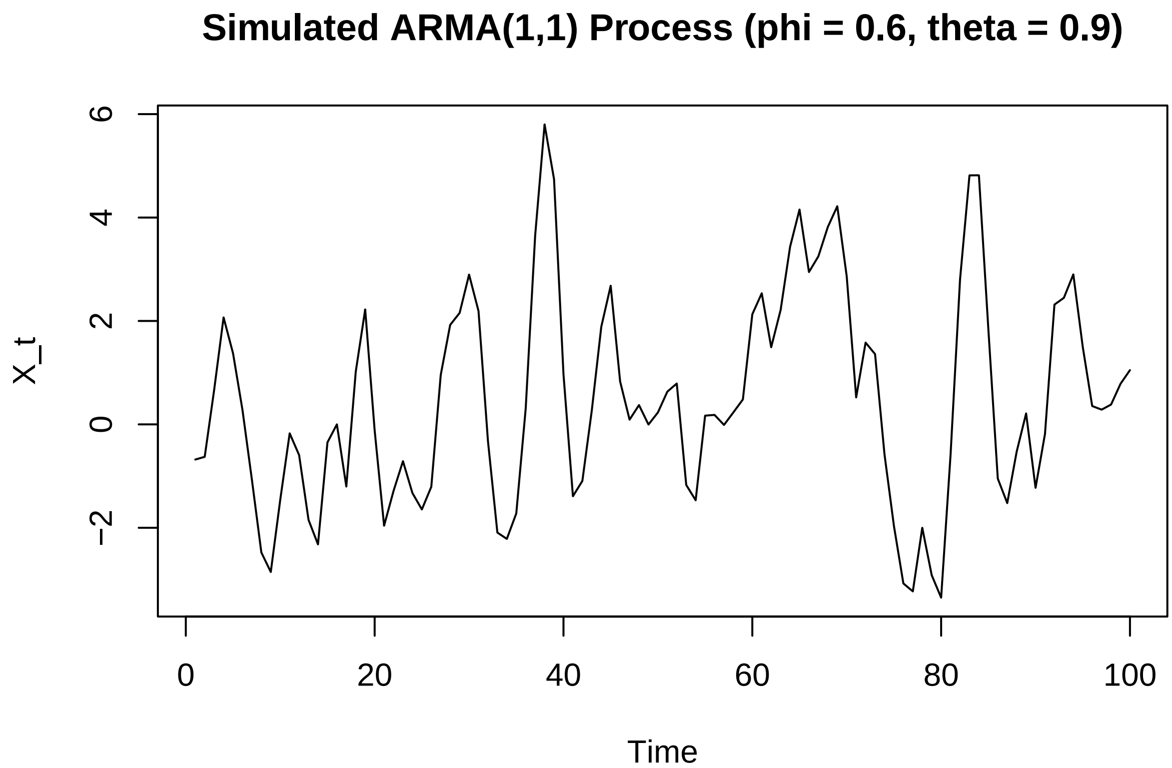
```r
set.seed(123)
n <- 100

arma10_sim <- arima.sim(n = n, list(ar = 0.6), sd = 1)
# Plot the ARMA(1,0) series
ts.plot(arma10_sim, main = "Simulated ARMA(1,0) Process (phi = 0.6)",
        ylab = "X_t", xlab = "Time")
```




```r
arma01_sim <- arima.sim(n = n, list(ma = 0.9), sd = 1)
#Plot the ARMA(0,1) series
ts.plot(arma01_sim, main = "Simulated ARMA(0,1) Process (theta = 0.9)",
        ylab = "X_t", xlab = "Time")
```

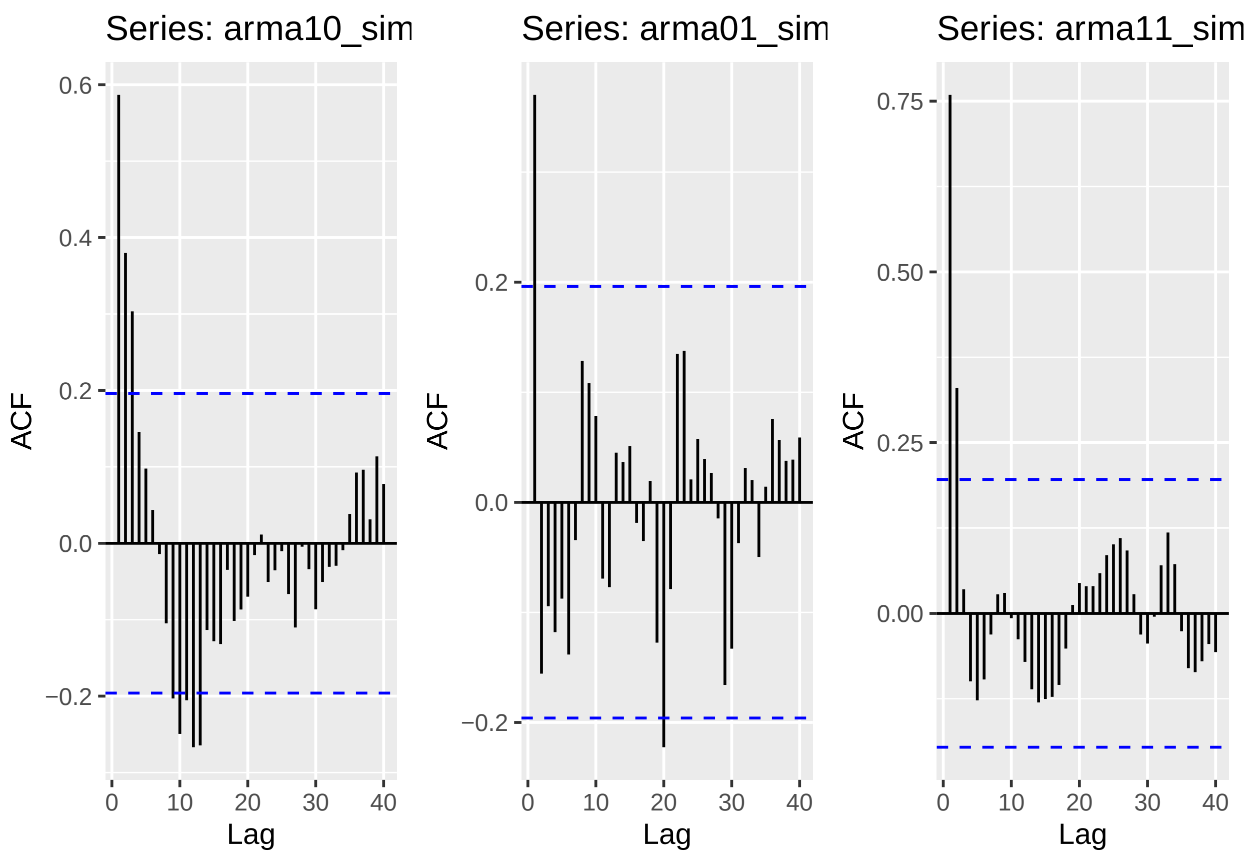



```r
arma11_sim <- arima.sim(n = n, list(ar =0.6, ma = 0.9), sd = 1)
# Plot the ARMA(1,1) series
ts.plot(arma11_sim, main = "Simulated ARMA(1,1) Process (phi = 0.6, theta = 0.9)",
        ylab = "X_t", xlab = "Time")
```



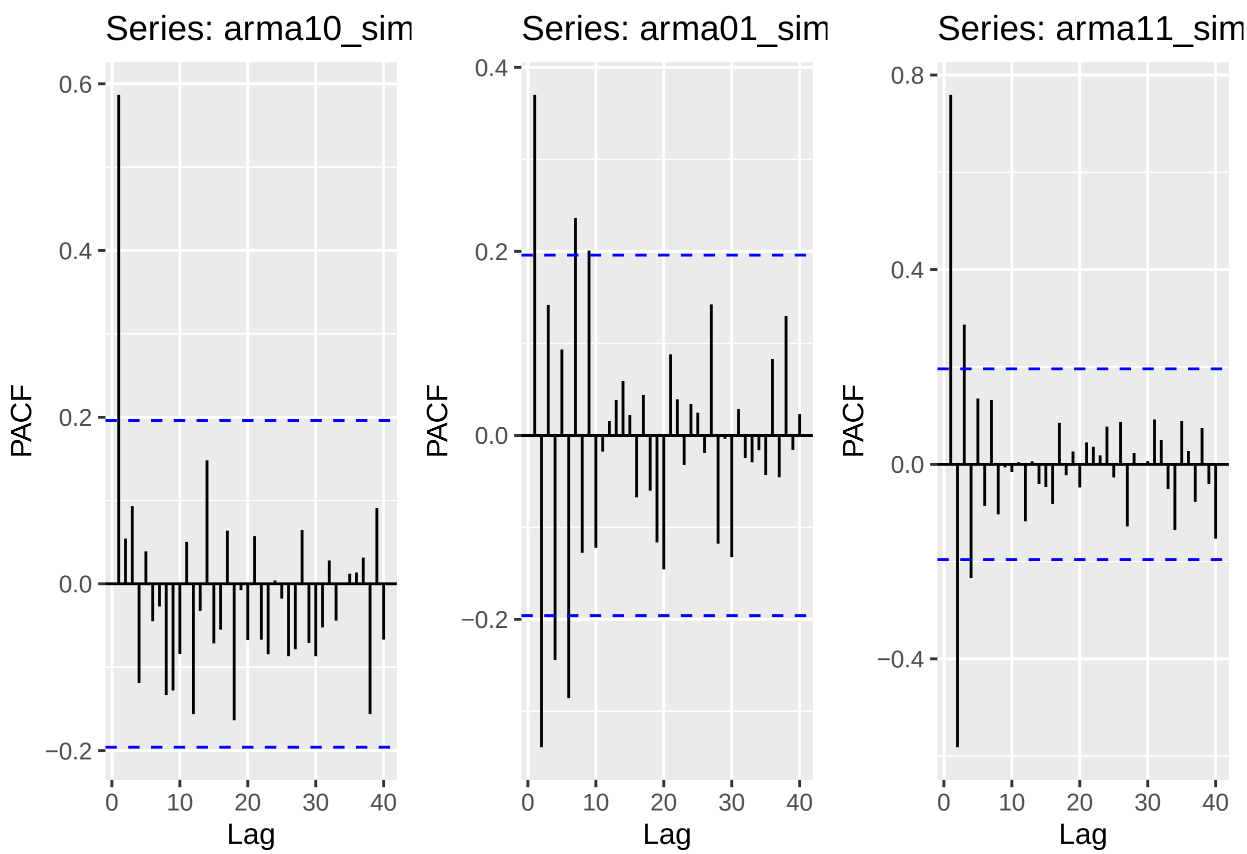
(b) Plot the sample ACF for each of these models in one window to facilitate comparison (Hint: use `cowplot::plot_grid()`).

```
plot_grid(
  autoplot(Acf(arma10_sim,lag.max=40, plot = FALSE)),
  autoplot(Acf(arma01_sim,lag.max=40, plot = FALSE)),
  autoplot(Acf(arma11_sim,lag.max=40, plot = FALSE)), nrow = 1
)
```



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

```
plot_grid(
  autoplot(Pacf(arma10_sim,lag.max=40, plot = FALSE)),
  autoplot(Pacf(arma01_sim,lag.max=40, plot = FALSE)),
  autoplot(Pacf(arma11_sim,lag.max=40, plot = FALSE)), nrow = 1
)
```



(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 the ARMA(1,0) plot, the ACF gradually decreases with significant spikes at initial lags while the PACF cuts off after lag 1, which are both characteristic of AR models. For the ARMA(0,1) plot, the ACF cuts off after lag 1 while the PACF has a gradual decay, both which are expected of an MA model. Lastly, for the ARMA(1,1) model, both the ACF and PACF has more gradual changes, making it look like a mix of AR and MA models.

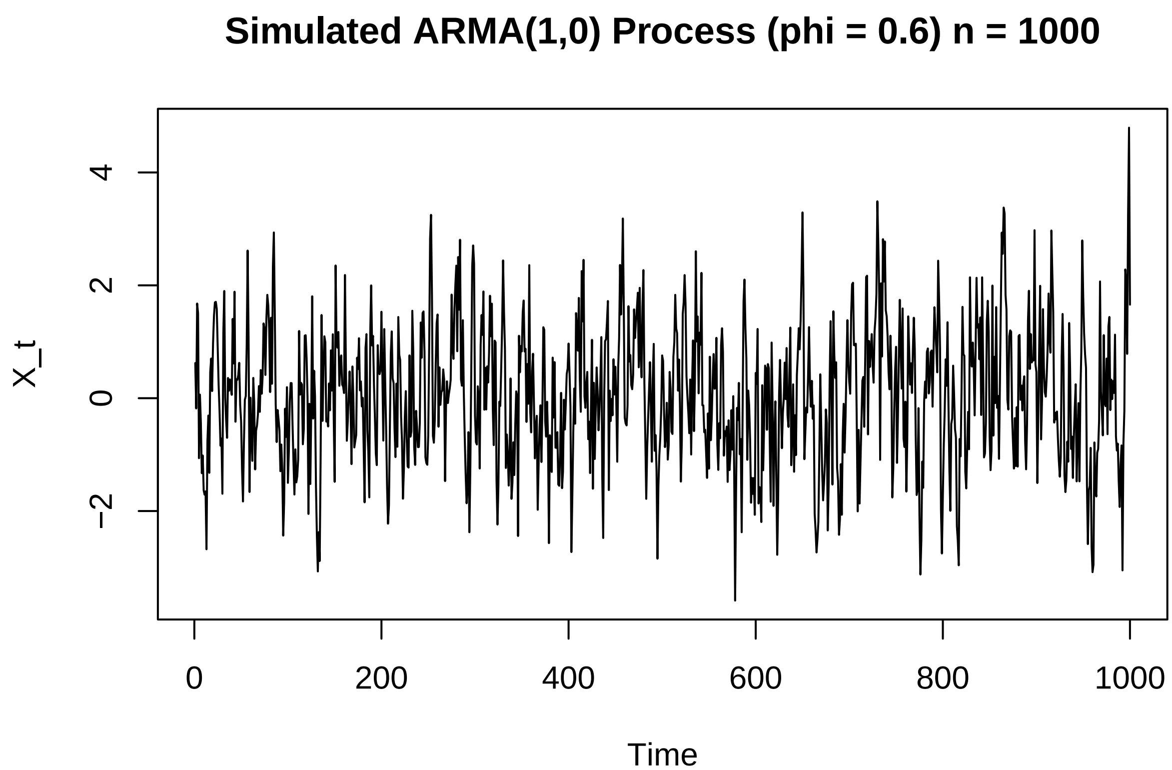
(e) Compare the PACF values R computed with the values you provided for the lag 1 correlation coefficient, i.e., does $\phi = 0.6$ match what you see on PACF for ARMA(1,0), and ARMA(1,1)? Should they match?

Answer: "It matches for ARMA(1,0), as expected, since in the PACF, lag 1 is the most significant and represents the autoregressive term that should be included in the model equation. For ARMA(1,1), however, the PACF at lag 1 does not have to match exactly because the moving average component also influences the correlation structure, causing a more gradual decay rather than a sharp cutoff. The presence of the MA(1) term means that past errors contribute to the process, altering the way the PACF behaves across lags.

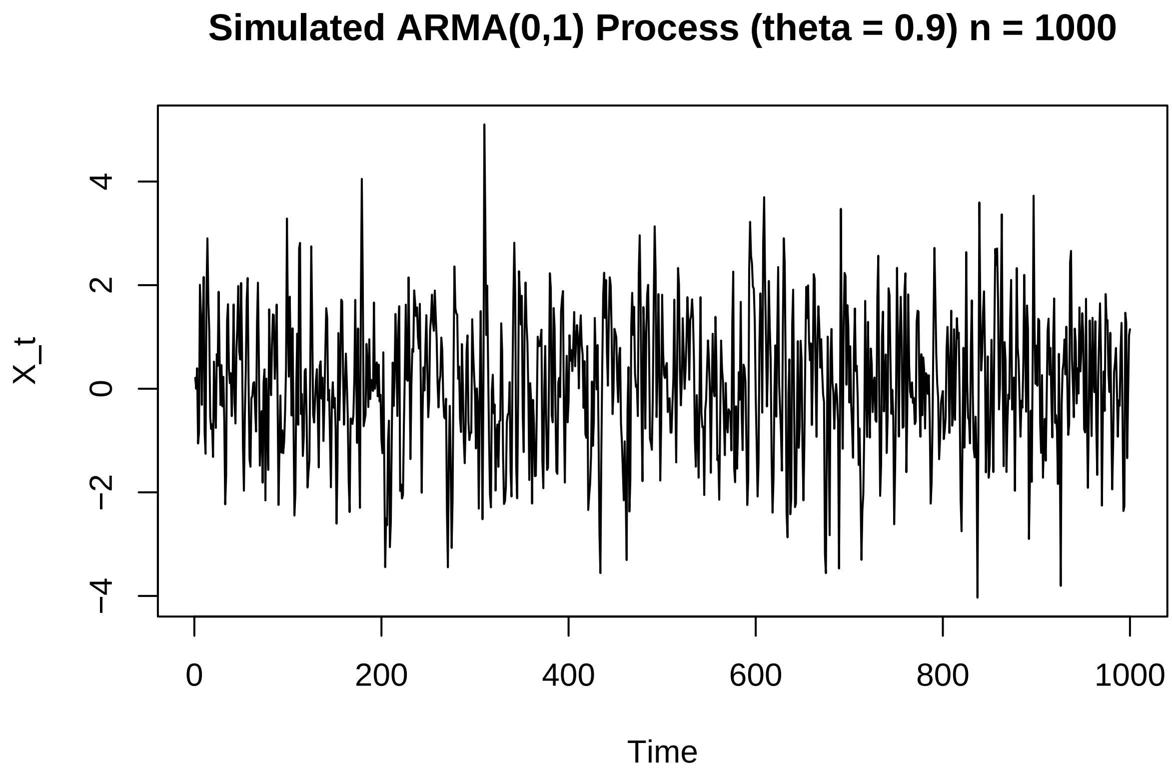
(f) Increase number of observations to $n = 1000$ and repeat parts (b)-(e).

```
set.seed(123)
n <- 1000
```

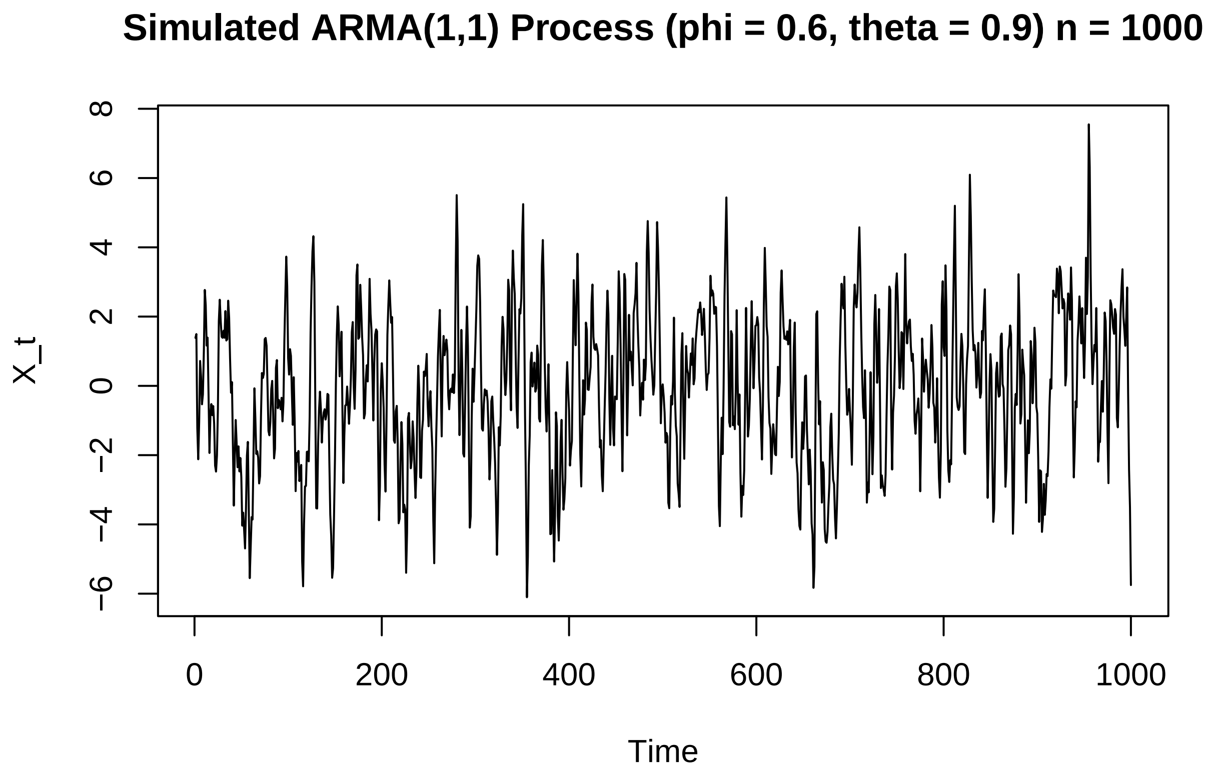
```
arma10_sim_2 <- arima.sim(n = n, list(ar = 0.6), sd = 1)
# Plot the ARMA(1,0) series
ts.plot(arma10_sim_2, main = "Simulated ARMA(1,0) Process (phi = 0.6) n = 1000",
        ylab = "X_t", xlab = "Time")
```



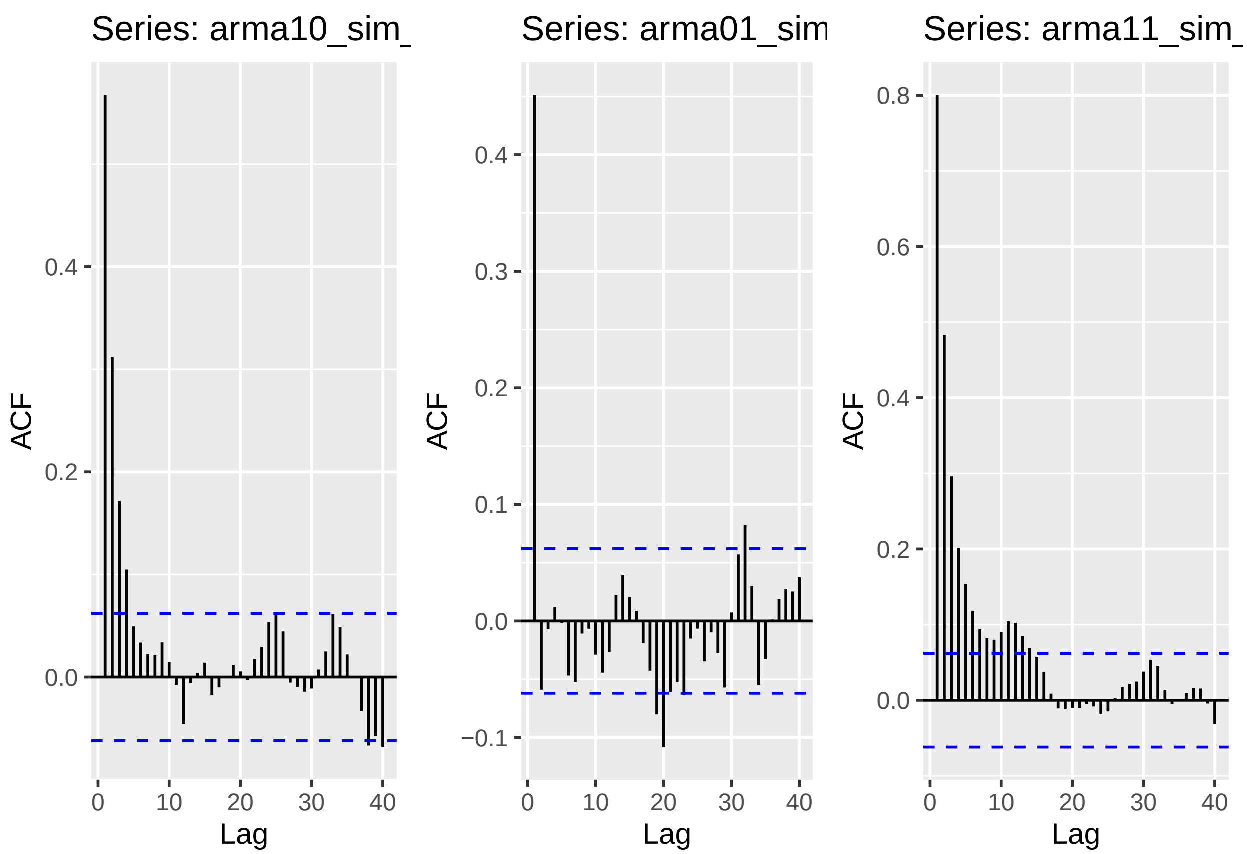

```
arma01_sim_2 <- arima.sim(n = n, list(ma = 0.9), sd = 1)
#Plot the ARMA(0,1) series
ts.plot(arma01_sim_2, main = "Simulated ARMA(0,1) Process (theta = 0.9) n = 1000",
        ylab = "X_t", xlab = "Time")
```



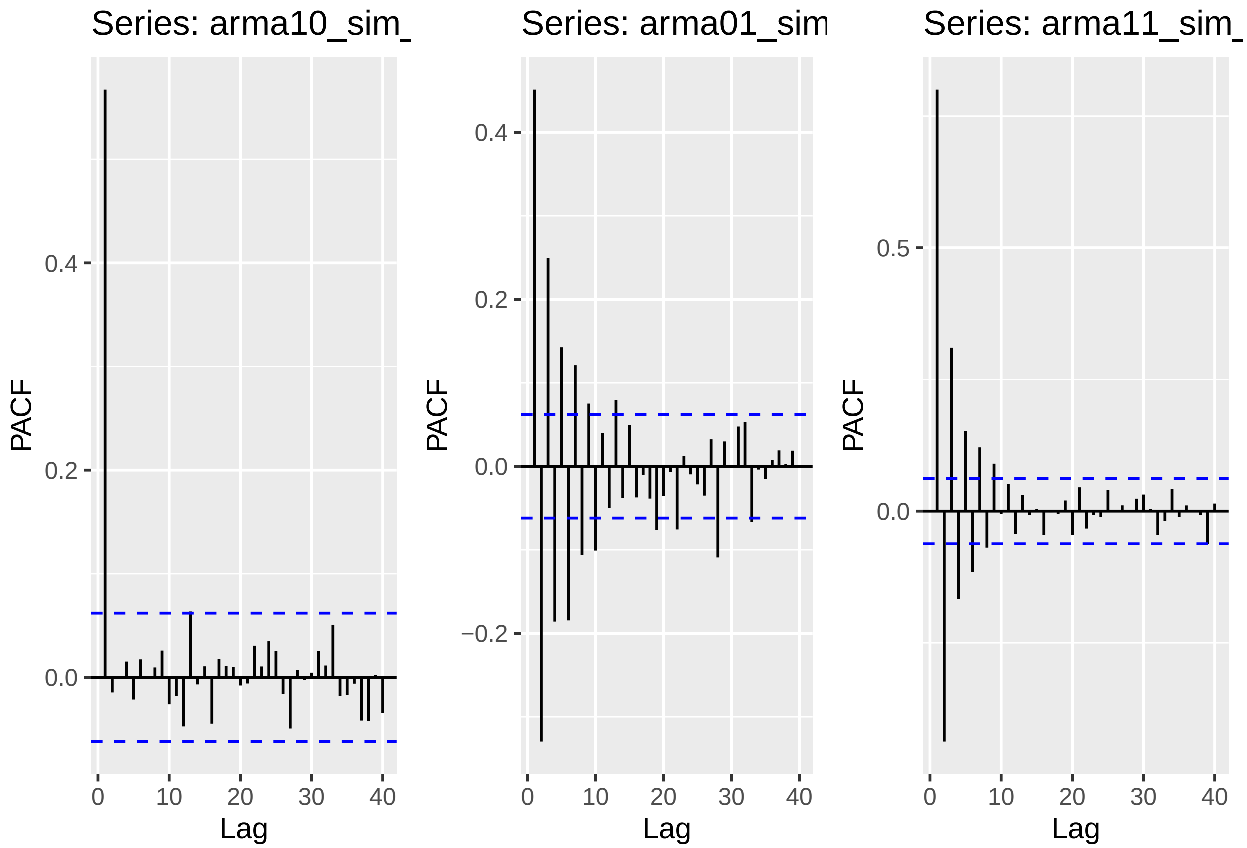

```
arma11_sim_2 <- arima.sim(n = n, list(ar =0.6, ma = 0.9), sd = 1)
# Plot the ARMA(1,1) series
ts.plot(arma11_sim_2, main = "Simulated ARMA(1,1) Process (phi = 0.6, theta = 0.9) n = 1000",
        ylab = "X_t", xlab = "Time")
```




```
plot_grid(
  autoplot(Acf(arma10_sim_2,lag.max=40, plot = FALSE)),
  autoplot(Acf(arma01_sim_2,lag.max=40, plot = FALSE)),
  autoplot(Acf(arma11_sim_2,lag.max=40, plot = FALSE)), nrow = 1
)
```




```r
plot_grid(
  autoplot(Pacf(arma10_sim_2,lag.max=40, plot = FALSE)),
  autoplot(Pacf(arma01_sim_2,lag.max=40, plot = FALSE)),
  autoplot(Pacf(arma11_sim_2,lag.max=40, plot = FALSE)), nrow = 1
)
```



Identification based off of the ACFs and PACF: For the first pair of charts (arma10_sim_2) there is a gradual decay in the ACF and the PACF sees a sharp cutoff at lag, both of which are characteristic of an AR(1) process. For the second pair of charts (arma01_sim_2) the ACF sees a sharp cutoff at lag 1 while the PACF has a gradual decay, which is in line with an MA(1) process. Lastly for the third pair of charts (arma11_sim_2), both the ACF and PACF have gradual decays, signaling a mix of AR and MA processes, suggesting that this is an ARMA(1,1) process.

Comparing the PACF Values with $\phi = 0.6$, the AR(1) model with this parameter at lag1 should be equal to 0.6. however, for an ARMA(1,1) model, the PACF at lag 1 is not exactly 0.6 because of the additional MA(1) component. These are both observed in the PACF charts for ARMA(1,0) and ARMA(1,1).

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.

p = 1 d = 0 q = 1 P = 1 D = 0 Q = 0 s = 12

ARIMA$(p, d, q)(P, D, Q)_s$ = ARIMA$(1, 0, 1)(1, 0, 0)_12$

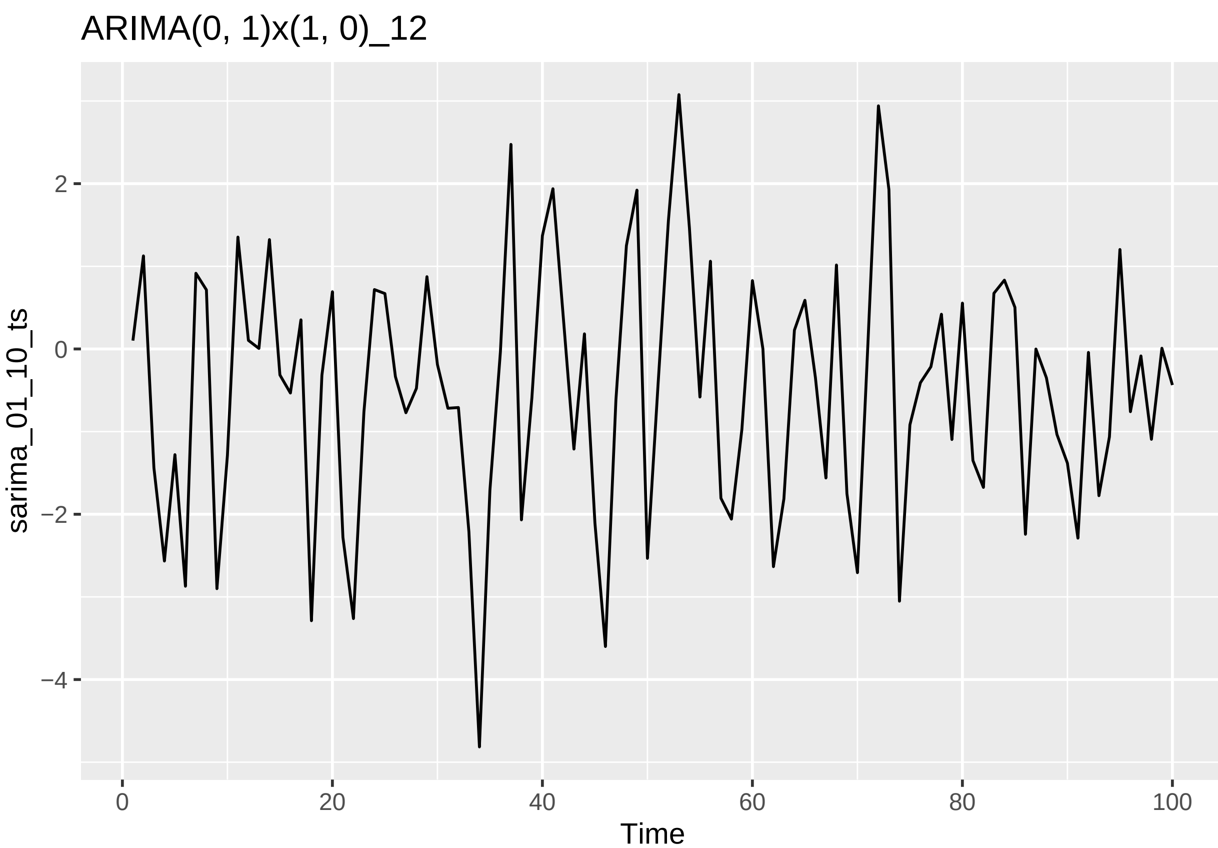
(b) Also from the equation what are the values of the parameters, i.e., model coefficients.

Non-seasonal AR model coefficient = 0.7 Seasonal AR model coefficient = -0.25 MA model coefficient = 0.10

Q4

Simulate a seasonal ARIMA$(0,1)\times(1,0)_{12}$ 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()`. Does it look seasonal?

```
sarima_01_10 <- sim_sarima(n = 100, model = list(ma = 0.5, ar = c(rep(0, 11), 0.8)))
sarima_01_10_ts <- ts(sarima_01_10)
autoplot(sarima_01_10_ts) +
  ggtitle("ARIMA(0, 1)x(1, 0)_12")
```

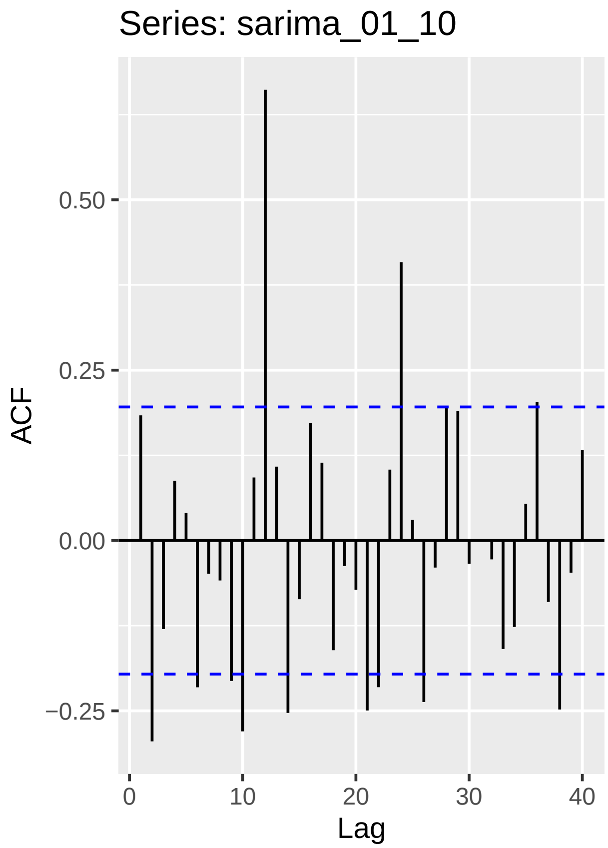


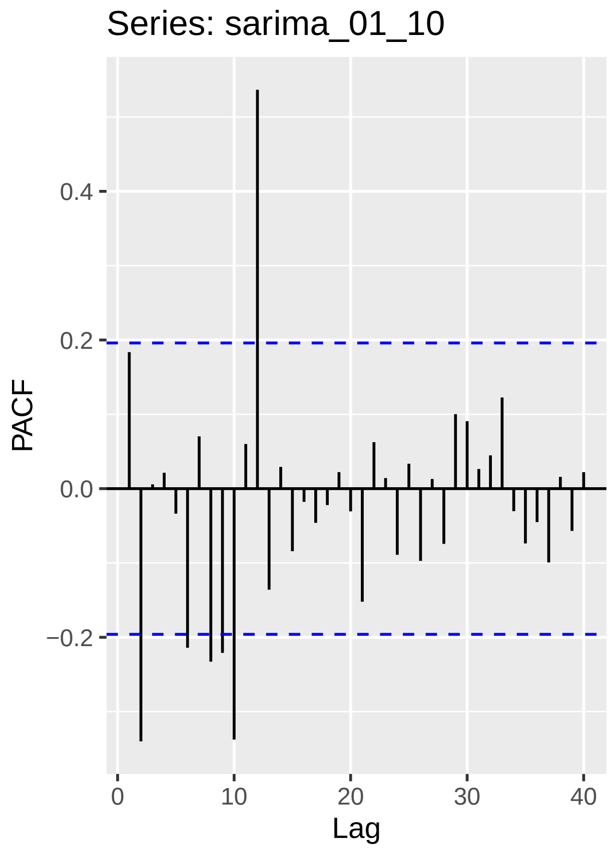
Yes the data looks seasonal, evident by the cyclical nature of the time series. There are peaks and troughs happening at regular intervals.

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.

```
plot_grid(
  autoplot(Acf(sarima_01_10,lag.max=40, plot = FALSE)),
  autoplot(Pacf(sarima_01_10,lag.max=40, plot = FALSE)), nrow = 1
)
```





The plots do not perfectly represent the model I simulated. While the strong spike at lag 12 in the ACF suggests a seasonal pattern, and the PACF spike at lag 12 supports the presence of the seasonal AR(1) term, some aspects of the plots are less clear. The ACF does not exhibit a strong spike at lag 1, which would typically be expected for an MA(1) process, and its pattern does not clearly resemble the usual behavior of either an MA or AR process. Additionally, while the PACF shows some gradual tailing off, which is characteristic of an MA process, it does not perfectly align with typical characteristics. These discrepancies suggest that while the seasonal structure is somewhat visible, the non-seasonal components may not be as clearly identifiable from these plots alone. Factors such as random variation or the possible interactions between seasonal and non-seasonal terms may be affecting the clarity of the ACF and PACF patterns.