Time Series Analysis
Assignment 3

Author

Ghassen Lassoued - s196609

April 23, 2020

Contents

1 Question 3.1

The data is plotted. The observations that will be used for testing are plotted in red.

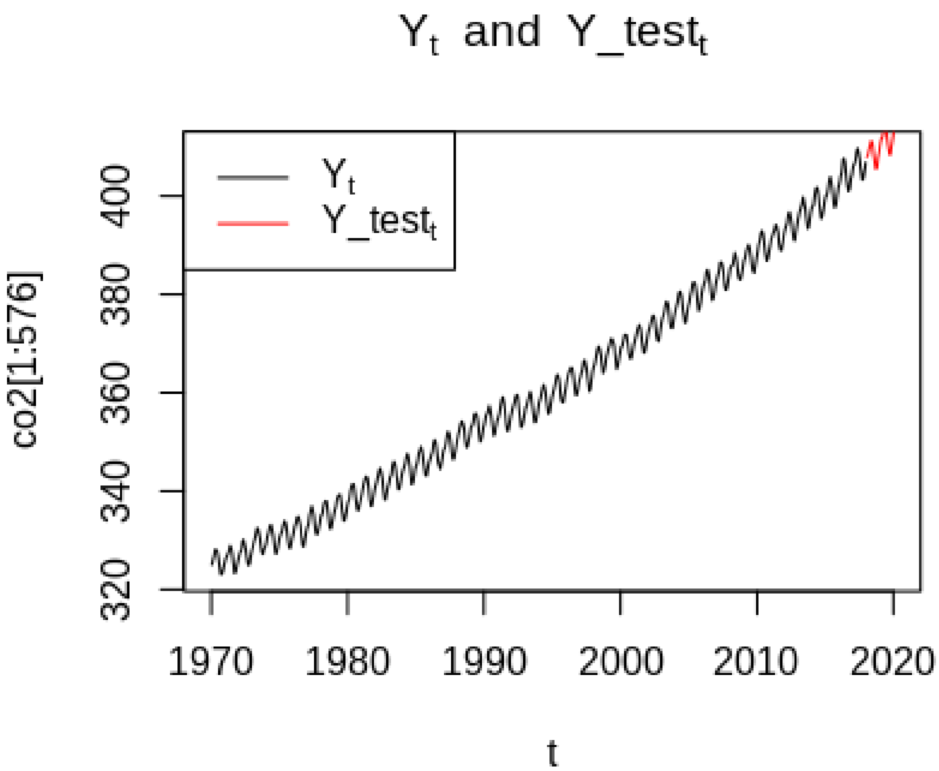

Figure 1: plotting data

2 Question 3.2

The autocorrelation function and the partial autocorrelation function of the CO2 concentration are plotted. Note that "S" denote Sampled. According to the book page 149 "In practical situations one will at most calculate the autocorrelation up to lag k = N/4 = 576/4 which is 144, hence for better vizualisation of plots, the lag max is 50 < 144.
Note that N is considered to be 576 because N is the number of observations that will be used to find models, so it does not include the last 24 observations for testing.

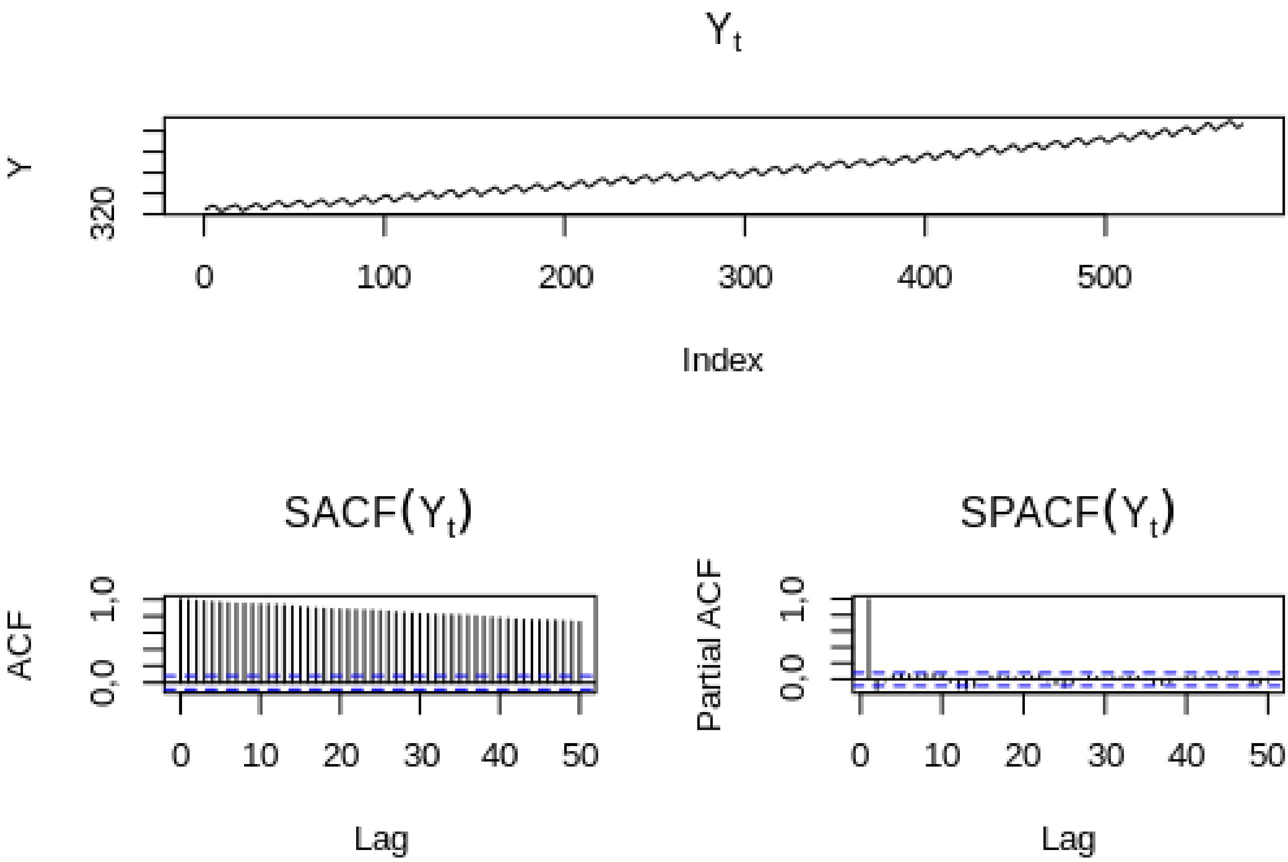

Figure 2: ACF and PACF for Y_t

It can be seen that the process is non-stationary. This is because SACF goes very slowly to zero.
Hence, a differentiation should be made. A first order differentiation is made first:

$$H_t = (1 - B)Y_t = \nabla Y_t \tag{1}$$

The ACF and PACF of the process H_t

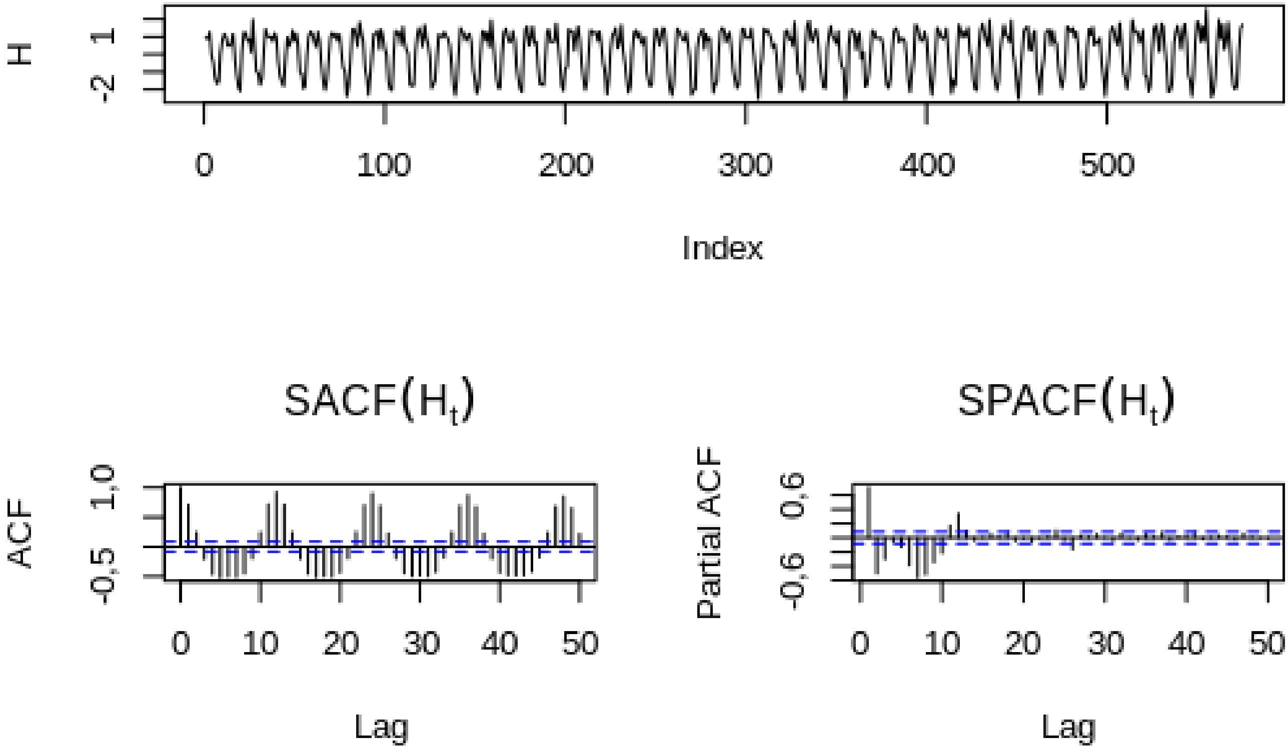

Figure 3: ACF and PACF for H_t

It can be seen from SACF of H_t that the problem concluded from SACF of Y_t is solved: there is no slowly decay towards zero. It can also be seen the sine function in SACF of H_t with peaks at multiples of 12. This denotes the seasonality of Y_t with 1 year. The sine function of H_t does not decrease exponentially towards zero, hence H_t is also non-stationary. A seasonal first order differentiation with period 12 is made on H_t:

$$Z_t = \nabla_{12} H_t = \nabla_{12} \nabla Y_t = (1 - B^{12})(1 - B) Y_t \tag{2}$$

The ACF and PACF of the process Z_t are plotted:

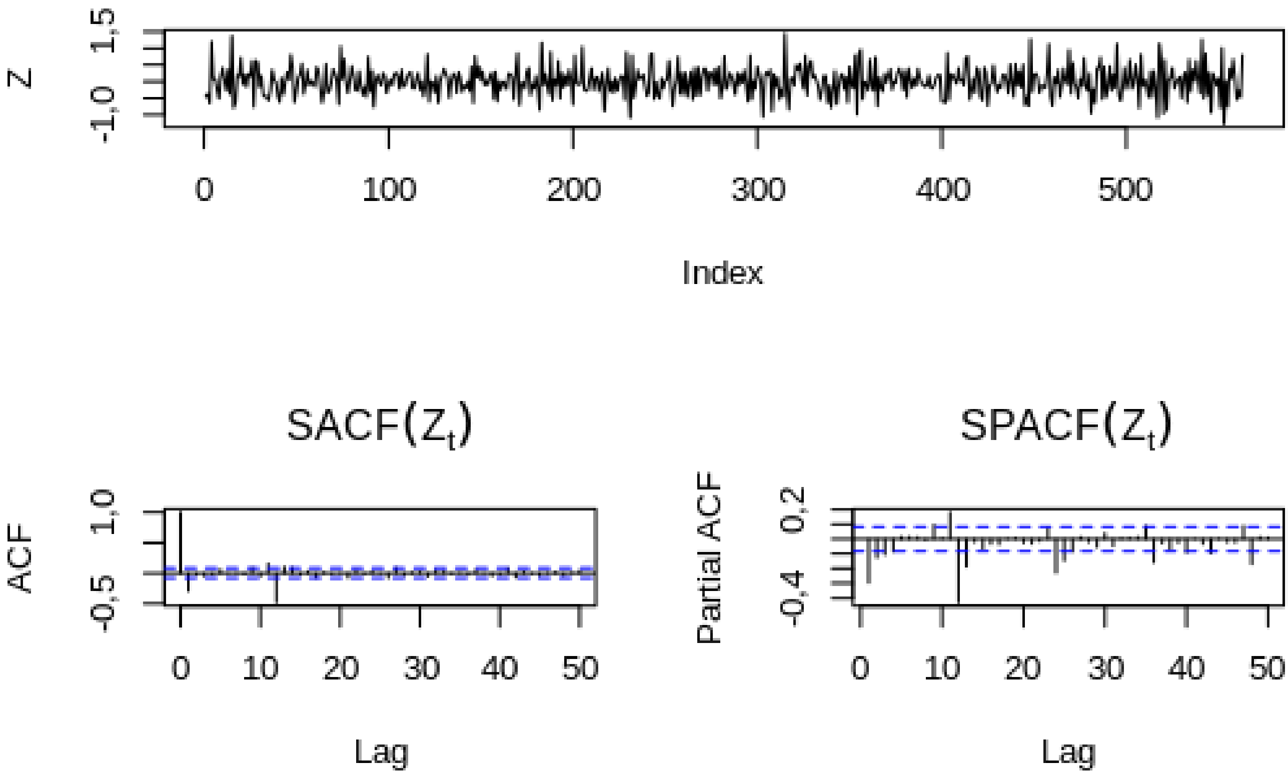

Figure 4: ACF and PACF for Z_t

The process Z_t is stationary because its SACF decreases quickly to zero. This process seems to be promising. In practical situations, as explained in the book page 153, d is selected as the lowest order of differencing for with SACF decreases sufficiently rapid towards 0. But why not try d=2 and see the results given. The model selection in the next part will compare these models. Furthermore, it is interesting to compare the prediction given by the different models having different d. This process is denoted:

$$X_t = \nabla Z_t = \nabla\nabla_{12}\nabla Y_t = \nabla^2\nabla_{12}Y_t \tag{3}$$

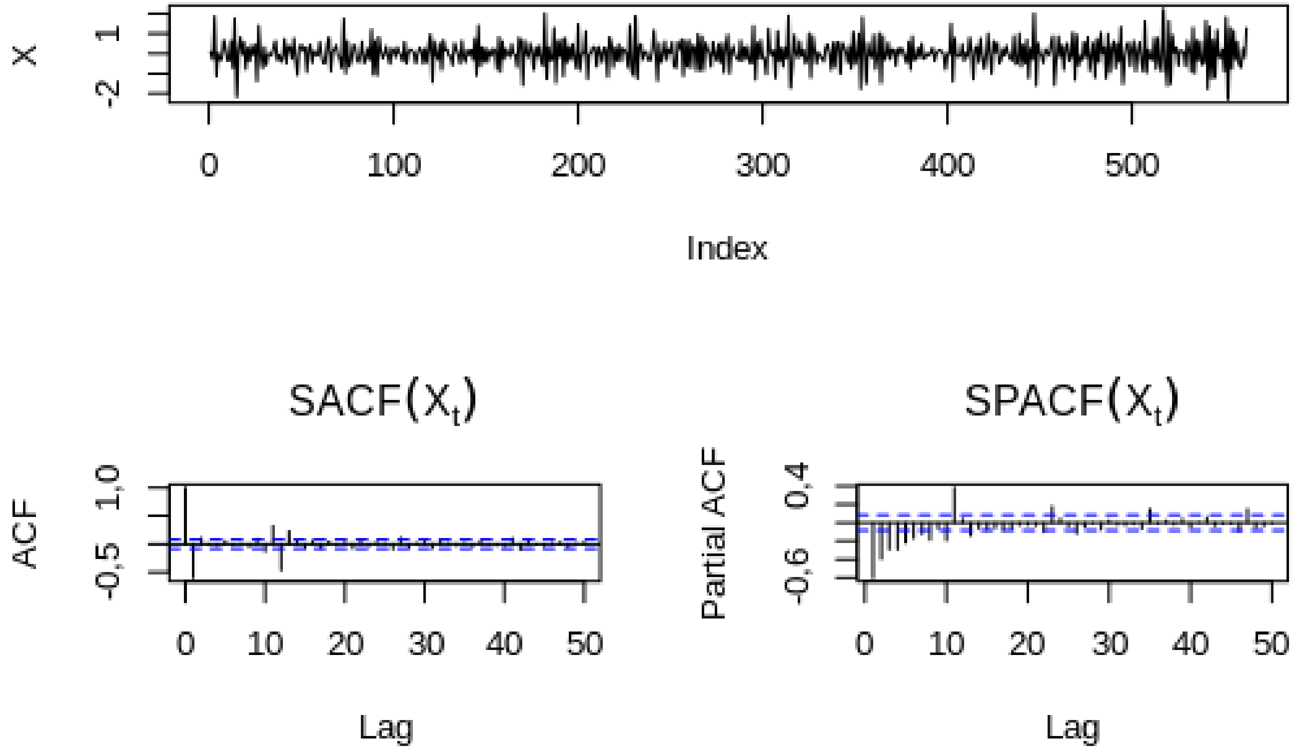

Figure 5: ACF and PACF for X_t

The process X_t is stationary because its SACF decreases quickly to zero.

3 Question 3.3

The ARIMA model with seasonal behaviour can be formulated as:

$$\phi(B)\Phi(B^s)\nabla^d\nabla_s^D(Y_t-\mu)=\theta(B)\Theta(B^s)\epsilon_t \tag{4}$$

Another notation for ARIMA model in Eq. 4 is $(p,d,q)\times(P,D,Q)_s$.
Note that the processes that will be used here are all univariate

The approach that that will be used for identifying the best model is :
1)Refer to table 6.1 page 155 to guess the model order
2)Test the significance of the estimated parameters: test whether a parameter θ_i is zero (6.5.2.2 p172):
H_0: $\theta_i=0$ against H_1: $\theta_i\neq 0$

Test statistic $T = \frac{\hat{\theta}_i}{\hat{\sigma}_{\hat{\theta}_i}}$ which follows a t-distribution under H_0 with $N - p - q$ degrees of freedom. N is the number of observation used for the estimation of θ_i.
H_0 is accepted if $2(1 - P\{T \leq |t|\}) > \alpha = 0.05$, else H_0 is rejected. If H_0 is accepted then $\theta_i = 0$ i.e θ_i is removed from the model.
3)Stopping Criteria
-Akaike Information Criterion (AIC), the method uses the maximum likelihood: AIC= -2log(max.likelihood) $+2n_i$. The most adequate model is the one that minimize AIC. AIC is used especially for large N. In this case one can say that N=576 is relatively large. Note that BIC(Bayesian Information Criterion) can also be used.
Note that AIC is predefined in ARIMA function in R. It is believed that AIC in R include already n_i the number of estimated parameter. If not, the function TONAMEHERE used for calculating the final Stopping Criteria can also return n_i.
-Ljung-Box test for residual analysis:
H_0: The residuals are independently distributed
against
H_1: The residuals are not independently distributed i.e they are correlated.
The p_value has to be above $\alpha = 0.05$ for each lag. It is noted $pval_{lag_k}$ Hence, it is logical to use an indicator function I.
I is defined by: I $= \infty$ if $pval_{lag_k} < 0.05$ and I=0 if $pval_{lag_i} > 0.05$.
The choice of this indicator function is justified by the definition of the final Stopping Criteria:
Final Stopping Criteria = AIC + I
This way, it is made sure that the most adequate model is the one that minimize the Stopping Criteria. Specifically, the idea is that if two models have the same AIC, then according the most adequate model is the one having more p_values > 0.05.
Note that I is proportional to the number of p_values< 0.05 .
Yet this should be done with precaution. For instance, consider model A having very low AIC with some p_values<0.05 , model B having a high AIC and many p_values>0.05 . If both models have the same Final Stopping Criteria, which is the best model? even worse, what if model B has a lower Final Stopping Criteria ? One should be careful when interpreting results and should see AIC alone and then the Final Stopping Criteria
This is confusing that is why it is worth saying it. But in this assignment it won't cause a problem. In general,if there is any contradiction, one would use AIC only.

4 Question 3.4

4.1 Model order

According to (6.23) from the book page 153, d = 1,or d=2, and according to (5.6.2) page 132 D=1 and s=12
The models having d=1 correspond to the use of Z_t.
First the seasonal structure is analyzed:

-SACF(Z_t) is equal to zero after lag12
-SPACF(Z_t) has an exponential decay after lag12.
-This suggest probably MA(1) with period=12
Secondly, one season is analyzed:
-SACF(Z_t) has an exponential decay after lag 0 or lag 1.
-SPACF(Z_t) has an exponential decay after lag 1.
-This suggest ARMA(1,1) or ARMA(2,1).
Finally, there are 2 models:
-model1: $(p = 1, d = 1, q = 1) \times (P = 0, D = 1, Q = 1)_12$.
-model2:$(p = 2, d = 1, q = 1) \times (P = 0, D = 1, Q = 1)_12$

The models having d=2 correspond to the use of X_t.
First the seasonal structure is analyzed:
-SACF(X_t) is equal to zero after lag12
-SPACF(X_t) has an exponential decay after lag12.
-This suggest probably MA(1) with period=12
Secondly, one season is analyzed:
-SACF(Z_t) has an exponential decay after lag 0 or lag 1.
-SPACF(Z_t) has an exponential decay after lag 1 or lag 0.
-This suggest ARMA(1,1) or ARMA(2,1).
Finally, there are 2 models:
-model3: $(p = 1, d = 2, q = 1) \times (P = 0, D = 1, Q = 1)_12$.
-model4:$(p = 1, d = 2, q = 2) \times (P = 0, D = 1, Q = 1)_12$

Note that these observations are not probably the same for another observer. This method is basically qualitative and open.

Note that a model5 will also be used later, which has the same $(p = 1, d = 2, q = 2) \times (P = 0, D = 1, Q = 1)_12$ as model 4. The difference is that in model5, the regressor used to estimate xreg is different. (the difference can be seen in the code in appendix)

4.2 Test whether a parameter is zero

Using the function "test_parameter_zero" in R, it is concluded that all the five models have no parameters to set to zero. All the hypothesis for all parameters were rejected.

4.3 Ljung-box test on residuals:

Using the function "my.tsdiag" in R, only model 3 could not pass the test. The other four models have all their p_values$>$ 0.05 for each lag.

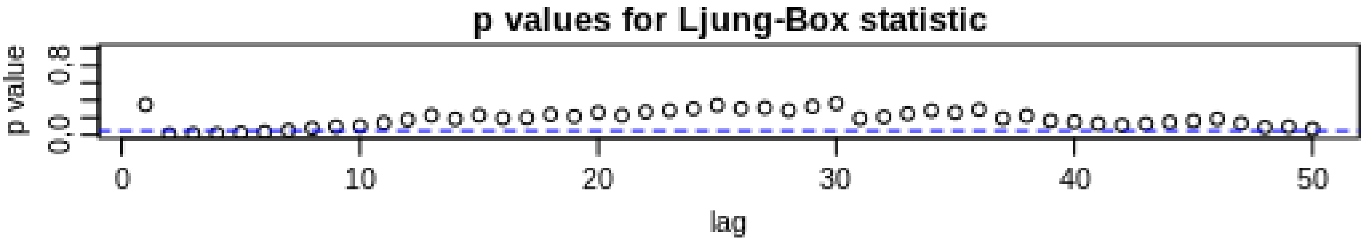

Figure 6: Ljung-box test for model3

4.4 Stopping Criteria

- "qqPlot in R:" It can be seen from "qqPlot" defined in R, that all the models seem to have residuals as white noise because the values in "qqPlot" follow the theoretical line. So this does not remove any model.
-Test in the autocorrelation function: (page 175) In all models, ACF is equal to zero after lag0 which is the case for white noise.
-Test in the autocorrelation function: In all models, PACF is equal to zero after lag0 which is the case for white noise.
-Ljung-box test: all the models passed the test as described before. All their p_values are above 0.05.
The resulats are similar in the four models in the previous tests. Based on these, one can not exclude any model.

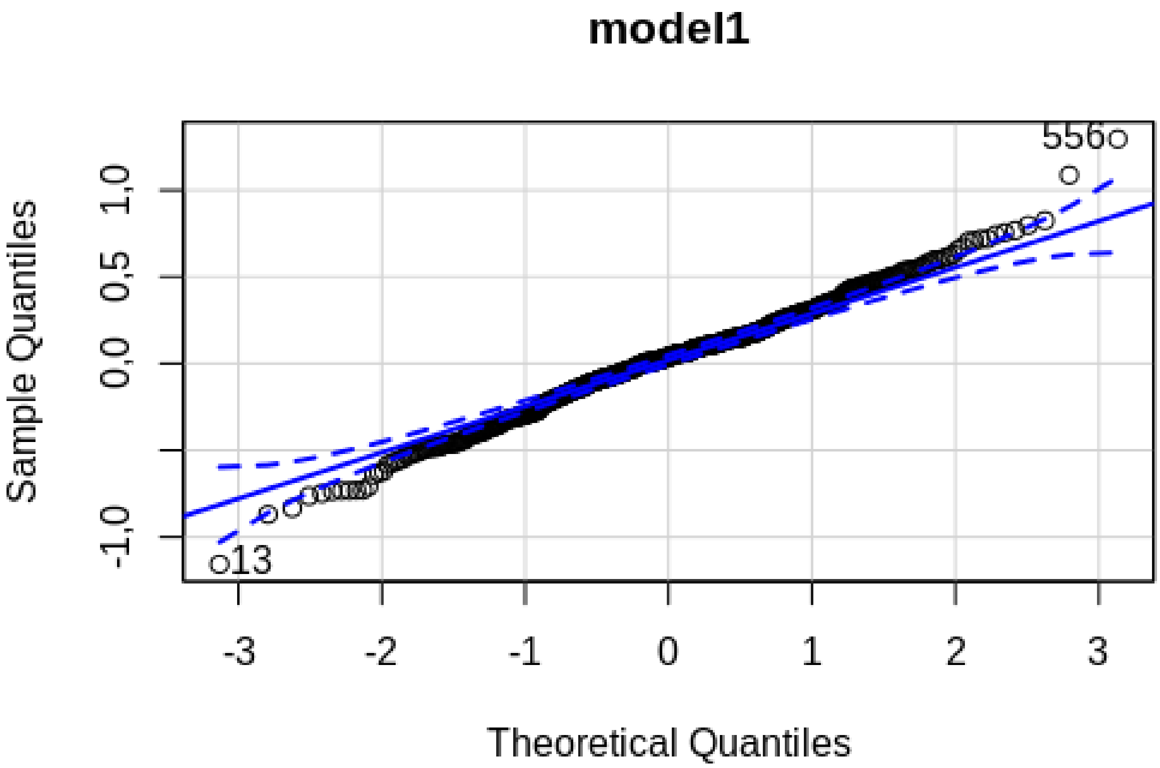
model1
Sample Quantiles
Theoretical Quantiles
-1,0
0,0
0,5
1,0
-3
-2
-1
0
1
2
3
556
13

Figure 7

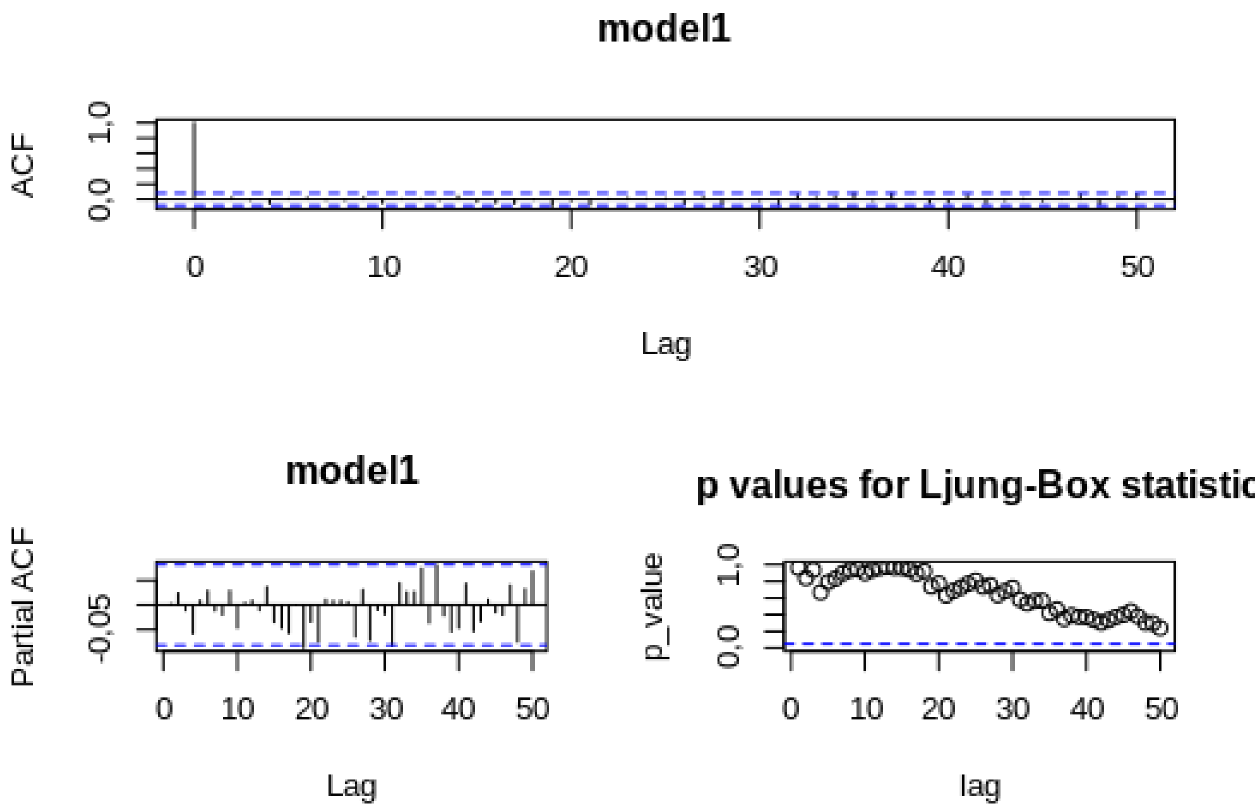
model1
ACF
0,0 1,0
0 10 20 30 40 50
Lag
model1
Partial ACF
-0,05
0 10 20 30 40 50
Lag
p values for Ljung-Box statistic
p_value
0,0 1,0
0 10 20 30 40 50
lag

Figure 8

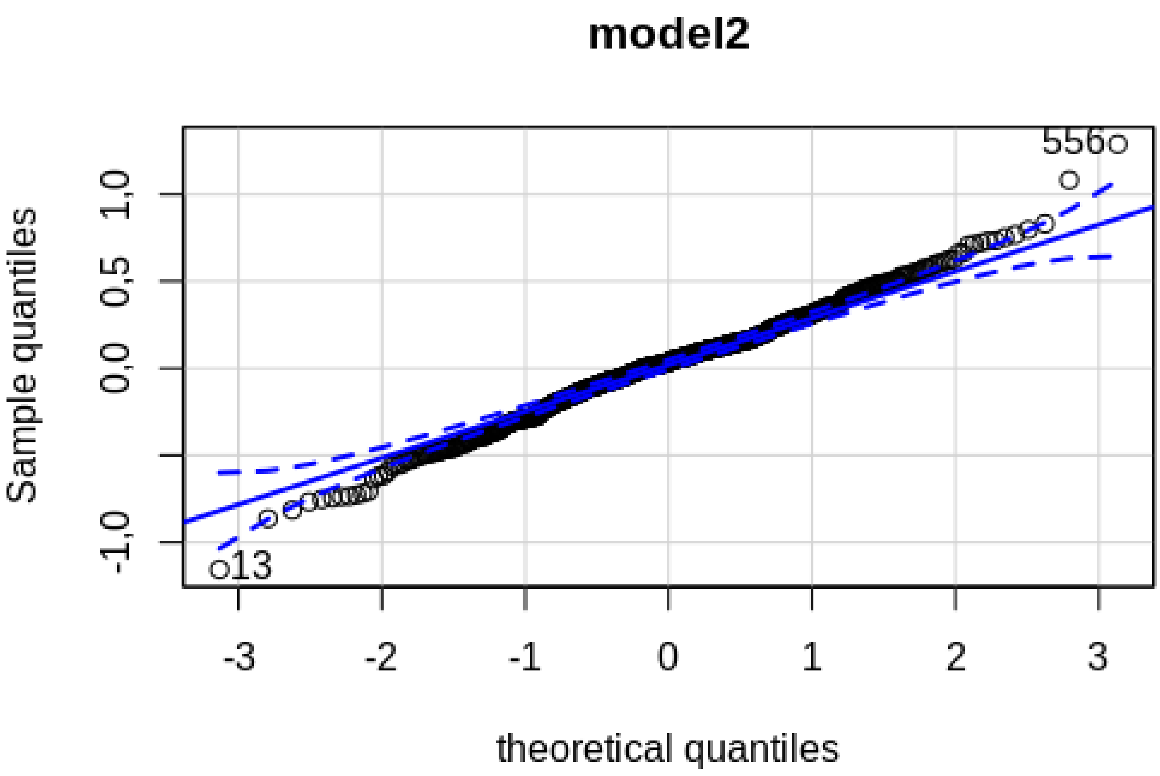
model2
Sample quantiles
theoretical quantiles
-1,0
0,0
0,5
1,0
-3
-2
-1
0
1
2
3
556
13

Figure 9

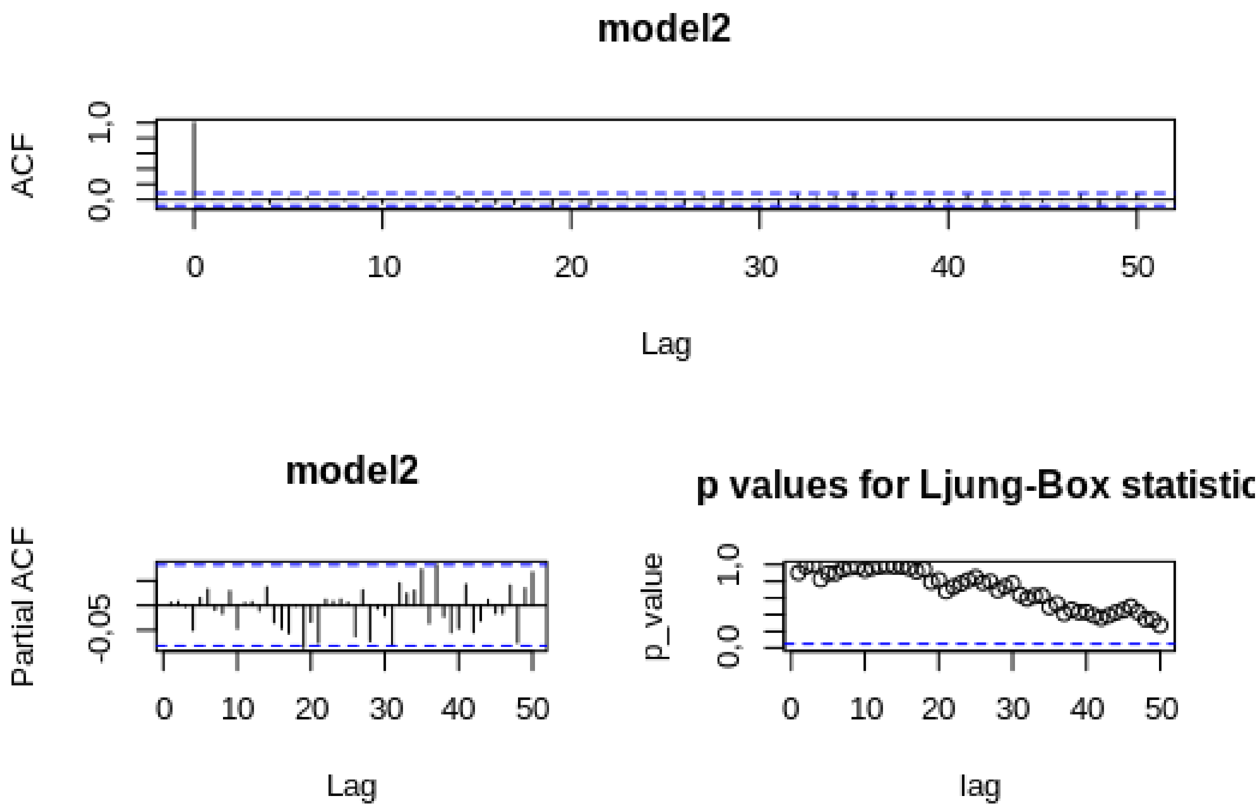
model2
ACF
0,0 1,0
0 10 20 30 40 50
Lag
model2
Partial ACF
-0,05
0 10 20 30 40 50
Lag
p values for Ljung-Box statistic
p_value
0,0 1,0
0 10 20 30 40 50
lag

Figure 10

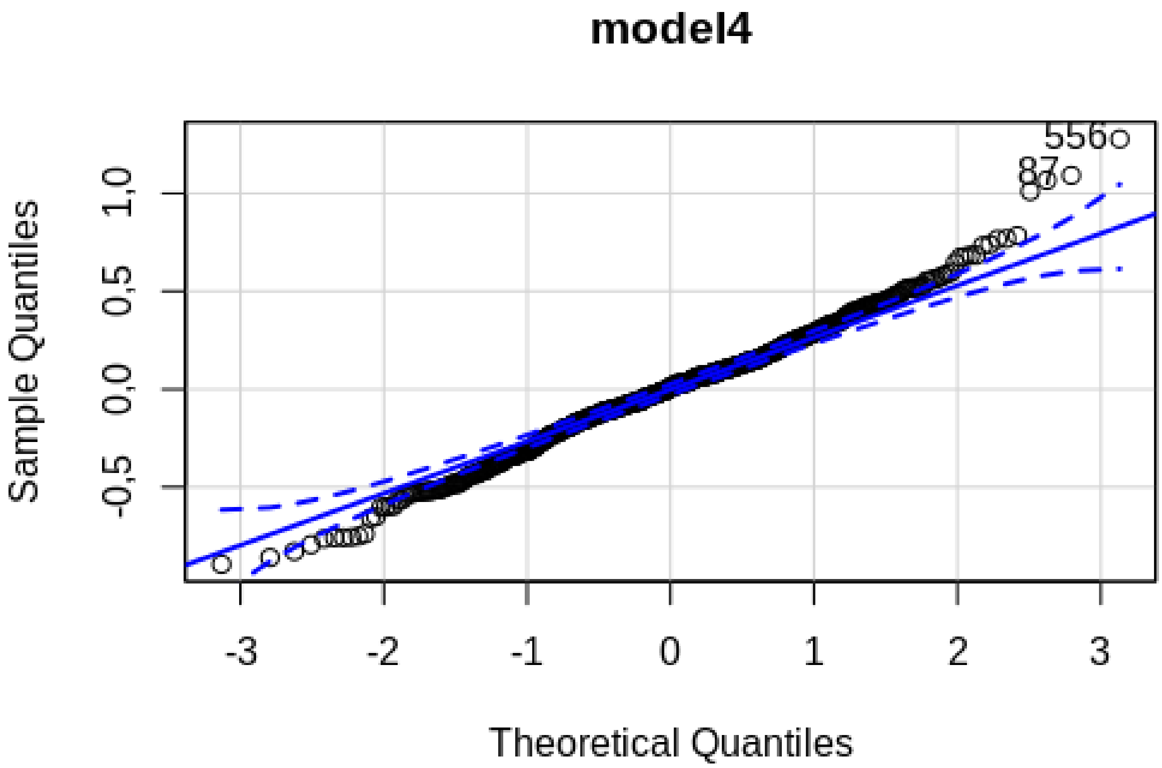
model4
Sample Quantiles
Theoretical Quantiles
556
87
-0,5
0,0
0,5
1,0
-3
-2
-1
0
1
2
3

Figure 11

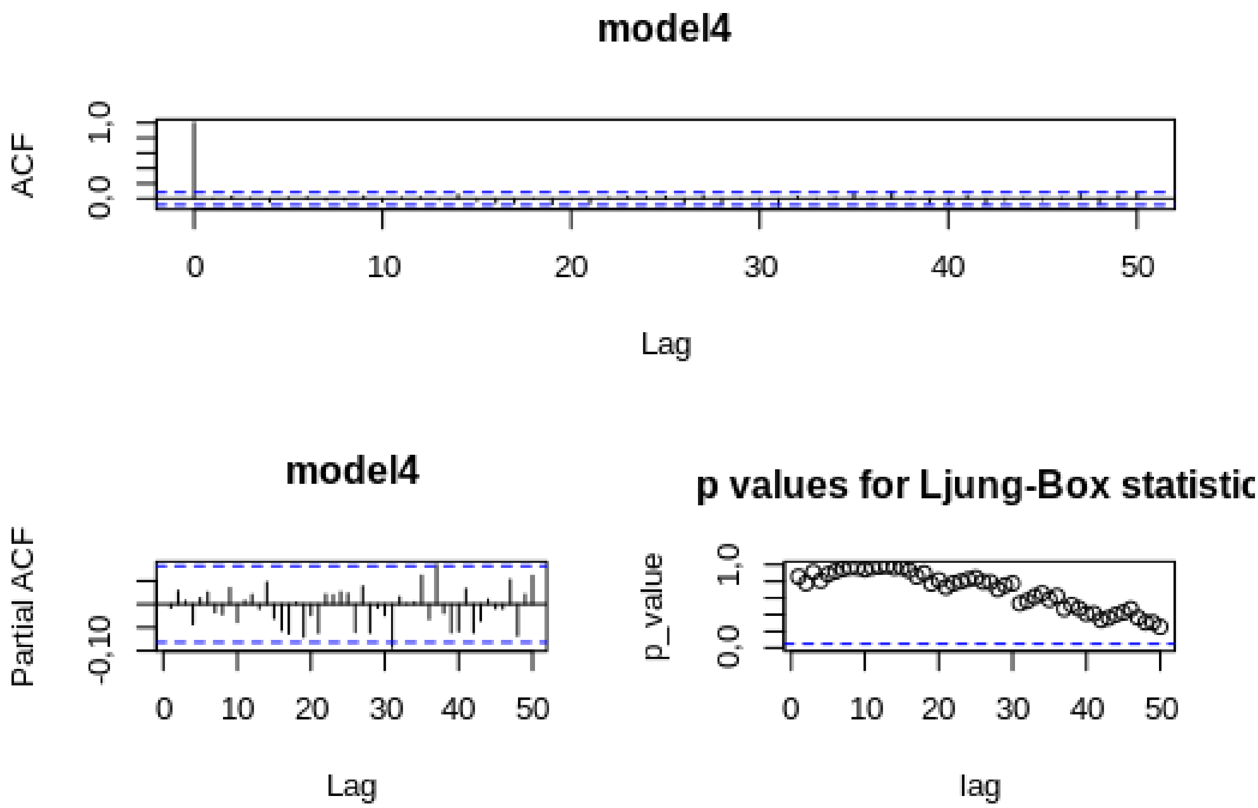
model4
ACF
0,0
1,0
0
10
20
30
40
50
Lag
model4
Partial ACF
-0,10
0
10
20
30
40
50
Lag
p values for Ljung-Box statistic
p_value
0,0
1,0
0
10
20
30
40
50
lag

Figure 12

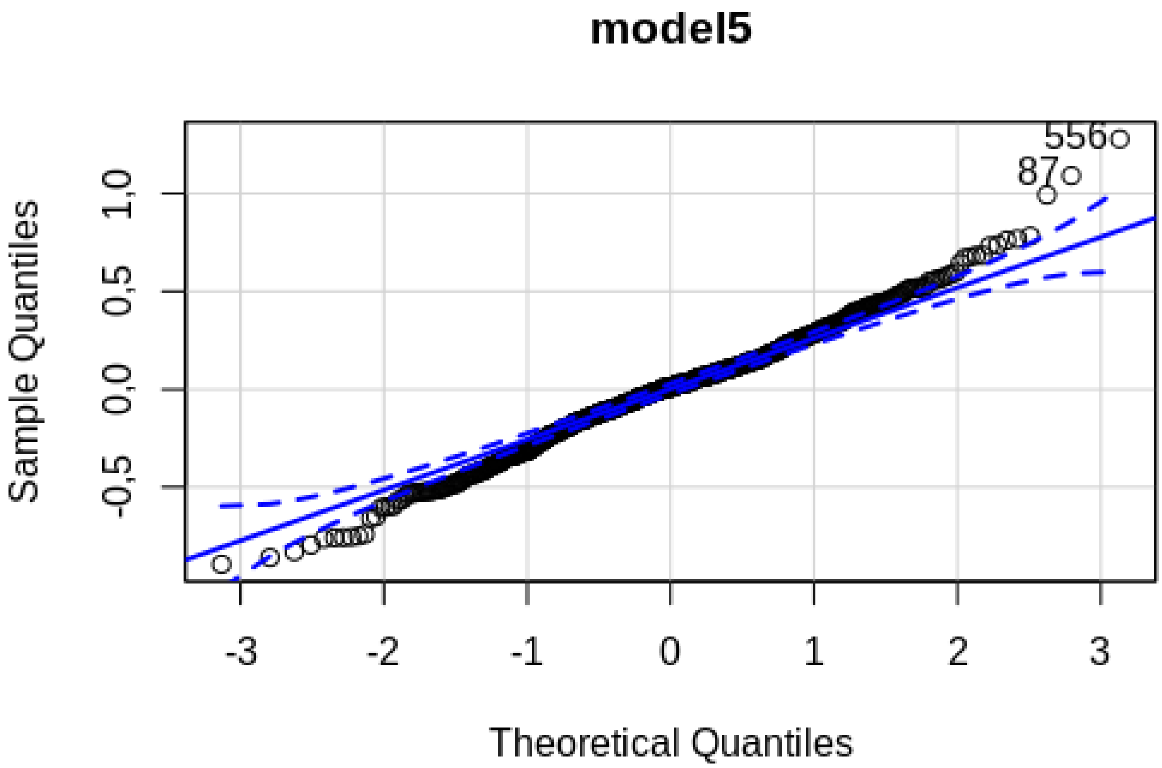
model5
Sample Quantiles
Theoretical Quantiles
556
87
-0,5
0,0
0,5
1,0
-3
-2
-1
0
1
2
3

Figure 13

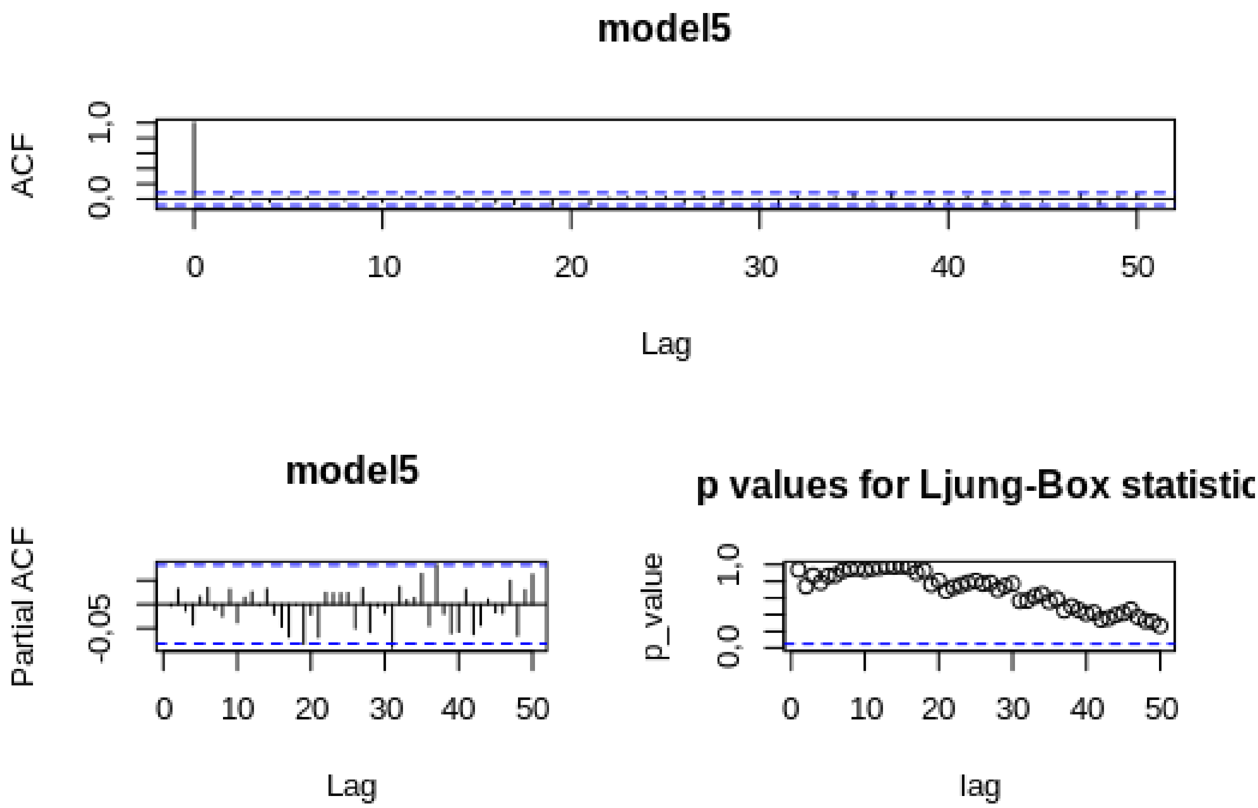

Figure 14

The table below summarize the Stopping Criteria for the models:

models	AIC	I	Final Stopping Criteria
model1	310,0384	0	310,0384
model2	311,6129	0	311,6129
model4	337,0638	0	337,0638
model5	325,0948	0	325,0948

Models 1,2,4 use the regressor xreg=1:576. Model5 uses as regressor xreg=time column in the data. Since the most adequate model is the model that minimize AIC, the chosen model is model1 even if there is no significant difference with AIC of model2.
It is interressant to see how model5 predicts since it does not have the same regressor. For this report, it is useful to compare the predictions results. In general, if one has to choose between model1 and model5, probably it will be model1 because it has the lowest AIC and lowest Final Stopping Criteria.

5 Question 3.5

For forcasting, the library(forcast) in R and the function "forecast" are used. The table below summarize the results
note that the observation number 576 is in 2017, and the observation 577 is in 2018, so the first prediction 1 month ahead is considered to be the prediction for the observation number 577. This clarification is stated because in the assignment it is said" You should not use the observations for years 2018 and 2019 (Last 24 observations)" and the last observation is in 2020.

months ahead	model1			model5			obersvation Y
	prediction	lower	upper	prediction	lower	upper	
1	407,9512	407,3313	408,5711	407,9817	407,3709	408,5925	407,96
2	408,6878	407,9473	409,4283	408,7437	408,0138	409,4736	408,32
6	410,9737	409,9433	412,0041	411,0932	410,0866	412,0999	410,79
12	408,9899	407,6411	410,3387	409,2144	407,8974	410,5314	409,07
24	411,2378	409,2893	413,1863	411,7015	409,7918	413,6112	411,76

It can be said that both models generalize well. The observations are always in the confidence.

A final idea to compare model1 and model5 is through the Sum Squared Error (SSE) of the prediction errors. The best model is the model that minimize SSE. The SSE is calculated on all the observations used for testing.

-model1: SSE=6,172915
-Model5: SSE=3,786148
It can be said according to SSE that model5 is better than model1.
One should not forget that in the previous section 4.4 it has been said that AIC for model 1 < AIC for model5.
Due to this the two models will be used for the next question.

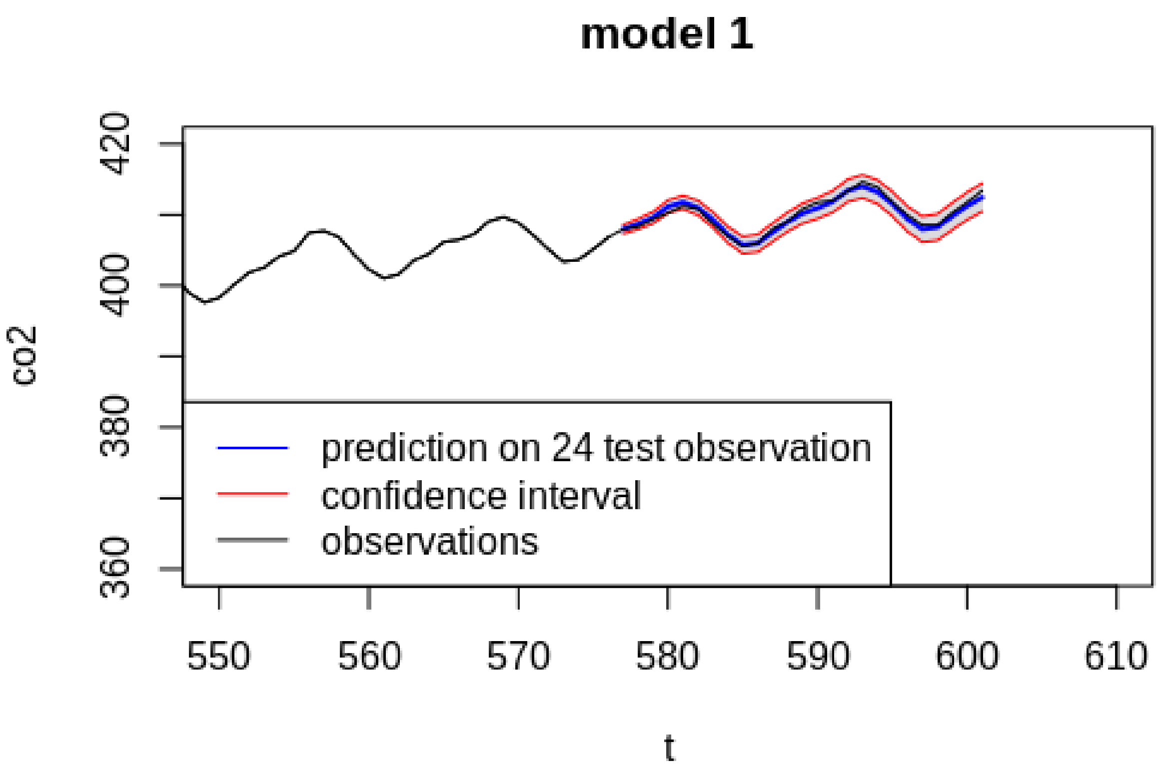
model 1
co2
420
400
380
360
prediction on 24 test observation
confidence interval
observations
550
560
570
580
590
600
610
t

Figure 15

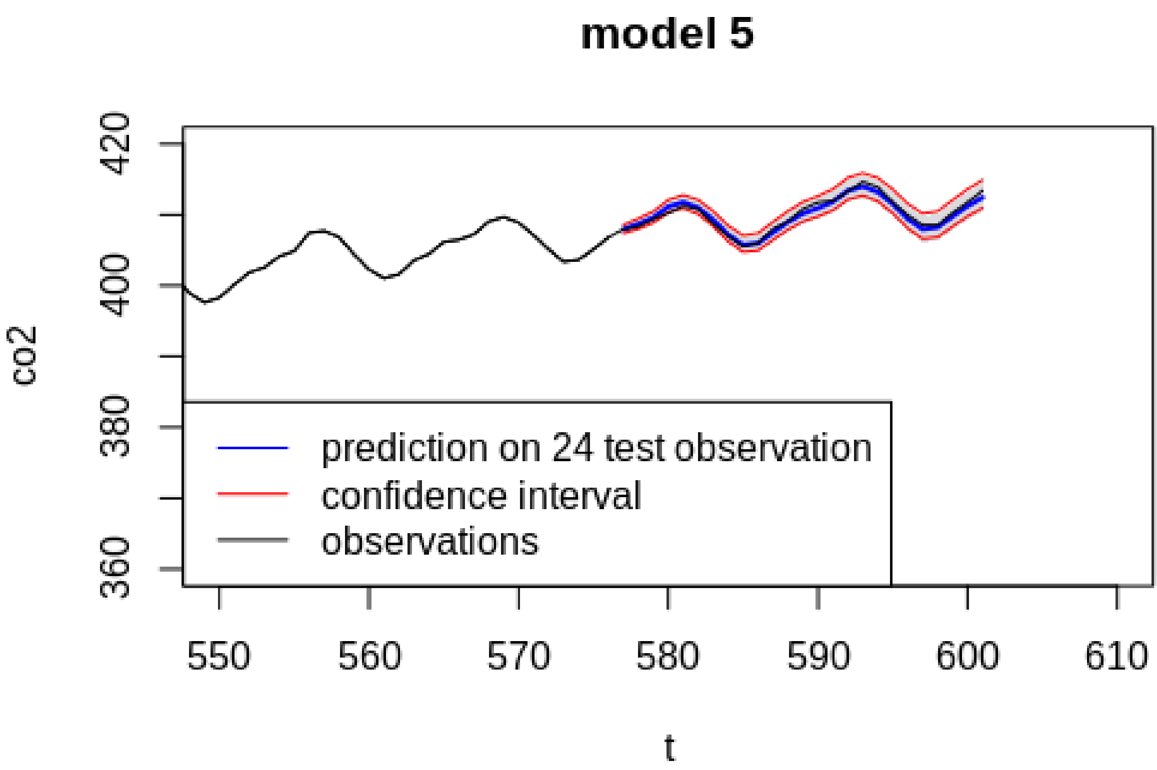

Figure 16

6 Question 3.6

The two processes are still trained only on the first 576 observations.
For model1:
The first time model1 reaches 460ppm is at t=844 which is equivalent to year 2040.54 .
The prediction was stopped at that value, which is exactly:
460,6040 with the confidence interval= [445.6792 ; 475.5287]
One can easily compute the model in order to have the lower value in the confidence interval reach 460 for the first time, but it is believed that this is not in the essence of the question.

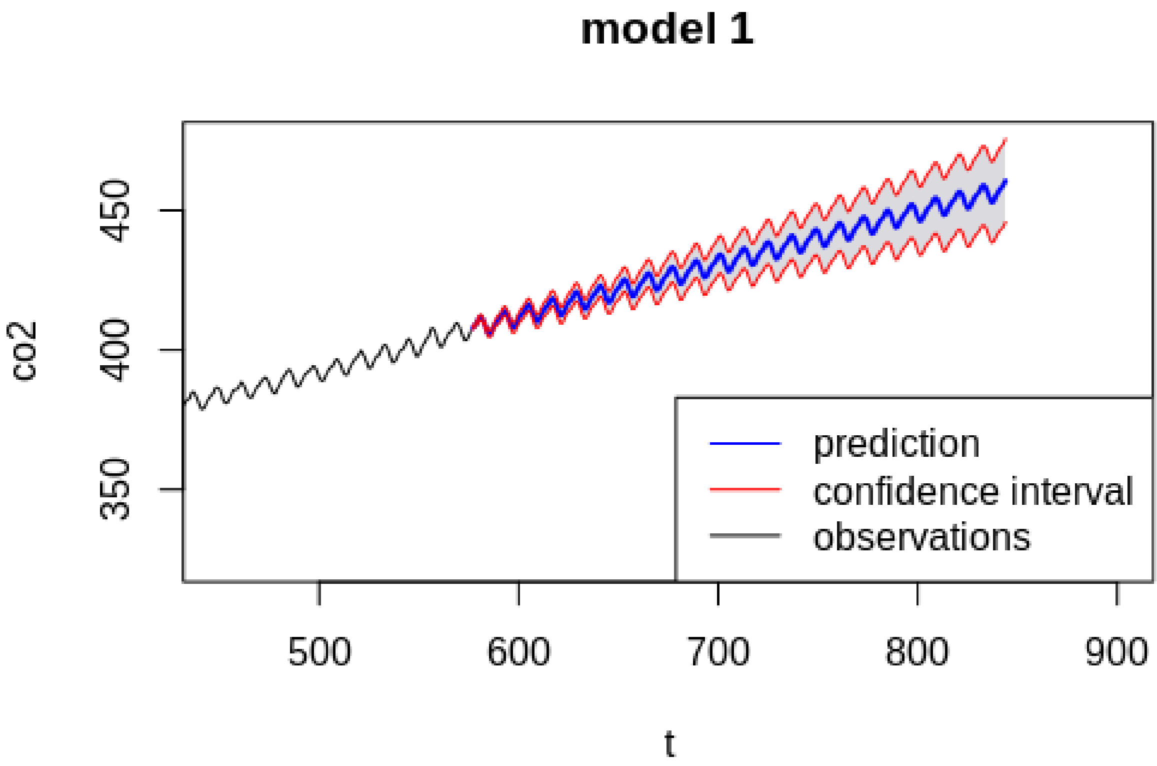

Figure 17

For model5:
The first time model1 reaches 460ppm is at year 2036.504 which correspond to t=796 .
The prediction was stopped at that value, which is exactly:
460.0428 with the confidence interval= [446.7369 ; 473.3488]
One can easily compute the model in order to have the lower value in the confidence interval reach 460 for the first time, but it is believed that this is not in the essence of the question.

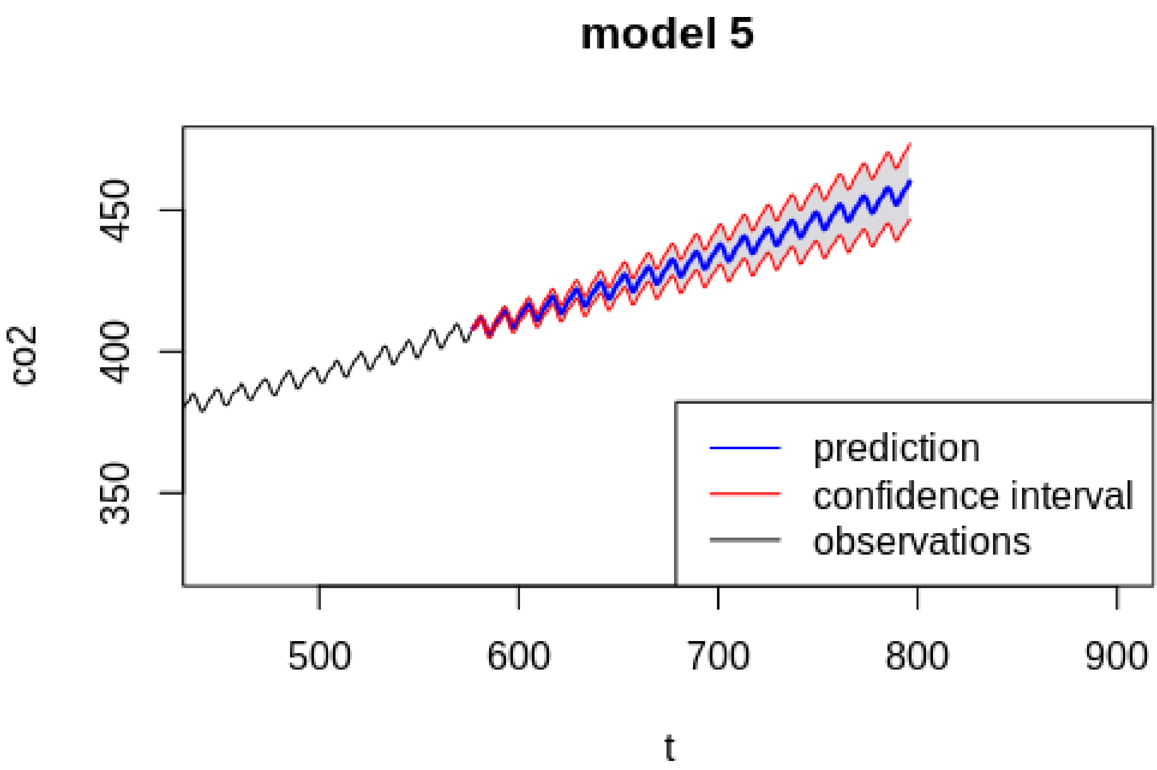

Figure 18

7 Code

```
rm(list=ls())
library(forecast)
library(car)

#Question1
    ################################################################################

data = read.table("/home/ghassen97/Desktop/S8/time series analysis/
    assignment/assignment 3/2020_A3_co2.txt", header = TRUE)
co2 = data$co2
t   = data$time
#test with last 24 observations, length(t)=601

plot(t[1:576], co2[1:576],xlim = c(1970,2020), type='l', main = expression(
    paste(Y[t], "  and  ", Y_test[t])), xlab = expression(t), col = 'black')
```

```
lines(t[577:601], co2[577:601], type='l', col='red')
Y = co2[1:576]
legend("topleft", legend = c(expression(Y[t]), expression(Y_test[t])), col =
    c('black', 'red'),  lty=1:1)

#Question2
    ####################################################################################

#attach(mtcars)
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
plot(Y, type = 'l', main = expression(Y[t]))
acf(Y,lag.max = 50, main = expression(SACF(Y[t])))
pacf(Y,lag.max = 50,  main = expression(SPACF(Y[t])))

#differencing Y_t
p= 1
H = diff(Y, lag = p, differences = 1)

layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
plot(H, type = 'l')
acf(H,lag.max = 50, main = expression(SACF(H[t])))
pacf(H,lag.max = 50,  main = expression(SPACF(H[t])))

#differencing H_t withperiod p=12
p = 12
Z = diff(H, lag = p, differences = 1)
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
plot(Z, type = 'l')
acf = acf(Z,lag.max=50,main = expression(SACF(Z[t])))
pacf =pacf(Z,lag.max=50, main = expression(SPACF(Z[t])))

p = 1
X = diff(Z, lag = p, differences = 1)
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
plot(X, type = 'l')
acf = acf(X, lag.max = 50, main = expression(SACF(X[t])))
pacf =pacf(X, lag.max = 50, main = expression(SPACF(X[t])))

#Question3
    ####################################################################################

#see report
#Question4
    ####################################################################################
```

Technical University of Denmark

```r
##definition of function that will be used######

#from R example week7:
my.tsdiag = function(model, nlag){
  oldpar = par(mfrow=c(4,1), mgp=c(2,0.7,0), mar=c(3,3,1.5,1))
  on.exit(par(oldpar))
  plot(fitted(model), col = 'blue', type = 'l')
  lines(model$x, col = 'red', type = 'l')
  model = model$residuals
  acf(model, lag.max = 50)
  pacf(model, lag.max = 50)

  pval = sapply(1:nlag, function(i) Box.test(model, i, type = "Ljung-Box")$p.value)
  plot(1L:nlag, pval, xlab = "lag", ylab = "p value", ylim = c(0,1), main = "p values for Ljung-Box statistic")
  abline(h = 0.05, lty = 2, col = "blue")

  return(pval)
}

# Test whether parameter=0
test_parameter_zero = function(model, a){
  TS = model$coef/(sqrt(abs(diag(model$var.coef))))
  pval = 2*(1-pt(abs(TS), df = (length(model$residuals) - length(model$coef)-1)))

  decision = symnum(pval[1:(length(pval))], decision(0, a, Inf), decision("Reject", "Accept"))
  print(pval[1:(length(pval))])
  print(decision)
}

# Stopping criteria function
stopping_criteria_final = function(model, nlag){
  pval = my.tsdiag(model, nlag)
  pval_lagk = sum((pval < 0.05))
  if(pval_lagk > 0)
  {
    I = 500 * pval_lagk
  }else{
    I = 0
  }
```

```
  ni = length(model$coef) - 1
  #stopping_criteria = model$aic + ni + I , this is the remark in report in
     Q3.3 regardind ni
  stopping_criteria = model$aic  + I
  print(I)
  return(stopping_criteria)
}


###
t = 1:(576)
#model_1:
model1 = Arima(Y[1:(576)], order = c(1,1,1), seasonal = list(order=c(0,1,1),
    period = 12), xreg = t)
#model_2
model2 = Arima(Y[1:(576)], order = c(2,1,1), seasonal = list(order=c(0,1,1),
    period = 12), xreg = t)

#model_3
model3 = Arima(Y[1:(576)], order = c(1,2,1), seasonal = list(order=c(0,1,1),
    period = 12), xreg = t)

#model_4:
model4 = Arima(Y[1:(576)], order = c(1,2,2), seasonal = list(order=c(0,1,1),
    period = 12), xreg = t)

#model_5= Model_4 with xreg =data$time[t]
model5 = Arima(Y[1:(576)], order = c(1,2,2), seasonal = list(order=c(0,1,1),
    period = 12), xreg = data$time[t])

#### test whether parameter=0 using test_parameter_zero
test_parameter_zero(model1,1)
test_parameter_zero(model2,1)
test_parameter_zero(model3,1)
test_parameter_zero(model4,1)
test_parameter_zero(model5,1)
####Ljung-box test using my.tsdiag
my.tsdiag(model1,50)
my.tsdiag(model2,50)
my.tsdiag(model3,50) #fail, has several p_values < 0.05
my.tsdiag(model4,50)
my.tsdiag(model5,50)
###stopping_criteria_final
stopping_criteria_final(model1,50)
```

```
stopping_criteria_final(model2,50)
stopping_criteria_final(model4,50)
stopping_criteria_final(model5,50)


nlag = 50
#model1
qqPlot(model1$residuals, main = "model1", ylab ="Sample Quantiles",xlab = "
   Theoretical Quantiles")
dat = model1
dat = dat$residuals
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
acf(dat, lag.max = 50, main = "model1")
pacf(dat, lag.max = 50, main = "model1")
pval = sapply(1:nlag, function(i) Box.test(dat, i, type = "Ljung-Box")$p.
   value) #from week7 R example
plot(1L:nlag, pval, xlab = "lag", ylab = "p_value", ylim = c(0,1), main = "p
    values for Ljung-Box statistic")
abline(h = 0.05, lty = 2, col = "blue")


#model2
qqPlot(model2$residuals, main = "model2", ylab ="Sample Quantiles",xlab = "
   Theoretical Quantiles")
dat = model2
dat = dat$residuals
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
acf(dat, lag.max = 50, main = "model2")
pacf(dat, lag.max = 50, main = "model2")
pval = sapply(1:nlag, function(i) Box.test(dat, i, type = "Ljung-Box")$p.
   value) #from week7 R example
plot(1L:nlag, pval, xlab = "lag", ylab = "p_value", ylim = c(0,1), main = "p
    values for Ljung-Box statistic")
abline(h = 0.05, lty = 2, col = "blue")


#model4
qqPlot(model4$residuals, main = "model4",ylab ="Sample Quantiles",xlab = "
   Theoretical Quantiles")
dat = model4
dat = dat$residuals
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
acf(dat, lag.max = 50, main = "model4")
pacf(dat, lag.max = 50, main = "model4")
```

```
pval = sapply(1:nlag, function(i) Box.test(dat, i, type = "Ljung-Box")$p.value) #from week7 R example
plot(1L:nlag, pval, xlab = "lag", ylab = "p_value", ylim = c(0,1), main = "p values for Ljung-Box statistic")
abline(h = 0.05, lty = 2, col = "blue")

#model5
qqPlot(model5$residuals, main = "model5", ylab ="Sample Quantiles",xlab = "Theoretical Quantiles")
dat = model5
dat = dat$residuals
layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
acf(dat, lag.max = 50, main = "model5")
pacf(dat, lag.max = 50, main = "model5")
pval = sapply(1:nlag, function(i) Box.test(dat, i, type = "Ljung-Box")$p.value) #from week7 R example
plot(1L:nlag, pval, xlab = "lag", ylab = "p_value", ylim = c(0,1), main = "p values for Ljung-Box statistic")
abline(h = 0.05, lty = 2, col = "blue")


#Question5 ##################################################################################

#model1
new_t= 577:601
forecast_model1<-forecast(model1,h=25,level=0.95, xreg = new_t)
lower_model1<-forecast_model1$lower
upper_model1<-forecast_model1$upper
plot(forecast_model1, main="model 1", xlim= c(550, 610),ylim=c(360,420),xlab = 't',ylab="co2")
lines(lower_model1, type = 'l', col='red' )
lines(upper_model1, type = 'l', col='red' )
lines(1:601, data$co2,type ='l',col='black')
legend("bottomleft", legend = c("prediction on 24 test observation","confidence interval","observations"), col = c('blue', 'red','black'), lty=1:1)

predictions_model1<-forecast_model1$mean
#SSE
SSE_model1 <- sum ( (data$co2[577:601] - predictions_model1)^2 )

#model5
```

```
xreg_model5<-data$time[new_t]
forecast_model5<-forecast(model5,h=25,level=0.95, xreg = xreg_model5)
lower_model5<-forecast_model5$lower
upper_model5<-forecast_model5$upper
plot(forecast_model1, main="model 5",xlim= c(550, 610),ylim=c(360,420),xlab = 't',ylab="co2")
lines(lower_model5, type = 'l', col='red' )
lines(upper_model5, type = 'l', col='red' )
lines(1:601, data$co2,type ='l',col='black')
legend("bottomleft", legend = c("prediction on 24 test observation","confidence interval","observations"), col = c('blue', 'red','black'), lty=1:1)

predictions_model5<-forecast_model5$mean
#SSE
SSE_model5 <- sum ( (data$co2[577:601] - predictions_model5)^2 )



#Question6 ####################################################################################

#model1
long<-844
new_t_long= 577:long
h_value<- long -577 + 1
forecast_model1_long<-forecast(model1,h=h_value,level=0.95, xreg = new_t_long)
# the first time model1 reaches 460 is in t=844
predictions_model1_long <- forecast_model1_long$mean
ind<-0
for (i in 1:length(predictions_model1_long) )
{
  if (predictions_model1_long[i]>460)
  {
    ind<-ind+1  #ind=1 making sure no 460 was missed
  }
}

lower_model1_long<-forecast_model1_long$lower
upper_model1_long<-forecast_model1_long$upper
plot(forecast_model1_long, main="model 1",xlim=c(450,900), xlab = 't',ylab="co2")
```

```
lines(lower_model1_long, type = 'l', col='red' )
lines(upper_model1_long, type = 'l', col='red' )
legend("bottomright", legend = c("prediction","confidence interval","
   observations"), col = c('blue', 'red','black'),  lty=1:1)



#model5
# first step, define the regressor
h_value=220
t = 577:(577+h_value-1) #t=577:796
delta_time = data$time[2:576] - data$time[1:575] # Delta
delta_time = delta_time[1:12]
delta_period_12 = (1:12)*delta_time
delta_period_12 = rep(delta_period_12, ceiling(h_value/12+1) )
delta_period_12 = delta_period_12[3:length(delta_period_12)]

year_start = 2018
xreg_model5_long = 0
for (i in seq(from = year_start, to= ceiling(2018+h_value/12), by = 1)) {
  years = rep(year_start, 12)
  xreg_model5_long = append(xreg_model5_long, years, length(xreg_model5_long
     ))
  year_start = year_start + 1
}
xreg_model5_long = xreg_model5_long[4:length(xreg_model5_long)]
xreg_model5_long = xreg_model5_long + delta_period_12
xreg_model5_long = xreg_model5_long[1:h_value]



h_value<-h_value

forecast_model5_long<-forecast(model5,h=h_value,level=0.95, xreg =
   xreg_model5_long)

predictions_model5_long <- forecast_model5_long$mean

ind<-0
for (i in 1:length(predictions_model5_long) )
{
  if (predictions_model5_long[i]>460)
  {
    ind<-ind+1  #indicator to make sure that 460 was reached once
```

```
  }
}


lower_model5_long<-forecast_model5_long$lower
upper_model5_long<-forecast_model5_long$upper
plot(forecast_model5_long, main="model 5",xlim=c(450,900), xlab = 't',ylab="
    co2")
lines(lower_model5_long, type = 'l', col='red' )
lines(upper_model5_long, type = 'l', col='red' )
#lines(1:601, data$co2,type ='l',col='black')
legend("bottomright", legend = c("prediction","confidence interval","
    observations"), col = c('blue', 'red','black'),  lty=1:1)
```

List of Figures