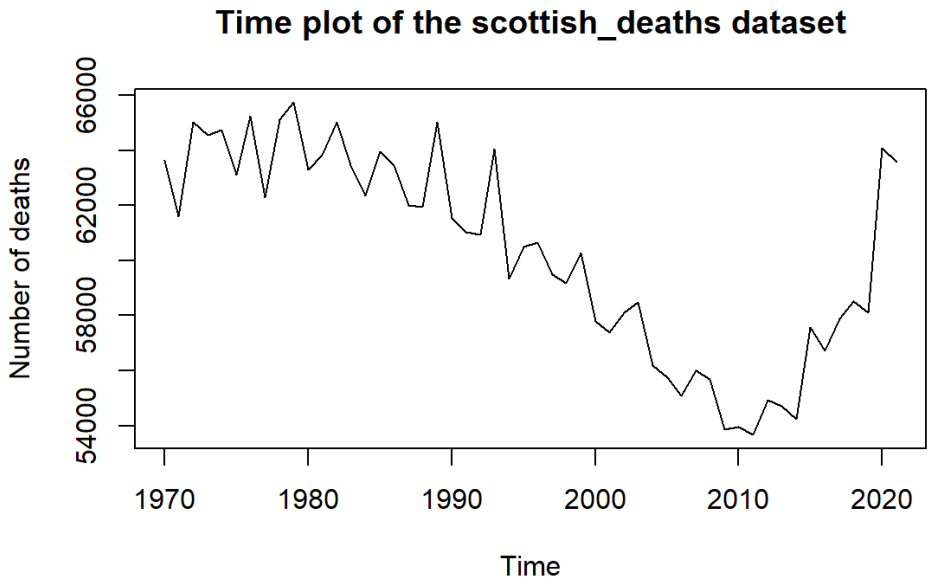
20217334-MATH3026-CW2

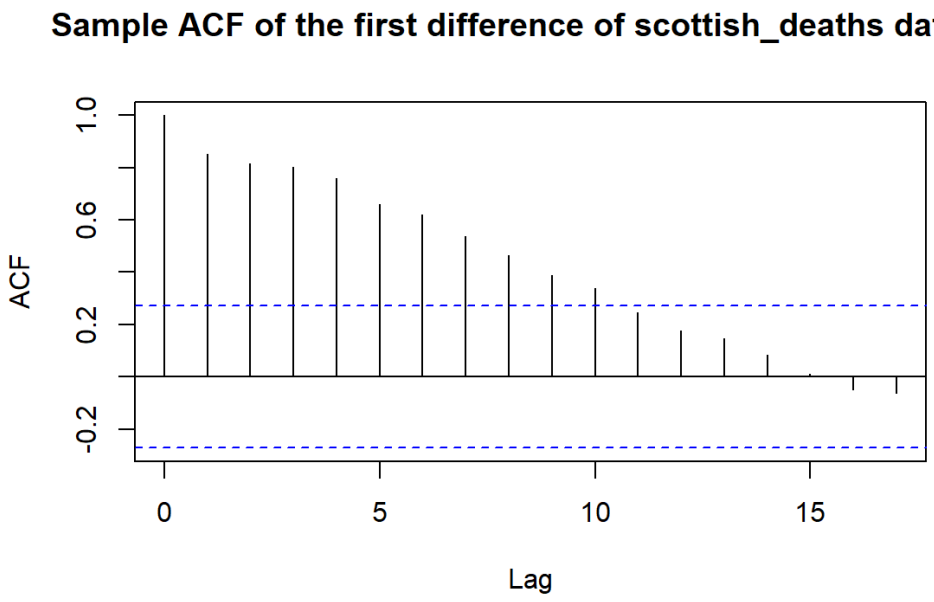
Zihang Yu

1. The scottish_deaths dataset.

First we look at the time plot of the time series.



From the time plot of the scottish deaths dataset, the time series doesn't seem to be stationary, and there is no obvious seasonal patterns.There is a decreasing trend in number of deaths from 1970 to 2010, and an increasing trend from 2010 to 2020.
Next we look at the sample ACF.



The sample ACF plot shows that the sample ACF values decrease steadily to zero as the lag increases from 1 to 10, and the ACF values seems to decrease linearly. Therefore, the time series might be non-stationary and we might consider differencing the time series.

We now prove the time series is not stationary using the Augmented Dickey-Fuller test.

```
##
##  Augmented Dickey-Fuller Test
##
## data:  scottish_deaths
## Dickey-Fuller = -1.955, Lag order = 0, p-value = 0.5925
## alternative hypothesis: stationary
```

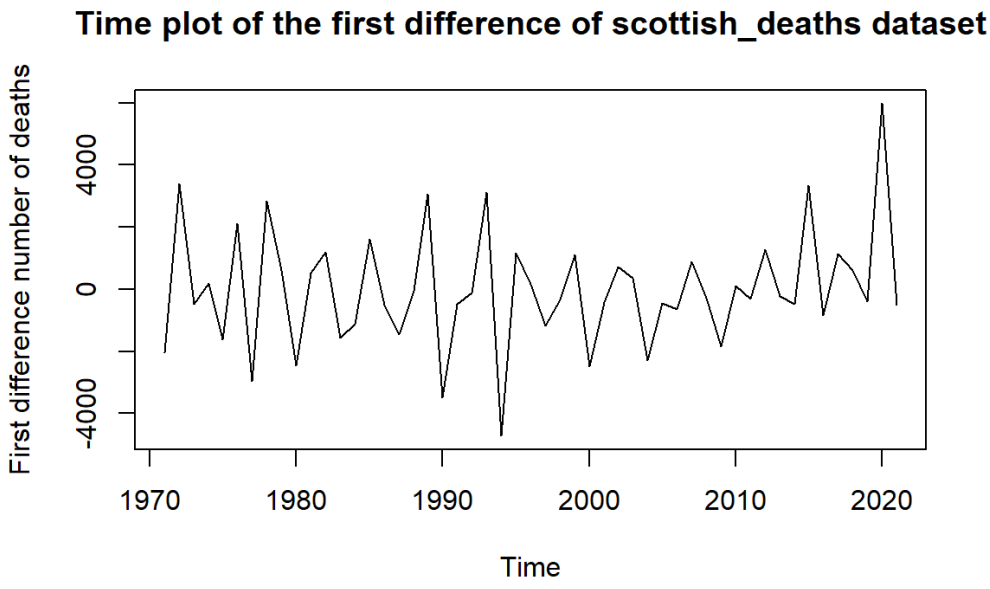
The p-value is 0.5925 which greater than 0.1 suggests that there is insufficient evidence to reject the null hypothesis which suppose the time series is non-stationary, therefore, the time series is non-stationary.

Since the time series is non-stationary, we try the first difference of the time series.

```
##
##  Augmented Dickey-Fuller Test
##
## data:  First_diff
## Dickey-Fuller = -10.685, Lag order = 0, p-value = 0.01
## alternative hypothesis: stationary
```

The p-value is 0.01, which indicates strong evidence against the null hypothesis. Therefore the first difference of the scottish deaths time series is stationary enough to do time series modelling.

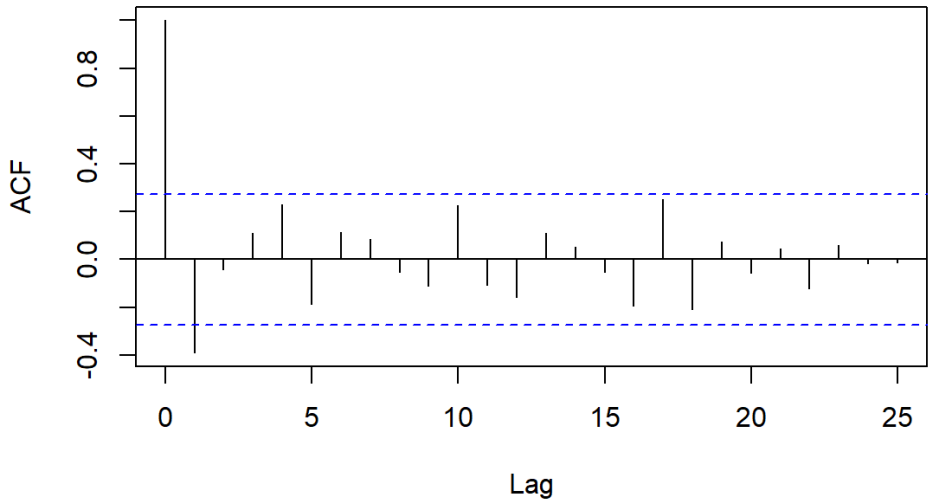
We now look at the time plot of the first difference of the time series.



The first difference of the scottish deaths time series appear to be stationary without obvious seasonal patterns and trend.

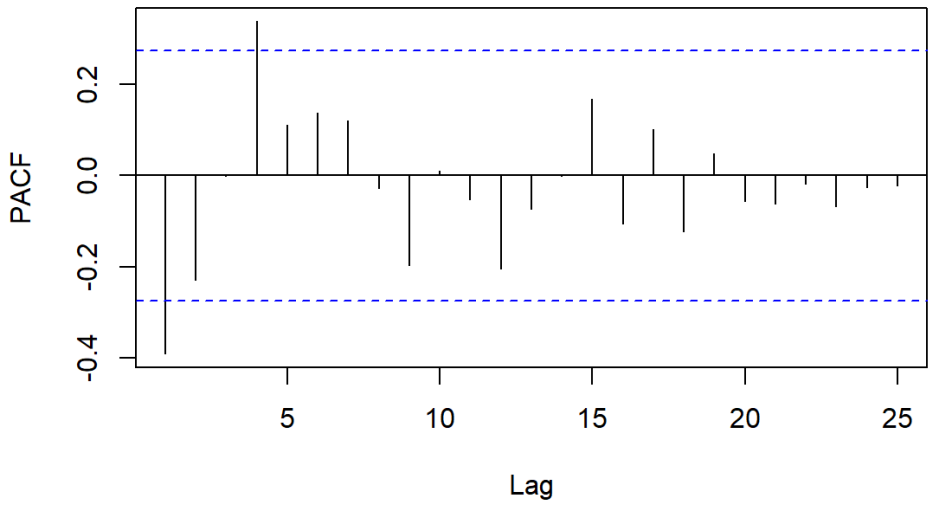
Next we look at the sample ACF and PACF of the first difference of the time series.

Sample ACF of the first difference of scottish_deaths datase



The sample ACF shows that there is a nagitive value exceed the significant boundary at lag 1, and the ACF cut off after lag 1. It suggests that an MA(1) term might be appropriate for this time series.

Sample PACF of the first difference of scottish_deaths datas



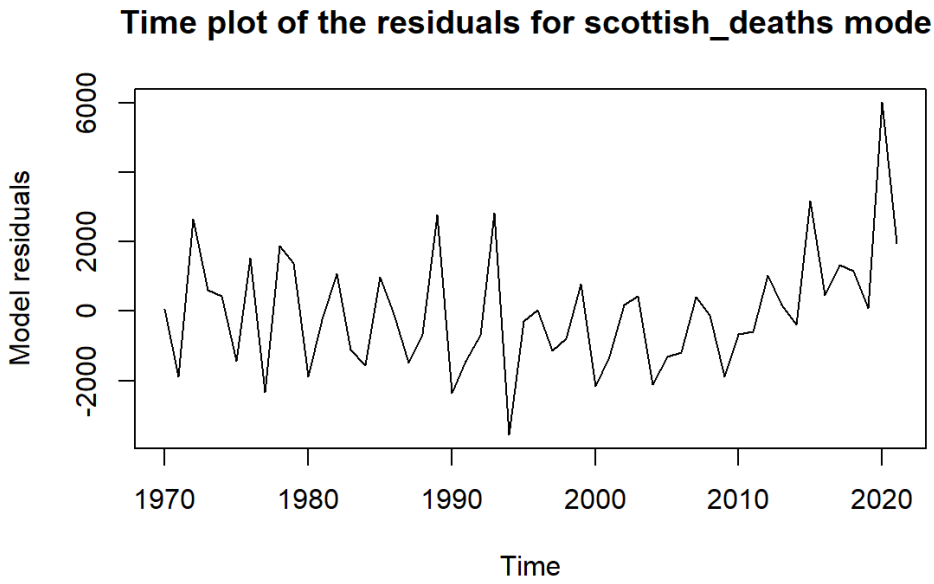
The sample PACF shows a negative value of approximately -0.4 at lag 1 and a small value of approximately 0.25 at lag 4. Overall, the PACF is consistent with the pattern for an MA(1) process as expected from the ACF plot, where there is a significant direct effect at lag 1 and no significant effects at any other lags.

From the information above, an autoregressive integrated moving average model with AR order = 0, degree of differencing = 1, MA order = 1, which in short an ARIMA(0,1,1) model for the scottish deaths time series might be appropriate.

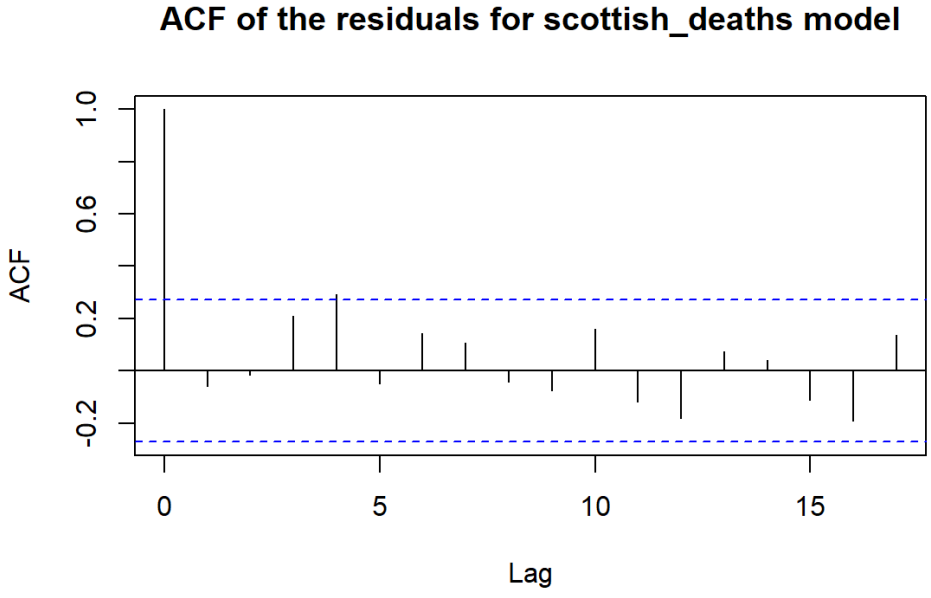
```
##
## Call:
## arima(x = scottish_deaths, order = c(0, 1, 1), method = "ML")
##
## Coefficients:
##           ma1
##       -0.4091
## s.e.   0.1073
##
## sigma^2 estimated as 2966522:  log likelihood = -452.48,  aic = 908.96
```

The coefficient estimated using "arima" function in R is -0.4091 and $\sigma_z^2 = 2966522$.

After fitting the model above, we check the fit of the model by looking at the plots of the model residuals.

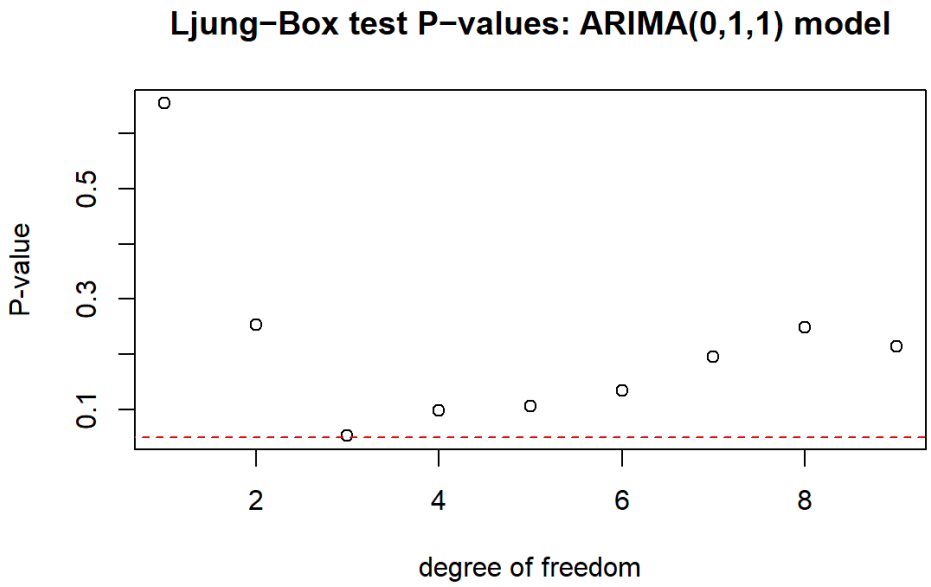


The time plot of the residuals for the model might be stationary without obvious seasonal patterns and trend.



The ACF of the residuals for the scottish_deaths model is approximatly zero for all lags except for lag0 which equals to 1. This suggests that the residuals are independent. Therefore, from the time plot and ACF, it is reasonable to conclude that the residuals for scottish_deaths model obtained is a white noise process. This means that the model fits the data well.

Finally we do a Ljung-Box test for different lags.



All P-values are greater than 0.05 and this suggests that the ARIMA(0,1,1) provides a good fit to the data.

To check the result we get, there is a function in R named “auto.arima” in the “forecast” package which automatically choose the model best fit the given time series data.

```
## Series: scottish_deaths
## ARIMA(0,1,1)
##
## Coefficients:
##           ma1
##       -0.4088
## s.e.   0.1074
##
## sigma^2 = 3025951:  log likelihood = -452.48
## AIC=908.96   AICc=909.21   BIC=912.83
```

We get a very similar result using the “auto.arima” function, which is an ARIMA(0,1,1). There is only a slight difference in the coefficients, -0.4088 compared to -0.4091.

In conclusion, an ARIMA(0,1,1) model might be a good fit of the scottish_deaths dataset, and the equation is:

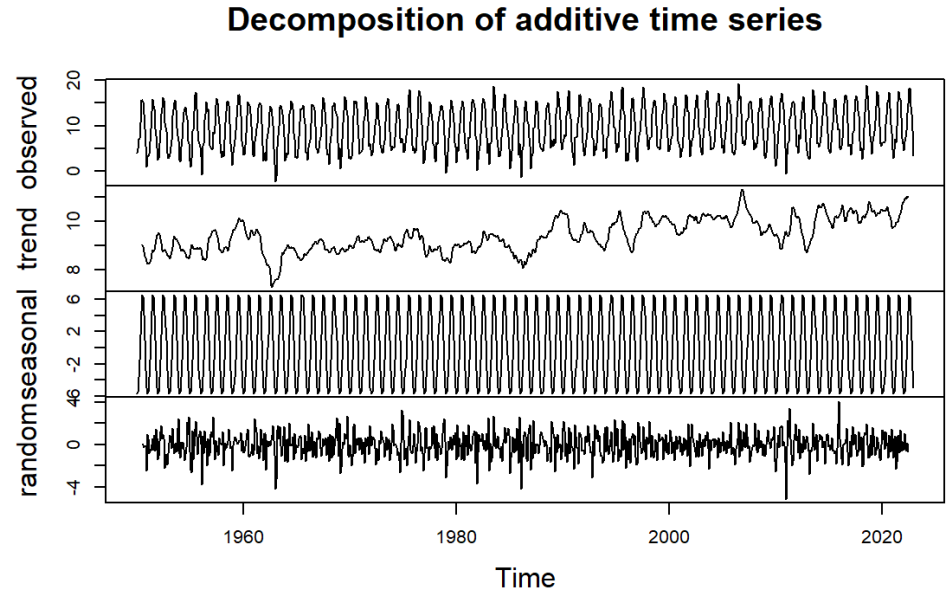
$$X_t - X_{t-1} = Z_t - 0.4091 Z_{t-1}$$

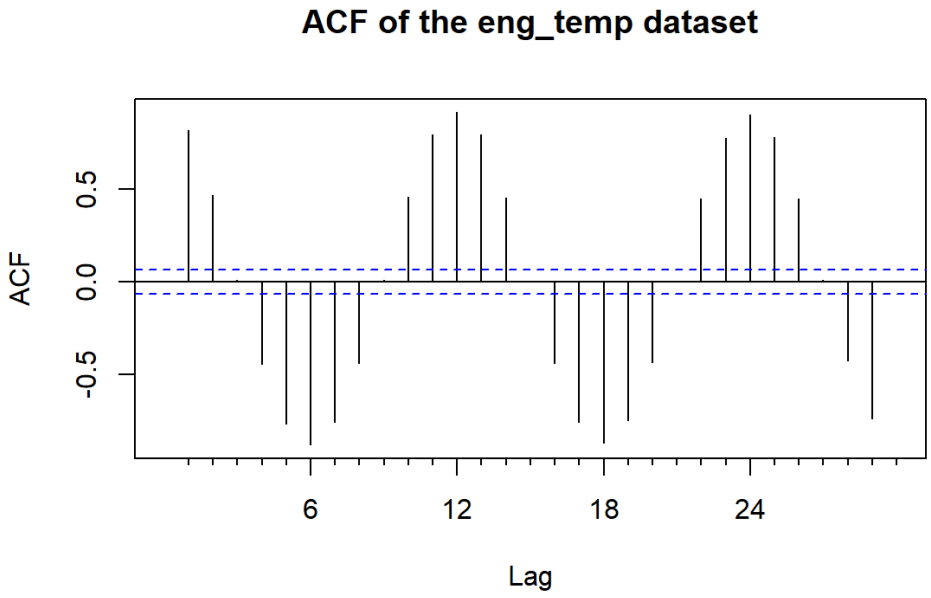
Where,

Z_t is a white noise process with mean 0 and $\sigma_z^2 = 2966522$

2. The eng_temp dataset

To get insights into the underlying patterns and structure of the time series, we first look at the decomposition and ACF of the dataset of the monthly average temperature in England from January 1950 to December 2022.

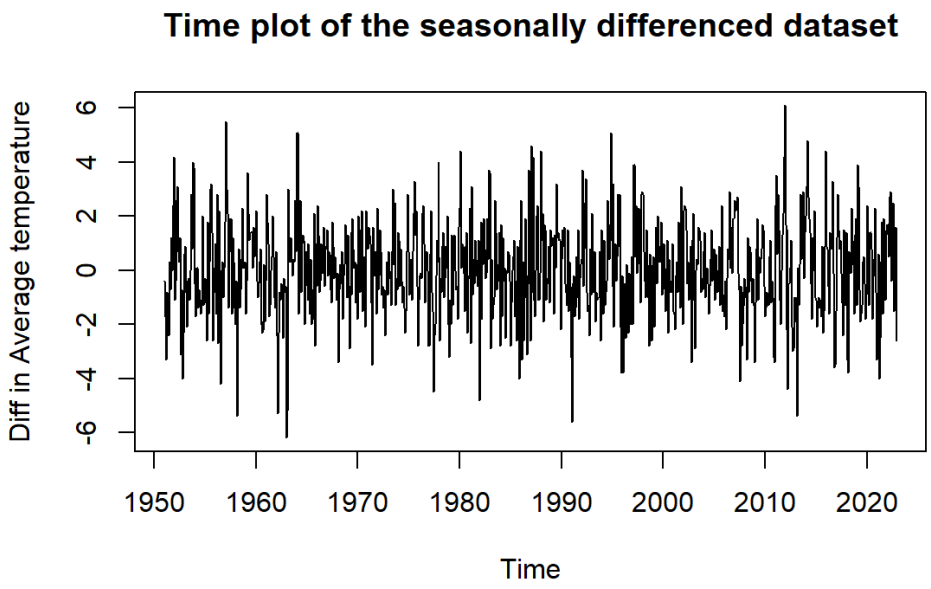


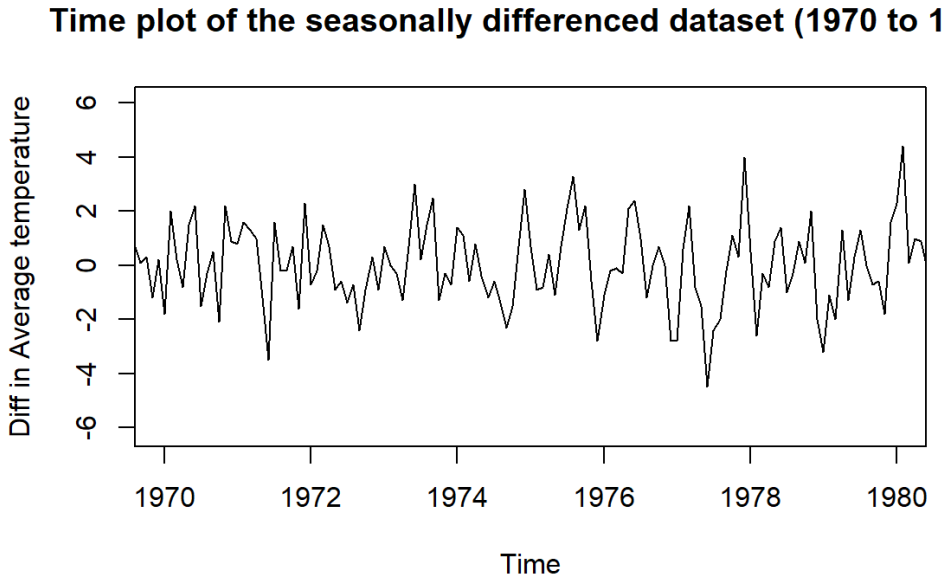


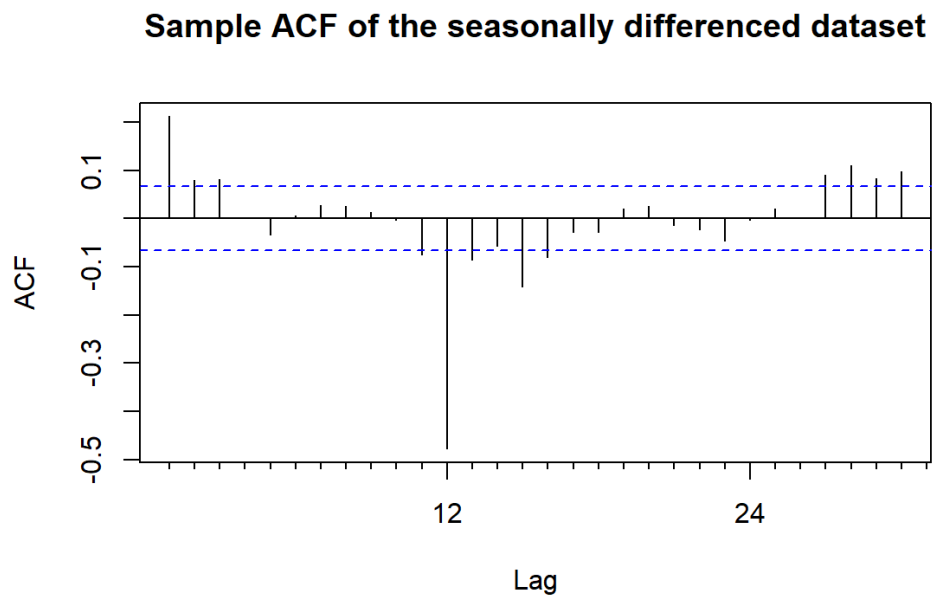
The time plot of the entire dataset might indicate stationarity, and there is no obvious trend as we can see from the trend of decomposition. We can see clear seasonal pattern where the average temperature reach the highest in the middle of a year and low at the end or start of a year. The random component does not show any significantly unusual variations in the data.

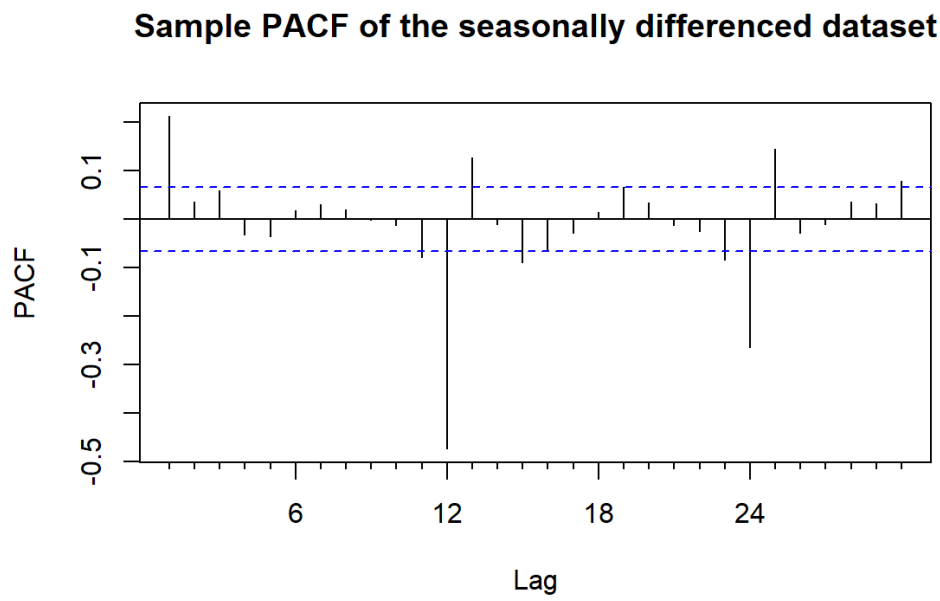
The ACF does not decay quickly and shows clear peaks at lag 0, 12, 24, and troughs at 6, 18, 30. We noticed that the data have been collected monthly. Therefore, the period of the time series is 12 and it suggests that we should difference the data with lag 12.

We next look at the time plot, acf and pacf plot of lag 12 seasonally differenced dataset.





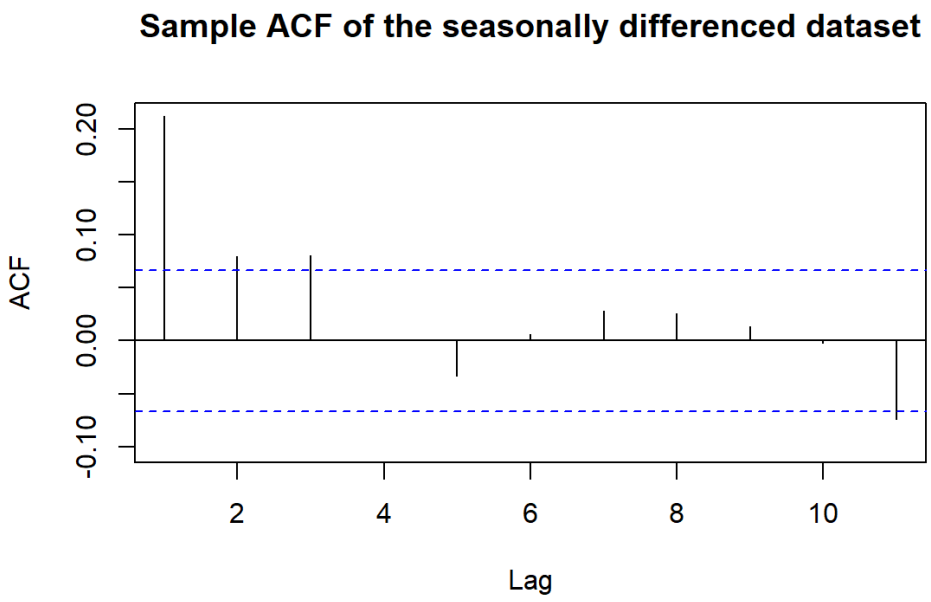


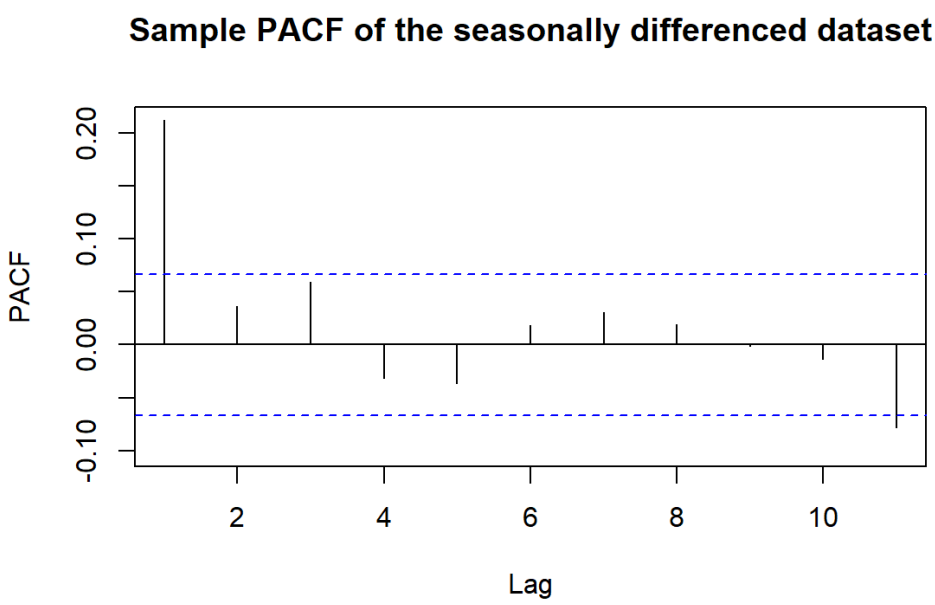


The time plot from 1970 to 1980 is also included since the time plot of the entire time zone is hard to see if the seasonal effect have been successfully removed. The differenced time series seems to be stationary with the seasonal effect removed, and we consider fitting an ARIMA model.

The ACF shows a spike at lag 12, and the ACF of other lags are approximately under the significant boundary.The PACF shows a spike at lag 12 and lag 24. Since the period is 12, the result from the ACF and PACF plot might suggests a MA(1) process for the seasonal part of the model.

We also need to decide the non-seasonal part of the model.



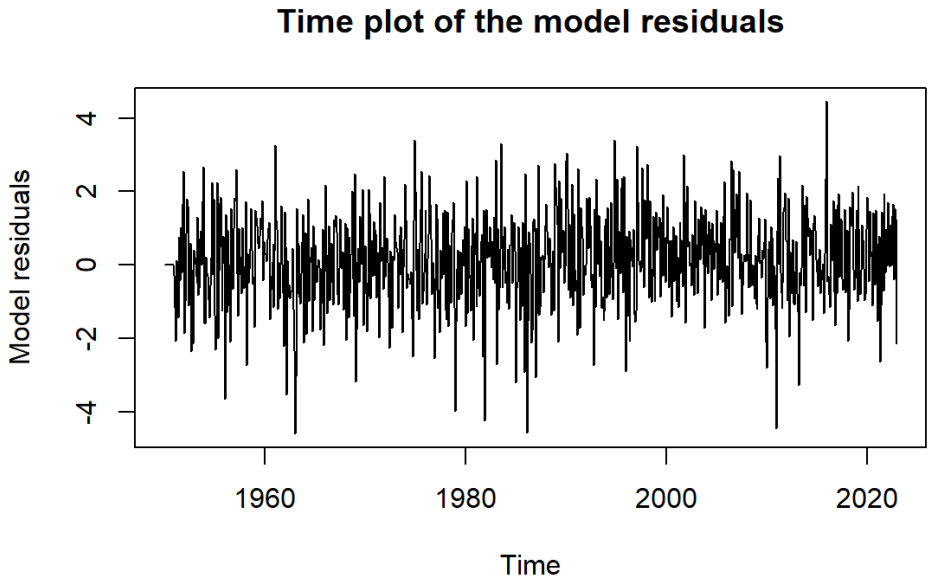


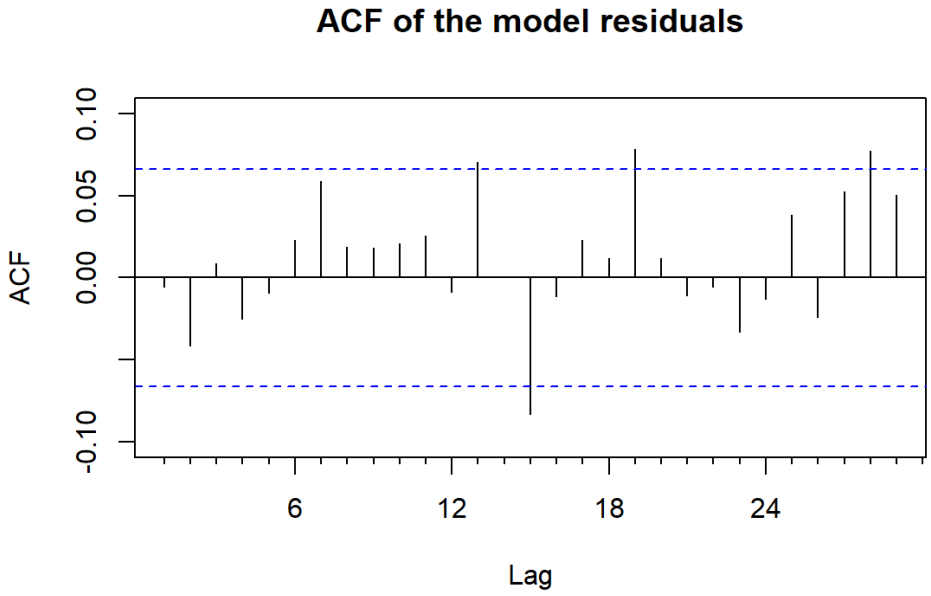
The ACF seems to be cut off after lag 1, and PACF also seems to be cut off after lag 1. Therefore, there should be a MA(1) term, and an AR(1) term in the model. An ARMA(1,1) model might be a good model for the non-seasonal part of the dataset.

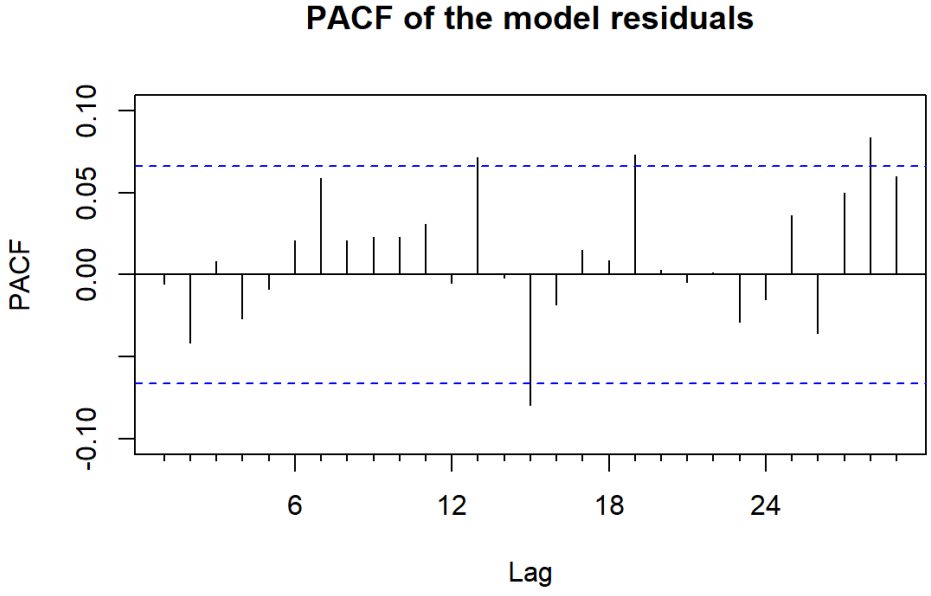
Our final model is a $SARIMA(1,0,1) \times (0,1,1)_{12}$ model.

```
##
## Call:
## arima(x = eng_temp, order = c(1, 0, 1), seasonal = list(order = c(0, 1, 1),
##     period = 12))
##
## Coefficients:
##          ar1      ma1     sma1
##       0.5302  -0.2720  -0.9319
## s.e.  0.1220   0.1377   0.0145
##
## sigma^2 estimated as 1.584:  log likelihood = -1436.82,  aic = 2881.63
```

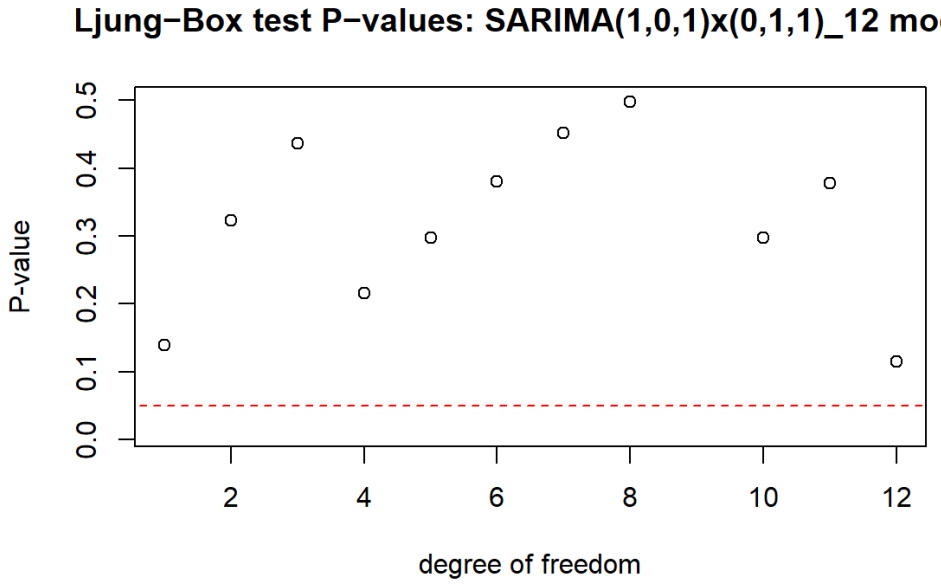
Next, we look at plots of the model residuals and perform the Ljung-Box test for different lags.


Time plot of the model residuals
Model residuals
4
2
0
-2
-4
1960
1980
2000
2020
Time


ACF of the model residuals
ACF
0.10
0.05
0.00
-0.10
6
12
18
24
Lag


PACF of the model residuals
PACF
0.10
0.05
0.00
-0.10
6
12
18
24
Lag

The time plot, ACF/PACF plot shows that the model residuals is fairly a white noies process. Therefore, the model mighe be a good fit of the eng_temp dataset.



All the p-values are greater than 0.05, thus suggests a $SARIMA(1,0,1) \times (0,1,1)_{12}$ model is a good fit of the dataset.

Again, we use the “auto.arima” function to look at the model that R automatically choose for the eng_temp dataset.

```
## Series: eng_temp
## ARIMA(1,0,0)(1,1,0)[12] with drift
##
## Coefficients:
##          ar1     sar1   drift
##       0.2403  -0.4981  0.0018
## s.e.  0.0331   0.0297  0.0037
##
## sigma^2 = 2.228:  log likelihood = -1572.29
## AIC=3152.58   AICc=3152.63   BIC=3171.63
```

It gives a $SARIMA(1,0,0) \times (1,1,0)_{12}$ model, which turns out to be a slightly different model compared to our result. This might also be a model that fits the eng_temp dataset well. However, we notice that the result of AIC value using the “auto.arima” function is bigger than that of our model, which is a $SARIMA(1,0,1) \times (0,1,1)_{12}$ model. Since the model with a lower AIC value is better, the model we get is prefered.

In conclusion, a $SARIMA(1,0,1) \times (0,1,1)_{12}$ model might be a good fit of the eng_temp dataset. The equation is:

$$(1-0.5302B)(1-B^{12})X_t = (1-0.272B)(1-0.9319B^{12})Z_t$$

Rearranging, we get

$$X_t = X_{t-12} - 0.5302X_{t-13} + 0.5302X_{t-1} + Z_t - 0.9319Z_{t-12} - 0.272Z_{t-1} + 0.2535Z_{t-13}$$

Where, Z_t is a white noise process with mean 0 and $\sigma_z^2 = 1.584$

```
# Appendix with R code
# 1.The scottish death dataset

load("scottish_deaths.rda")
#load the datasets into R
ts.plot(scottish_deaths, xlab = "Time", ylab = "Number of deaths", main = "Time plot of the scottish_deaths dat
aset")
# Plot a time plot of the scottish deaths dataset
acf(scottish_deaths,main = "Sample ACF of the first difference of scottish_deaths dataset")

library(tseries)
adf.test(scottish_deaths, alternative="stationary", k=0)
#Augmented Dickey-Fuller test for the time series

First_diff <- diff(scottish_deaths, differences = 1)
# The data is non-stationary, check the first difference

adf.test(First_diff, alternative="stationary", k=0)
#Augmented Dickey-Fuller test for the first difference of the time series

ts.plot(First_diff, xlab="Time", ylab = "First difference number of deaths", main = "Time plot of the first dif
ference of scottish_deaths dataset")

acf(First_diff, lag.max = 25, xlab = "Lag", ylab = "ACF", main = "Sample ACF of the first difference of scottis
h_deaths dataset")
#plot the sample acf up to lag 25

pacf(First_diff, lag.max = 25,  xlab = "Lag", ylab = "PACF", main = "Sample PACF of the first difference of sco
ttish_deaths dataset")
#plot the sample pacf up to lag 25

M1 <- arima(scottish_deaths,order = c(0,1,1),method = "ML")
# fit an ARIMA(0,1,1) model, degree of differencing = 1
M1

Z1 <- residuals(M1)
# extract the model residuals
ts.plot(Z1, xlab = "Time", ylab = "Model residuals", main = "Time plot of the residuals for scottish_deaths mod
el")
# time plot of the residuals

acf(Z1, main = "ACF of the residuals for scottish_deaths model")
# ACF of the residuals

#This LB_test function is the same as that on the Moodle.
#Function to produce P-values for the Ljung-Box test for different lags
#where an ARMA(p,q) model has been fitted.
#Note that k must be > p+q (See Lecture 9 slides)
#Number of degrees of freedom for the test = k-p-q

#Arguments for the function "LB_test"
#resid = residuals from a fitted ARMA(p,q) model.

#max.k = the maximum value of k at which we perform the test
#Note that the minimum k is set at p+q+1 (corresponding to a test with one degree
#of freedom)

#p = Order of the AR part of the model
#q = Order of the MA part of the model

#The function returns a table with one column showing the number of degrees
#of freedom for the test and the other the associated P-value.

LB_test<-function(resid,max.k,p,q){
```

```
 lb_result<-list()
 df<-list()
 p_value<-list()
  for(i in (p+q+1):max.k){
   lb_result[[i]]<-Box.test(resid,lag=i,type=c("Ljung-Box"),fitdf=(p+q))
   df[[i]]<-lb_result[[i]]$parameter
   p_value[[i]]<-lb_result[[i]]$p.value
  }
 df<-as.vector(unlist(df))
 p_value<-as.vector(unlist(p_value))
 test_output<-data.frame(df,p_value)
 names(test_output)<-c("deg_freedom","LB_p_value")
 return(test_output)
}

LB_value = LB_test(Z1,10,0,1)
# call the above function
plot(LB_value,xlab = "degree of freedom", ylab="P-value",main = "Ljung-Box test P-values: ARIMA(0,1,1) model",
cex=1)
# plot the p-values of Ljung-Box test
lines(c(0:10),rep(0.05,11),lty=2,col="red")
# add a line y = 0.05


library(forecast)
fit <- auto.arima(scottish_deaths, seasonal = FALSE)
# R will automatically choose the model bet fit the given data
fit

# 2. The eng_temp dataset
load("eng_temp.rda")
#load the datasets into R
library(forecast)
plot(decompose(eng_temp))
#plot the decomposition of the time series
Acf(eng_temp,xlim=c(0,30),main="ACF of the eng_temp dataset")
# plot the acf up to lag 30

diff12 <- diff(eng_temp,differences = 1, lag = 12)
# seasonally difference the data at lag 12

ts.plot(diff12, xlab="Time", ylab = "Diff in Average temperature", main = "Time plot of the seasonally differen
ced dataset")

ts.plot(diff12, xlim=c(1970,1980), xlab="Time", ylab = "Diff in Average temperature", main = "Time plot of the
seasonally differenced dataset")

Acf(diff12,lag.max = 30, xlab = "Lag", ylab = "ACF", main = "Sample ACF of the seasonally differenced dataset")
#plot the sample acf up to lag 30

Pacf(diff12, lag.max = 30,  xlab = "Lag", ylab = "PACF", main = "Sample PACF of the seasonally differenced data
set")
#plot the sample pacf up to lag 30

Acf(diff12,lag.max = 11, xlab = "Lag", ylab = "ACF", main = "Sample ACF of the seasonally differenced dataset")
#plot the sample acf up to lag 11
Pacf(diff12,lag.max = 11, xlab = "Lag", ylab = "PACF", main = "Sample PACF of the seasonally differenced datase
t")
#plot the sample pacf up to lag 11

M2 <- arima(eng_temp, order = c(1, 0, 1), seasonal = list(order = c(0, 1, 1), period = 12))
# fit the SARIMA model
M2
Z2 <- residuals(M2)
# look at the residuals of the model
```

```
ts.plot(Z2,xlab = "Time", ylab = "Model residuals", main = "Time plot of the model residuals")
# the time plot of the residuals of the model
Acf(Z2,xlab = "Lag", ylab = "ACF", main = "ACF of the model residuals")
# ACF of the residuals of the model
Pacf(Z2,xlab = "Lag", ylab = "PACF", main = "PACF of the model residuals")
# PACF of the residuals of the model

#This LB_test_SARIMA function is the same as that on the Moodle.
#Function to produce P-values for the Ljung-Box test for different lags
#where an ARIMA(p,d,q)x(P,D,Q)_h model has been fitted.
#Note that k must be > p+q+P+Q
#Number of degrees of freedom for the test = k-p-q-P-Q

#Arguments for the function "LB_test"
#resid = residuals from a fitted ARIMA(p,d,q)x(P,D,Q)_h model

#max.k = the maximum value of k at which we perform the test
#Note that the minimum k is set at p+q+P+Q+1 (corresponding to a test with one degree
#of freedom)

#p = Order of the non-seasonal AR part of the model
#q = Order of the non-seasonal MA part of the model
#P = Order of the seasonal AR part of the model
#Q = Order of the seasonal MA part of the model

#The function returns a table with one column showing the number of degrees
#of freedom for the test and the other the associated P-value.

LB_test_SARIMA<-function(resid,max.k,p,q,P,Q){
 lb_result<-list()
 df<-list()
 p_value<-list()
  for(i in (p+q+P+Q+1):max.k){
   lb_result[[i]]<-Box.test(resid,lag=i,type=c("Ljung-Box"),fitdf=(p+q+P+Q))
   df[[i]]<-lb_result[[i]]$parameter
   p_value[[i]]<-lb_result[[i]]$p.value
  }
 df<-as.vector(unlist(df))
 p_value<-as.vector(unlist(p_value))
 test_output<-data.frame(df,p_value)
 names(test_output)<-c("deg_freedom","LB_p_value")
 return(test_output)
}

LB_value2 <- LB_test_SARIMA(Z2,15,1,1,0,1)
plot(LB_value2,xlab = "degree of freedom", ylab="P-value",ylim = c(0.01,0.5), main = "Ljung-Box test P-values:
SARIMA(1,0,1)x(0,1,1)_12 model",cex=1)
# plot the p-values of Ljung-Box test
lines(c(0:15),rep(0.05,16),lty=2,col="red")
# add a line y = 0.05
fit1 <- auto.arima(eng_temp, seasonal = TRUE)
# R will automatically choose the model bet fit the given data
fit1
```