ENV 790.30 - Time Series Analysis for Energy Data | Spring 2023

Assignment 6 - Due date 03/06/23

Hugh Cipparone

Directions

You should open the .rmd file corresponding to this assignment on RStudio. The file is available on our class repository on Github. And to do so you will need to fork our repository and link it to your RStudio.

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_Sp23.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: "xlsx" or "readxl", "ggplot2", "forecast","tseries", and "Kendall". 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.

Questions

This assignment has general questions about ARIMA Models.

Packages needed for this assignment: "forecast","tseries". Do not forget to load them before running your script, since they are NOT default packages.\

```
#Load/install required package here

library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.5.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(dplyr)
library(forecast)
```

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

```r
library(tseries)
library(ggplot2)
library(astsa)
```

```
##
## Attaching package: 'astsa'
##
## The following object is masked from 'package:forecast':
##
##     gas
```

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: In order to identify a given time series dataset as being best modeled by an AR(2) function, you must look at the ACF and PACF plots. If an AR(2) function is the best model to capture the nature of the time series dataset, the ACF plot will decay exponentially over time and the PACF plot will display a significant drop in PACF value after the second lag (technically the second lag after the first non-1 valued lag)

- MA(1)

  Answer: In order to identify a given time series dataset as being best modeled by an MA(1) function, you must look at the ACF and PACF plots. If an MA(1) function is the best model to capture the nature of the time series dataset, the PACF plot will decay exponentially over time and the ACF plot will display a significant drop in ACF value after the first lag (technically the first lag after the first non-1 valued lag)

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). Consider three models: ARMA(1,0), ARMA(0,1) and ARMA(1,1) with parameters $\phi = 0.6$ (look at the PACF - is lag 1 .6?) and $\theta = 0.9$. The ϕ refers to the AR coefficient and the θ refers to the MA coefficient (NOT the coefficient on the ACF. can't state this for MA - can only do comparison for AR model). Use R to generate $n = 100$ observations from each of these three models

```r
arma.1.0<-arima.sim(n=100, model=list(ar=c(0.6)))
arma.0.1<-arima.sim(n=100, model=list(ma=c(0.9)))
arma.1.1<-arima.sim(n=100, model=list(ar=c(0.6), ma=c(0.9)))

arma.1.0
```

```
## Time Series:
## Start = 1
## End = 100
## Frequency = 1
##   [1] -0.33286203 -0.91527864 -0.80862591 -1.05639721 -0.66430303 -1.90806119
##   [7]  0.25988137  0.34243803  0.53913120 -2.33620579  0.01636150 -0.34077714
##  [13]  1.55737268  1.19598420  0.56782265 -1.55168783 -0.75995264 -1.50036428
##  [19]  0.33258794  0.33170710 -0.03177840  1.32295554 -0.79643943  1.50858856
```

```
## [25] -0.99500009  0.22164445  0.75433427 -0.80903755  0.37688890  0.35843152
## [31]  0.24957967  0.03190461 -0.83400337 -0.63917750  0.40805061  0.67438353
## [37]  0.18933950  0.79466559  1.67012584 -0.47812957  0.23382393  2.39943805
## [43]  1.11046188 -0.67527673 -0.57633716 -0.39630254 -0.94688142 -1.26544805
## [49] -0.89556776  0.94580301 -0.39071729 -1.15577729 -1.88497587 -2.19752266
## [55] -2.77820554 -0.79520141 -0.77528865 -0.26215578  0.98114349  0.57114029
## [61]  1.52153333  1.77007748  1.11873995  1.12810739  0.72523620  0.10720575
## [67] -0.64251590  1.36665863 -1.67939040 -1.41076023 -0.89867370 -0.53825702
## [73] -0.07871062 -0.09468532 -0.42254123 -0.68518424 -0.65294105 -1.07845061
## [79] -3.18975916 -2.10891309 -2.13375707 -2.02385281 -2.14382409 -1.71067698
## [85] -1.94362237 -0.97046016 -0.50578939 -1.86898446 -1.04085360 -1.85310891
## [91] -2.51844626 -0.84055413 -1.38593654 -1.12399019  1.34168132  0.02268207
## [97] -0.59559388 -0.37552292 -0.92096494 -0.63349737
```

```
arma.0.1
```

```
## Time Series:
## Start = 1
## End = 100
## Frequency = 1
##   [1] -0.24562900  0.22730049  0.78894617 -0.22536966 -1.43598949  1.00662213
##   [7]  0.21999350  0.77769395  3.09591852 -0.43820104 -1.14447898  0.10398551
##  [13] -1.83033422 -2.34127557 -0.07797734  2.43914443  0.88753005 -2.35524201
##  [19] -0.53154447  2.28073056  0.01721867 -1.19392993 -1.06833550 -0.06204229
##  [25]  0.70241734  0.50938286  2.56214860  4.11851579  2.57972748  0.70635964
##  [31] -1.19813548 -1.46071357  0.33458908  0.09838538  0.21735161  1.51088658
##  [37]  1.88359363 -1.40210127 -3.99185755 -2.21850227  1.10870998  0.44100272
##  [43] -1.12348670  0.55328785  0.58797535  0.89435520  0.53003021 -2.51977828
##  [49] -2.44077083 -1.06739911 -0.90437041  0.92081294  0.46070740 -1.84491719
##  [55] -0.44025742 -0.62265204  0.10593290  0.89373881 -0.13739301 -1.10298418
##  [61] -1.06473655  1.60777645  1.52962375  1.04419979 -0.14572183 -2.47504080
##  [67] -1.60172319  0.00561623 -0.46656310  0.68928284  0.50145863 -0.92486402
##  [73]  0.94860620  1.13991935 -0.17207194 -1.52258608  1.05044043  2.61906981
##  [79] -0.58957549 -1.39187645 -0.47913786 -1.48233336 -1.25415165 -0.66328052
##  [85] -1.97476285 -2.45425237 -1.16245015  1.91122009  1.37251460  0.20075623
##  [91]  1.08806855 -0.36972980 -1.63146298 -1.31195222  0.09161476  1.89274979
##  [97] -0.24728493 -1.12934674  0.14704005  0.26563384
```

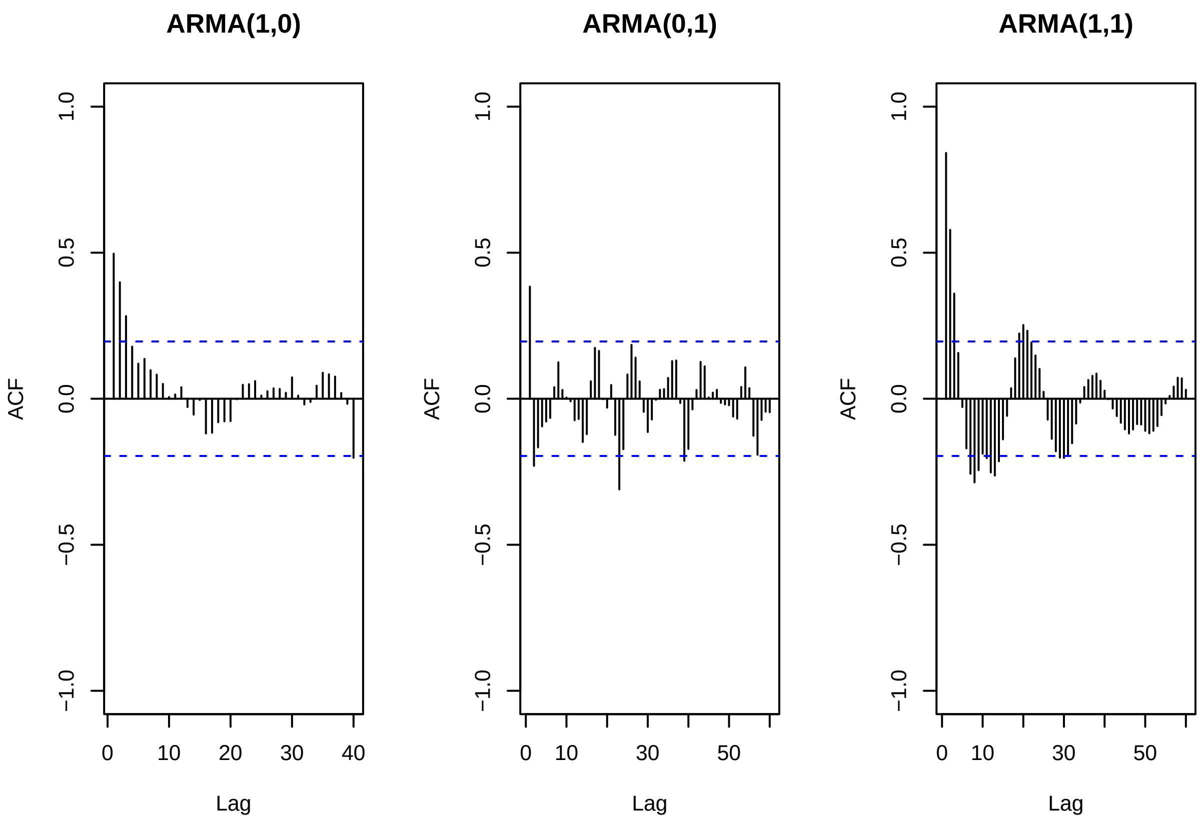
```
arma.1.1
```

```
## Time Series:
## Start = 1
## End = 100
## Frequency = 1
##   [1]  0.394201189  2.467324775  2.411130143  0.916021702  0.092904760
##   [6]  0.303492098  1.470451484  0.766999498 -0.109164591  1.061132531
##  [11]  2.751515029  4.425661072  3.967215482  3.046257498  1.850120451
##  [16] -0.177212315 -0.497025993 -1.260203215 -0.945967001  0.471772561
##  [21]  1.601542018  0.801446690  1.146771368  0.461851464 -2.600327553
##  [26] -2.947978465 -3.690022976 -4.536290278 -3.712581518 -1.405766860
##  [31]  0.350714081  0.427337859  0.319267029  1.341217287  1.103363439
##  [36] -0.158962330 -1.556817732 -1.986904336 -1.822039551 -2.545417368
##  [41] -4.110024263 -5.476433151 -4.850279506 -2.842767596  0.311277621
##  [46]  0.345146924 -0.460507586  1.128348348  2.314830411  3.304981017
##  [51]  3.010809485  0.846508264 -0.348957704  0.781553114  2.674655843
##  [56]  1.113055412 -0.818579097 -2.152506585 -2.088086380 -2.207846783
```

```
## [61] -3.171832281 -3.953913822 -5.510049148 -5.195782869 -4.846826762
## [66] -3.881414961 -1.525022955 -0.354117534  1.792818338  4.144458405
## [71]  4.286959985  4.309136384  4.783834340  3.420680883  1.935223947
## [76]  0.357847699 -0.230583082  0.385175111  1.252949983  1.143407643
## [81]  0.195145994 -0.598322002 -2.254826110 -2.497599403 -1.041359864
## [86] -0.606297970 -1.046116970 -1.298260724  1.525997892  3.088766729
## [91]  0.379807809 -0.002989742 -0.334089172 -1.658036036 -2.863133833
## [96] -0.820580951  0.524386139 -0.100577565  1.423117120  3.333355276
```

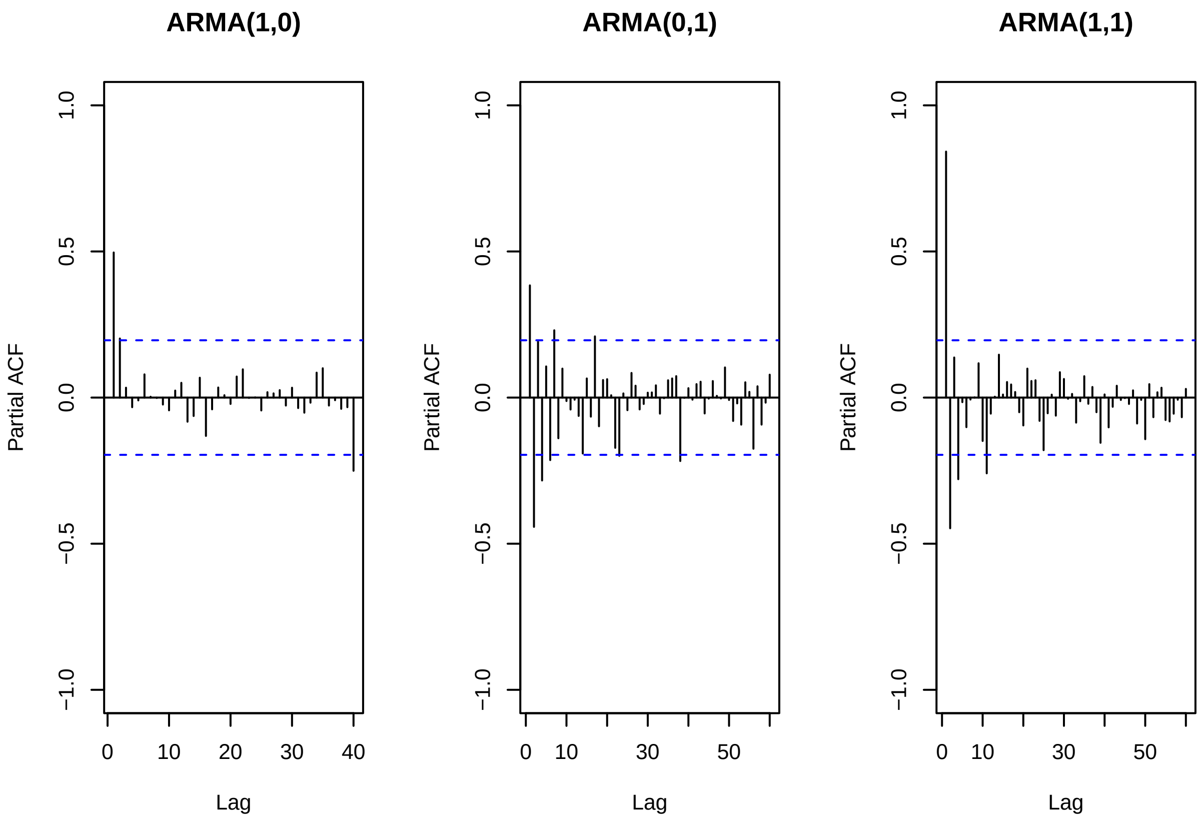
(a) Plot the sample ACF for each of these models in one window to facilitate comparison (Hint: use command $par(mfrow = c(1, 3))$ that divides the plotting window in three columns).

```
par(mfrow=c(1,3))
Acf(arma.1.0,lag.max=40,main="ARMA(1,0)",ylim=c(-1,1))
Acf(arma.0.1,lag.max=60,main="ARMA(0,1)",ylim=c(-1,1))
Acf(arma.1.1,lag.max=60,main="ARMA(1,1)",ylim=c(-1,1))
```



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

```
par(mfrow=c(1,3))
Pacf(arma.1.0,lag.max=40,main="ARMA(1,0)",ylim=c(-1,1))
Pacf(arma.0.1,lag.max=60,main="ARMA(0,1)",ylim=c(-1,1))
Pacf(arma.1.1,lag.max=60,main="ARMA(1,1)",ylim=c(-1,1))
```



(c) 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 identify them correctly? Explain your answer.

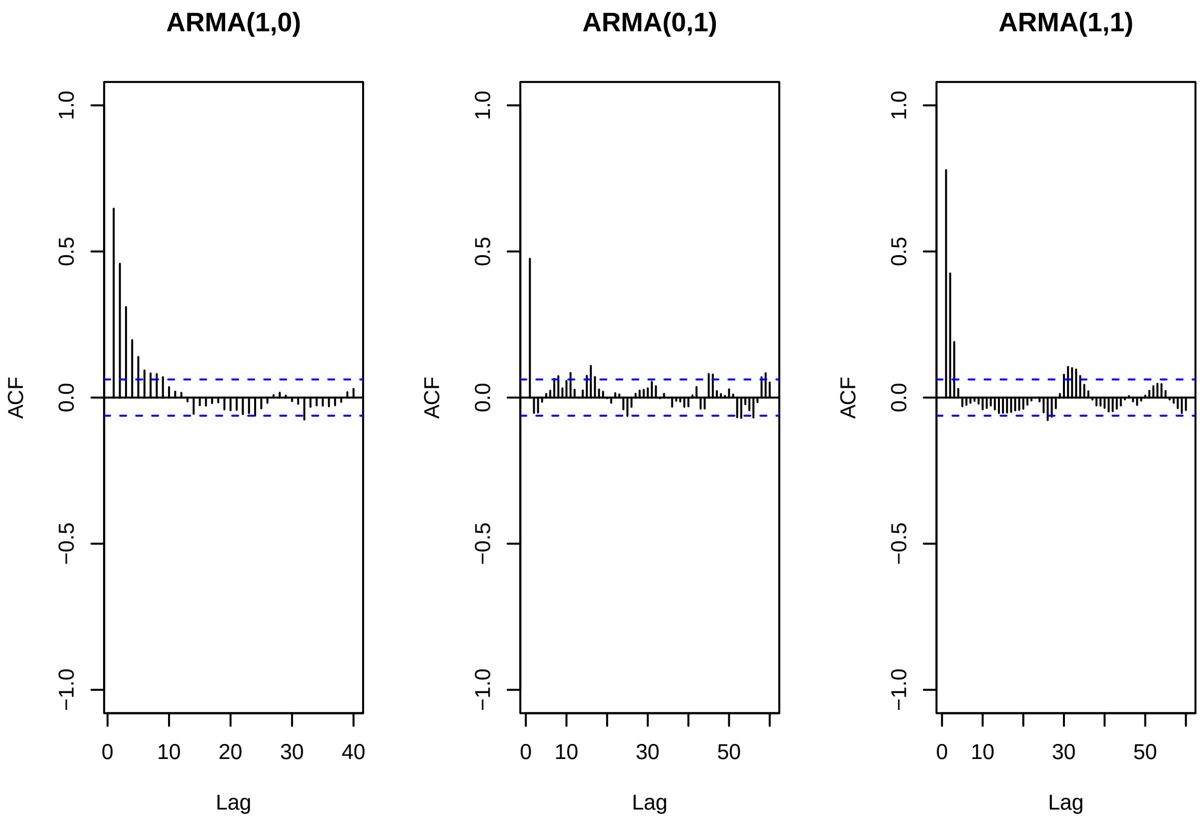
Answer: I think that I could identify these plots as ARMA plots because I can see that there's not only an exponential decline in the ACF but there's also an exponential decline in lag value in the PACF. Normally this exponential decline only occurs in one or the other. However I don't think that I could identify the order. In order to identify the order for the AR components of the models I'm looking at the PACF. The first model (AR(1,0)) broadly displays an AR(1) but the second one appears to display AR(2) (which shouldn't display an AR order at all). The third I could see having an order of 1 but maybe even an AR of 2. Meanwhile for the ACF - the segment that displays the order for the MA models - I see an order 1 for the first chart (when the order should be 0), an order 1 for the second (which should be a 1) and maybe an order one for the third (which should be an order 1 for the MA component).

(d) Compare the ACF and PACF values R computed with the theoretical values you provided for the coefficients. Do they match? Explain your answer.

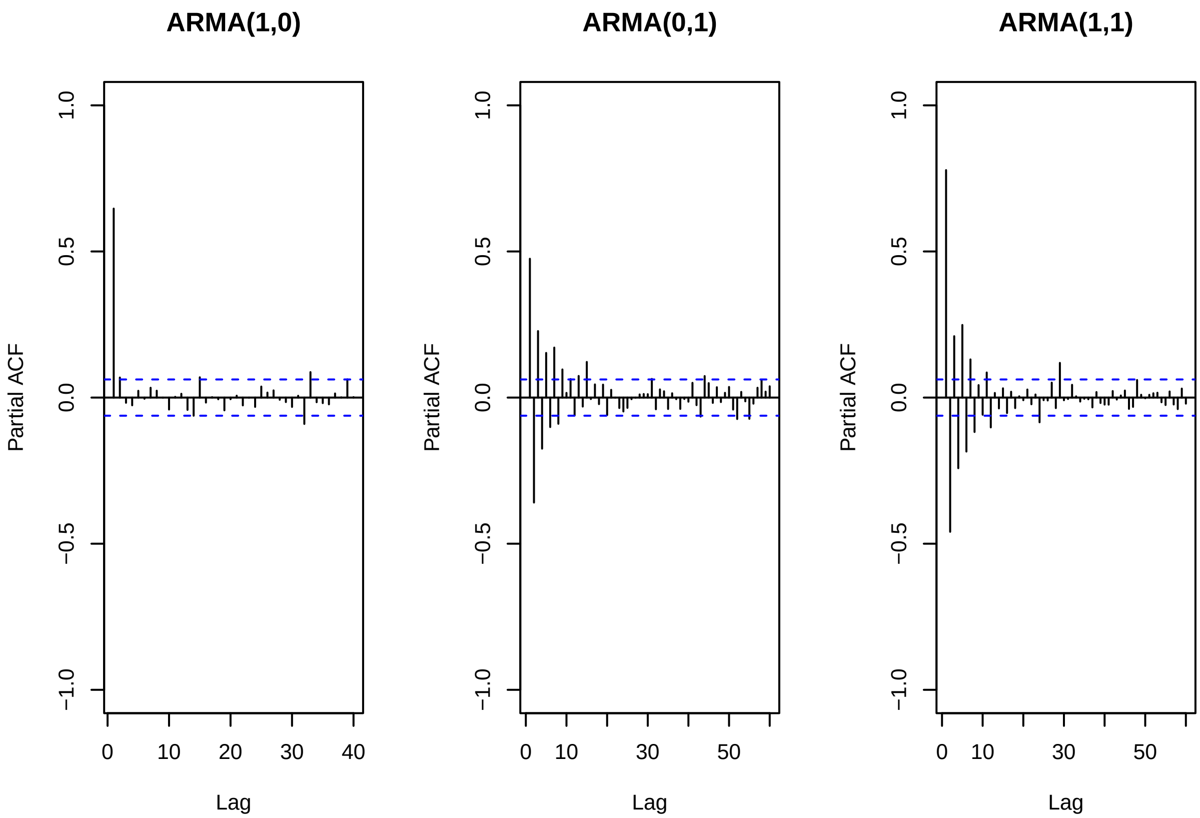
Answer: I simulated a phi of 0.6 - however the PACFs display around 0.5 and then far above 0.5 - maybe around 0.8 - for ARMA(1,1). I simulated a theta of 0.9 but the ACF orders display 0.5, 0.5 and something around 0.8. So they do not match

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

```
par(mfrow=c(1,3))
Acf(arma.1.0.n,lag.max=40,main="ARMA(1,0)",ylim=c(-1,1))
Acf(arma.0.1.n,lag.max=60,main="ARMA(0,1)",ylim=c(-1,1))
Acf(arma.1.1.n,lag.max=60,main="ARMA(1,1)",ylim=c(-1,1))
```




```r
par(mfrow=c(1,3))
Pacf(arma.1.0.n,lag.max=40,main="ARMA(1,0)",ylim=c(-1,1))
Pacf(arma.0.1.n,lag.max=60,main="ARMA(0,1)",ylim=c(-1,1))
Pacf(arma.1.1.n,lag.max=60,main="ARMA(1,1)",ylim=c(-1,1))
```



Answer: I could once again identify them as ARMA models but once again could not identify the models - I'm seeing the AR having orders of: 1, 2, 2 and the MA having orders of 1, 1, 1, which if the order of the models are ARMA(1,0), ARMA(0,1) and ARMA(1,1) my order estimates are all off. However the phi values appear closer - the PACF's lag is 0.6 for the ARMA(1,0), and is a little closer to 0.6 for the other models. The theta values do not appear similar to 0.9 at all across any of the models.

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.

#think about the constants here. this is usually an indication of whether or not differenced and how many times. 0 or 1 depending on whether or not there is a constant in the equation. First part is finding values of the orders. second part is looking for the regression coefficients. If an AR(1), what is the coefficient? specifically what is multiplying by the AR1.

#There is no constant above that indicates that it's been differenced. so D = 1 or 2.

#cant compare seasonal vs non seasonal - particularly when you consider constant seasonal component vs. non-constant seasonal component

ARIMA(1,0,1)(1,0,0)_12

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

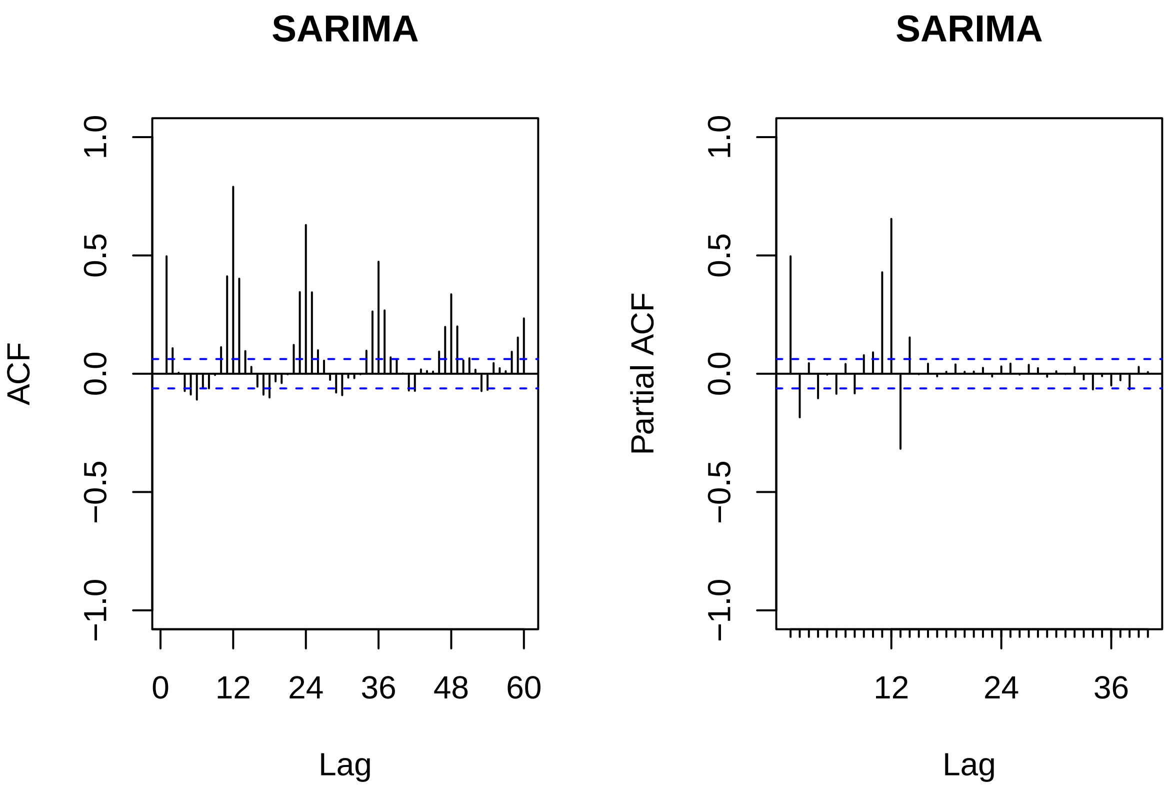
p=0.7 d=0 q=0.1 P=-0.25 D=0 Q=0

Q4

Plot the ACF and PACF of a seasonal ARIMA$(0,1)\times(1,0)_{12}$ model with $\phi = 0.8$ and $\theta = 0.5$ using R. 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 ACF and PACF for the simulated data. 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.

```
sarima<-sarima.sim(ar=0,d=0,ma=.5,sar=.8,D=0,sma=0,S=12,n=1000)

par(mfrow=c(1,2))
Acf(sarima,lag.max=60,main="SARIMA",ylim=c(-1,1))
Pacf(sarima,lag.max=40,main="SARIMA",ylim=c(-1,1))
```



Answer: I could see the non-seasonal component having an order 1 for the MA, although I see an order 1 for the non-seasonal AR. For the seasonal section I could see an SAR of 1 and an SMA of 1 as well. So not particularly well specified.