Macroeconomic Forcasting in R

charles MAWUSI

27/03/2022

This applied study in Time Series follows the **EdX** Macroeconomic forecasting model. The course was originally taught in Eviews. I replicated the codes in R following insight from Struya. Packages used includes among others;

- Forecast
- Tidyverse
- timetk
- lubricate

Data Processing

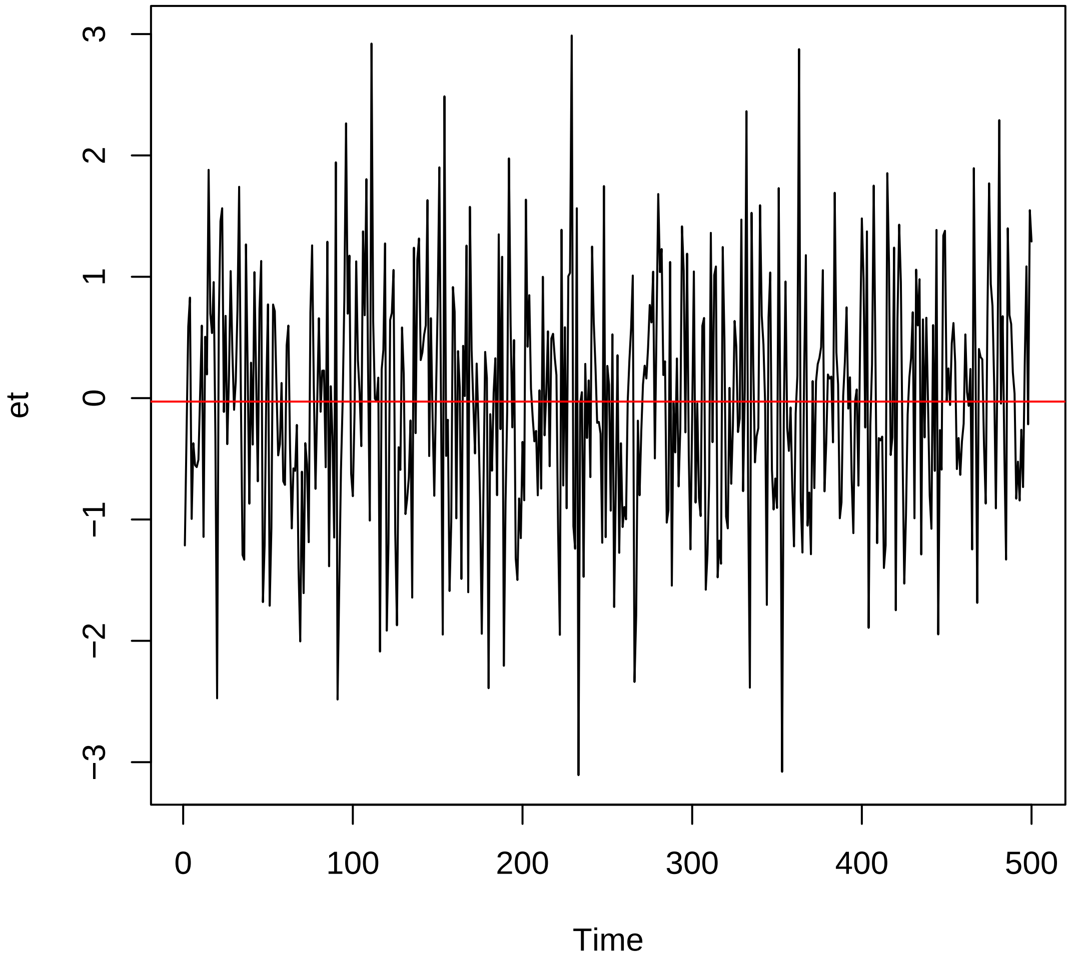
```
library(tidyverse)
library(ggplot2)
library(timetk)
library(dplyr)
library(forecast)
```

```
ARMA_et<-read.csv(file = file.choose(), header = T)
str(ARMA_et)
```

```
'data.frame':   500 obs. of  1 variable:
 $ et: num  -1.214 -0.285 0.59 0.829 -0.995 ...
```

The mean of the data series in 3 decimal places is **-0.028** and the standard deviation is **0.982**. Missing values in the data series is = **0**

```
ARMA_et %>% plot.ts() %>% abline(h=mean(ARMA_et$et), col="red")
```



Simulating AR(1), MA(1), ARMA(1,1) Process

```
# simulating an AR(1) process: yt  = 3.0 + 0.55*yt-1 + et; y0 = 0

y_AR<-as.vector(1)
y_AR[1]<-0
n=500
for (i in 2:500){
  y_AR[i]<-3 + 0.55*(y_AR[i-1])+ARMA_et$et[i]
}
```

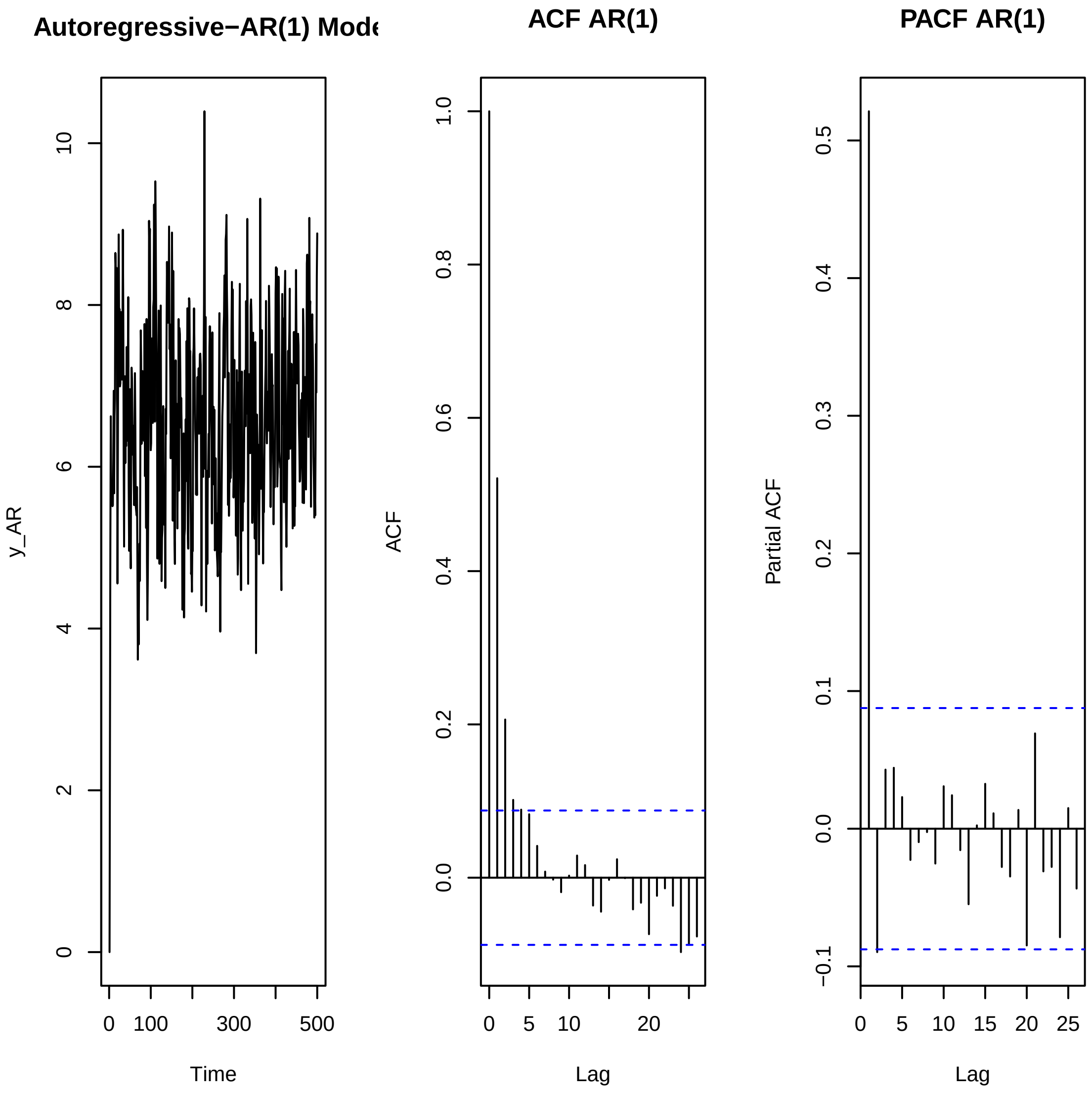
```
# Generating an MA process  yt = -2.5 + et + 0.70 et-1

y_MA<-as.vector(1)
y_MA[1]<-0
n=500
for (i in 2:500){
  y_MA[i]<--2.5 + ARMA_et$et[i]+0.7*ARMA_et$et[i-1]
}
```

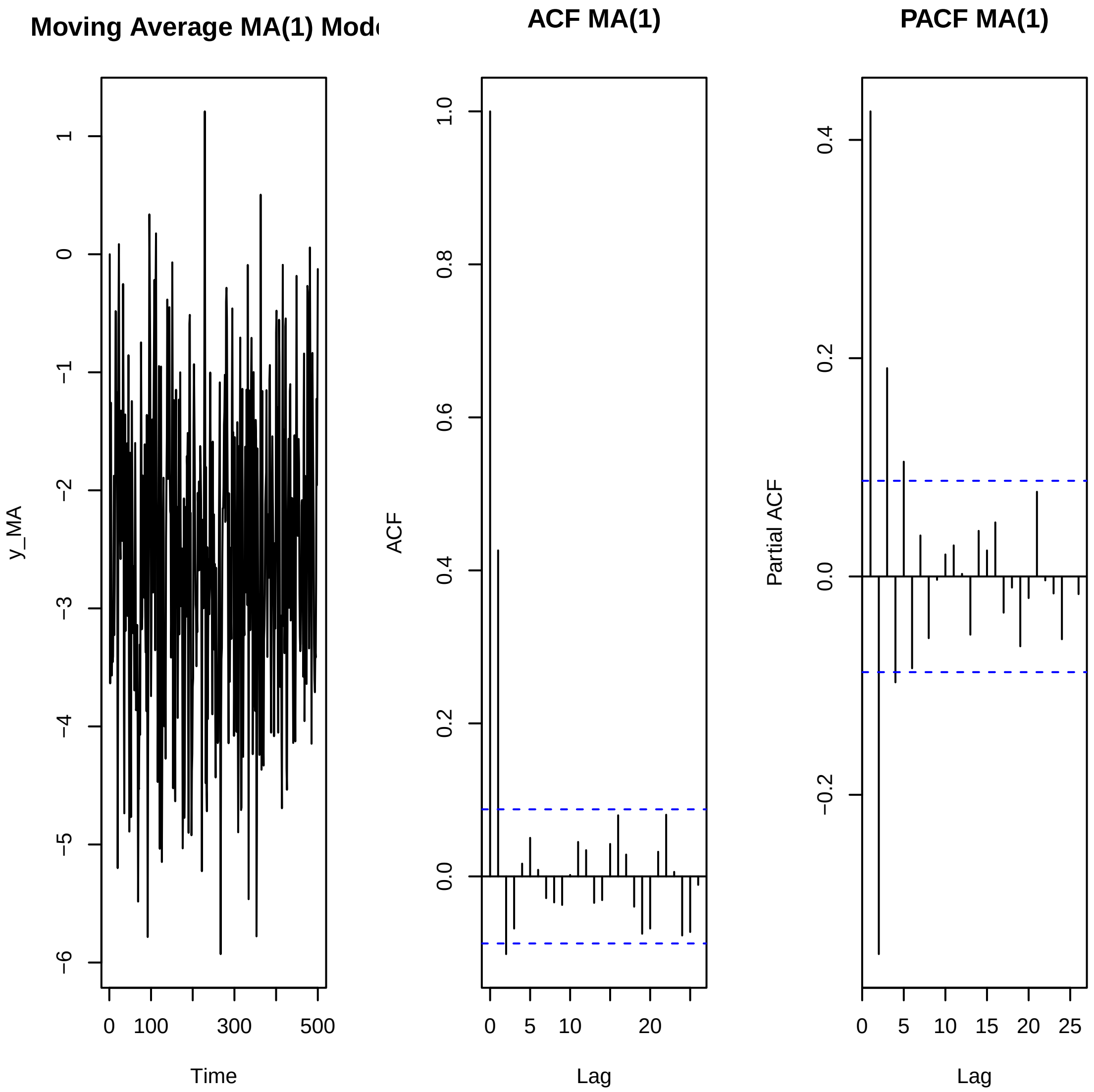
The fifth observation of the MA(1) model is **-2.915** and that of the AR(1) model is **5.648**

```
# Plot the ACF and PAC for the AR(1) and MA(1) Processes

par(mfrow =c(1, 3))
ts.plot(y_AR, main="Autoregressive-AR(1) Model")
acf(y_AR, main="ACF AR(1)")
pacf(y_AR, main="PACF AR(1)")
```



```
par(mfrow =c(1, 3))
ts.plot(y_MA, main="Moving Average MA(1) Model")
acf(y_MA, main ="ACF MA(1)")
pacf(y_MA, main="PACF MA(1)")
```



```
# Generating an ARMA process  yt = 0.5 + 0.5yt-1 +0.70et-1 + et

ARMA<-as.vector(1)
ARMA[1]<-0
n=500
for (i in 2:n){
  ARMA[i]<-0.5+0.55*ARMA[i-1]+ARMA_et$et[i]+0.7*ARMA_et$et[i-1]
}
```

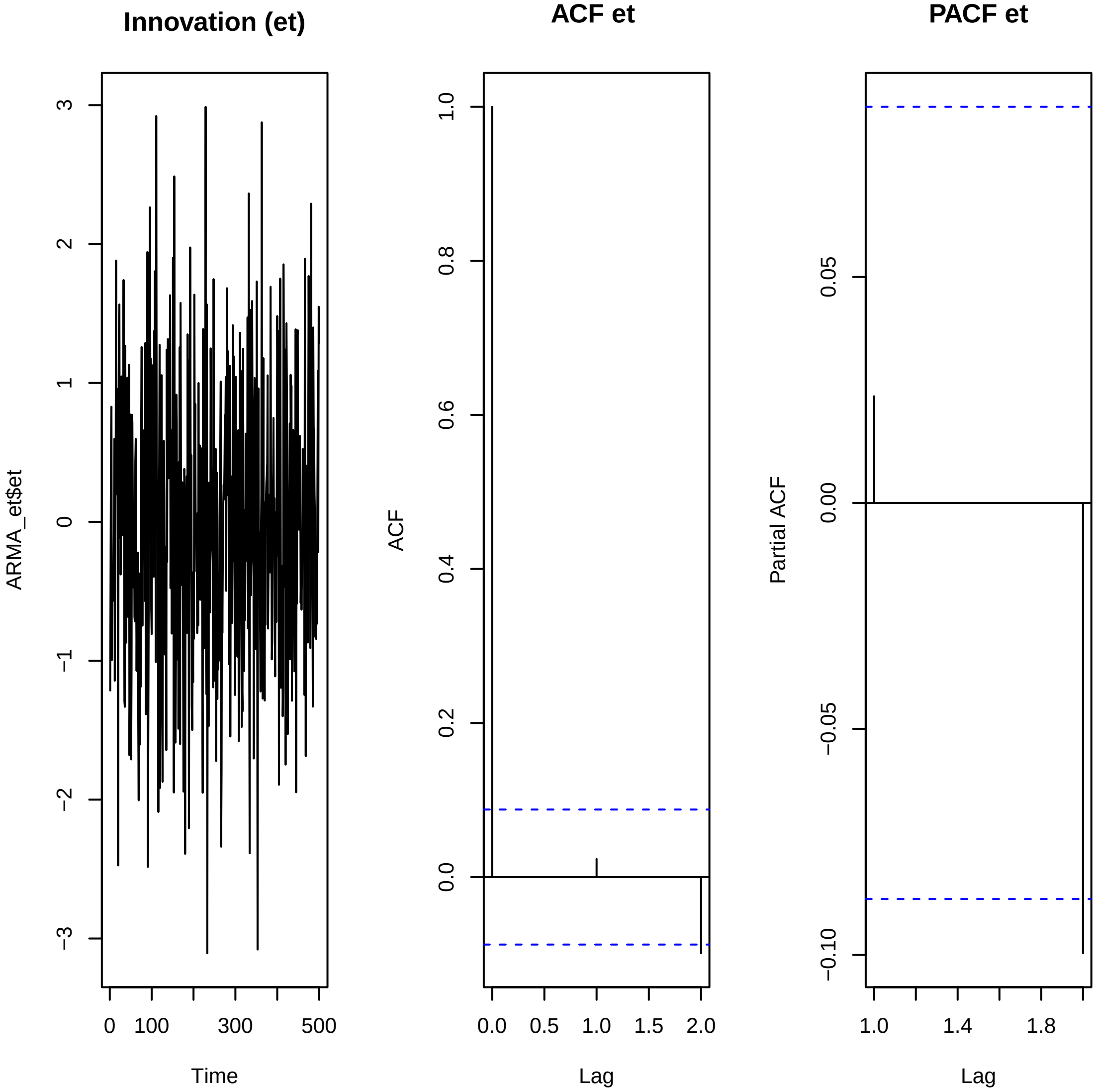
The fifth observation of the simulate ARMA is **1.206**

Limit all Series ACF and PACF to A Maximum of 2 lags

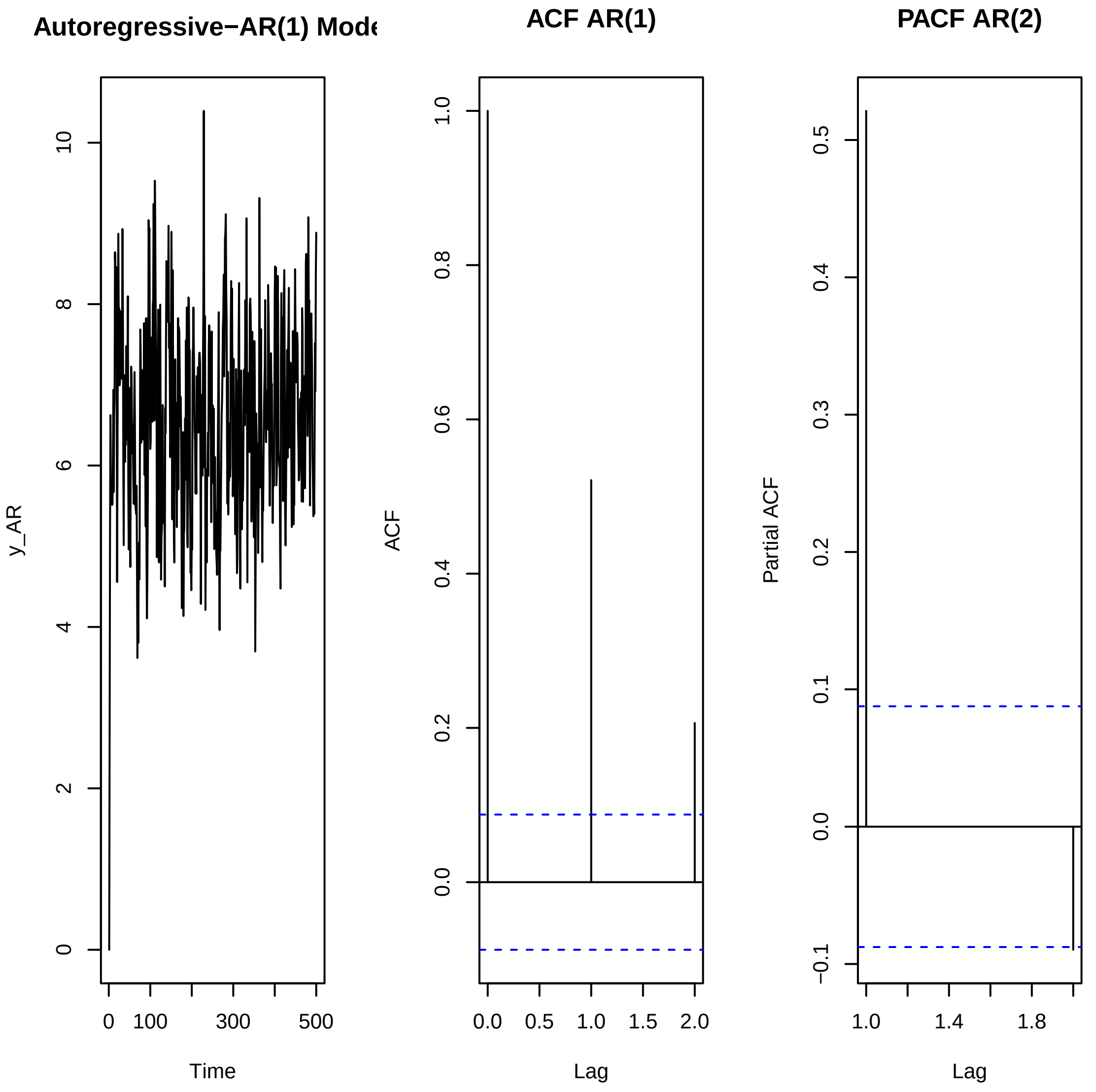
```
# Plot ACF and PACF for all the series with maximun lags 1 and 2

par(mfrow =c(1, 3))
ts.plot(ARMA_et$et,  main="Innovation (et)")
```

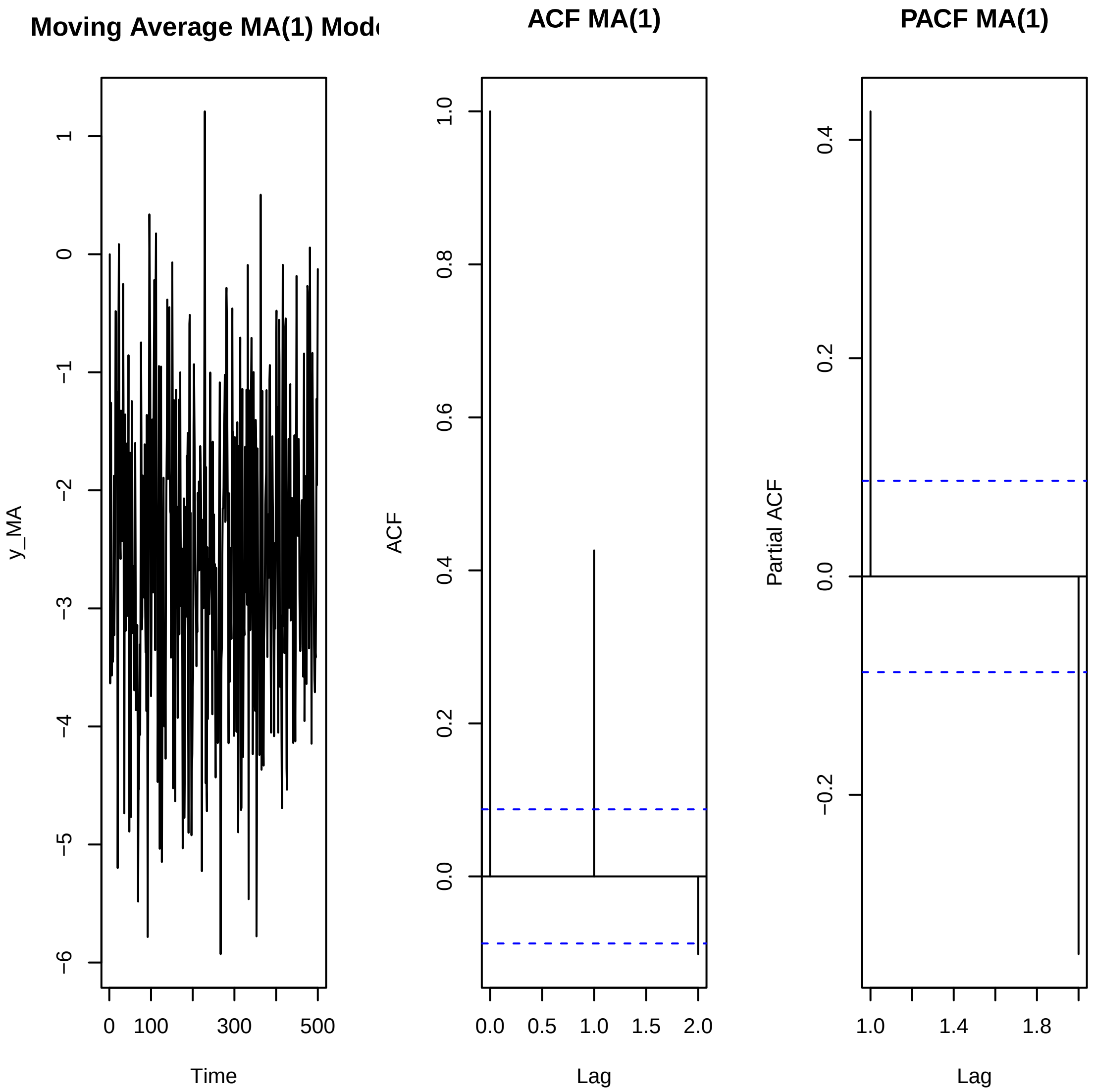
```
acf(ARMA_et$et, lag.max = 2, main="ACF et ")
pacf(ARMA_et$et,lag.max = 2, main="PACF et")
```



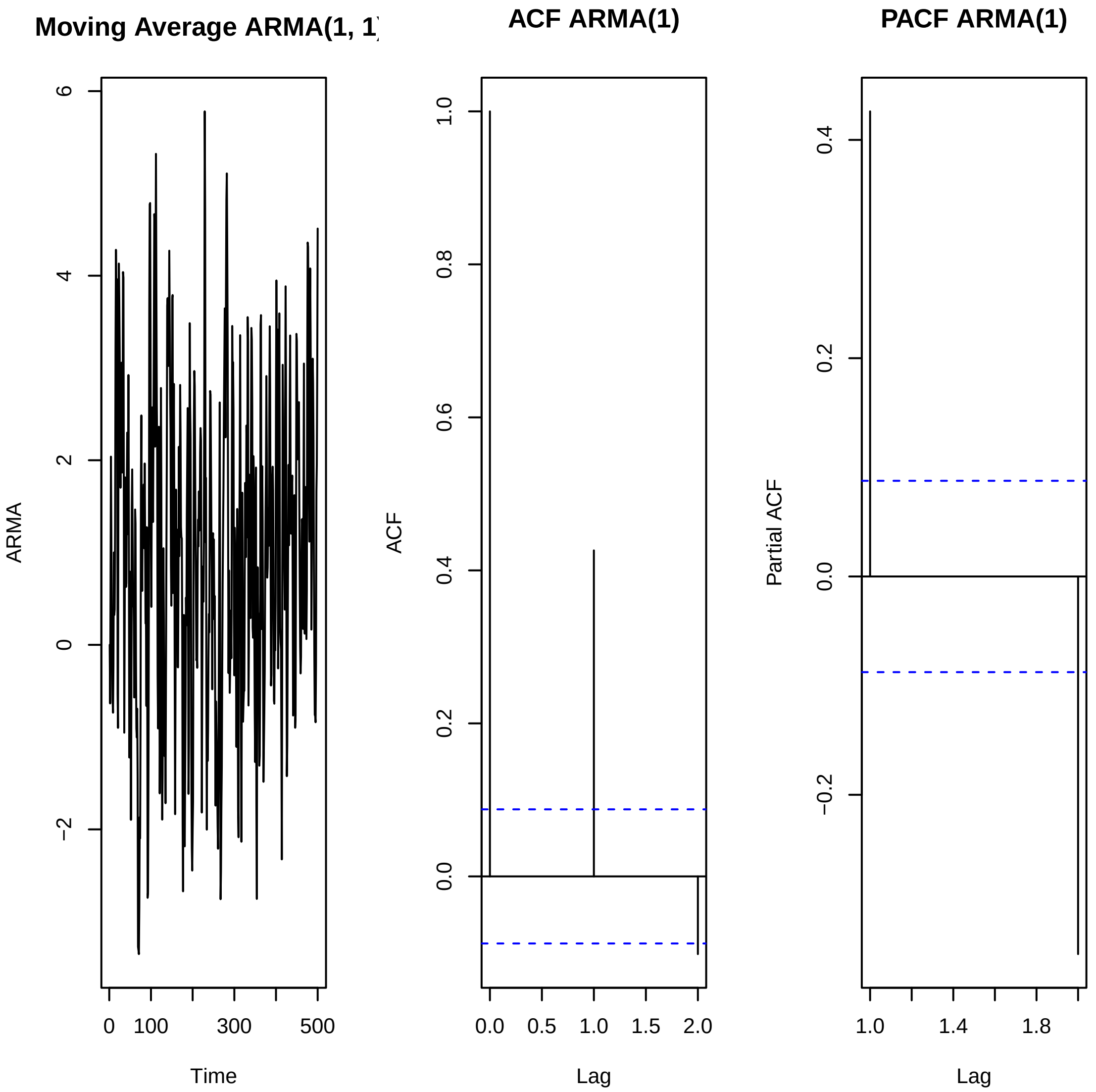
```
par(mfrow =c(1, 3))
ts.plot(y_AR, main="Autoregressive-AR(1) Model")
acf(y_AR,lag.max = 2, main="ACF AR(1)")
pacf(y_AR,lag.max = 2,  main="PACF AR(2)")
```



```
par(mfrow =c(1, 3))
ts.plot(y_MA, main="Moving Average MA(1) Model")
acf(y_MA, lag.max = 2, main ="ACF MA(1)")
pacf(y_MA, lag.max = 2,  main="PACF MA(1)")
```



```
par(mfrow =c(1, 3))
ts.plot(ARMA, main="Moving Average ARMA(1, 1) ")
acf(y_MA,lag.max = 2,  main ="ACF ARMA(1)")
pacf(y_MA, lag.max = 2, main="PACF ARMA(1)")
```



```
library(tidyr)
lm(y_AR~lag(y_AR, 1)+lag(y_AR, 2)) %>% summary()  # max of 2 lags
```

```
Call:
lm(formula = y_AR ~ lag(y_AR, 1) + lag(y_AR, 2))

Residuals:
     Min       1Q   Median       3Q      Max 
-3.15867 -0.70643  0.02768  0.60408  3.10145 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)   3.42376    0.28313  12.093   <2e-16 ***
lag(y_AR, 1)  0.56986    0.04484  12.709   <2e-16 ***
lag(y_AR, 2) -0.08834    0.04356  -2.028   0.0431 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.9798 on 495 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.2814,    Adjusted R-squared:  0.2785
F-statistic: 96.93 on 2 and 495 DF,  p-value: < 2.2e-16
```