ORIE5550_HW3_Markdown

Luis Alonso Cendra Villalobos (lc2234)

2024-02-20

Problem 1

(a) Check if the following AR models are causal

See handwritten notes for further calculations.

```r
# Model 1
phi1_1 <- -0.2
phi2_1 <- 0.48
polyroot(c(1, -1 * phi1_1, -1 * phi2_1))
```

```
## [1]  1.666667+0i -1.250000-0i
```

```r
# Model 2
phi1_2 <- -0.2
phi2_2 <- -0.48
polyroot(c(1, -1 * phi1_2, -1 * phi2_2))
```

```
## [1] -0.208333+1.428261i -0.208333-1.428261i
```

```r
# Model 3
phi1_3 <- -1.8
phi2_3 <- -0.81
polyroot(c(1, -1 * phi1_3, -1 * phi2_3))
```

```
## [1] -1.111111-0i -1.111111+0i
```

(b) For those models in (a) that are causal, write their ACFs explicitly; Compare your result with the computation by ARMAacf; Include R code.

(c) For those models in (a) that are causal, generate a realization of length 200; Plot the realization and correlogram; Overlap the theoretical ACF curve derived at (b) onto the drawn correlogram.

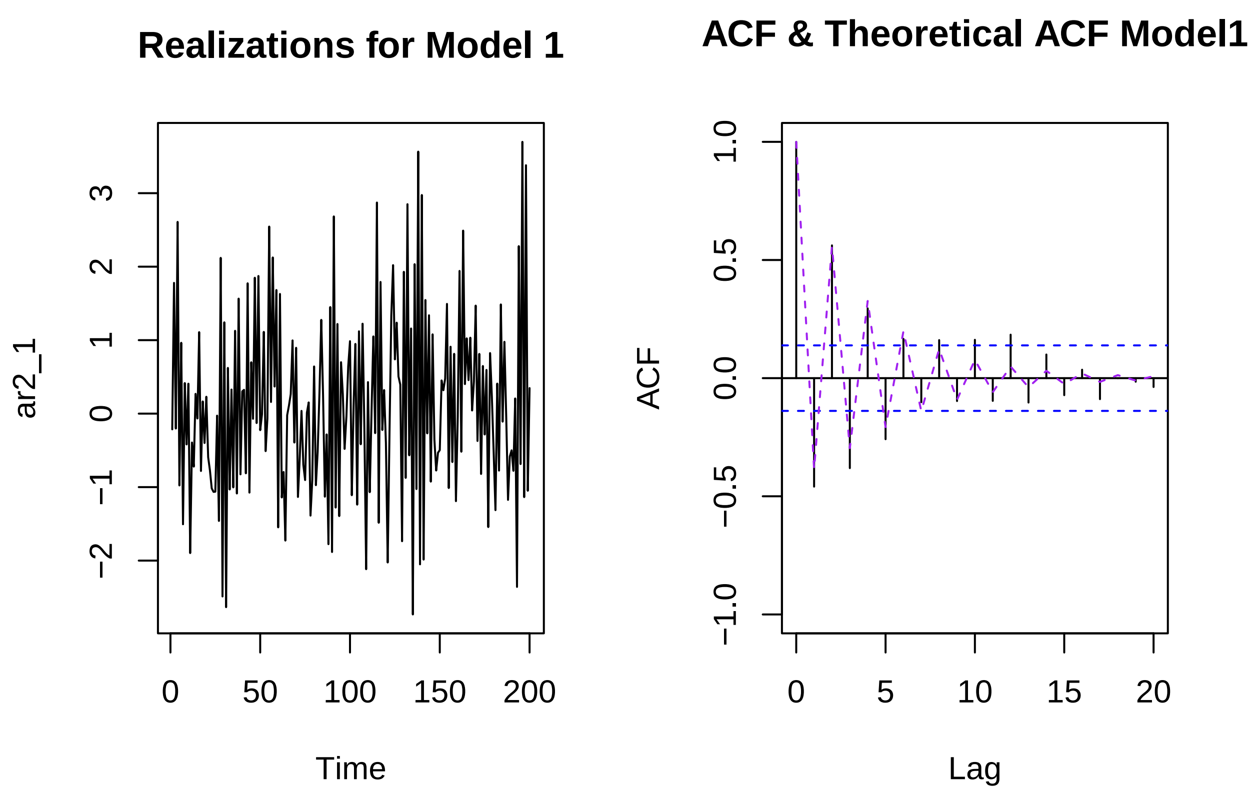
Both (b) and (c) will be calculated in a single block of code for each model.

For explicit ACFs requested in part (b), see handwritten notes.

```r
h <- seq(0, 20, by = 1)

# Model 1
phi1_1 <- -0.2
phi2_1 <- 0.48
ARMAacf(ar=c(phi1_1,phi2_1))
```

```
##          0          1          2
##  1.0000000 -0.3846154  0.5569231
```

```r
ar2_1 <- arima.sim(list(order = c(2,0,0), ar = c(phi1_1,phi2_1) ), n = 200)
theoretical_acf1 <- ((64/91)*(-5/4)^(-h) + (27/91)*(5/3)^(-h))
par(mfrow = c(1, 2))
plot(ar2_1, type="l", main="Realizations for Model 1")
acf(ar2_1, lag.max=20, ylim=c(-1, 1), main="ACF & Theoretical ACF Model1")
lines(h,  theoretical_acf1, lty="dashed", col="purple")
```




```r
# re-shape for comparisson
h <- seq(1, 20, by = 1)
theoretical_acf1 <- ((64/91)*(-5/4)^(-h) + (27/91)*(5/3)^(-h))
ARMAacf_model1 <- ARMAacf(ar=c(phi1_1,phi2_1), lag.max = 20)
stopifnot(all.equal(unname(ARMAacf_model1), c(1, theoretical_acf1)))
equal_1 <- all.equal(unname(ARMAacf_model1), c(1, theoretical_acf1))
cat("Are the theoretical ACF and ARMAacf equal for Model 1? R/", equal_1)
```

```
## Are the theoretical ACF and ARMAacf equal for Model 1? R/ TRUE
```

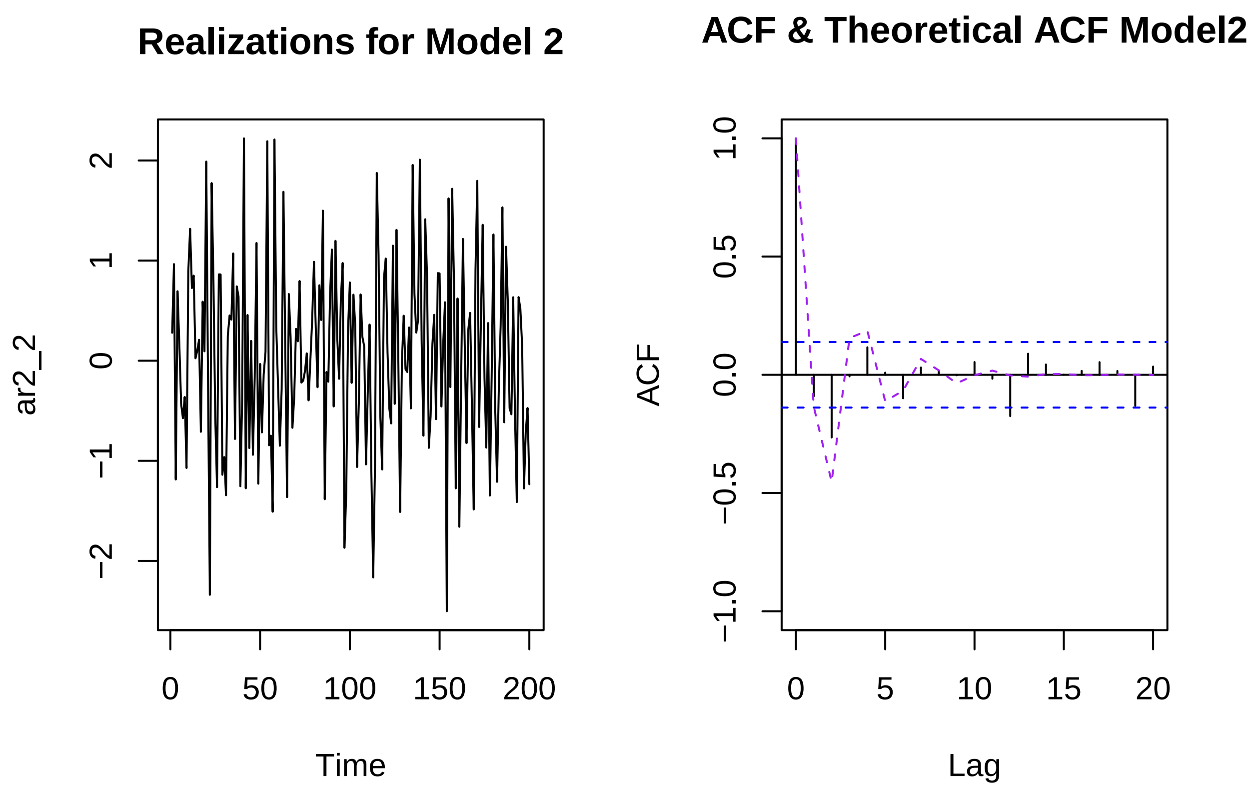
```
h <- seq(0, 20, by = 1)

# Model 2
phi1_2 <- -0.2
phi2_2 <- -0.48
ARMAacf(ar=c(phi1_2,phi2_2))
```

```
##         0          1          2
##  1.0000000 -0.1351351 -0.4529730
```

```
ar2_2 <- arima.sim(list(order = c(2,0,0), ar = c(phi1_2,phi2_2) ), n = 200)
# solving system of equations
rho_0 = 1
rho_1 = phi1_2/(1-phi2_2)
A <- matrix(c(1, 1, 1/polyroot(c(1, -1 * phi1_2, -1 * phi2_2))[1], 1/polyroot(c(1, -1 * phi1_2, -1 * phi
B <- c(rho_0, rho_1)
coeff <- solve(A, B)
print(coeff)
```

```
## [1] 0.5-0.0256249i 0.5+0.0256249i
```

```
theoretical_acf2 <- Re(coeff[1] * (polyroot(c(1, -1 * phi1_2, -1 * phi2_2))[1])**(-h) + coeff[2] * (pol
par(mfrow = c(1, 2))
plot(ar2_2, type="l", main="Realizations for Model 2")
acf(ar2_2, lag.max=20, ylim=c(-1, 1),  main="ACF & Theoretical ACF Model2")
lines(h,  theoretical_acf2, lty="dashed", col="purple")
```




```
# re-shape for comparisson
h <- seq(1, 20, by = 1)
theoretical_acf2 <- Re(coeff[1] * (polyroot(c(1, -1 * phi1_2, -1 * phi2_2))[1])**(-h) + coeff[2] * (pol
ARMAacf_model2 <- ARMAacf(ar=c(phi1_2,phi2_2),lag.max=20)
stopifnot(all.equal(unname(ARMAacf_model2), c(1, theoretical_acf2)))
equal_2 <- all.equal(unname(ARMAacf_model2), c(1, theoretical_acf2))
cat("Are the theoretical ACF and ARMAacf equal for Model 2? R/", equal_2)
```

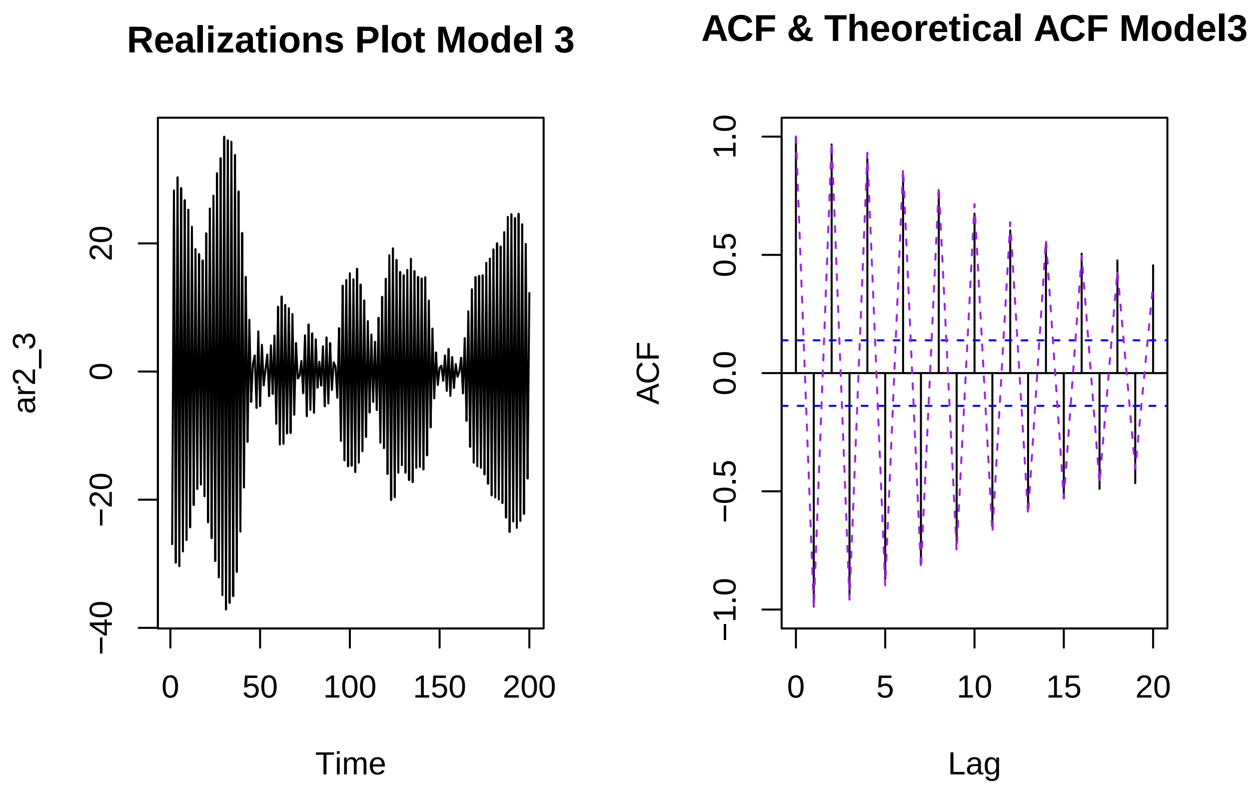
```
## Are the theoretical ACF and ARMAacf equal for Model 2? R/ TRUE
```

```
h <- seq(0, 20, by = 1)

# Model 3
phi1_3 <- -1.8
phi2_3 <- -0.81
ARMAacf(ar=c(phi1_3,phi2_3))
```

```
##          0          1          2
##  1.0000000 -0.9944751  0.9800552
```

```
ar2_3 <- arima.sim(list(order = c(2,0,0), ar = c(phi1_3,phi2_3) ), n = 200)
theoretical_acf3 <- ((-10/9)^(-h) * (1+(19/181)*h))
par(mfrow = c(1, 2))
plot(ar2_3, type="l", main="Realizations Plot Model 3")
```

```r
acf(ar2_3, lag.max=20, ylim=c(-1, 1), main="ACF & Theoretical ACF Model3")
lines(h,  theoretical_acf3, lty="dashed", col="purple")
```




```r
# re-shape for comparisson
h <- seq(1, 20, by = 1)
theoretical_acf3 <- ((-10/9)^(-h) * (1+(19/181)*h))
ARMAacf_model3 <- ARMAacf(ar=c(phi1_3,phi2_3),lag.max=20)
stopifnot(all.equal(unname(ARMAacf_model3), c(1, theoretical_acf3)))
equal_3 <- all.equal(unname(ARMAacf_model3), c(1, theoretical_acf3))
cat("Are the theoretical ACF and ARMAacf equal for Model 3? R/", equal_3)
```

```
## Are the theoretical ACF and ARMAacf equal for Model 3? R/ TRUE
```

Problem 2

(a) Compute the ACVF and ACF of X_t

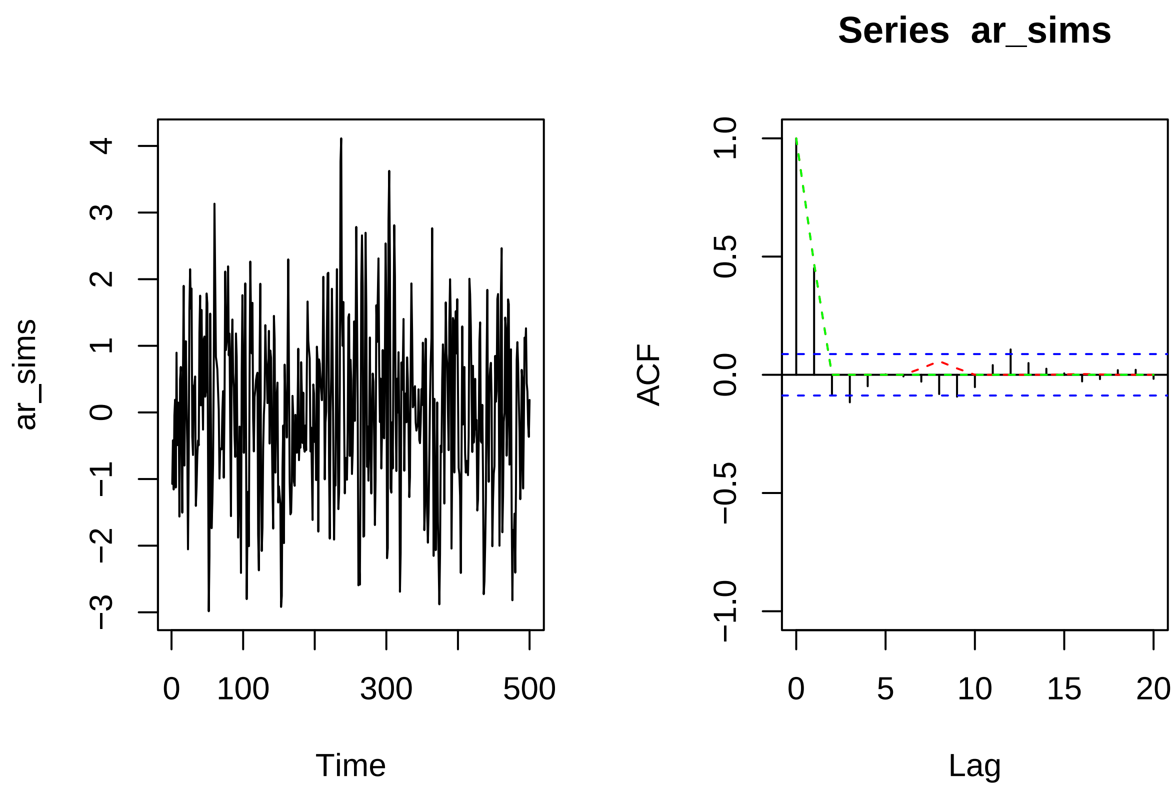
See handwritten notes.

(b) Generate a realization of length 500 from the AR(7) model. Plot the correlogram for the realization up to lag 20; For each plot, overlap the theoretical ACF curve of the MA(1) model and the corresponding AR(7) model with different colors; Repeat the same exercise by generating the realization from the MA(1). [Note: You can use ARMAacf for both AR(7) and MA(1) in this problem.]

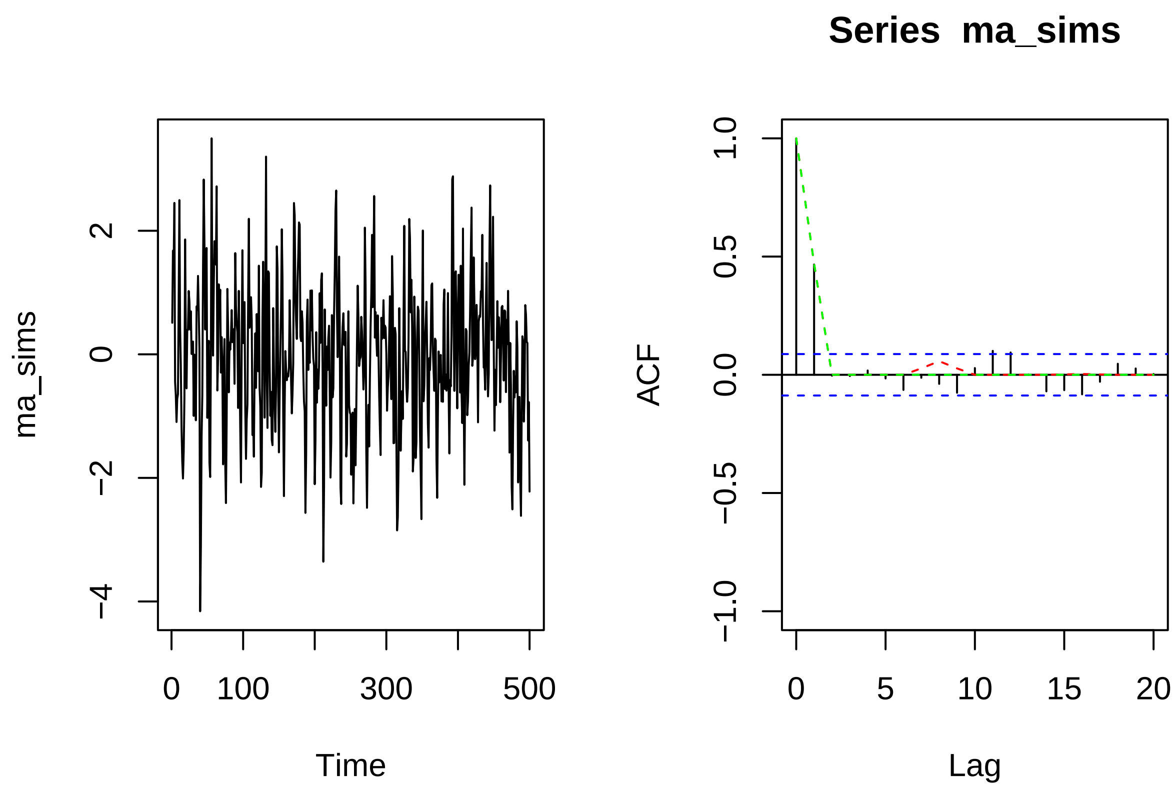
```
phi1 <-  0.7
phi2 <- -0.7 ** 2
phi3 <-  0.7 ** 3
phi4 <- -0.7 ** 4
phi5 <-  0.7 ** 5
phi6 <- -0.7 ** 6
phi7 <-  0.7 ** 7

ARMAacf(ar=c(phi1,phi2, phi3, phi4, phi5, phi6, phi7))
```

```
##             0             1             2             3             4
##  1.000000e+00  4.697987e-01 -2.139571e-19 -1.170855e-18 -6.979772e-20
##             5             6             7
## -1.868048e-17 -3.711439e-18  2.708296e-02
```

```
ar_sims <- arima.sim(list(order = c(7,0,0), ar = c(phi1,phi2, phi3, phi4, phi5, phi6, phi7) ), n = 500)
par(mfrow = c(1, 2))
plot(ar_sims, type="l")
acf(ar_sims, lag.max=20, ylim=c(-1, 1))
lines(seq(0,20),ARMAacf(ar=c(phi1,phi2, phi3, phi4, phi5, phi6, phi7),lag.max=20),lty="dashed",col="red
lines(seq(0,20),ARMAacf(ma=c(0.7),lag.max=20),lty="dashed",col="green")
```




```r
# MA(1) Model
ma_sims <- arima.sim(model= list(ma = phi1), n = 500)
par(mfrow = c(1, 2))
plot(ma_sims, type="l")
acf(ma_sims, lag.max=20, ylim=c(-1, 1))
lines(seq(0,20),ARMAacf(ar=c(phi1,phi2, phi3, phi4, phi5, phi6, phi7),lag.max=20),lty="dashed",col="red
lines(seq(0,20),ARMAacf(ma=c(0.7),lag.max=20),lty="dashed",col="green")
```



Problem 3

See handwritten notes.

Problem 4

Consider the time series of sunspot numbers "SUNSPOTS.txt" posted on Canvas under Homework folder. Do the following.

```r
# import library for auto.arima and forecast functions
#install.packages("forecast")
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.3.2

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

```r
# import library for coeftest function
#install.packages("lmtest")
library(lmtest)
```

```
## Warning: package 'lmtest' was built under R version 4.3.2
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.3.2
```

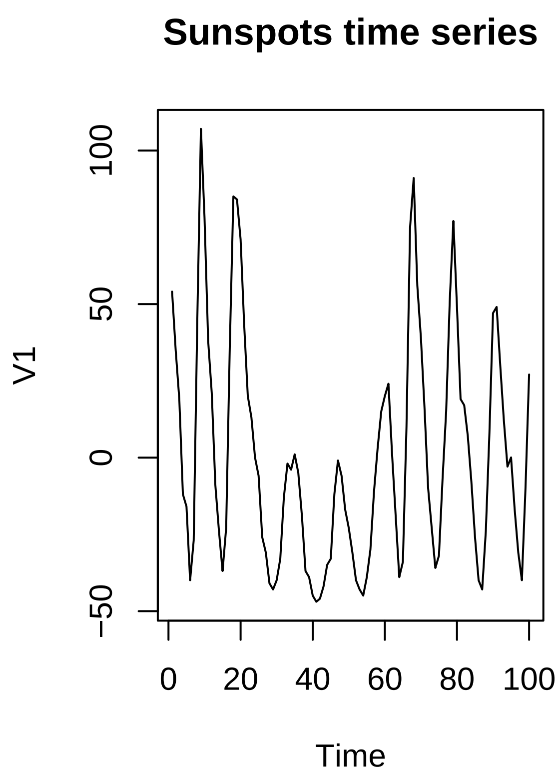
```
##
## Attaching package: 'zoo'
```

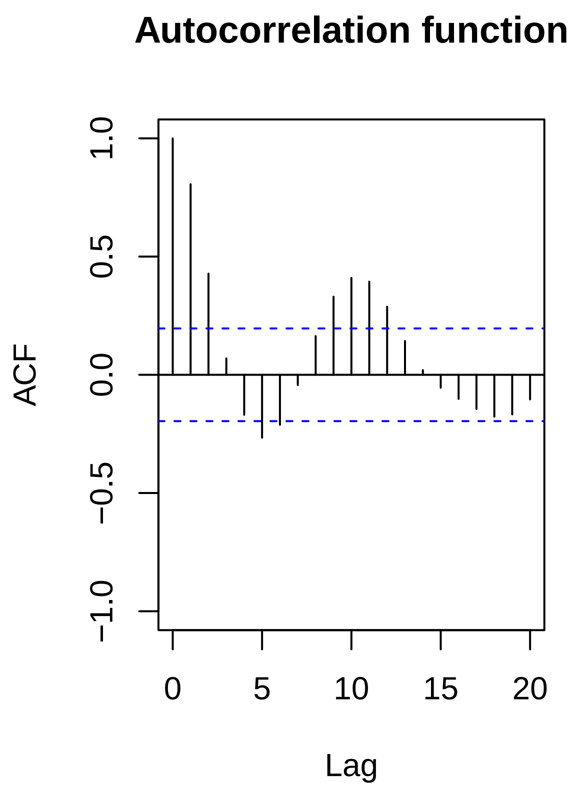
```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
file_path <- "C:\\Users\\alons\\OneDrive - Cornell University\\Cornell University\\Spring 2024\\ORIE 55

data <- read.table(file_path, header = FALSE, sep = "\t")

sunspotsTS_original = ts(data, frequency = 1)
```

(a) Demean the series and produce a time plot and a correlogram of the time series.

```
# demean series
sunspotsTS <- sunspotsTS_original - mean(sunspotsTS_original)
par(mfrow = c(1,2))
plot.ts(sunspotsTS, main = "Sunspots time series")
acf(sunspotsTS, lag.max=20, ylim=c(-1, 1), main = "Autocorrelation function")
```





(b) What order p is suggested by AIC and BIC information criteria, and the sample PACF? Include the outputs for these

```
auto.arima(sunspotsTS,max.p=5,max.q=0,ic="aic",allowmean = FALSE) # AIC
```

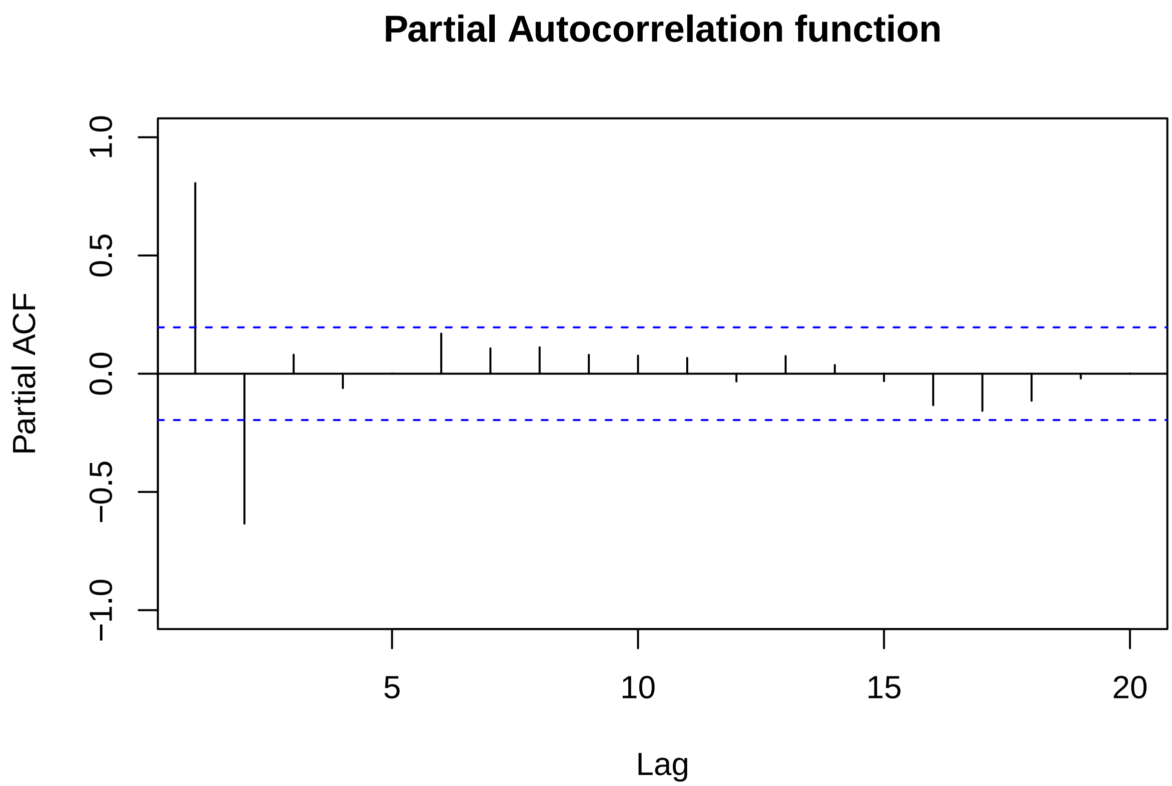
```
## Series: sunspotsTS
## ARIMA(4,0,0) with zero mean
##
## Coefficients:
##          ar1      ar2     ar3      ar4
##       1.5844  -1.1492  0.4344  -0.1462
## s.e.  0.0995   0.1835  0.1853   0.1009
##
## sigma^2 = 222.6:  log likelihood = -411.49
## AIC=832.98   AICc=833.62   BIC=846.01
```

```
auto.arima(sunspotsTS,max.p=5,max.q=0,ic="bic",allowmean = FALSE) # BIC
```

```
## Series: sunspotsTS
## ARIMA(2,0,0) with zero mean
##
## Coefficients:
##          ar1      ar2
```

```
##        1.4076  -0.7131
## s.e.   0.0705   0.0701
##
## sigma^2 = 232.7:  log likelihood = -414.65
## AIC=835.3   AICc=835.55   BIC=843.12
```

```
pacf(sunspotsTS, lag.max=20, ylim=c(-1, 1), main = "Partial Autocorrelation function") # PACF
```



Based on the BIC and PACF, we choose an AR(2) model. Additionally, when fitting an AR(4), some coefficients were not reported as significant, thus, discarding the AIC suggestion.

(c) Fit the AR(p) model to the time series for the order p from part (b); Check if all AR coefficients are significant. If any of the coefficients is not significant, choose a different lag order.

```
arma.model <- arima(sunspotsTS, order=c(2,0,0), include.mean = FALSE, method = "ML")
summary(arma.model) #
```

```
##
## Call:
## arima(x = sunspotsTS, order = c(2, 0, 0), include.mean = FALSE, method = "ML")
##
## Coefficients:
```

```
##          ar1      ar2
##       1.4076  -0.7131
## s.e.  0.0705   0.0701
##
## sigma^2 estimated as 228.1:  log likelihood = -414.65,  aic = 835.3
##
## Training set error measures:
##                     ME     RMSE      MAE      MPE     MAPE      MASE      ACF1
## Training set 0.2789515 15.10212 11.90218 149.4492 325.4567 0.6943522 0.1339231
```

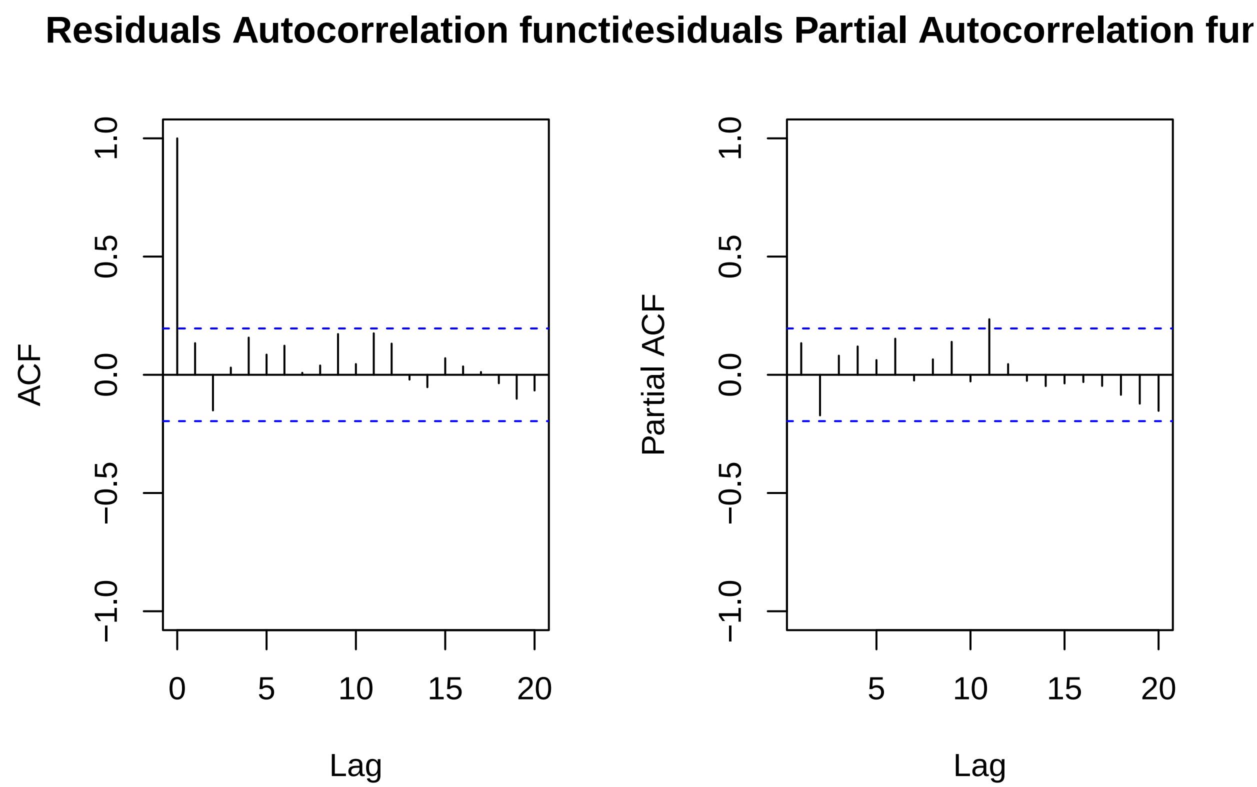
```
coeftest(arma.model)# get p-values
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value  Pr(>|z|)
## ar1  1.407593   0.070468  19.975 < 2.2e-16 ***
## ar2 -0.713116   0.070134 -10.168 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
phi1_q4 <- arma.model$coef[1]
phi2_q4 <- arma.model$coef[2]
```

(d) Produce the sample ACF and PACF of the model residuals; Are the model residuals consistent with the assumptions of normal distribution? Support your explanation with any method.

```
par(mfrow = c(1,2))
acf(arma.model$residuals, lag.max=20, ylim=c(-1, 1), main = "Residuals Autocorrelation function")
pacf(arma.model$residuals, lag.max=20, ylim=c(-1, 1), main = "Residuals Partial Autocorrelation functio
```

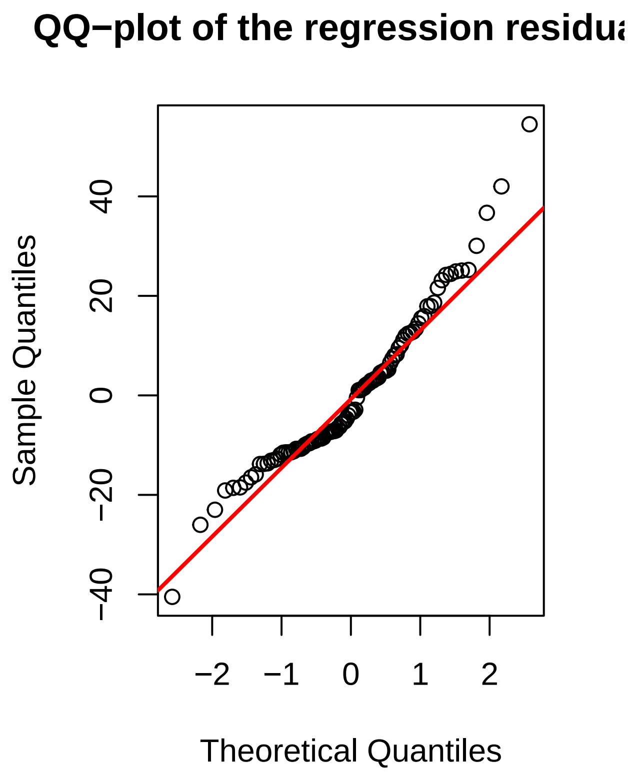



```r
# QQ plot
qqnorm(arma.model$residuals, main = "QQ-plot of the regression residuals")
qqline(arma.model$residuals, col = "red", lwd = 2)

# Shapiro-Wilks test. Ho is normality
shapiro.test(arma.model$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  arma.model$residuals
## W = 0.95061, p-value = 0.0009106
```

```r
#Box-Ljung test
Box.test(arma.model$residuals, lag=20, type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arma.model$residuals
## X-squared = 21.809, df = 20, p-value = 0.3509
```



The QQ-Plots exhibit heavy tails, rejecting normality.

For the Shapiro–Wilk test, the null hypothesis is that the sample comes from a normal distribution, and our results reject this H_0, again, rejecting normality.

For Box-Ljung Test, the null hypothesis for a series of residuals exhibits no autocorrelation for a fixed number of lags. Our results do not reject the null hypothesis of no autocorrelation up to lag 20.

(e) Write down the exact model that was fit to the original time series [Note: In this model, do not forget to include the removed mean].

My result is an AR(2) model, when including the removed mean, we have the general form:

$$(X_t - \mu) = \phi_1(X_{t-1} - \mu) + \phi_2(X_{t-2} - \mu) + Z_t \text{ with } \{Z_t : t \in \mathbb{Z}\} \sim WN(0, \sigma_z^2)$$

Substituting the values for μ, $\hat{\phi}_1$ and $\hat{\phi}_2$ estimated above we have:

$$(X_t - 46.93) = 1.41(X_{t-1} - 46.93) + -0.71(X_{t-2} - 46.93) + Z_t \text{ with } \{Z_t : t \in \mathbb{Z}\} \sim WN(0, \sigma_z^2)$$

(f) Forecast the time series for 5 steps into the future and provide confidence intervals for the forecasts; Produce the forecast time plot with this information.

```
h = 5
arma.forecast <- predict(arma.model, h)
round(arma.forecast$pred, 3)
```

```
## Time Series:
## Start = 101
## End = 105
## Frequency = 1
## [1] 45.185 44.298 30.131 10.823 -6.253
```

```r
round(arma.forecast$se, 3)
```

```
## Time Series:
## Start = 101
## End = 105
## Frequency = 1
## [1] 15.102 26.076 32.354 34.439 34.565
```

```r
lower <- arma.forecast$pred - (1.96 * arma.forecast$se)
upper <- arma.forecast$pred + (1.96 * arma.forecast$se)

forecasted_periods <- c(101:105)

par(mfrow = c(1,1))
plot(sunspotsTS, xlim=c(1,105), main = "Forecasts for the demeaned time series, horizon = 5")
lines(forecasted_periods, arma.forecast$pred, lwd = 2)
lines(forecasted_periods, lower, lty=2, lwd=2)
lines(forecasted_periods, upper, lty=2, lwd=2)
```