HW6 solution

Question 1

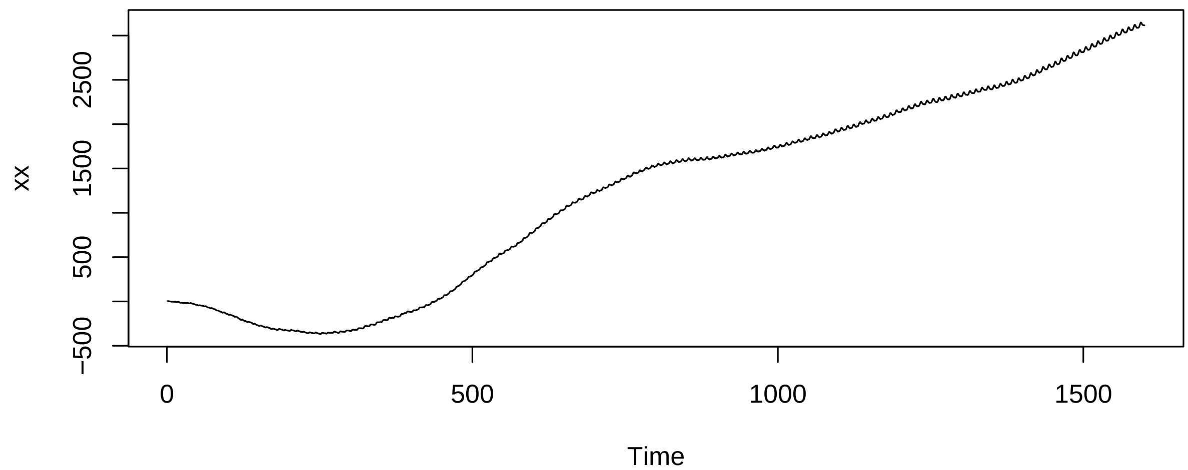
Summary: for the series X_t, I choose the model SARIMA $(1,1,0) \times (0,1,2)_{10}$ excluding the intercept. See the table below for the parameter estimates, standard errors, and p-values.

Coefficients	Estimates	SE	pvalue
AR1	0.5105	0.0216	0
SMA1	-0.8193	0.0228	0
SMA2	0.4436	0.0238	0

Explanation: For this dataset we do not need to do transformation to stabilize the variance. The ACF plot suggests that differencing is needed and we take the first difference. By observing the ACF and PACF plots of the differenced data, we take a seasonal difference with the seasonal period $s = 10$. The ACF is reasonable so we take $d = 1$ and $D = 1$. Then we try different models, with the max $p = 3$, the max $q = 4$, the max $P = 1$ and the max $Q = 2$. By checking the AICc and BICs, we think $(1,0) \times (0,2)$ is the leader. After doing examinations of many of the other AICc and BIC hits and their diagnostics and looking at the ACF/PACF plots of residuals, we seem to still arrive at the $(1,0) \times (0,2)$ fit. The intercept term is not significant according to its p-value so we exclude it.

Output

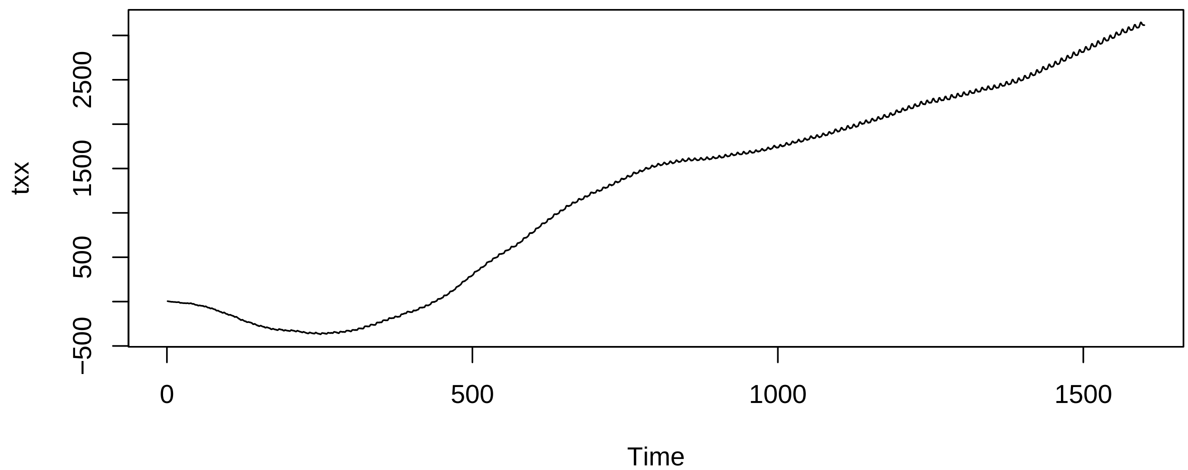
```r
xx <- dat1
plot(xx)
```



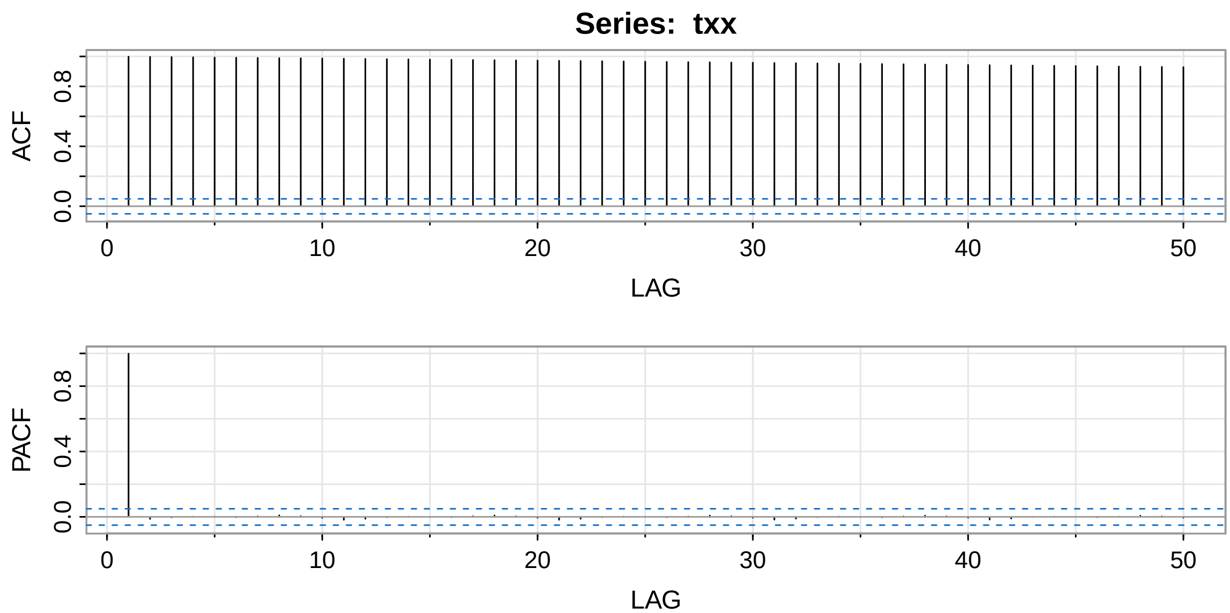

```r
nn <- length(xx)
```

Note: We tried some transformations of xx and use txx to denote them and conclude that txx should be replaced just by xx.

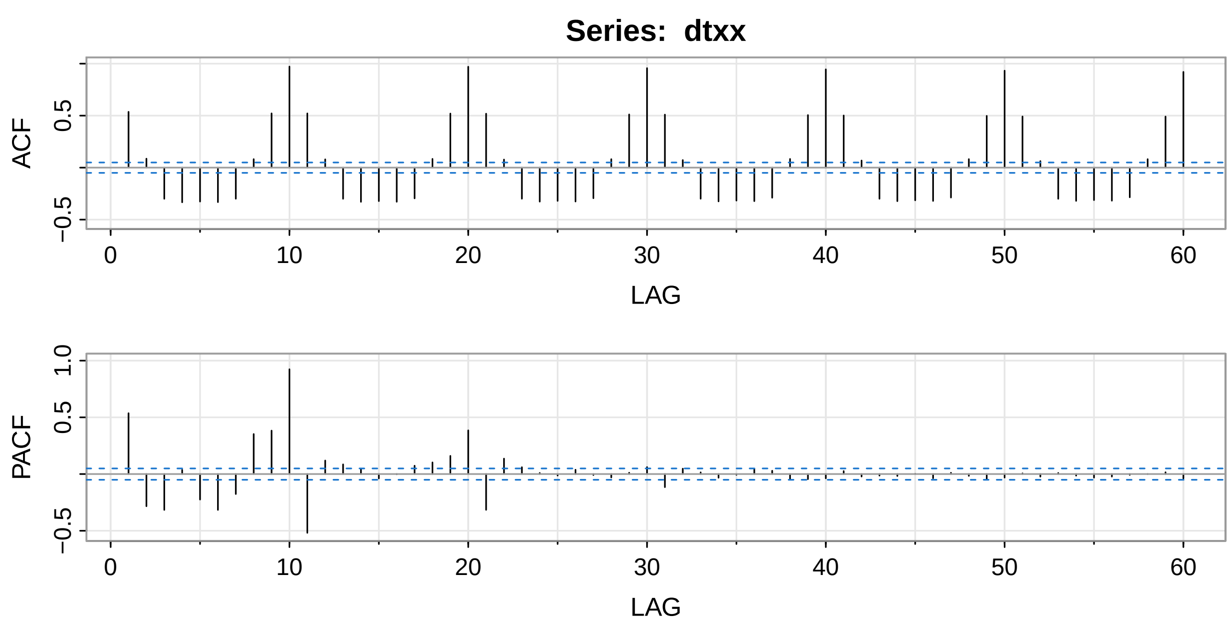
```r
txx <- xx
plot(txx)
```



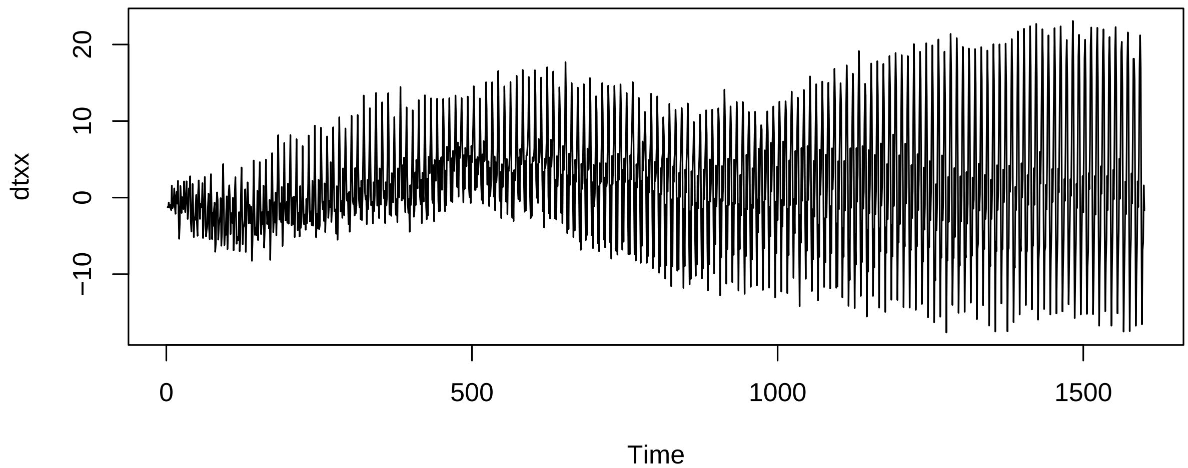

```r
invisible(acf2(txx))
```




```r
dtxx <- diff(txx)
invisible(acf2(dtxx, max.lag = 60))
```



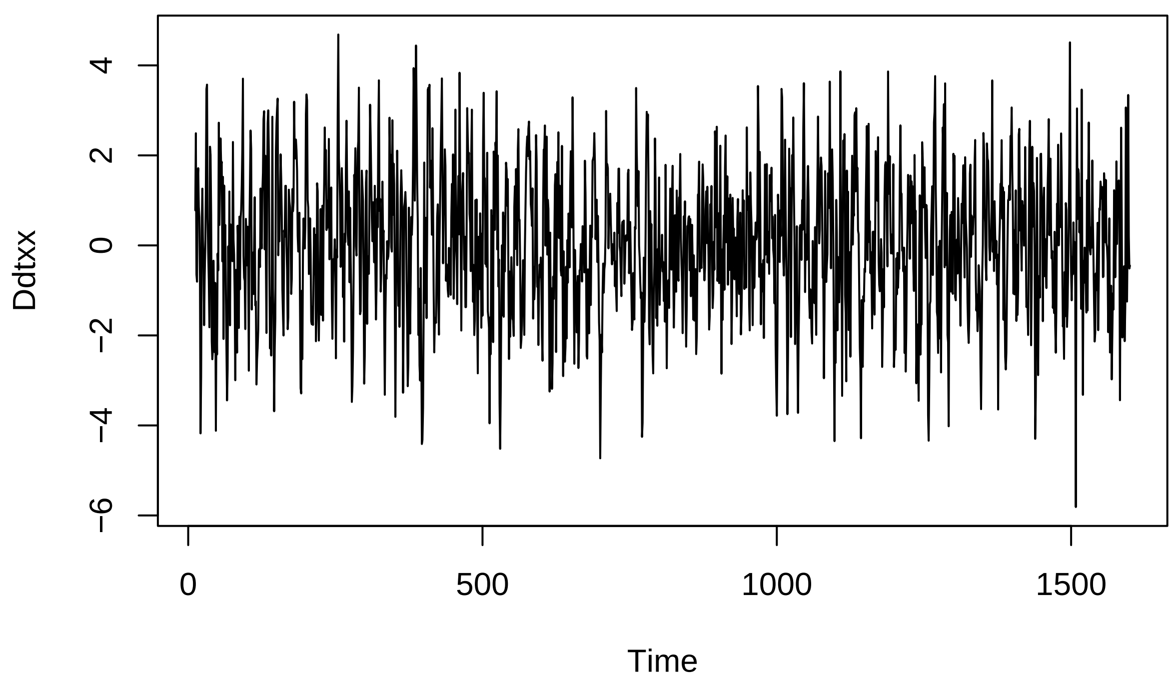

```r
plot(dtxx)
```



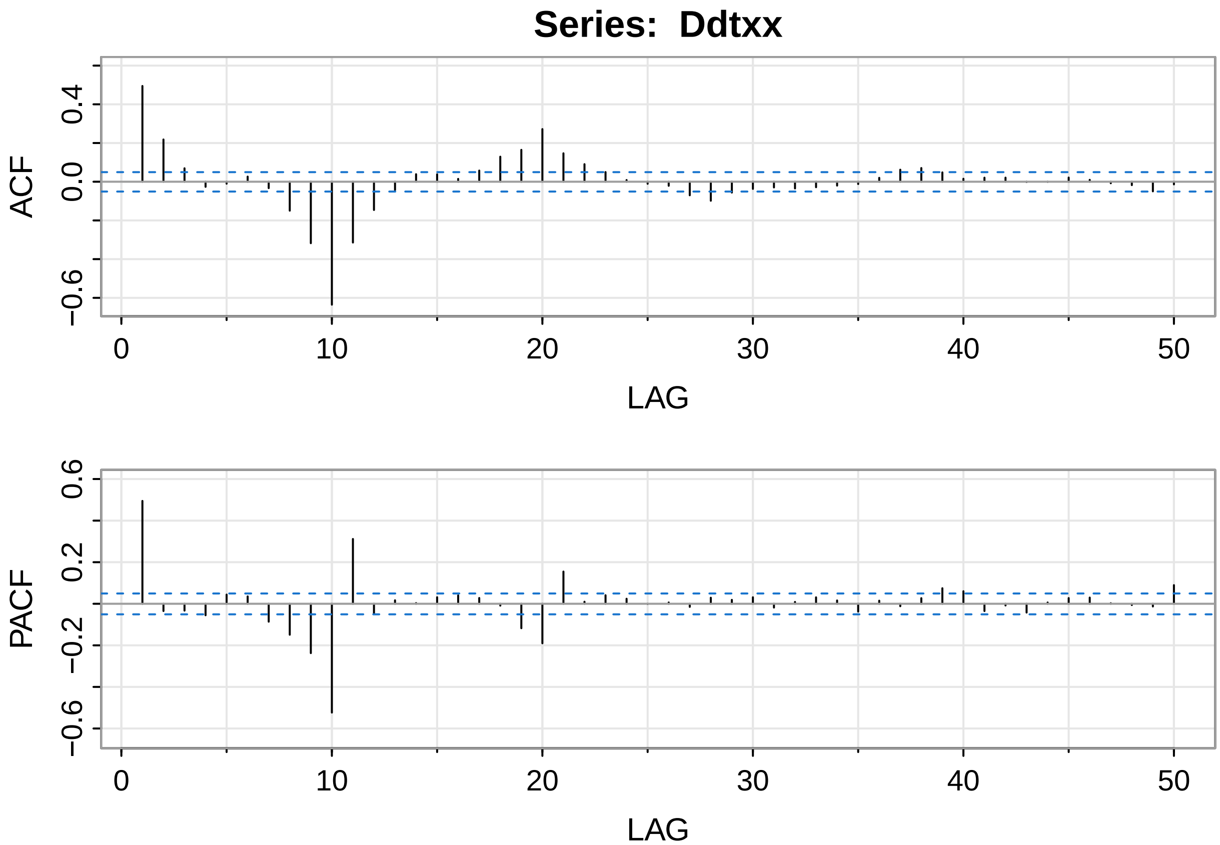

```
ss <- 10
```

From ACF/PACF plots notice and set seasoanlity frequency of 10.

```
Ddtxx <- diff(dtxx, lag = ss)
plot(Ddtxx)
```



```r
invisible(acf2(Ddtxx))
```




```r
maxAR <- 4
maxMA <- 5
maxSAR <- 2
maxSMA <- 3
fits_all <- array(list(), dim = c(maxAR, maxMA, maxSAR, maxSMA))
AICmin <- BICmin <- Inf
for (ii in 1:maxAR) {
for (jj in 1:maxMA) {
for (kk in 1:maxSAR) {
for (ll in 1:maxSMA) {
fits_all[[ii, jj, kk, ll]] <- astsa::sarima(Ddtxx, p = ii - 1, d = 0,
q = jj - 1, P = kk - 1, D = 0, Q = ll - 1, S = ss, no.constant = F,
details = F)
if (fits_all[[ii, jj, kk, ll]]$AICc < AICmin)
AICmin <- fits_all[[ii, jj, kk, ll]]$AICc
if (fits_all[[ii, jj, kk, ll]]$BIC < BICmin)
BICmin <- fits_all[[ii, jj, kk, ll]]$BIC
}
}
}
}
```

```
## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1
```

```
## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in log(s2): NaNs produced

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in log(s2): NaNs produced

## Warning in log(s2): NaNs produced

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in sqrt(diag(fitit$var.coef)): NaNs produced

## Warning in sqrt(diag(fitit$var.coef)): NaNs produced

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1

## Warning in arima(xdata, order = c(p, d, q), seasonal = list(order = c(P, :
## possible convergence problem: optim gave code = 1
```

```r
print("AICc's")
```

```
## [1] "AICc's"
```

```r
for (ii in 1:maxAR) {
for (jj in 1:maxMA) {
for (kk in 1:maxSAR) {
for (ll in 1:maxSMA) {
```

```r
AICdelta <- ((fits_all[[ii, jj, kk, ll]])$AICc - AICmin) * nn
if (AICdelta < 10) {
print(paste0("(", ii - 1, ", ", jj - 1, ") x (", kk - 1, ", ",
ll - 1, ")"))
print(AICdelta)
}
}
}
}
}
```

```
## [1] "(0, 3) x (0, 2)"
## [1] 6.050654
## [1] "(0, 3) x (1, 2)"
## [1] 7.954165
## [1] "(0, 4) x (0, 2)"
## [1] 6.408969
## [1] "(0, 4) x (1, 2)"
## [1] 8.270975
## [1] "(1, 0) x (0, 2)"
## [1] 4.056733
## [1] "(1, 0) x (1, 2)"
## [1] 5.905048
## [1] "(1, 1) x (0, 2)"
## [1] 4.7771
## [1] "(1, 1) x (1, 2)"
## [1] 6.63682
## [1] "(1, 2) x (0, 2)"
## [1] 6.415583
## [1] "(1, 2) x (1, 2)"
## [1] 8.253558
## [1] "(1, 3) x (0, 2)"
## [1] 6.551492
## [1] "(1, 3) x (1, 2)"
## [1] 8.412377
## [1] "(1, 4) x (0, 2)"
## [1] 3.909605
## [1] "(2, 0) x (0, 2)"
## [1] 4.85532
## [1] "(2, 0) x (1, 2)"
## [1] 6.716924
## [1] "(2, 1) x (0, 2)"
## [1] 5.578119
## [1] "(2, 1) x (1, 2)"
## [1] 7.423921
## [1] "(2, 2) x (0, 2)"
## [1] 7.597632
## [1] "(2, 2) x (1, 2)"
## [1] 9.438807
## [1] "(2, 3) x (0, 2)"
## [1] 8.578259
## [1] "(3, 0) x (0, 2)"
## [1] 6.551762
## [1] "(3, 0) x (1, 2)"
```

```
## [1] 8.393039
## [1] "(3, 1) x (0, 2)"
## [1] 7.595551
## [1] "(3, 1) x (1, 2)"
## [1] 9.436589
## [1] "(3, 2) x (0, 2)"
## [1] 8.992351
## [1] "(3, 3) x (0, 2)"
## [1] 0
## [1] "(3, 3) x (1, 2)"
## [1] 1.428054
## [1] "(3, 4) x (0, 2)"
## [1] 1.323542
## [1] "(3, 4) x (1, 2)"
## [1] 8.793893
```

```
print("BICc's")
```

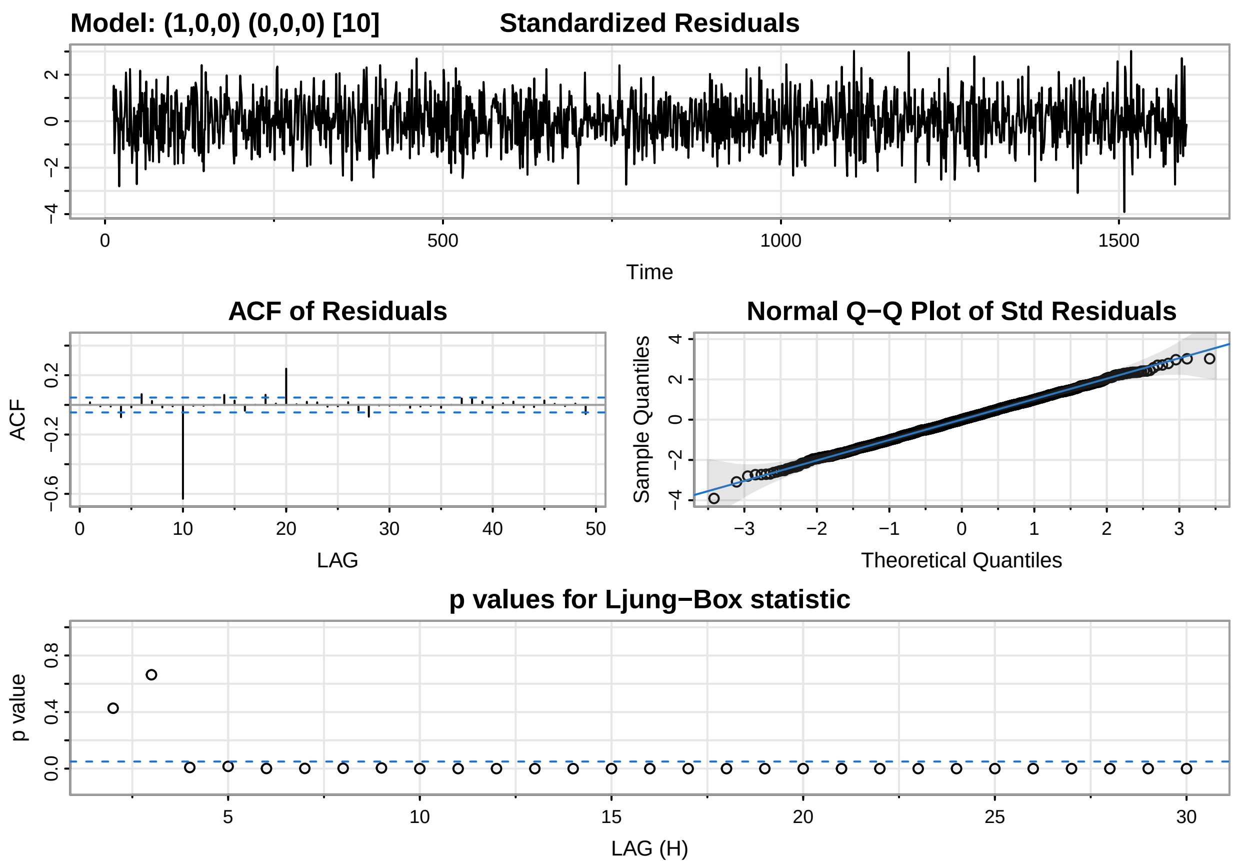
```
## [1] "BICc's"
```

```
for (ii in 1:maxAR) {
for (jj in 1:maxMA) {
for (kk in 1:maxSAR) {
for (ll in 1:maxSMA) {
BICdelta <- ((fits_all[[ii, jj, kk, ll]])$BIC - BICmin) * nn
if (BICdelta < 14) {
print(paste0("(", ii - 1, ", ", jj - 1, ") x (", kk - 1, ", ",
ll - 1, ")"))
print(BICdelta)
}
}
}
}
}
```

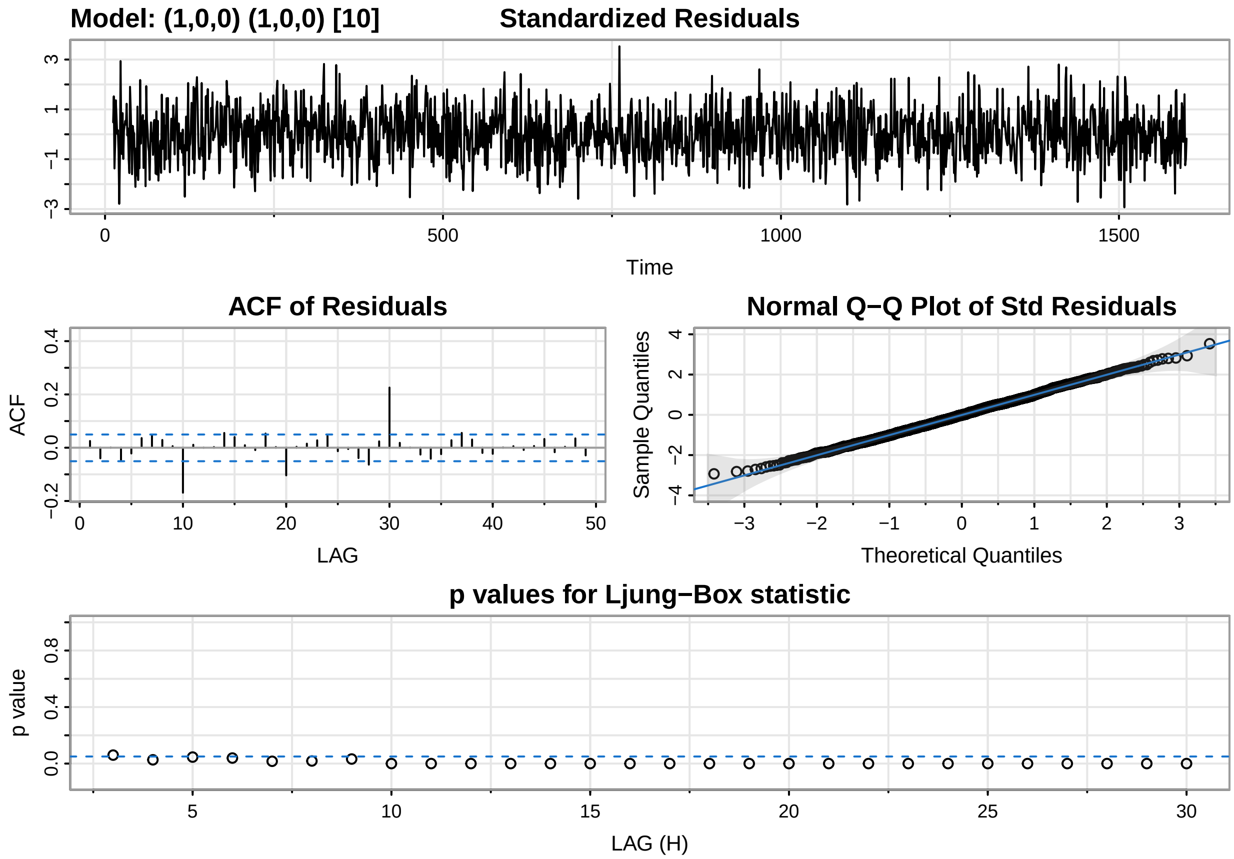
```
## [1] "(0, 3) x (0, 2)"
## [1] 12.78196
## [1] "(1, 0) x (0, 2)"
## [1] 0
## [1] "(1, 0) x (1, 2)"
## [1] 7.243617
## [1] "(1, 1) x (0, 2)"
## [1] 6.115669
## [1] "(1, 1) x (1, 2)"
## [1] 13.36813
## [1] "(1, 2) x (0, 2)"
## [1] 13.14689
## [1] "(2, 0) x (0, 2)"
## [1] 6.19389
## [1] "(2, 0) x (1, 2)"
## [1] 13.44823
## [1] "(2, 1) x (0, 2)"
## [1] 12.30943
## [1] "(3, 0) x (0, 2)"
## [1] 13.28307
```

AICc hits: $(3,3) \times (0,2)$; $(4,2) \times (0,2)$; Also $(3,3) \times (1,2)$; $(3,4) \times (0,2)$; Finally: $(1,0) \times (0,2)$ 4.056; $(1,1) \times (0,2)$ at 4.77. BIC hits: $(1,0) \times (0,2)$; next best is at $(1,1) \times (0,2)$ (around 6) and $(2,0) \times (0,2)$ (also around 6). $(3,3) \times (0,2)$ has deltaBIC of around 23 which rules it out for me. Thus so far I think $(1,0) \times (0,2)$ is the leader. After doing examinations of many of the above fits and their diagnostics (not all displayed here), and looking at the ACF/PACF plots of residuals, we seem to still arrive at the $(1,0) \times (0,2)$ fit.

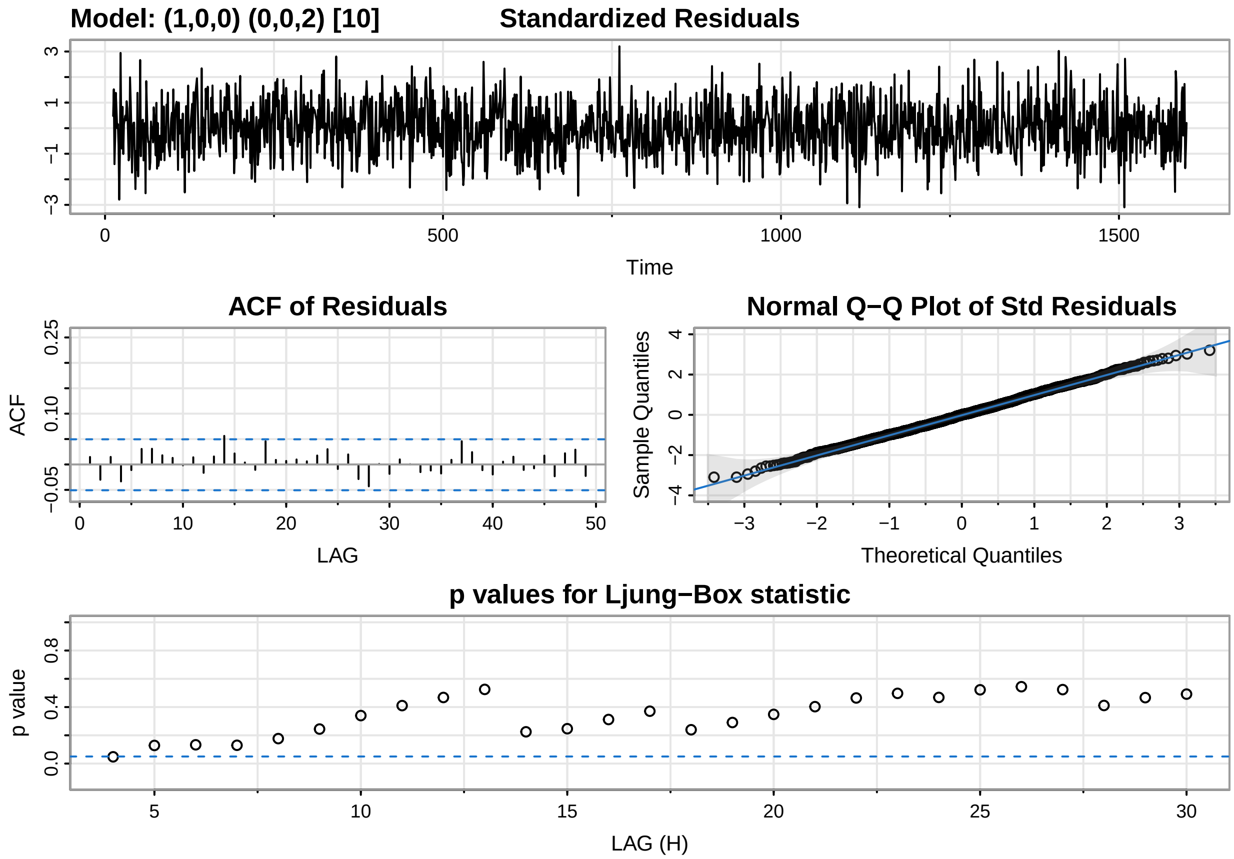
```
invisible(capture.output(astsa::sarima(Ddtxx, p = 1, d = 0, q = 0, P = 0, D = 0,
Q = 0, S = ss)))
```



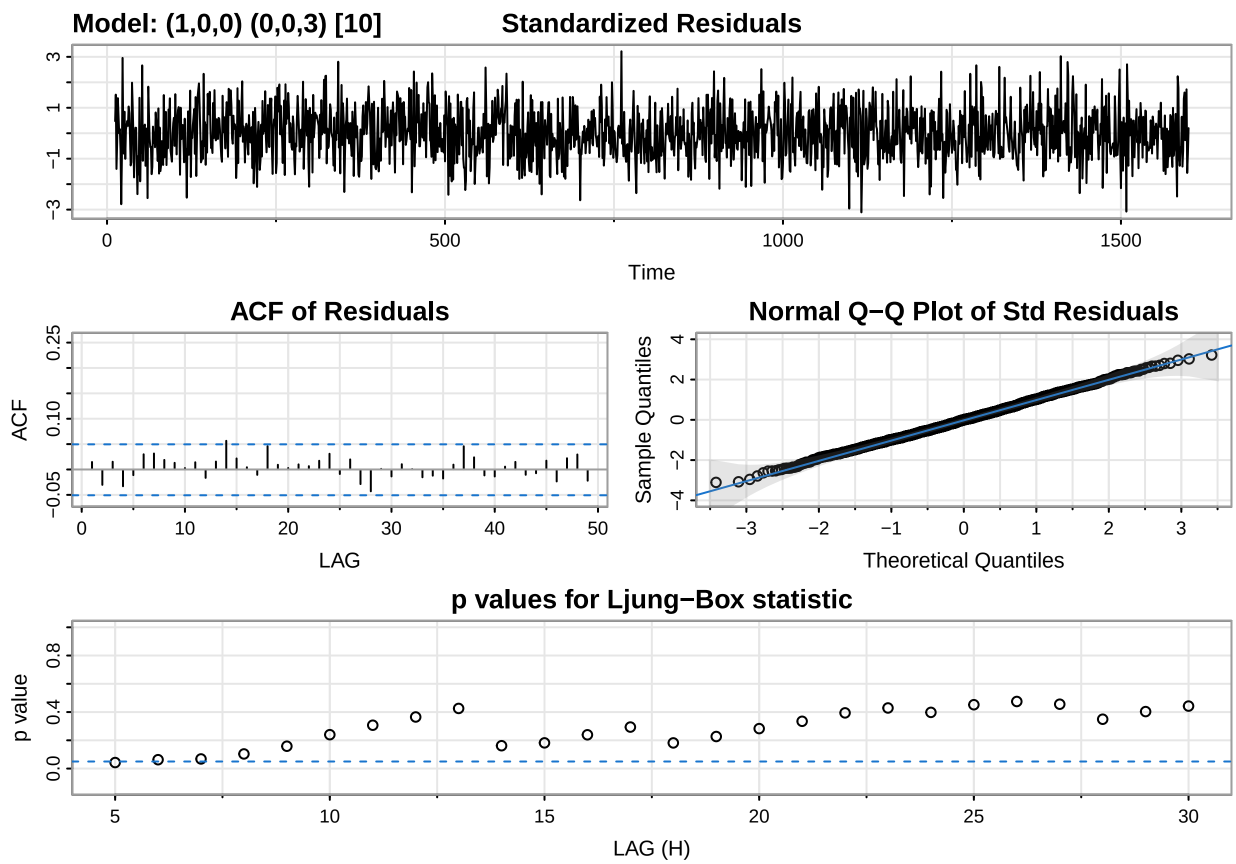

```
invisible(capture.output(astsa::sarima(Ddtxx, p = 1, d = 0, q = 0, P = 1, D = 0,
Q = 0, S = ss)))
```




```
invisible(capture.output(astsa::sarima(Ddtxx, p = 1, d = 0, q = 0, P = 0, D = 0,
Q = 2, S = ss)))
```




```
invisible(capture.output(astsa::sarima(Ddtxx, p = 1, d = 0, q = 0, P = 0, D = 0,
Q = 3, S = ss)))
```



We choose the model $(1,1,0) \times (0,1,2)_{10}$ and now check if we should include the intercept.

```
astsa::sarima(Ddtxx, p = 1, d = 0, q = 0, P = 0, D = 0, Q = 2, S = ss, no.constant = F,
details = F)$ttable
```

```
##        Estimate     SE  t.value p.value
## ar1      0.5101 0.0216  23.6214   0.000
## sma1    -0.8194 0.0228 -35.9471   0.000
## sma2     0.4434 0.0238  18.6072   0.000
## xmean    0.0206 0.0320   0.6450   0.519
```

```
astsa::sarima(Ddtxx, p = 1, d = 0, q = 0, P = 0, D = 0, Q = 2, S = ss, no.constant = T,
details = F)$ttable
```

```
##      Estimate     SE  t.value p.value
## ar1    0.5105 0.0216  23.6466       0
## sma1  -0.8193 0.0228 -35.9358       0
## sma2   0.4436 0.0238  18.6249       0
```

we exclude the intercept

Question 2

Summary: for the weather data, I choose the model SARIMAX $(1, 0, 1) \times (0, 0, 1)_{24}$.

Coefficients	Estimates	SE	pvalue
AR1	0.5514	0.1040	0.0000
MA1	-0.3018	0.1178	0.0106
SMA1	-0.1403	0.0402	0.0005
Intercept	-129.0752	30.2603	0.0000
Year	0.0719	0.0152	0.0000
Feb	5.2034	0.6189	0.0000
Mar	17.9402	0.6690	0.0000
Apr	32.4310	0.6949	0.0000
May	44.8709	0.7082	0.0000
Jun	54.6486	0.7148	0.0000
Jul	59.6278	0.7166	0.0000
Aug	56.7990	0.7148	0.0000
Sep	47.8598	0.7086	0.0000
Oct	34.4354	0.6954	0.0000
Nov	19.3847	0.6700	0.0000
Dec	5.5402	0.6207	0.0000

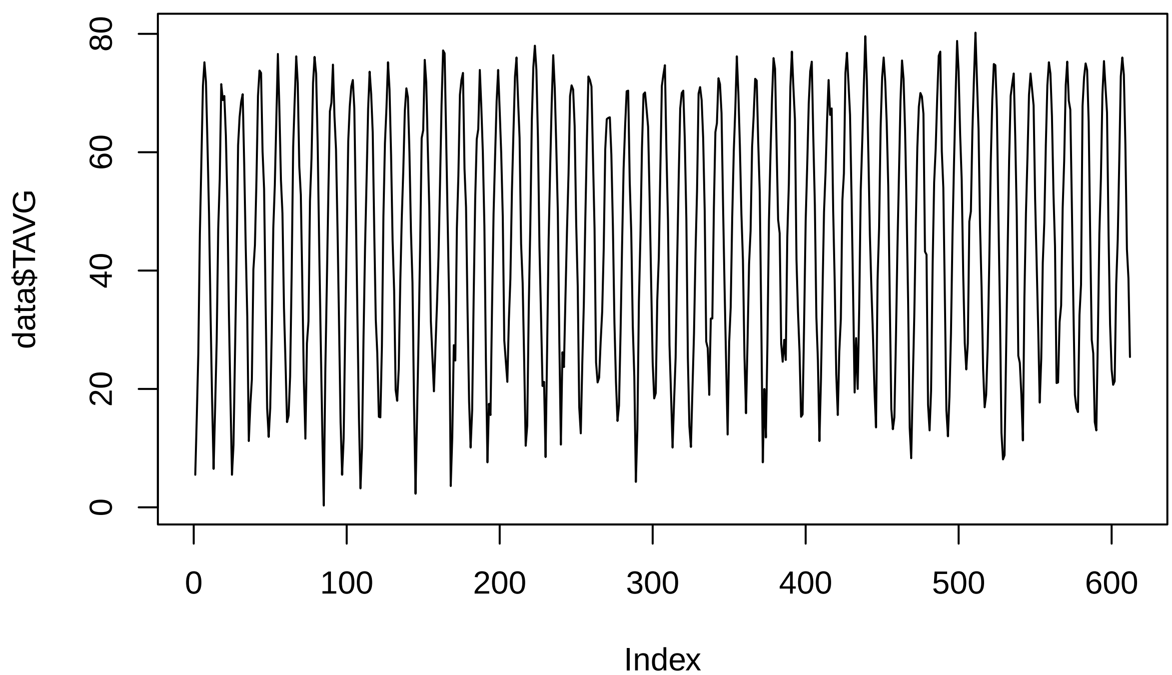
Explaination: From the ACF/PACF plots, no differencing seems necessary andwe can notice a seasonality frequency of 12 or 24. We fit the SARIMA with $s = 12$ and $s = 24$ respectively. For $s = 12$, the model chosen is $(1, 0, 1) \times (1, 0, 1)_{12}$, and for $s = 24$, the model is $(1, 0, 1) \times (0, 0, 1)_{24}$. Since the later model has both better AIC and BIC, we decide to consider it as the best model among all candidates.

Output:

```r
library(lubridate)
```

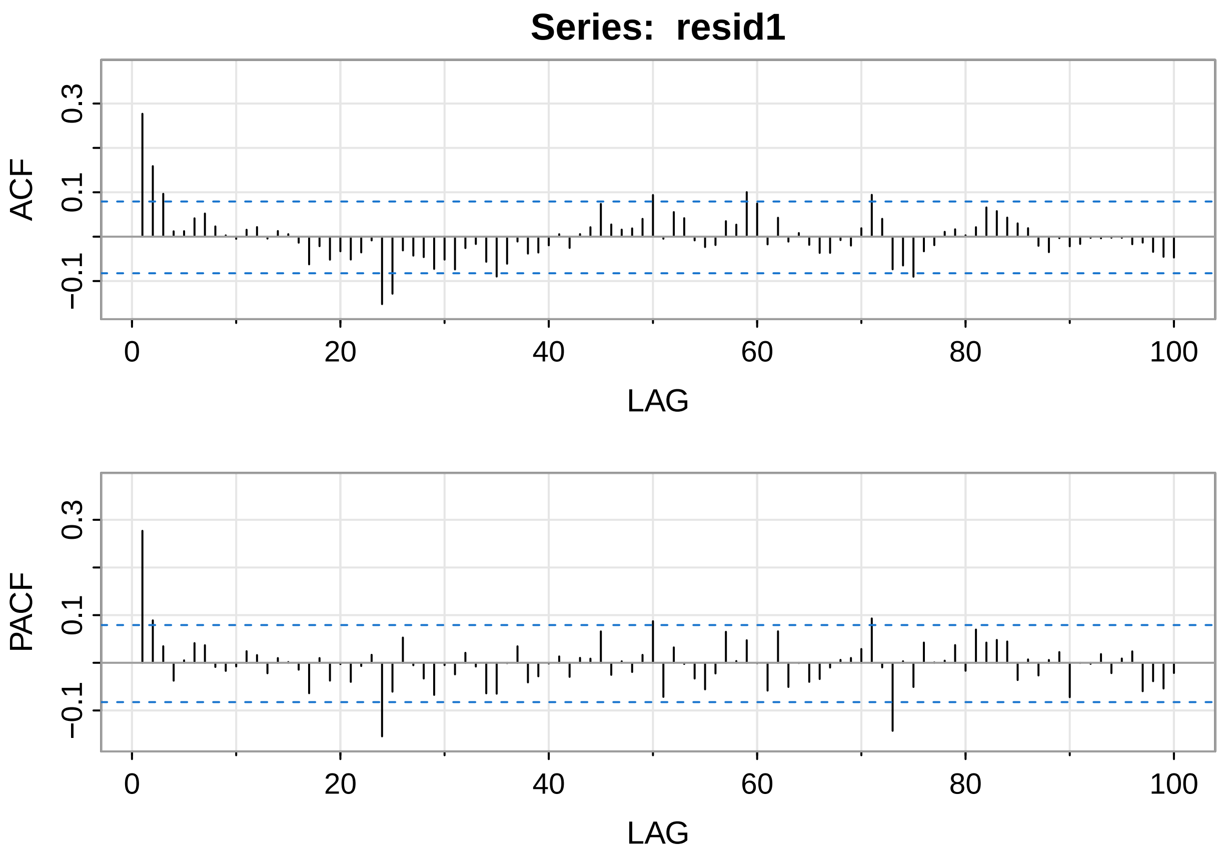
```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```r
library(astsa)
data <- read.csv('MSP_monthly.csv')
data <- data[, c(3, 17)]
data$DATE <- ym(data$DATE)
data$time <- year(data$DATE)
data$month <- as.factor(month(data$DATE))
TAVG <- data$TAVG
nn <- length(TAVG)
plot(data$TAVG, type='l')
```



To fit a SARIMA model with exogenous covariates, we can first fit the corresponding linear model and investigate the temporal dependence by looking at the ACF/PACF plots of its residuals:

```
XX <- model.matrix(~ -1 + year(DATE) +as.factor(month(DATE)), data = data)
lm1 <- lm(TAVG~time+month, data=data)
resid1 <- resid(lm1)
summary(lm1)
acf2(resid1, 100)
```



As shown in the plot, some significant correlations appear at lag $24, 50, 60, 71, \ldots$. Thus, we might have a periodicity of 12 or 24. To narrow down the number of candidate models, we first use AIC and BIC to filter some models out. Note that to properly fit the model with factor covariates *month*, we need to use *model.matrix*() to create a design matrix since *sarima*() doesn't automatically recognize the factor variables.

```r
maxAR <- 1
maxMA <- 1
maxSAR <- 2
maxSMA <- 3
fits_all <- array(list(), dim=c(maxAR + 1, maxMA + 1, maxSAR + 1, maxSMA+ 1))
ss = 12 # 12 or 24?
## fits_all <- vector("list", length=maxAR)
AICmin <- BICmin <- Inf
for (ii in 1:(maxAR+1)){ ## ii-1, jj-1 are orders
  for (jj in 1:(maxMA+1)){
    for (kk in 1:(maxSAR+1)){ ## ii-1, jj-1 are orders
      for (ll in 1:(maxSMA+1)){
        ## fits_all[[ii]][[jj]][[kk]][[ll]] <-
        fits_all[[ii,jj,kk,ll]] <-
          astsa::sarima(TAVG,
                        p=ii-1, d=0, q=jj-1,
                        P=kk-1,D=0,Q=ll-1, S=ss,
                        xreg=XX[,c(1, 3:13)],
                        no.constant=T,
                        details=F)
        if (fits_all[[ii,jj,kk,ll]]$AICc < AICmin)
          AICmin <- fits_all[[ii,jj,kk,ll]]$AICc
        if (fits_all[[ii,jj,kk,ll]]$BIC < BICmin)
          BICmin <- fits_all[[ii,jj,kk,ll]]$BIC
      }
    }
  }
}
## fits_all_24 = fits_all

for (ii in 1:(maxAR+1)){ ## ii-1, jj-1 are orders
  for (jj in 1:(maxMA+1)){
    for (kk in 1:(maxSAR+1)){ ## ii-1, jj-1 are orders
      for (ll in 1:(maxSMA+1)){
        AICdelta <- ((fits_all[[ii, jj, kk, ll]])$AICc - AICmin) * nn
        ##BICdelta <- ((fits_all[[ii]][[jj]])$ICs["BIC"] - BICmin) * nn
        try(
          {if (AICdelta < 15){
            print(paste0("(",
                         ii-1, ", ", jj-1, ") x (",
                         kk-1, ", ", ll-1, ")"))
            print(AICdelta)
          }}
        )
      }
    }
  }
}
```

```
## [1] "(0, 1) x (1, 3)"
## [1] 11.9625
## [1] "(1, 0) x (0, 2)"
## [1] 8.263387
## [1] "(1, 0) x (0, 3)"
```

```
## [1] 10.01508
## [1] "(1, 0) x (1, 1)"
## [1] 7.309233
## [1] "(1, 0) x (1, 2)"
## [1] 10.10005
## [1] "(1, 0) x (1, 3)"
## [1] 3.044127
## [1] "(1, 0) x (2, 0)"
## [1] 8.085826
## [1] "(1, 0) x (2, 1)"
## [1] 9.430186
## [1] "(1, 0) x (2, 2)"
## [1] 8.26408
## [1] "(1, 0) x (2, 3)"
## [1] 8.885735
## [1] "(1, 1) x (0, 0)"
## [1] 12.83479
## [1] "(1, 1) x (0, 1)"
## [1] 14.55455
## [1] "(1, 1) x (0, 2)"
## [1] 5.042878
## [1] "(1, 1) x (0, 3)"
## [1] 6.917973
## [1] "(1, 1) x (1, 0)"
## [1] 14.66023
## [1] "(1, 1) x (1, 1)"
## [1] 4.33878
## [1] "(1, 1) x (1, 2)"
## [1] 6.9553
## [1] "(1, 1) x (1, 3)"
## [1] 0
## [1] "(1, 1) x (2, 0)"
## [1] 5.085338
## [1] "(1, 1) x (2, 1)"
## [1] 6.514926
## [1] "(1, 1) x (2, 2)"
## [1] 6.08753
## [1] "(1, 1) x (2, 3)"
## [1] 6.233087
```

```
print("BICc's")
```

```
## [1] "BICc's"
```

```
for (ii in 1:(maxAR+1)){ ## ii-1, jj-1 are orders
  for (jj in 1:(maxMA+1)){
    for (kk in 1:(maxSAR+1)){ ## ii-1, jj-1 are orders
      for (ll in 1:(maxSMA+1)){
        BICdelta <- ((fits_all[[ii, jj, kk, ll]])$BIC - BICmin) * nn
        try({
        if (BICdelta < 10){
          print(paste0("(",
                       ii-1, ", ", jj-1, ") x (",
                       kk-1, ", ", ll-1, ")"))
          print(BICdelta)
```

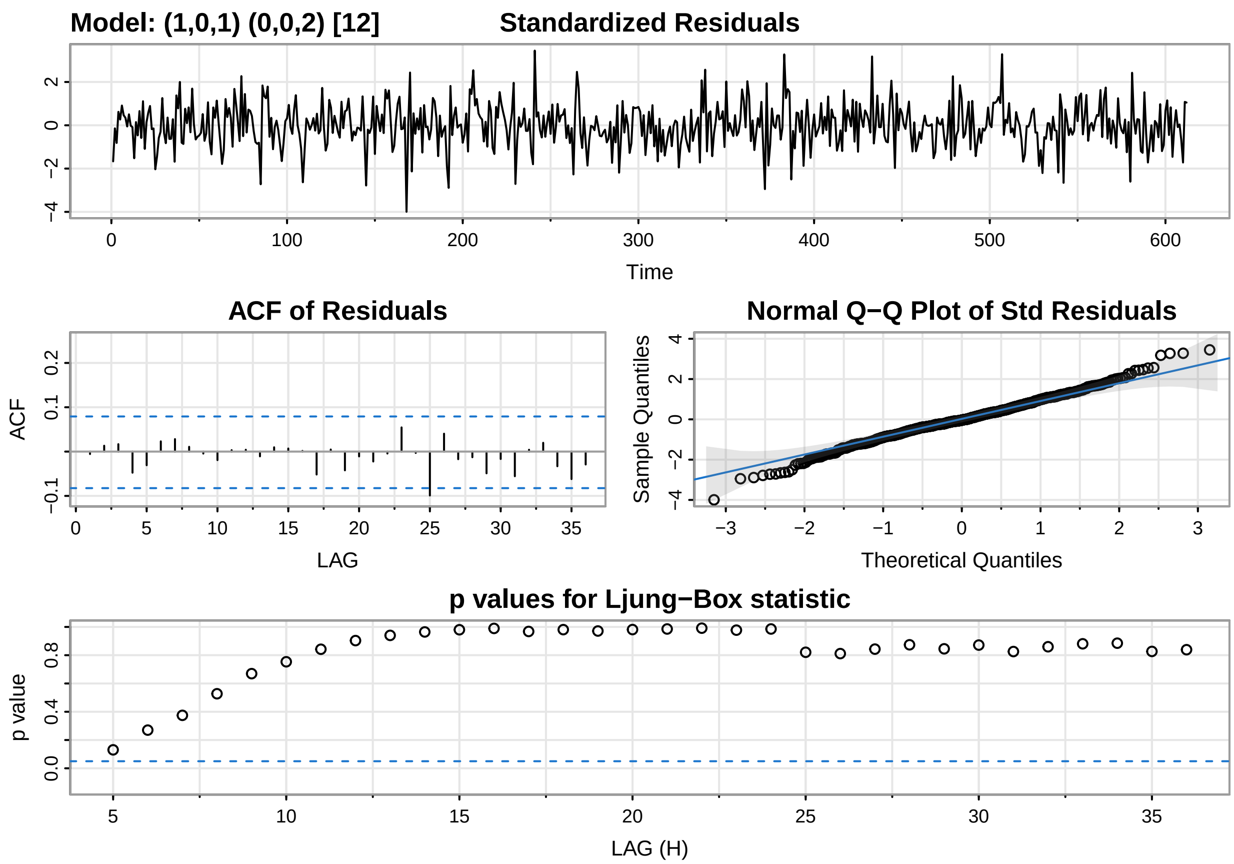
```
        }
        })
      }
    }
  }
}
```

```
## [1] "(0, 1) x (0, 0)"
## [1] 8.94008
## [1] "(0, 1) x (0, 2)"
## [1] 9.715465
## [1] "(0, 1) x (1, 1)"
## [1] 8.989881
## [1] "(0, 1) x (2, 0)"
## [1] 9.327161
## [1] "(1, 0) x (0, 0)"
## [1] 0.05384352
## [1] "(1, 0) x (0, 1)"
## [1] 6.067459
## [1] "(1, 0) x (0, 2)"
## [1] 0.9541546
## [1] "(1, 0) x (0, 3)"
## [1] 7.006558
## [1] "(1, 0) x (1, 0)"
## [1] 6.177334
## [1] "(1, 0) x (1, 1)"
## [1] 0
## [1] "(1, 0) x (1, 2)"
## [1] 7.091531
## [1] "(1, 0) x (1, 3)"
## [1] 4.329188
## [1] "(1, 0) x (2, 0)"
## [1] 0.776593
## [1] "(1, 0) x (2, 1)"
## [1] 6.421668
## [1] "(1, 0) x (2, 2)"
## [1] 9.54914
## [1] "(1, 1) x (0, 0)"
## [1] 1.21774
## [1] "(1, 1) x (0, 1)"
## [1] 7.245313
## [1] "(1, 1) x (0, 2)"
## [1] 2.03436
## [1] "(1, 1) x (0, 3)"
## [1] 8.203033
## [1] "(1, 1) x (1, 0)"
## [1] 7.350995
## [1] "(1, 1) x (1, 1)"
## [1] 1.330262
## [1] "(1, 1) x (1, 2)"
## [1] 8.240361
## [1] "(1, 1) x (1, 3)"
## [1] 5.571466
## [1] "(1, 1) x (2, 0)"
```

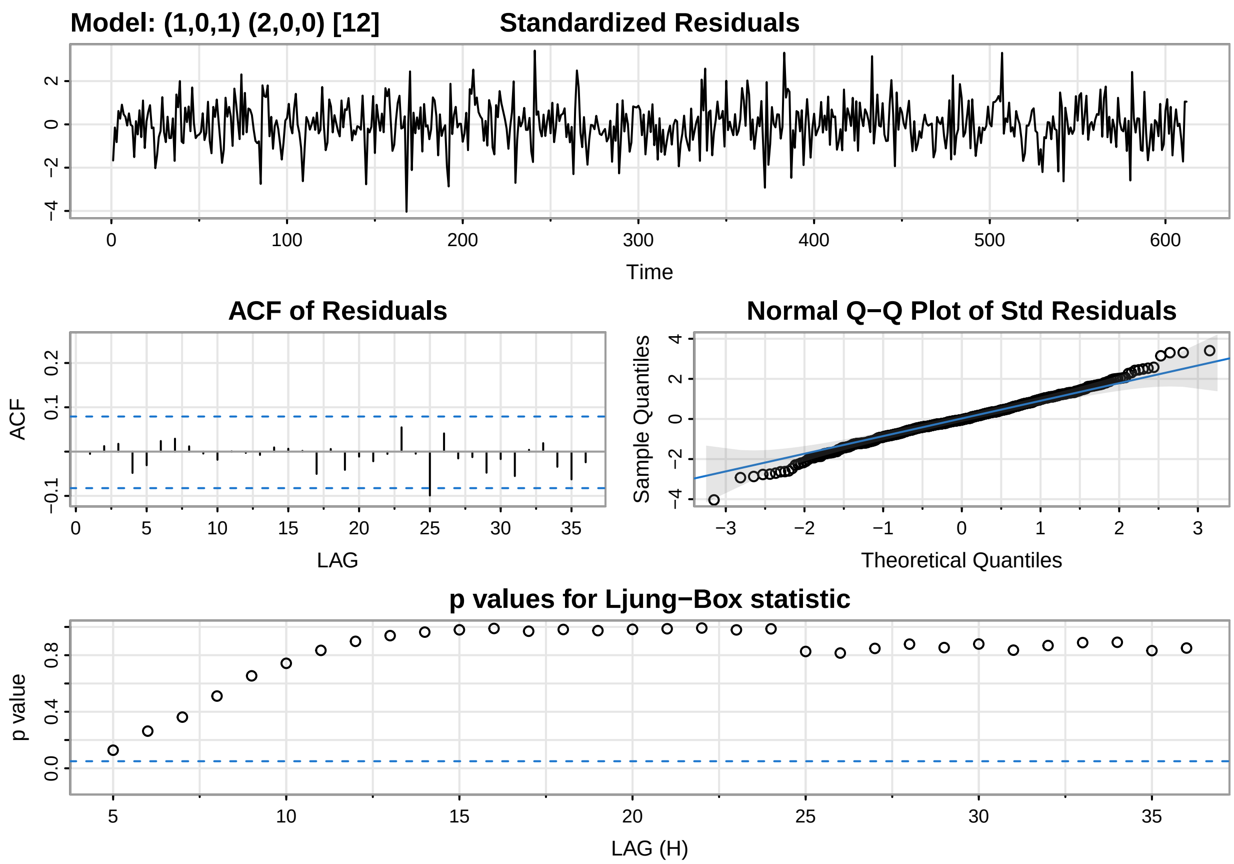
```
## [1] 2.07682
## [1] "(1, 1) x (2, 1)"
## [1] 7.799987
```

For $s = 12$, we narrow the models down to $(1, 1) \times (0, 2)$, $(1, 1) \times (2, 0)$ and $(1, 1) \times (1, 1)$ based on the their smaller AICc and BIC.

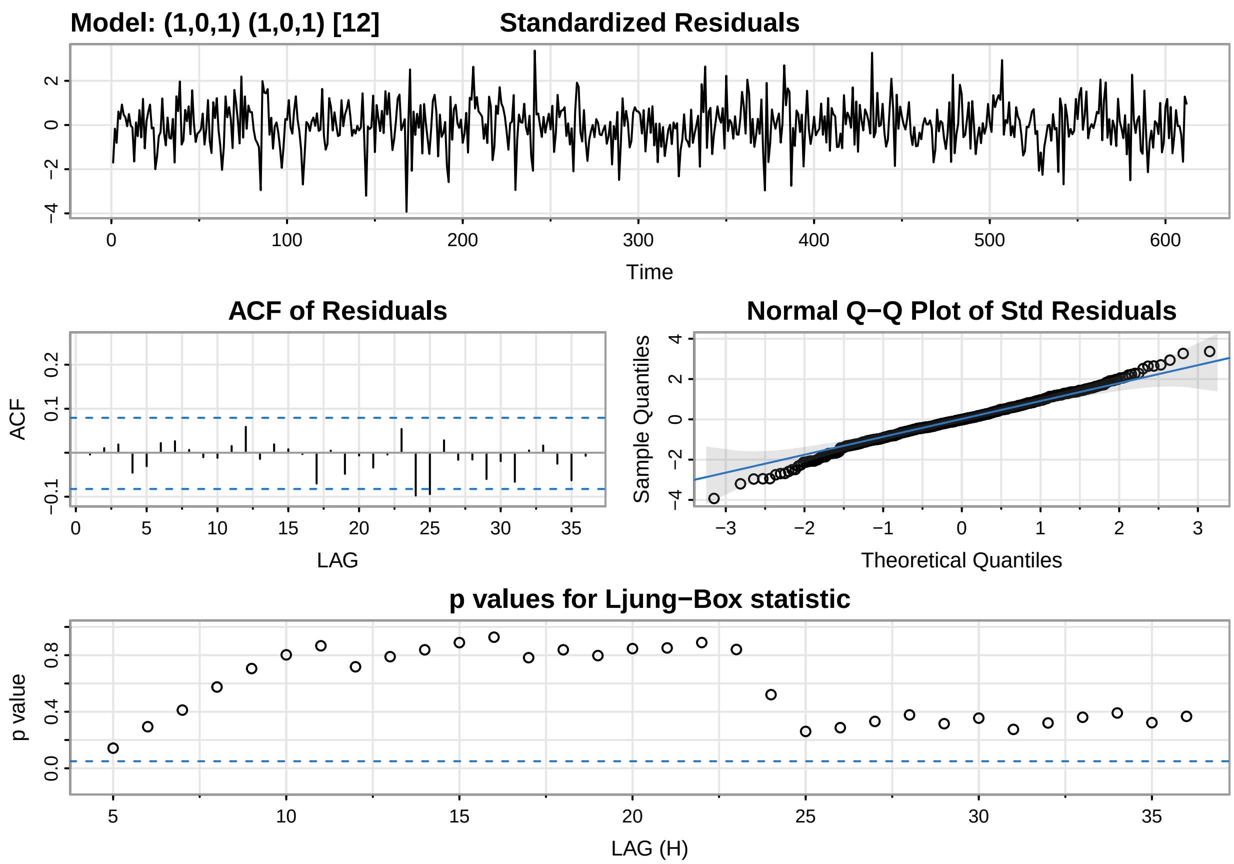
```
model <- sarima(TAVG,  p=1, d=0, q=1,
                P=0, D=0, Q=2, S=12,
                xreg = XX[,c(1, 3:13)])
```



```r
model <- sarima(TAVG,  p=1, d=0, q=1,
                P=2, D=0, Q=0, S=12,
                xreg = XX[,c(1, 3:13)])
```



```
model <- sarima(TAVG,  p=1, d=0, q=1,
                P=1, D=0, Q=1, S=12,
                xreg = XX[,c(1, 3:13)])
```



In the end, we choose $(1,0,1)\times(0,0,2)_{12}$ since it has a slightly better AIC and BIC than $(1,0,1)\times(2,0,0)_{12}$ and its ACF of residuals is better than $(1,0,1)\times(1,0,1)_{12}$. Next, let's consider $s=24$.

```r
maxAR <- 2
maxMA <- 2
maxSAR <- 1
maxSMA <- 2
fits_all <- array(list(), dim=c(maxAR + 1, maxMA + 1, maxSAR + 1, maxSMA+ 1))
ss = 24 # 12 or 24?
## fits_all <- vector("list", length=maxAR)
AICmin <- BICmin <- Inf
for (ii in 1:(maxAR+1)){ ## ii-1, jj-1 are orders
  for (jj in 1:(maxMA+1)){
    for (kk in 1:(maxSAR+1)){ ## ii-1, jj-1 are orders
      for (ll in 1:(maxSMA+1)){
        ## fits_all[[ii]][[jj]][[kk]][[ll]] <-
        fits_all[[ii,jj,kk,ll]] <-
          astsa::sarima(TAVG,
                        p=ii-1, d=0, q=jj-1,
                        P=kk-1,D=0,Q=ll-1, S=ss,
                        xreg=XX[,c(1, 3:13)],
                        no.constant=T,
                        details=F)
        if (fits_all[[ii,jj,kk,ll]]$AICc < AICmin)
          AICmin <- fits_all[[ii,jj,kk,ll]]$AICc
        if (fits_all[[ii,jj,kk,ll]]$BIC < BICmin)
          BICmin <- fits_all[[ii,jj,kk,ll]]$BIC
      }
    }
  }
}
```

```
## Warning in sqrt(diag(fitit$var.coef)): NaNs produced

## Warning in sqrt(diag(fitit$var.coef)): NaNs produced
```

```r
## fits_all_24 = fits_all

for (ii in 1:(maxAR+1)){ ## ii-1, jj-1 are orders
  for (jj in 1:(maxMA+1)){
    for (kk in 1:(maxSAR+1)){ ## ii-1, jj-1 are orders
      for (ll in 1:(maxSMA+1)){
        AICdelta <- ((fits_all[[ii, jj, kk, ll]])$AICc - AICmin) * nn
        ##BICdelta <- ((fits_all[[ii]][[jj]])$ICs["BIC"] - BICmin) * nn
        try(
          {if (AICdelta < 15){
            print(paste0("(",
                         ii-1, ", ", jj-1, ") x (",
                         kk-1, ", ", ll-1, ")"))
            print(AICdelta)
          }}
        )
      }
    }
  }
}
```

```
## [1] "(0, 1) x (1, 2)"
```

```
## [1] 7.475103
## [1] "(0, 2) x (0, 1)"
## [1] 9.135195
## [1] "(0, 2) x (0, 2)"
## [1] 11.19929
## [1] "(0, 2) x (1, 0)"
## [1] 9.279918
## [1] "(0, 2) x (1, 1)"
## [1] 11.21736
## [1] "(0, 2) x (1, 2)"
## [1] 0
## [1] "(1, 0) x (0, 1)"
## [1] 8.271571
## [1] "(1, 0) x (0, 2)"
## [1] 10.25711
## [1] "(1, 0) x (1, 0)"
## [1] 8.283116
## [1] "(1, 0) x (1, 1)"
## [1] 10.30298
## [1] "(1, 0) x (1, 2)"
## [1] 12.282
## [1] "(1, 1) x (0, 0)"
## [1] 14.78847
## [1] "(1, 1) x (0, 1)"
## [1] 5.058695
## [1] "(1, 1) x (0, 2)"
## [1] 7.148242
## [1] "(1, 1) x (1, 0)"
## [1] 5.263448
## [1] "(1, 1) x (1, 1)"
## [1] 7.159188
## [1] "(1, 1) x (1, 2)"
## [1] 9.190729
## [1] "(1, 2) x (0, 1)"
## [1] 6.736898
## [1] "(1, 2) x (0, 2)"
## [1] 8.842481
## [1] "(1, 2) x (1, 0)"
## [1] 6.982185
## [1] "(1, 2) x (1, 1)"
## [1] 8.849485
## [1] "(1, 2) x (1, 2)"
## [1] 10.98768
## [1] "(2, 0) x (0, 1)"
## [1] 4.996558
## [1] "(2, 0) x (0, 2)"
## [1] 7.093197
## [1] "(2, 0) x (1, 0)"
## [1] 5.231357
## [1] "(2, 0) x (1, 1)"
## [1] 7.100722
## [1] "(2, 0) x (1, 2)"
## [1] 9.153078
## [1] "(2, 1) x (0, 1)"
```

```
## [1] 6.881493
## [1] "(2, 1) x (0, 2)"
## [1] 8.985282
## [1] "(2, 1) x (1, 0)"
## [1] 7.115737
## [1] "(2, 1) x (1, 1)"
## [1] 8.992985
## [1] "(2, 1) x (1, 2)"
## [1] 11.04532
## [1] "(2, 2) x (0, 1)"
## [1] 8.413005
## [1] "(2, 2) x (0, 2)"
## [1] 10.52232
## [1] "(2, 2) x (1, 0)"
## [1] 9.495037
## [1] "(2, 2) x (1, 1)"
## [1] 11.40874
## [1] "(2, 2) x (1, 2)"
## [1] 12.5976
```

```r
print("BICc's")
```

```
## [1] "BICc's"
```

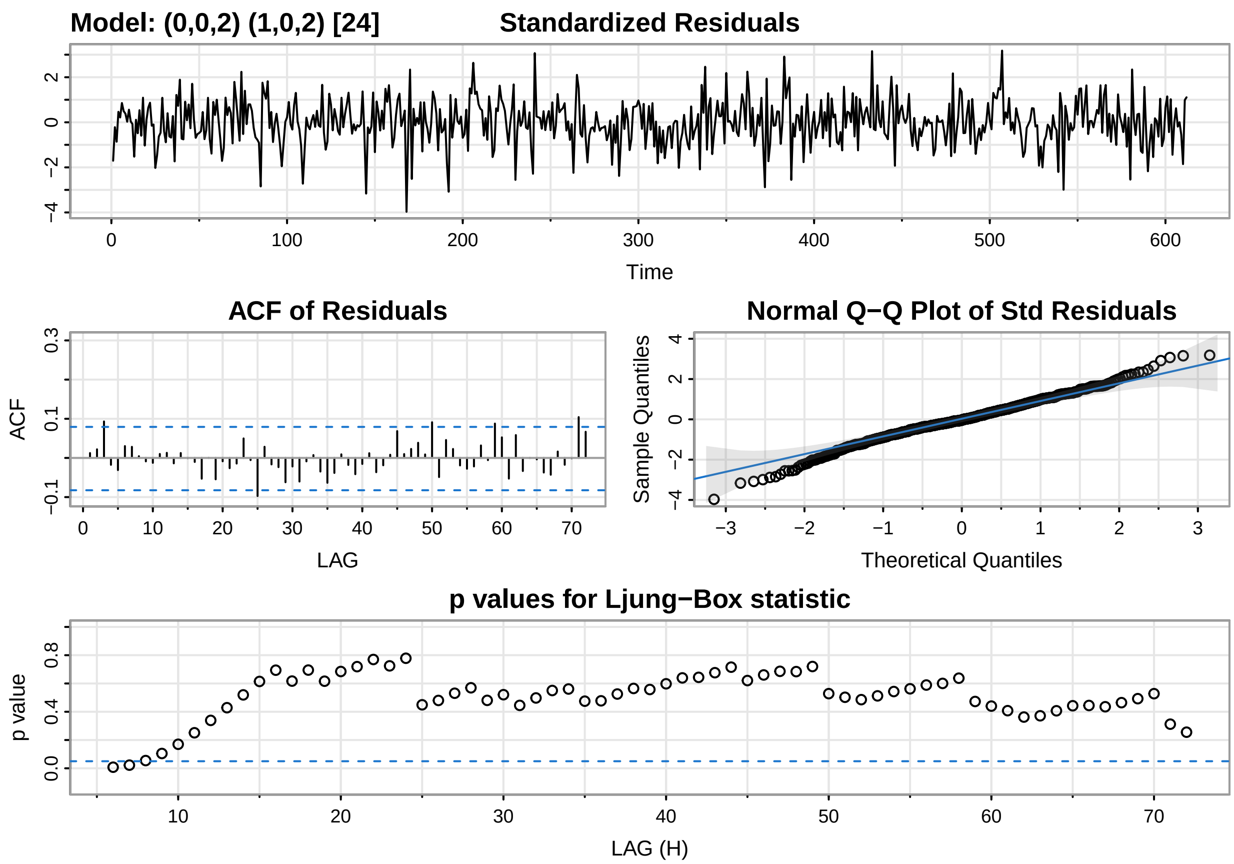
```r
for (ii in 1:(maxAR+1)){ ## ii-1, jj-1 are orders
  for (jj in 1:(maxMA+1)){
    for (kk in 1:(maxSAR+1)){ ## ii-1, jj-1 are orders
      for (ll in 1:(maxSMA+1)){
        BICdelta <- ((fits_all[[ii, jj, kk, ll]])$BIC - BICmin) * nn
        try({
        if (BICdelta < 10){
          print(paste0("(",
                       ii-1, ", ", jj-1, ") x (",
                       kk-1, ", ", ll-1, ")"))
          print(BICdelta)
        }
        })
      }
    }
  }
}
```

```
## [1] "(0, 1) x (0, 1)"
## [1] 8.72622
## [1] "(0, 1) x (1, 0)"
## [1] 8.562569
## [1] "(0, 1) x (1, 2)"
## [1] 7.812063
## [1] "(0, 2) x (0, 1)"
## [1] 5.17144
## [1] "(0, 2) x (1, 0)"
## [1] 5.316163
## [1] "(0, 2) x (1, 2)"
## [1] 4.630538
## [1] "(1, 0) x (0, 0)"
```

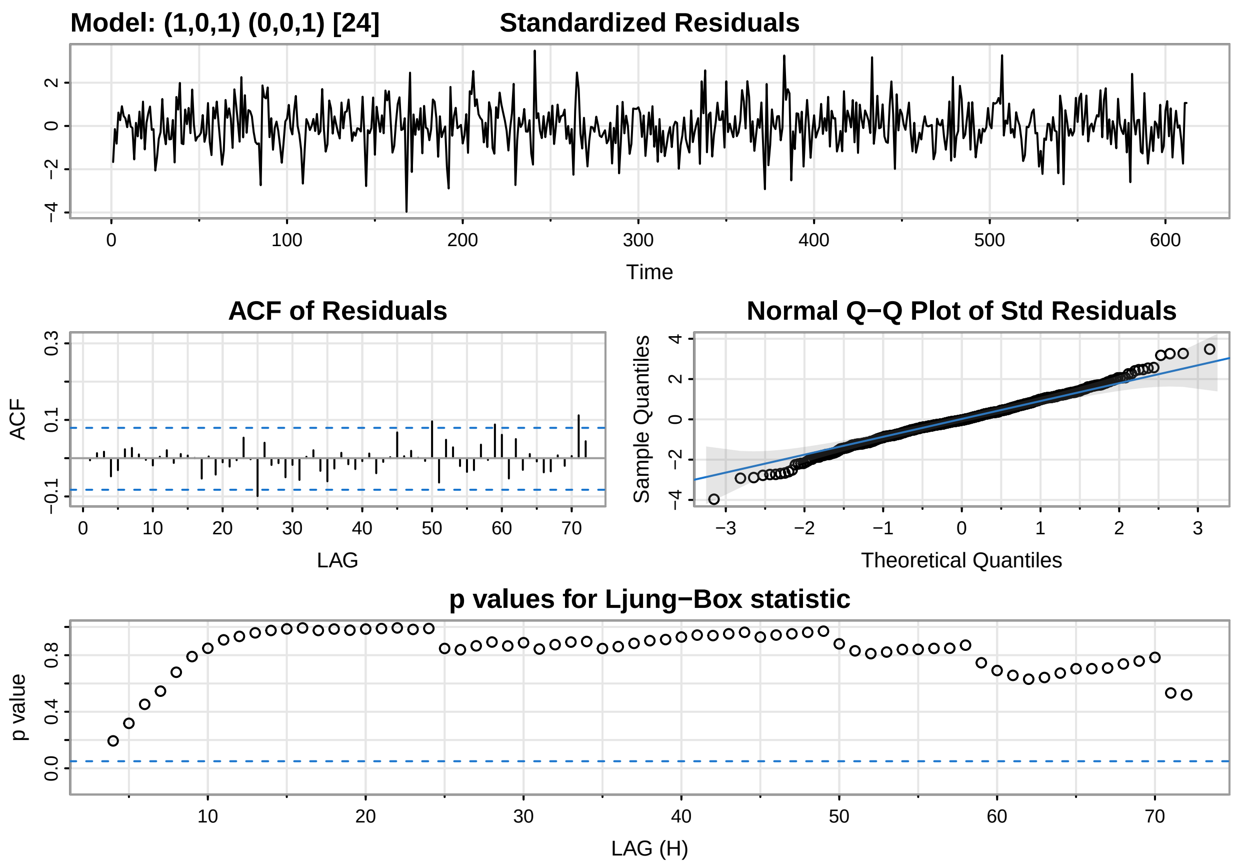
```
## [1] 5.353
## [1] "(1, 0) x (0, 1)"
## [1] 0
## [1] "(1, 0) x (0, 2)"
## [1] 6.29335
## [1] "(1, 0) x (1, 0)"
## [1] 0.01154564
## [1] "(1, 0) x (1, 1)"
## [1] 6.33923
## [1] "(1, 1) x (0, 0)"
## [1] 6.516897
## [1] "(1, 1) x (0, 1)"
## [1] 1.09494
## [1] "(1, 1) x (0, 2)"
## [1] 7.485202
## [1] "(1, 1) x (1, 0)"
## [1] 1.299693
## [1] "(1, 1) x (1, 1)"
## [1] 7.496148
## [1] "(1, 2) x (0, 1)"
## [1] 7.073858
## [1] "(1, 2) x (1, 0)"
## [1] 7.319145
## [1] "(2, 0) x (0, 0)"
## [1] 6.838106
## [1] "(2, 0) x (0, 1)"
## [1] 1.032804
## [1] "(2, 0) x (0, 2)"
## [1] 7.430157
## [1] "(2, 0) x (1, 0)"
## [1] 1.267602
## [1] "(2, 0) x (1, 1)"
## [1] 7.437682
## [1] "(2, 1) x (0, 1)"
## [1] 7.218453
## [1] "(2, 1) x (1, 0)"
## [1] 7.452697
```

Similarly, $(0,2) \times (1,2)$, $(1,1) \times (0,1)$ and $(1,1) \times (1,0)$ are reasonale models based on the their smaller AICc and BIC. Then check their diagnostic plots.

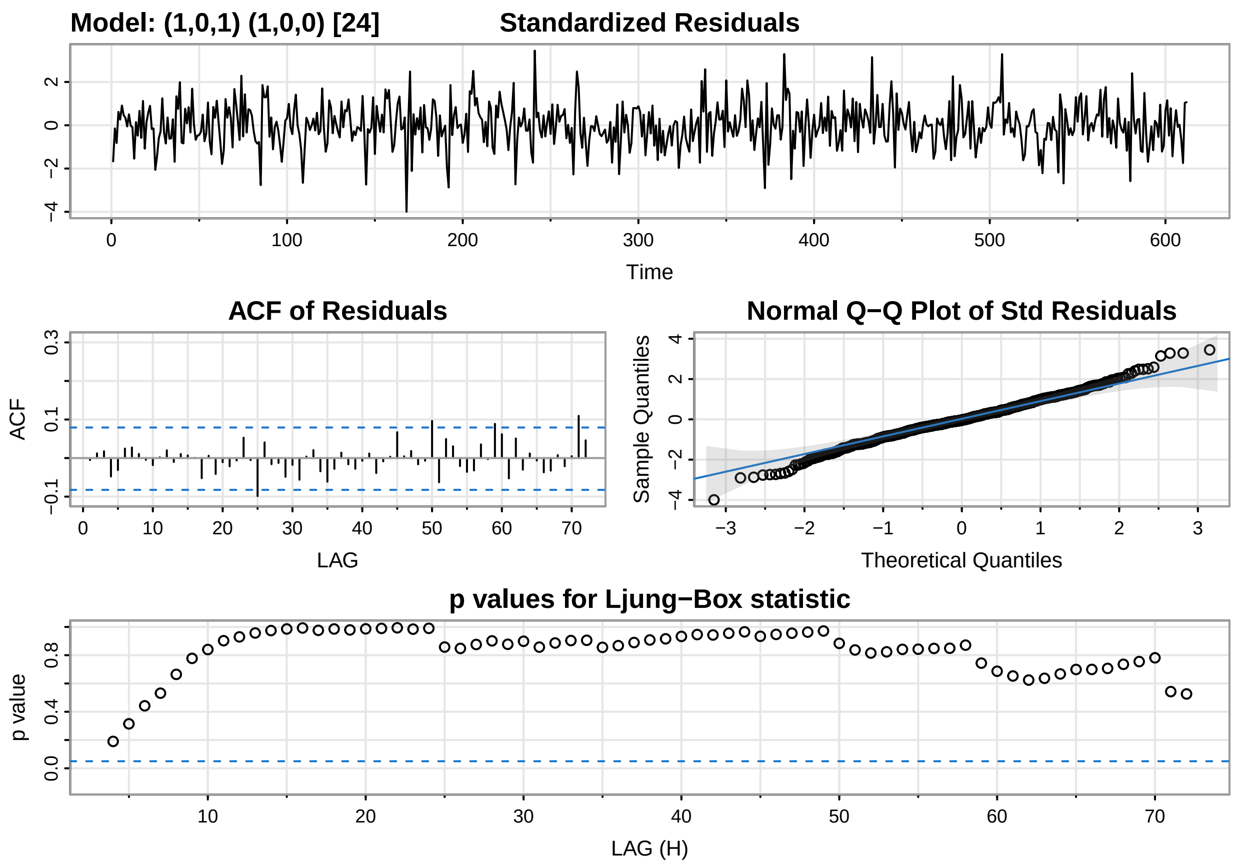
```
model <- sarima(TAVG,  p=0, d=0, q=2,
                P=1, D=0, Q=2, S=24,
                xreg = XX[,c(1, 3:13)])
```



```r
model.final <- sarima(TAVG,  p=1, d=0, q=1,
                 P=0, D=0, Q=1, S=24,
                 xreg = XX[,c(1, 3:13)])
```



```r
model <- sarima(TAVG,  p=1, d=0, q=1,
                P=1, D=0, Q=0, S=24,
                xreg = XX[,c(1, 3:13)])
```



As shown in the diagnostic plots, $(0,0,2)\times(1,0,2)_{24}$ model has several significant LB statistics so it is ruled out. Since, $(1,0,1)\times(0,0,1)_{24}$ has both slightly better AIC and BIC, so it is our final model. It is also reasonable to consider $(1,0,1)\times(1,0,0)_{24}$ since AR model is often easier to work with. In addition, this model beats the previous model with $s=12$. The model estimate is as follows:

```r
model.final$ttable
```

```
##                            Estimate      SE t.value p.value
## ar1                          0.5514  0.1040  5.3011  0.0000
## ma1                         -0.3018  0.1178 -2.5623  0.0106
## sma1                        -0.1403  0.0402 -3.4941  0.0005
## intercept                 -129.0752 30.2603 -4.2655  0.0000
## year(DATE)                   0.0719  0.0152  4.7438  0.0000
## as.factor(month(DATE))2      5.2034  0.6189  8.4081  0.0000
## as.factor(month(DATE))3     17.9402  0.6690 26.8145  0.0000
## as.factor(month(DATE))4     32.4310  0.6949 46.6711  0.0000
## as.factor(month(DATE))5     44.8709  0.7082 63.3561  0.0000
## as.factor(month(DATE))6     54.6486  0.7148 76.4559  0.0000
## as.factor(month(DATE))7     59.6278  0.7166 83.2147  0.0000
## as.factor(month(DATE))8     56.7990  0.7148 79.4655  0.0000
## as.factor(month(DATE))9     47.8598  0.7086 67.5455  0.0000
## as.factor(month(DATE))10    34.4354  0.6954 49.5188  0.0000
## as.factor(month(DATE))11    19.3847  0.6700 28.9313  0.0000
```

```
## as.factor(month(DATE))12    5.5402  0.6207  8.9254  0.0000
```