

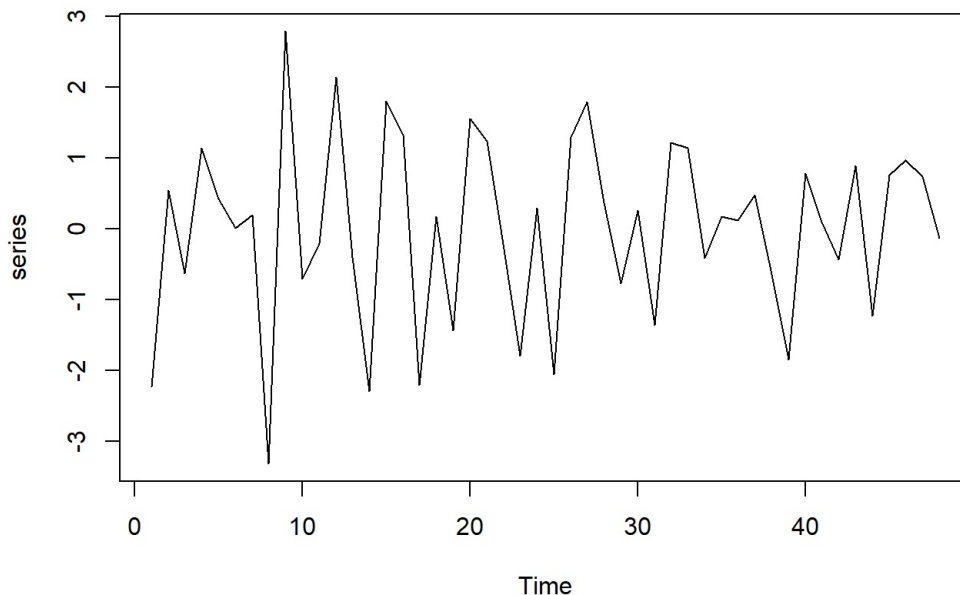
# Assignment 6

Siddharth Nilakhe

2023-11-20

Question 2

```
series <- arima.sim(n = 48, list(ma = -0.8))
plot(series)
```



```
# Part a
mom_estimate <- function(x) {
  z <- acf(x, plot = FALSE)$acf[2]
  if (abs(z) < 0.5) {
    return((-1 + sqrt(1 - 4 * z^2)) / (2 * z))
  } else {
    return(NA)
  }
}

moment_estimate_result <- mom_estimate(series)
cat("Part a:", moment_estimate_result, "\n")
```

```
## Part a: 0.3234414
```

```
# Part b
arma_model_css <- arima(series, order = c(0, 0, 1), method = 'CSS')
cat("Part b:", summary(arma_model_css), "\n")
```

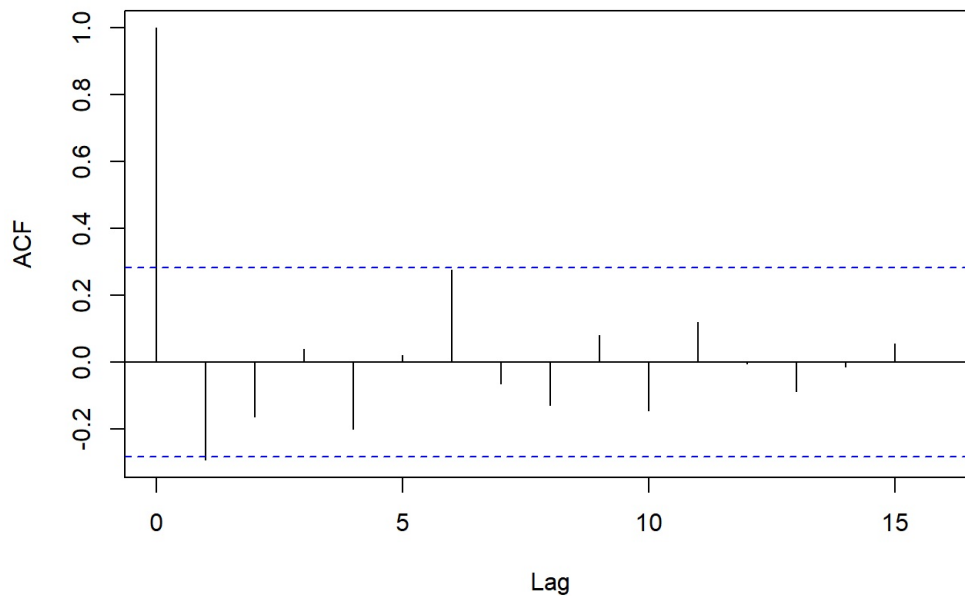
```
## Part b: 2 1 4 2 1 1 7 48 4 1 1 1 1 10 -none- -none- -none- -none- -none- -none- -none- ts -none- -
none- -none- -none- -none- -none- numeric numeric numeric logical numeric logical numeric numeric call character
numeric numeric numeric list
```

```
# Part c
arma_model_ml <- arima(series, order = c(0, 0, 1), method = 'ML')
cat("Part c:", summary(arma_model_ml), "\n")
```

```
## Part c: 2 1 4 2 1 1 7 48 4 1 1 1 1 10 -none- -none- -none- -none- -none- -none- -none- ts -none- -
none- -none- -none- -none- -none- numeric numeric numeric logical numeric numeric numeric numeric call character
numeric numeric numeric list
```

```
# Part d
acf_result <- acf(series)
```

## Series series



```
print("ACF plot suggests the model is MA1")
```

```
## [1] "ACF plot suggests the model is MA1"
```

```
# Part e
newmal_mean <- numeric(500)
newmal_moment_estimate <- numeric(500)

for (i in 1:500) {
  newmal <- arima.sim(n = 48, list(ma = -0.8))
  newmal_mean[i] <- mean(newmal)
  newmal_moment_estimate[i] <- mom_estimate(newmal)
}

cat("Part e - Mean of newmal:", mean(newmal_mean), "\n")
```

```
## Part e - Mean of newmal: 0.0005801644
```

```
cat("Part e - Moment Estimate for newmal:", mean(newmal_moment_estimate), "\n")
```

```
## Part e - Moment Estimate for newmal: NA
```

```
# Part f
variance_formula_result <- (1 - (-0.8)^2) / 48
variance_sd_result <- sd(newmal)^2
cat("Part f - Variance (Using Formula):", variance_formula_result, "\n")
```

```
## Part f - Variance (Using Formula): 0.0075
```

```
cat("Part f - Variance (Using sd(newmal)^2):", variance_sd_result, "\n")
```

```
## Part f - Variance (Using sd(newmal)^2): 1.338285
```

### Question 4

```
library(tseries)
```

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

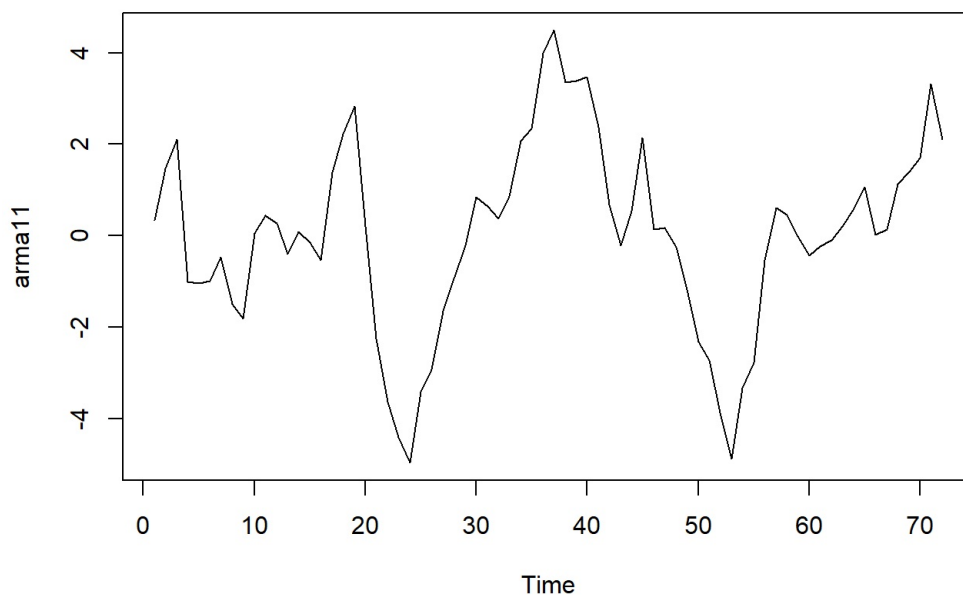
```
library(ggplot2)
library(TSA)
```

```
##
## Attaching package: 'TSA'
```

```
## The following objects are masked from 'package:stats':
##
## acf, arima
```

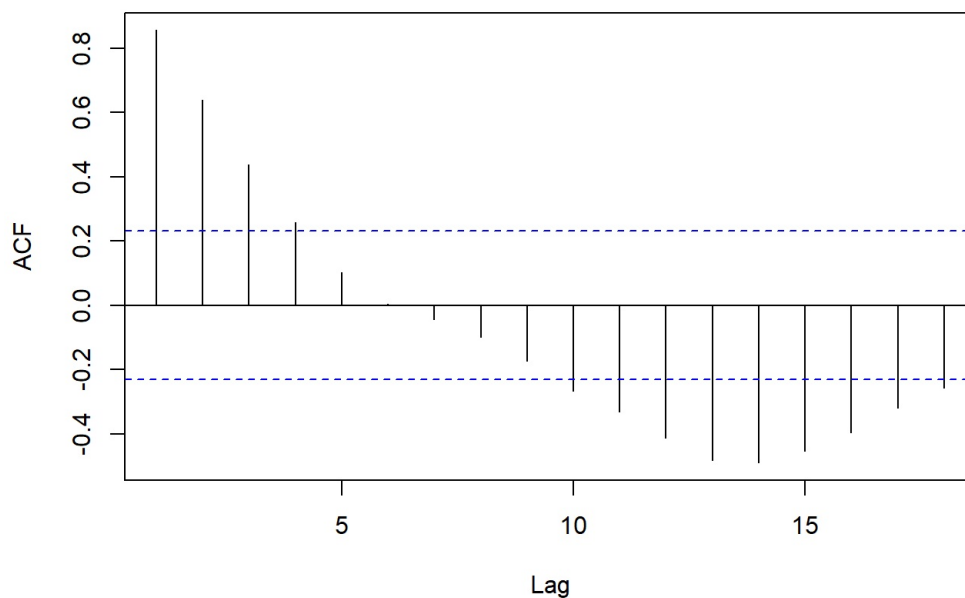
```
## The following object is masked from 'package:utils':
##
## tar
```

```
arma11 <- arima.sim(n = 72, list(ar = 0.7, ma = 0.4))
plot(arma11)
```



```
# Part a
acf_result <- acf(arma11)$acf
```

**Series arma11**



```
acf_result
```

```
## , , 1
##
##          [,1]
## [1,]  0.856481601
## [2,]  0.639865244
## [3,]  0.439074029
## [4,]  0.258194703
## [5,]  0.102958577
## [6,]  0.002945086
## [7,] -0.044435927
## [8,] -0.099116325
## [9,] -0.173440992
## [10,] -0.266291839
## [11,] -0.331330332
## [12,] -0.412506112
## [13,] -0.483578373
## [14,] -0.490096774
## [15,] -0.454488095
## [16,] -0.396331113
## [17,] -0.318890815
## [18,] -0.257993653
```

```
# Part b
arma_model_css <- arima(arma11, order = c(1, 0, 1), method = 'CSS')
summary(arma_model_css)
```

```
##          Length Class  Mode
## coef         3    -none- numeric
## sigma2        1    -none- numeric
## var.coef       9    -none- numeric
## mask          3    -none- logical
## loglik         1    -none- numeric
## aic            1    -none- numeric
## arma          7    -none- numeric
## residuals    72     ts    numeric
## call          4    -none- call
## series        1    -none- character
## code          1    -none- numeric
## n.cond        1    -none- numeric
## nobs          1    -none- numeric
## model         10    -none- list
```

```
# Part c
arma_model_ml <- arima(arma11, order = c(1, 0, 1), method = 'ML')
summary(arma_model_ml)
```

```
##          Length Class  Mode
## coef         3    -none- numeric
## sigma2        1    -none- numeric
## var.coef       9    -none- numeric
## mask          3    -none- logical
## loglik         1    -none- numeric
## aic            1    -none- numeric
## arma          7    -none- numeric
## residuals    72     ts    numeric
## call          4    -none- call
## series        1    -none- character
## code          1    -none- numeric
## n.cond        1    -none- numeric
## nobs          1    -none- numeric
## model         10    -none- list
```

```
# Part d
eacf_result <- eacf(arma11)
```

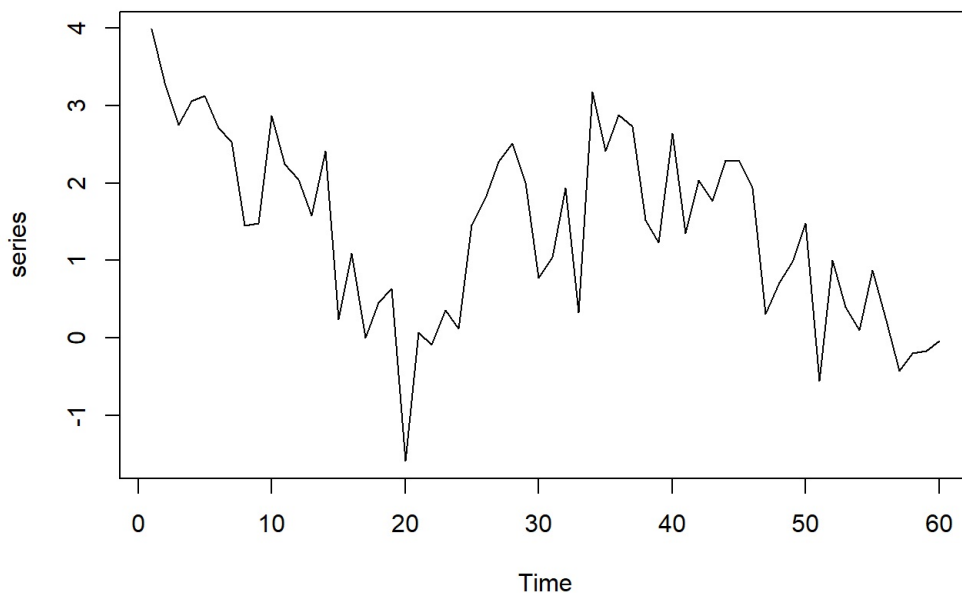
```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x x x x 0 0 0 0 0 x x x x x
## 1 x 0 0 0 0 0 0 0 0 0 0 0 0 0
## 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## 3 x 0 0 0 0 0 0 0 0 0 0 0 0 0
## 4 x 0 0 0 0 0 0 0 0 0 0 0 0 0
## 5 x x 0 0 0 0 0 0 0 0 0 0 0 0
## 6 x x 0 0 0 0 0 0 0 0 0 0 0 0
## 7 0 x 0 0 0 0 0 0 0 0 0 0 0 0
```

```
cat("EACF tells us it is an ARMA(1,1) model\n")
```

```
## EACF tells us it is an ARMA(1,1) model
```

### Question 3

```
series <- arima.sim(model = list(ar = c(0.6, 0.3)), 60)
plot(series)
```



```
# Part a
ar_model_yw <- ar(series, order.max = 2, AIC = FALSE, method = 'yw')
cat("Part a:", summary(ar_model_yw))
```

```
## Part a: 1 2 1 1 3 1 1 1 2 60 1 1 1 5 4 -none- -none- -none- -none- -none- -none- -none- -none- -
none- ts -none- -none- -none- -none- -none- numeric numeric numeric numeric numeric numeric numeric numer
ic numeric character character numeric call numeric
```

```
# Part b
ar_model_ols <- ar(series, order.max = 2, AIC = FALSE, method = 'ols')
cat("Part b:", summary(ar_model_ols))
```

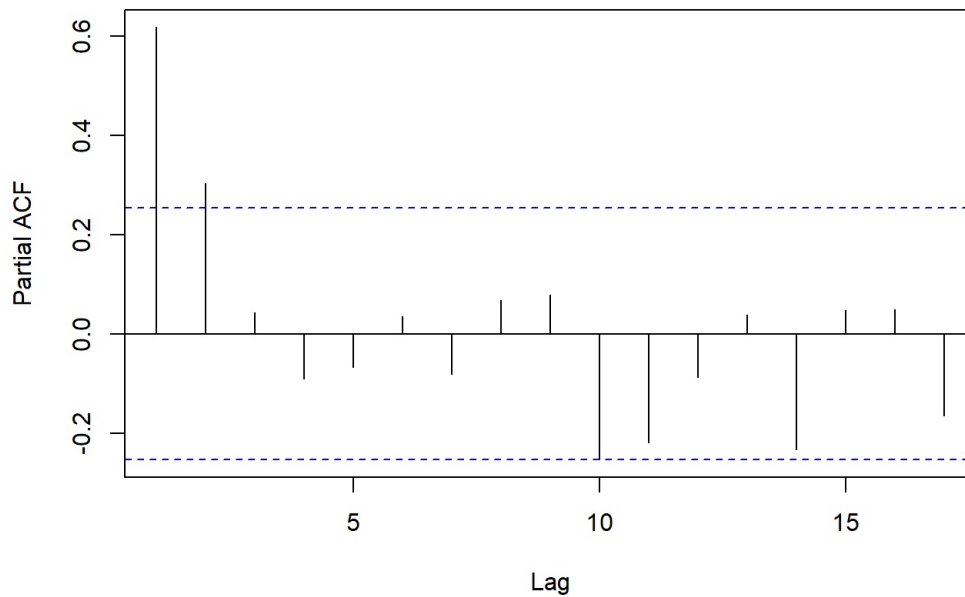
```
## Part b: 1 2 1 1 1 3 1 1 1 0 60 1 1 1 5 2 -none- -none- -none- -none- -none- -none- -none- -none
- -none- -none- ts -none- -none- -none- -none- -none- numeric numeric numeric numeric numeric numeric num
eric numeric NULL numeric character character numeric call list
```

```
# Part c
ar_model_mle <- ar(series, order.max = 2, AIC = FALSE, method = 'mle')
cat("Part c: \n", summary(ar_model_mle))
```

```
## Part c:
## 1 2 1 1 3 1 1 1 0 60 1 1 1 5 4 -none- -none- -none- -none- -none- -none- -none- -none- t
s -none- -none- -none- -none- -none- numeric numeric numeric numeric numeric numeric numeric numeric NULL numeric
character character numeric call numeric
```

```
# Part d
pacf_result <- pacf(series)
```

Series series



```
cat("Part d: \nIt is an AR2 process")
```

```
## Part d:
## It is an AR2 process
```

```
# Part e - Simulation and Analysis
newar2_mean <- numeric(500)
for (i in 1:500) {
  newar2 <- arima.sim(model = list(ar = c(0.6, 0.3)), 60)
  newar2_mean[i] <- mean(newar2)
}
cat("Part e: \n", mean(newar2_mean))
```

```
## Part e:
## 0.06635431
```

```
ar_model_newar2_yw <- ar(newar2, order.max = 2, AIC = FALSE, method = 'yw')
cat("Part e: \n", summary(ar_model_newar2_yw))
```

```
## Part e:
## 1 2 1 1 3 1 1 1 2 60 1 1 1 5 4 -none- -none- -none- -none- -none- -none- -none- -none- -none- t
s -none- -none- -none- -none- -none- numeric numeric numeric numeric numeric numeric numeric numeric nume
ric character character numeric call numeric
```

```
ar_model_newar2_ols <- ar(newar2, order.max = 2, AIC = FALSE, method = 'ols')
cat("Part e: \n", summary(ar_model_newar2_ols))
```

```
## Part e:
## 1 2 1 1 1 3 1 1 1 0 60 1 1 1 5 2 -none- -none- -none- -none- -none- -none- -none- -none- -none-
- -none- ts -none- -none- -none- -none- -none- numeric numeric numeric numeric numeric numeric numeric nu
meric NULL numeric character character numeric call list
```

```
ar_model_newar2_mle <- ar(newar2, order.max = 2, AIC = FALSE, method = 'mle')
```

```
## Warning in arima0(x, order = c(i, 0L, 0L), include.mean = demean): possible
## convergence problem: optim gave code = 1
```

```
cat("Part e: \n", summary(ar_model_newar2_mle))
```

## Part e:

```
## 1 2 1 1 3 1 1 1 0 60 1 1 1 5 4 -none- -none- -none- -none- -none- -none- -none- -none- -none- t
s -none- -none- -none- -none- -none- numeric numeric numeric numeric numeric numeric numeric numeric NULL numeric
character character numeric call numeric
```

Q5)

Since the dataset robot was having issues at my machine, I had to run it on my friend's machine. Please find the code and the screenshot of the output attached below

```
# Part a
data(robot)
arima_model_a <- arima(robot, order = c(1, 0, 0))
cat("Part a: \n", summary(arima_model_a))

# Part b
arima_model_b <- arima(robot, order = c(0, 1, 1))
cat("Part b: \n", summary(arima_model_b))

# Part c
if (AIC(arima_model_a) < AIC(arima_model_b)) {
  cat("Part c - AIC of Part a is lower than Part b\n")
} else {
  cat("Part c - AIC of Part b is lower than Part a\n")
}
```

## **OUTPUT**

```
## Part a:
##      2      1      4      2      1      1      7 324      3      1      1      1      1      10 -none- -none- -none- -none- -none- -none- -
##
## Part b:
##      1      1      1      1      1      1      7 324      3      1      1      1      1      10 -none- -none- -none- -none- -none- -none- -
##
## Part c - AIC of Part b is lower than Part a
```