US Flu prediction

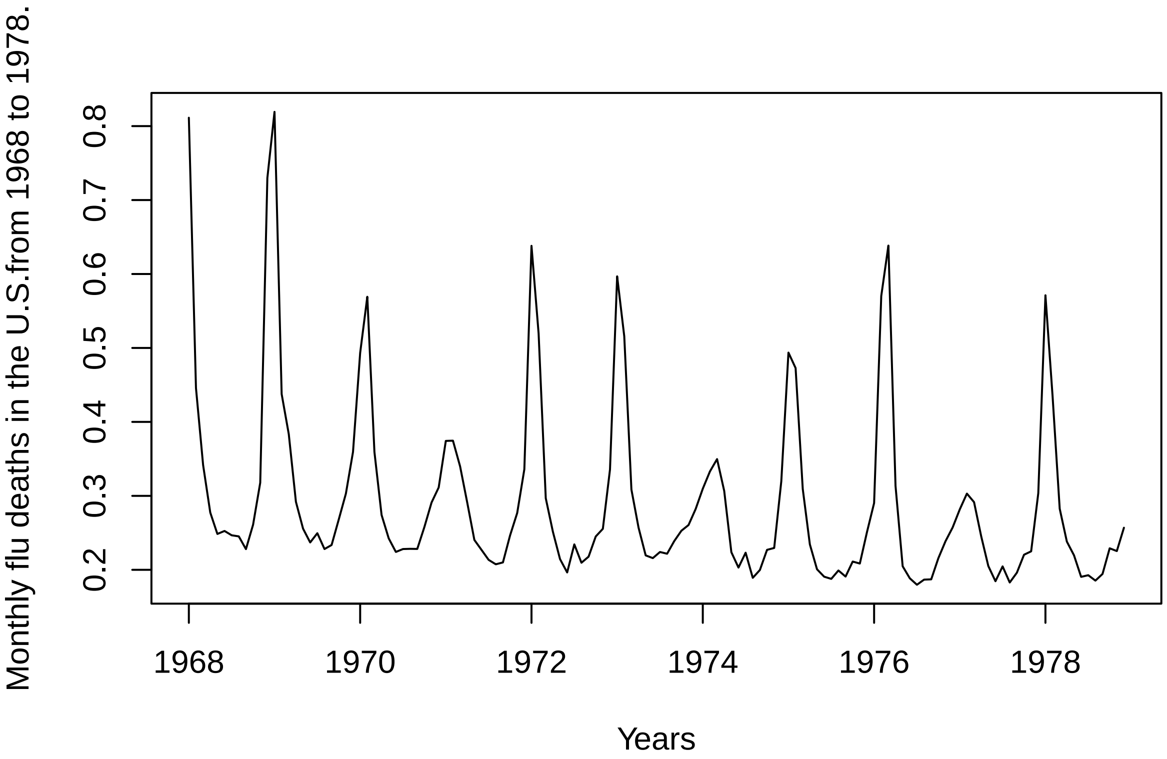
Yinzhi Chen

2022-04-07

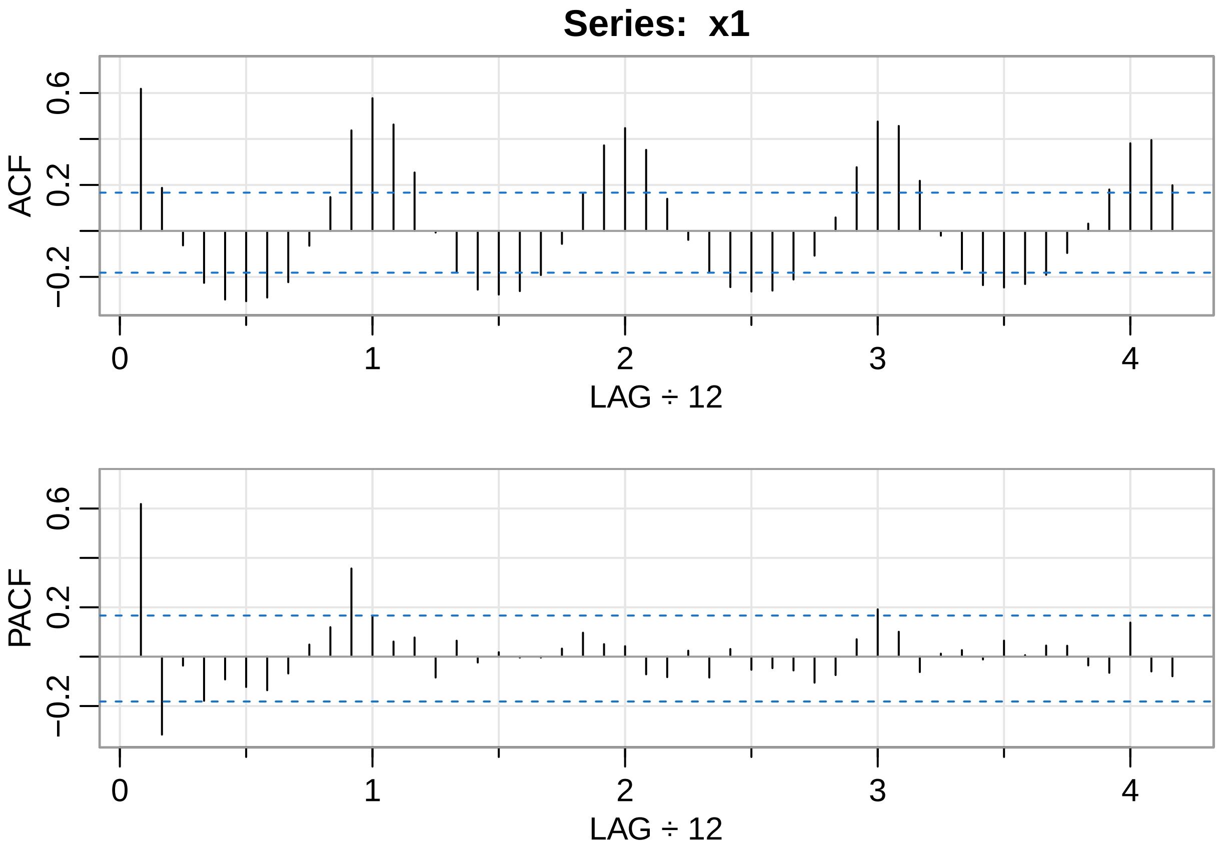
```
library(astsa)

length(flu)
```

```
## [1] 132
```

```
x1 = flu

plot.ts(x1, xlab = "Years", ylab = "Monthly flu deaths in the U.S.from 1968 to 1978.") #
```


Monthly flu deaths in the U.S.from 1968 to 1978.
0.2
0.3
0.4
0.5
0.6
0.7
0.8
1968
1970
1972
1974
1976
1978
Years

```
acf2(x1, 50)
```



```
##       [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11] [,12]
## ACF  0.62  0.19 -0.06 -0.23 -0.30 -0.31 -0.29 -0.22 -0.06  0.15  0.44  0.58
## PACF 0.62 -0.32 -0.04 -0.18 -0.09 -0.12 -0.14 -0.07  0.05  0.12  0.36  0.16
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
## ACF   0.46  0.25 -0.01 -0.18 -0.26 -0.28 -0.26 -0.19 -0.06  0.16  0.37  0.45
## PACF  0.06  0.08 -0.08  0.06 -0.02  0.02  0.00  0.00  0.03  0.10  0.05  0.04
##      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36]
## ACF   0.35  0.14 -0.04 -0.18 -0.24 -0.26 -0.26 -0.21 -0.11  0.06  0.28  0.48
## PACF -0.07 -0.08  0.02 -0.08  0.03 -0.05 -0.05 -0.06 -0.11 -0.07  0.07  0.19
##      [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.46  0.22 -0.02 -0.17 -0.24 -0.25 -0.23 -0.19 -0.10  0.03  0.18  0.38
## PACF  0.10 -0.06  0.01  0.03 -0.01  0.06  0.01  0.05  0.04 -0.04 -0.07  0.14
##      [,49] [,50]
## ACF   0.40  0.20
## PACF -0.06 -0.08
```

First, pick data “flu” from package “astsa”, the dataset includes Monthly pneumonia and influenza deaths in the U.S., 1968 to 1978. As the ongoing pandemic Covid 19, which also appears similar symptoms as flu, so I intend to use this dataset. Second, checking the length of the dataset, it has 132 observations which is more than 100 observations so it satisfied. Third, by observing the process is not stationary, and by observing the ACF plot which clearly displays a seasonal pattern, and ACF has a very slow decay to zero, so it indicates a differencing is needed to make it stationary.

```
diff1 = diff(x1)
plot.ts(diff1)
```

```
diff1 <- acf2(diff(x1))
```

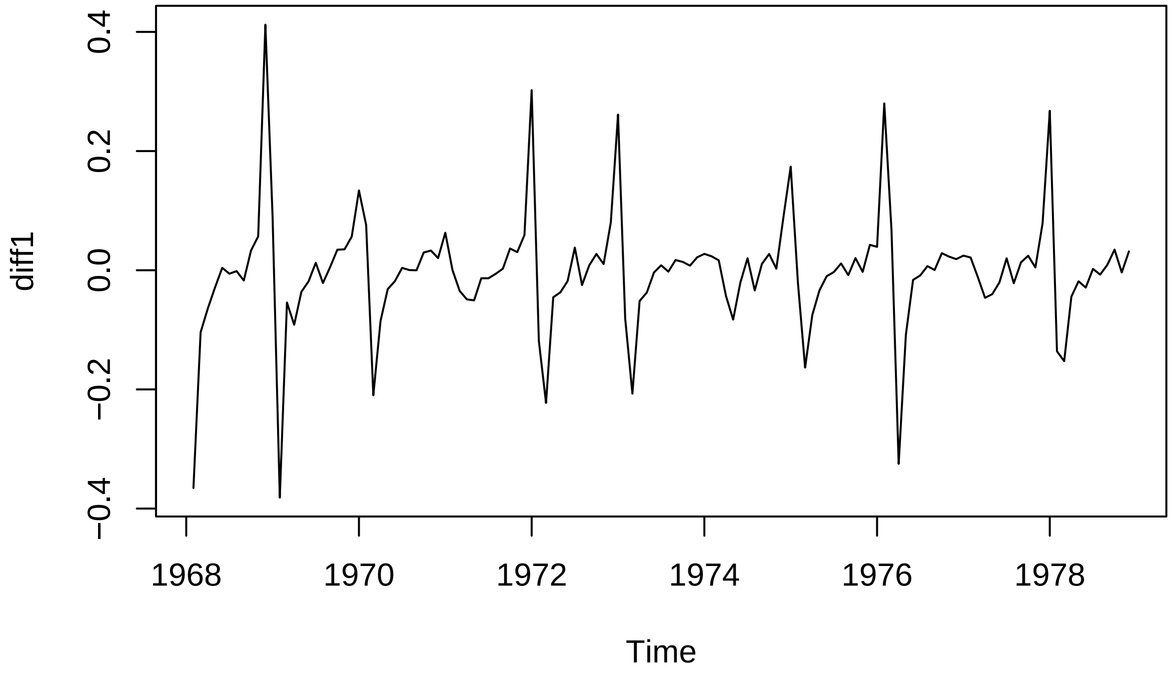
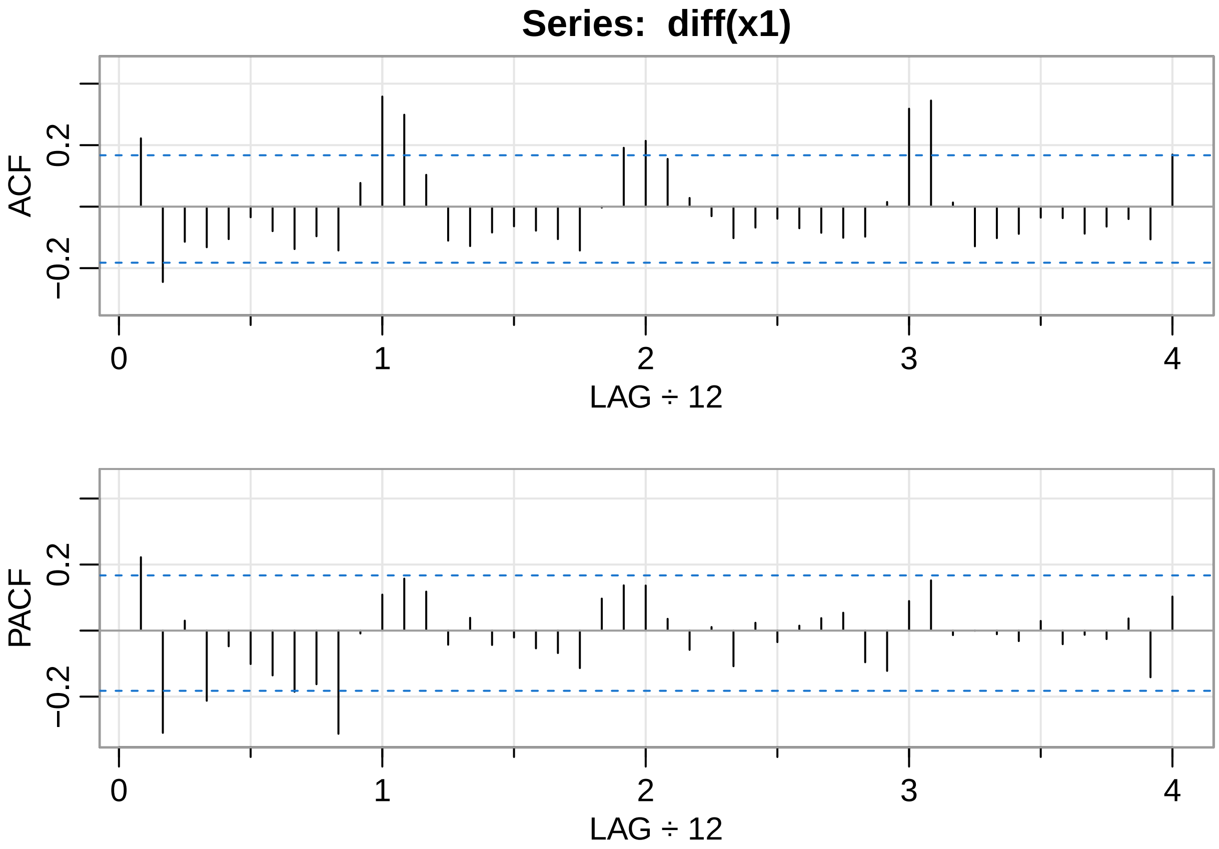
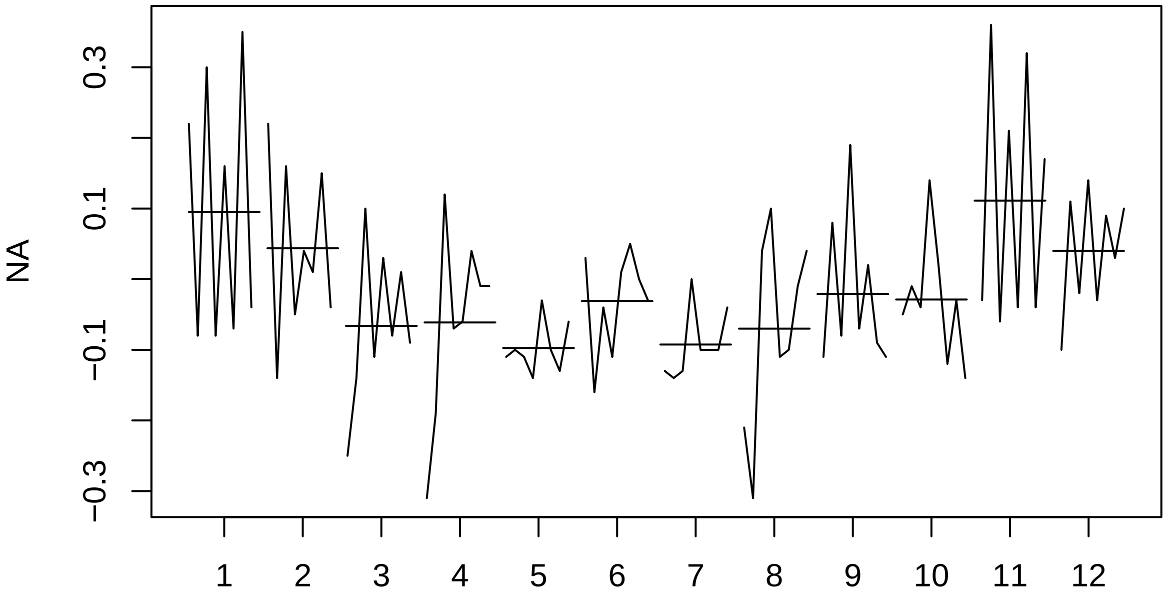


Figure 1: Monthly pneumonia and influenza death in the US from 1968 to 1978

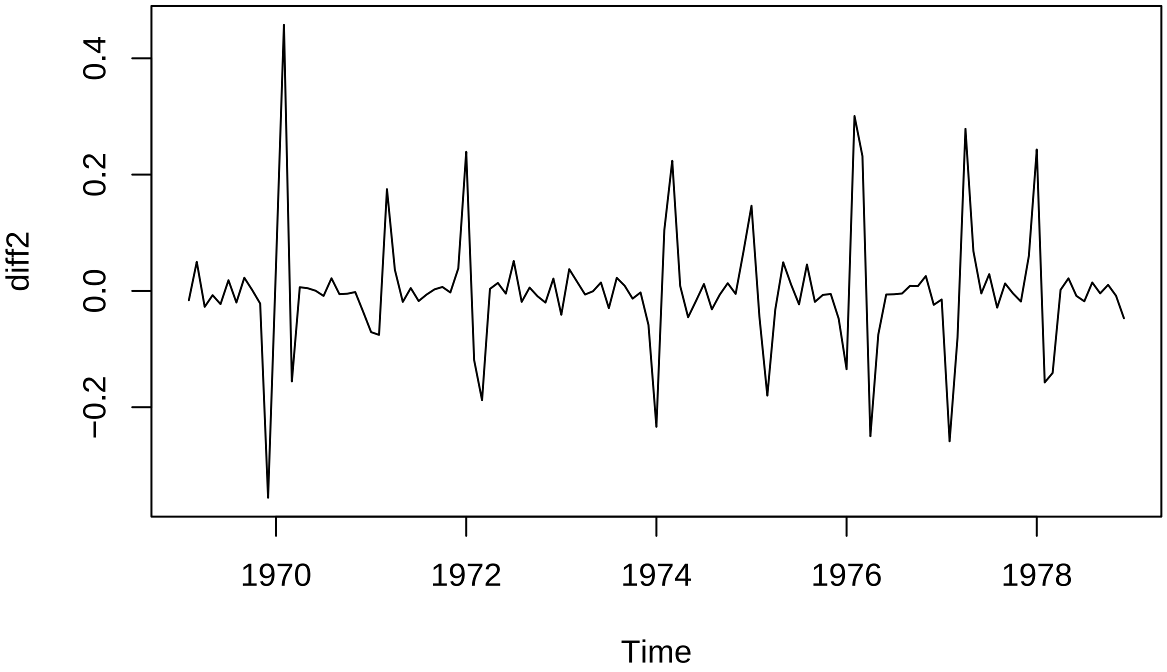


```
monthplot(diff1)
```



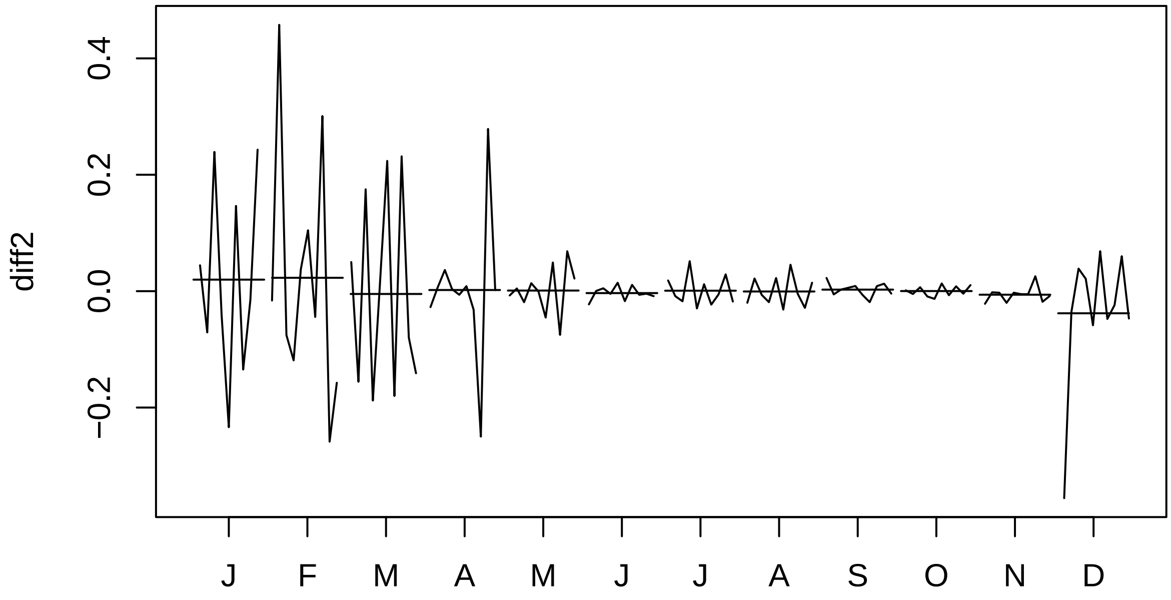
By observing at the ACF and monthplot after regular differencing, it is clear the there is still persistence in the seasons which we should get rid of the effect of it.

```
diff2 = diff(diff(x1, lag = 12))
plot(diff2)
```



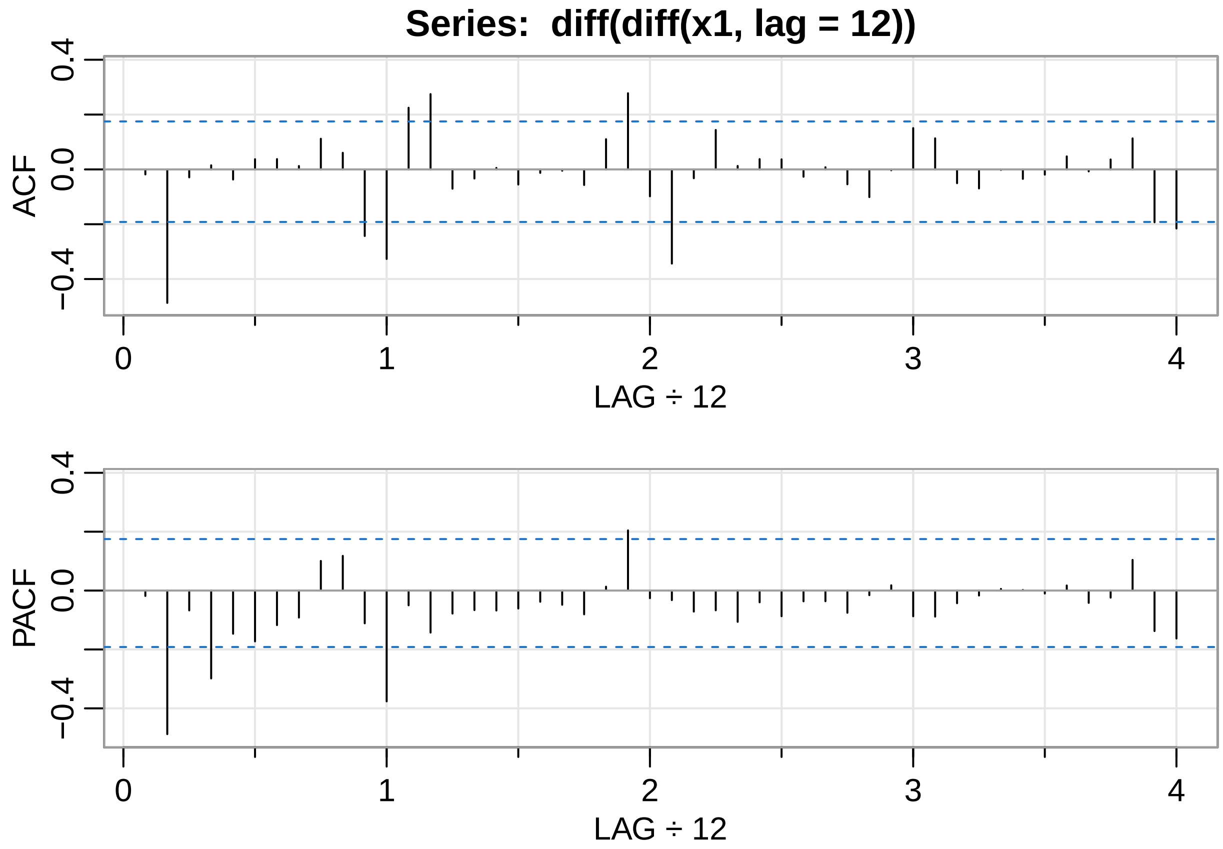
Then plot the new month plot after seasonal differencing

```
monthplot(diff2)
```



By observing the monthplot, the transformed data appears no seasonal trend and the transformed data appears to be stationary which is sutiable for us to fit model.

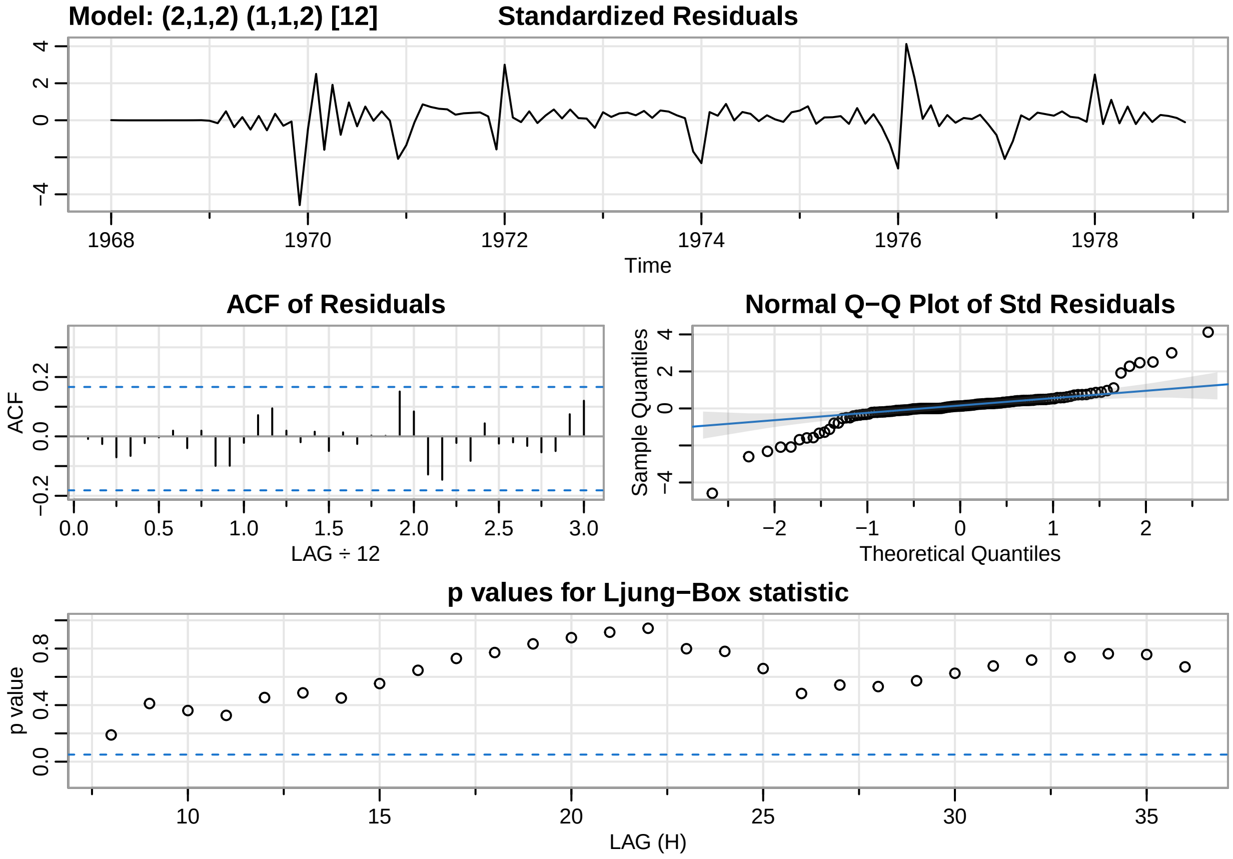
```
diff2 <- acf2(diff(diff(x1, lag = 12)))
```



The ACF cuts off after lag 2, PACF tails off which indicates MA(2) => p=2 PACF cuts off after lag 2, ACF tails off which indicates AR(2) => q=2 Then we used regular differencing once and seasonal differencing once for the flu data so our d and D values equal to 1. ACF cuts off after 2s, PACF tails off which indicates SMA(2) => Q = 2 PACF cuts off after 1s, ACF tails off which indicates SAR(1) => P = 1 Thus the first model is ARIMA = (2, 1, 2) x (1, 1, 2)s

```
sarima(x1, 2, 1, 2, 1, 1, 2, 12)
```

```
## initial  value -2.404760
## iter   2 value -2.582268
## iter   3 value -2.786351
## iter   4 value -2.787559
## iter   5 value -2.821686
## iter   6 value -2.866485
## iter   7 value -2.875810
## iter   8 value -2.884166
## iter   9 value -2.902373
## iter  10 value -2.907591
## iter  11 value -2.920110
## iter  12 value -2.923655
## iter  13 value -2.929914
## iter  14 value -2.934940
## iter  15 value -2.936766
## iter  16 value -2.939970
## iter  17 value -2.941181
## iter  17 value -2.941181
## iter  18 value -2.941685
## iter  19 value -2.941956
## iter  20 value -2.942208
## iter  21 value -2.942828
## iter  22 value -2.943101
## iter  23 value -2.943255
## iter  24 value -2.943464
## iter  25 value -2.944189
## iter  26 value -2.946355
## iter  26 value -2.946355
## iter  27 value -2.946357
## iter  27 value -2.946357
## iter  28 value -2.946357
## iter  28 value -2.946357
## iter  29 value -2.946357

## iter  29 value -2.946357

## iter  29 value -2.946357

## final  value -2.946357

## converged

## initial  value -2.629823

## iter   2 value -2.655819

## iter   3 value -2.662329

## iter   4 value -2.663195

## iter   5 value -2.663882

## iter   6 value -2.665723

## iter   7 value -2.667514

## iter   8 value -2.670653

## iter   9 value -2.675980

## iter  10 value -2.684194

## iter  11 value -2.687273

## iter  12 value -2.687912

## iter  13 value -2.688407

## iter  14 value -2.688853

## iter  15 value -2.688979

## iter  16 value -2.689129

## iter  17 value -2.689155

## iter  18 value -2.689211

## iter  19 value -2.689237

## iter  20 value -2.689247

## iter  21 value -2.689254

## iter  22 value -2.689257
## iter  23 value -2.689259
## iter  24 value -2.689260
## iter  25 value -2.689261
## iter  26 value -2.689261
## iter  27 value -2.689261
## iter  28 value -2.689262
## iter  29 value -2.689262
## iter  29 value -2.689262
## iter  29 value -2.689262
## final  value -2.689262
## converged
```



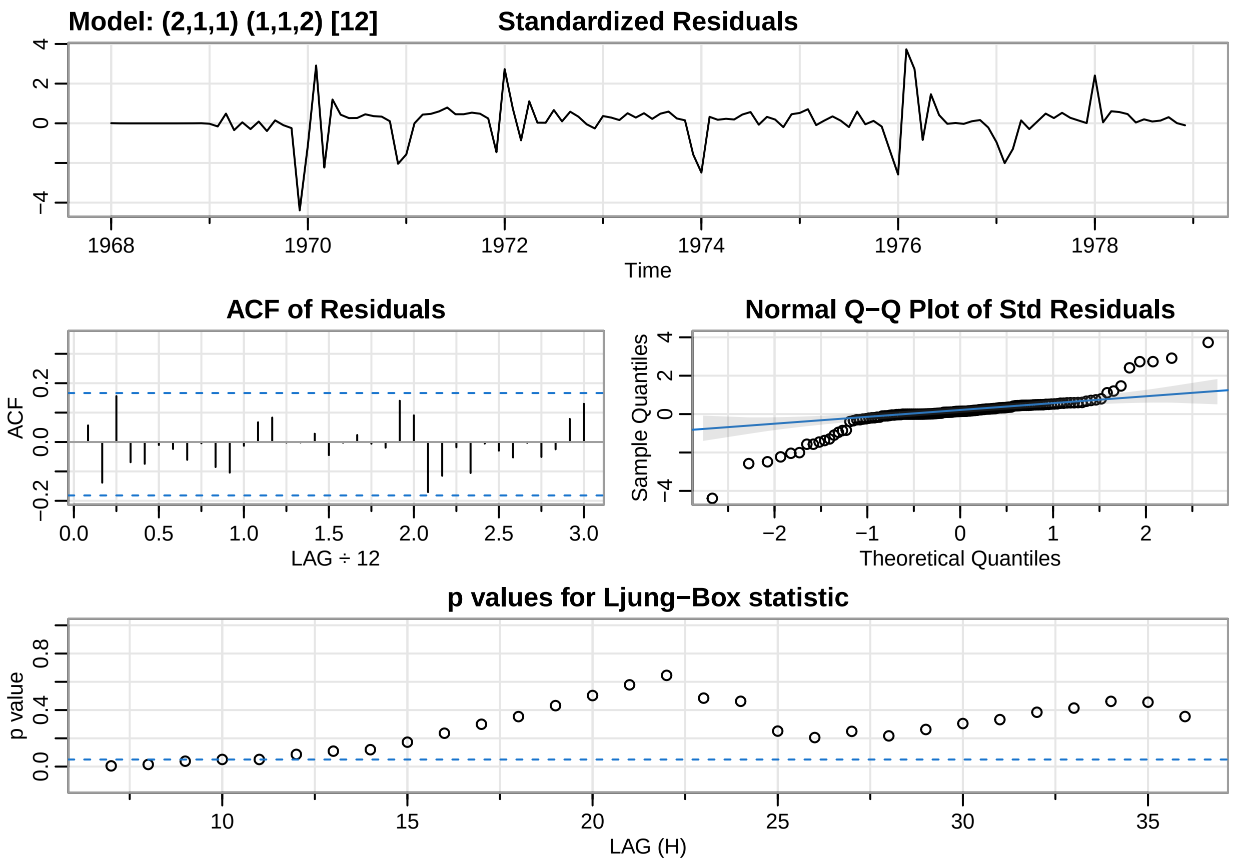
```
## $fit
```

```
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     include.mean = !no.constant, transform.pars = trans, fixed = fixed, optim.control
##         REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ar2      ma1      ma2     sar1    sma1     sma2
##       0.0047  -0.1163  -0.2751  -0.6849  -0.6714  0.2375  -0.7624
## s.e.  0.1613   0.1339   0.1450   0.1458   0.1015  0.3328   0.2625
##
## sigma^2 estimated as 0.003746:  log likelihood = 151.17,  aic = -286.34
##
## $degrees_of_freedom
## [1] 112
##
## $ttable
##      Estimate     SE t.value p.value
## ar1    0.0047 0.1613  0.0294  0.9766
## ar2   -0.1163 0.1339 -0.8690  0.3867
## ma1   -0.2751 0.1450 -1.8968  0.0604
## ma2   -0.6849 0.1458 -4.6977  0.0000
## sar1  -0.6714 0.1015 -6.6125  0.0000
## sma1   0.2375 0.3328  0.7136  0.4770
## sma2  -0.7624 0.2625 -2.9042  0.0044
##
## $AIC
## [1] -2.406192
##
## $AICc
## [1] -2.397713
##
## $BIC
## [1] -2.219361
```

The initial model have some insignificant p values, so we should try dropping one of the parameter from p and q. Try droppoing the q value by 1 and fit ARIMA = (2, 1, 1) x (1, 1, 2)s

```
mod1 = sarima(x1, 2, 1, 1, 1, 1, 2, 12, details = TRUE)
```

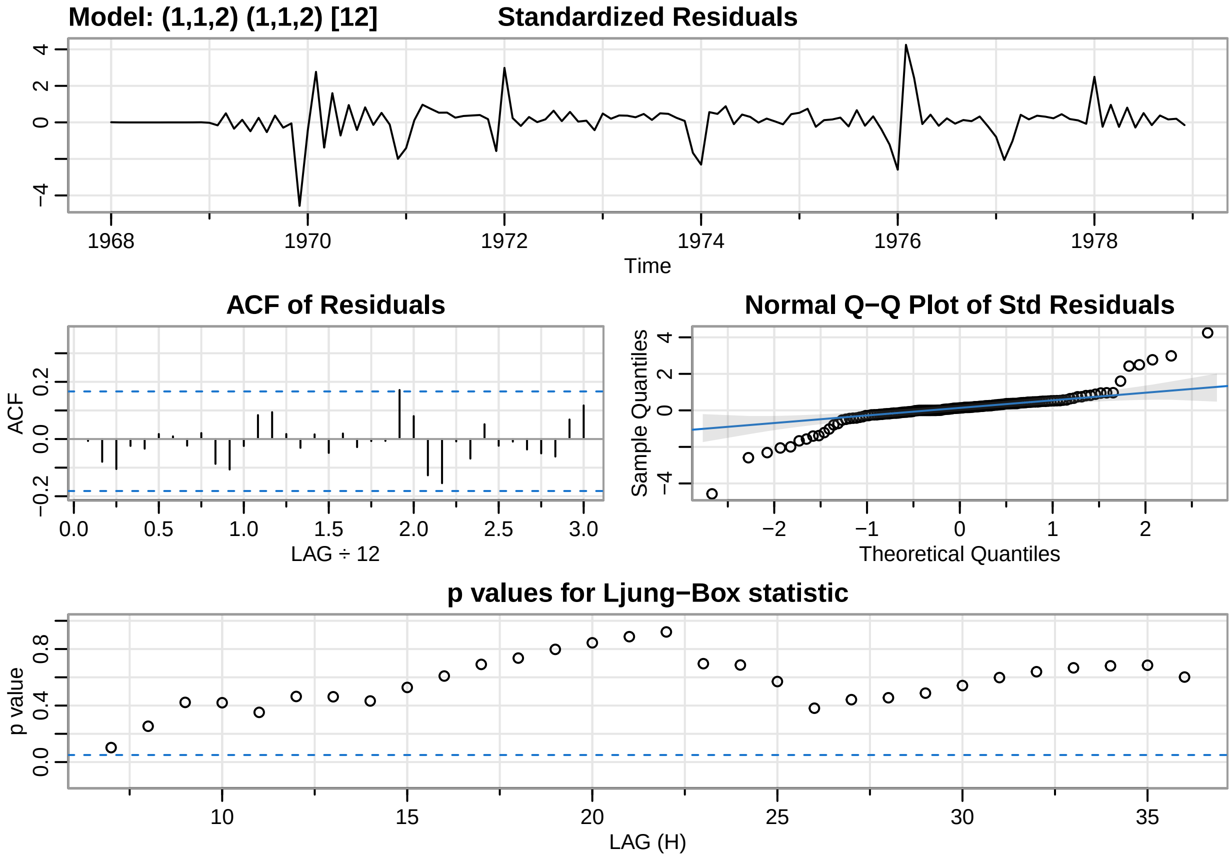
```
## initial  value -2.404760
## iter   2 value -2.612379
## iter   3 value -2.707888
## iter   4 value -2.717900
## iter   5 value -2.721599
## iter   6 value -2.722131
## iter   7 value -2.726635
## iter   8 value -2.728252
## iter   9 value -2.734883
## iter  10 value -2.737215
## iter  11 value -2.740616
## iter  12 value -2.746340
## iter  13 value -2.747080
## iter  14 value -2.747215
## iter  15 value -2.747233

## iter  16 value -2.747236

## iter  17 value -2.747248

## iter  18 value -2.747255

## iter  19 value -2.747261

## iter  20 value -2.747262

## iter  21 value -2.747263

## iter  22 value -2.747264

## iter  23 value -2.747264

## iter  23 value -2.747264

## iter  23 value -2.747264

## final  value -2.747264

## converged

## initial  value -2.506449

## iter   2 value -2.526514

## iter   3 value -2.528020

## iter   4 value -2.532156

## iter   5 value -2.540196

## iter   6 value -2.554595

## iter   7 value -2.583252

## iter   8 value -2.602565

## iter   9 value -2.638973

## iter  10 value -2.651585

## iter  11 value -2.653158

## iter  12 value -2.654103

## iter  13 value -2.654491

## iter  14 value -2.655442
## iter  15 value -2.655543
## iter  16 value -2.655689
## iter  17 value -2.655811
## iter  18 value -2.655892
## iter  19 value -2.655938
## iter  20 value -2.655970
## iter  21 value -2.655974
## iter  22 value -2.655981
## iter  23 value -2.655986
## iter  24 value -2.655988
## iter  25 value -2.655991
## iter  26 value -2.655992
## iter  27 value -2.655992
## iter  28 value -2.655993
## iter  29 value -2.655993
## iter  29 value -2.655993
## iter  29 value -2.655993
## final  value -2.655993
## converged
```



Try droppoing the p value by 1 and fit ARIMA = (1, 1, 2) x (1, 1, 2)s

```
sarima(x1, 1, 1, 2, 1, 1, 2, 12)
```

```
## initial  value -2.395550
## iter   2 value -2.723254
## iter   3 value -2.776145
## iter   4 value -2.827836
## iter   5 value -2.833472
## iter   6 value -2.862915
## iter   7 value -2.865292
## iter   8 value -2.868129
## iter   9 value -2.869548
## iter  10 value -2.870624
## iter  11 value -2.871507
## iter  12 value -2.872171
## iter  13 value -2.872332
## iter  14 value -2.872340
## iter  15 value -2.872341
## iter  16 value -2.872341
## iter  17 value -2.872341
## iter  18 value -2.872341
## iter  19 value -2.872341
## iter  19 value -2.872341
## iter  19 value -2.872341
## final  value -2.872341
## converged
## initial  value -2.575255
## iter   2 value -2.649796
## iter   3 value -2.655056
## iter   4 value -2.657873
## iter   5 value -2.659139
## iter   6 value -2.659648
## iter   7 value -2.661586
## iter   8 value -2.666519
## iter   9 value -2.668699
## iter  10 value -2.671587
## iter  11 value -2.681762
## iter  12 value -2.683192
## iter  13 value -2.684242
## iter  14 value -2.685606
## iter  15 value -2.685885

## iter  16 value -2.686008

## iter  17 value -2.686020

## iter  18 value -2.686037

## iter  19 value -2.686044

## iter  20 value -2.686062

## iter  21 value -2.686074

## iter  22 value -2.686081

## iter  23 value -2.686082

## iter  24 value -2.686082

## iter  25 value -2.686082

## iter  26 value -2.686082

## iter  26 value -2.686082

## iter  26 value -2.686082

## final  value -2.686082

## converged
```

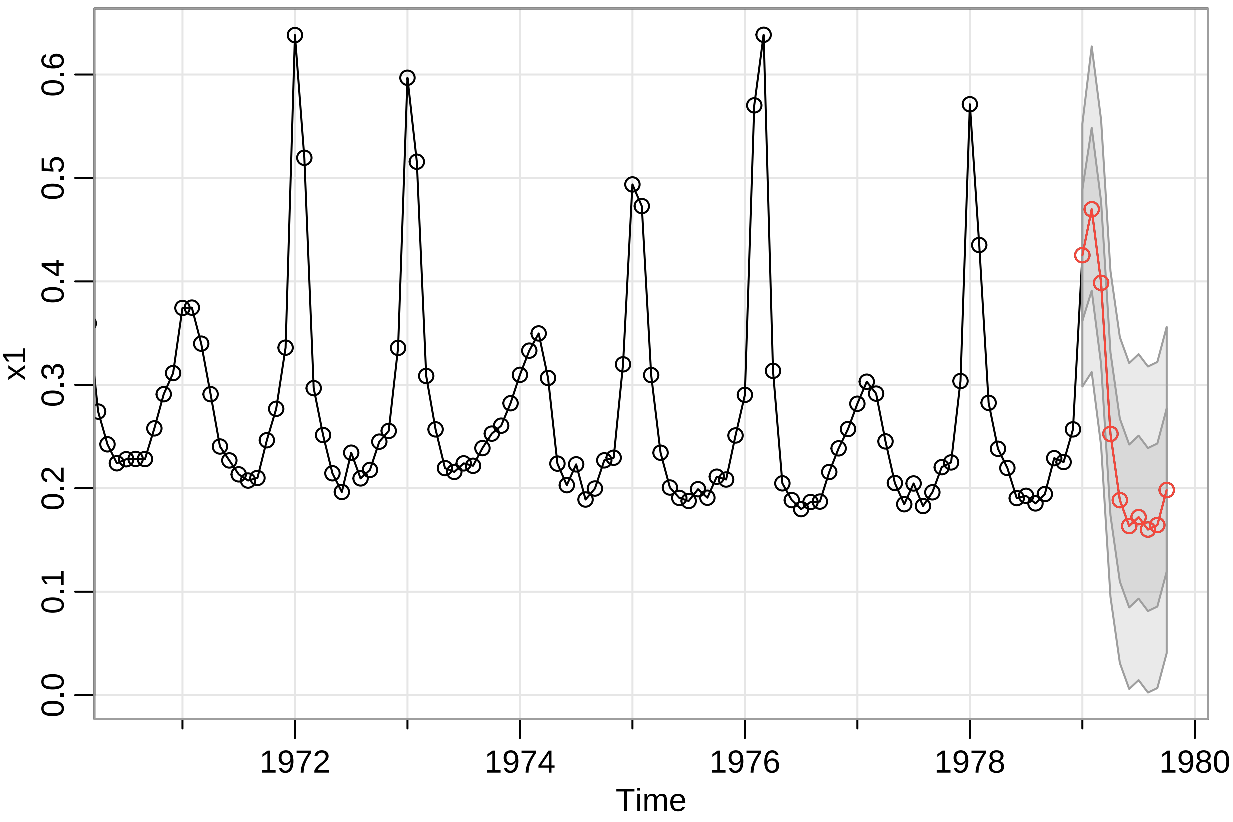


```
## $fit
##
## Call:
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),
##     include.mean = !no.constant, transform.pars = trans, fixed = fixed, optim.control
##         REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ma1      ma2     sar1    sma1     sma2
##      -0.0646  -0.2057  -0.7666  -0.6786  0.2422  -0.7576
## s.e.  0.1266   0.0997   0.0883   0.1006  0.4220   0.3266
##
## sigma^2 estimated as 0.003766:  log likelihood = 150.79,  aic = -287.58
```

```
##
## $degrees_of_freedom
## [1] 113
##
## $ttable
##      Estimate     SE t.value p.value
## ar1   -0.0646 0.1266 -0.5099  0.6111
## ma1   -0.2057 0.0997 -2.0634  0.0414
## ma2   -0.7666 0.0883 -8.6819  0.0000
## sar1  -0.6786 0.1006 -6.7481  0.0000
## sma1   0.2422 0.4220  0.5739  0.5672
## sma2  -0.7576 0.3266 -2.3195  0.0222
##
## $AIC
## [1] -2.416641
##
## $AICc
## [1] -2.410338
##
## $BIC
## [1] -2.253163
```

Comparing the 3 fitted model, the first model have some insignificant p values, then by dropping parameters to ensure the p values are significant. The second model drops q value by 1 which more p values become significant but there has some p value points for Ljung-Box statistic are below the blue line which doesn't satisfy p-test. The third model drops p value by 1 which more p values become significant and all of the p values points for Ljung-Box statistic are above the blue line. Thus, the third ARIMA model ARIMA = (1, 1, 2) x (1, 1, 2)s is chosen. By observing the new fitted model, standard residual doesn't follow any patterns or trend, ACF also lie between the upper and lower blue lines, normal QQ plots are mostly around the residual line with very few outliers, all of the residual diagnostics are above the blue line, which means the results are significant, so the model is feasible for future predictions.

```
pred1 <- sarima.for(x1, 10, 1, 1, 2, 1, 1, 2, 12 )
```

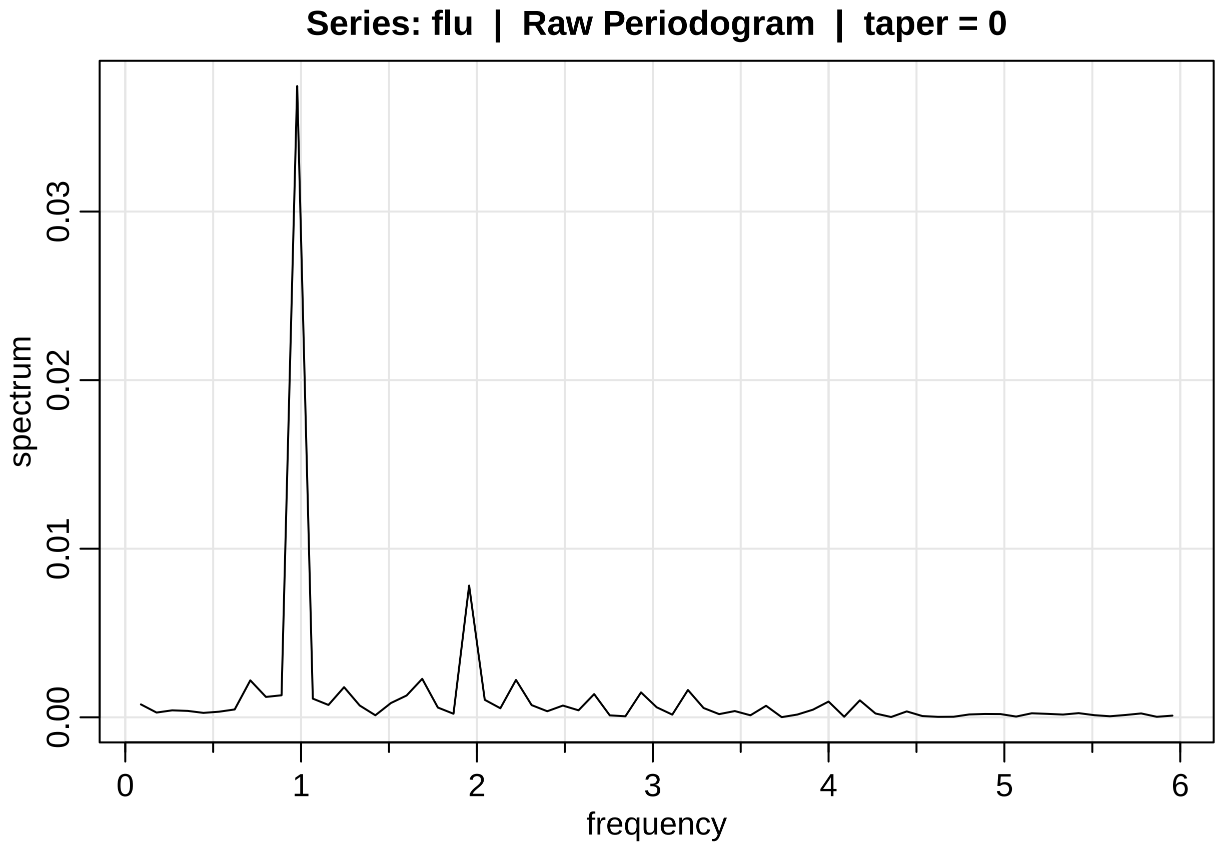


```
year <- c(1:10)
upper = pred1$pred+qnorm(0.975)*pred1$se # 5% upper Prediction interval
lower = pred1$pred-qnorm(0.975)*pred1$se # 5% lower Prediction interval
(data.frame("Prediction"=pred1$pred,"95% PI Lower Bound"=lower,"95% PI Upper Bound"=uppe
```

```
##    Prediction X95..PI.Lower.Bound X95..PI.Upper.Bound
## 1   0.4253207          0.300891364          0.5497500
## 2   0.4697623          0.315466934          0.6240576
## 3   0.3985773          0.244307420          0.5528472
## 4   0.2525818          0.098267410          0.4068962
## 5   0.1885257          0.034178284          0.3428732
## 6   0.1634951          0.009113839          0.3178763
## 7   0.1720719          0.017657031          0.3264869
## 8   0.1600831          0.005634464          0.3145317
## 9   0.1644375          0.009955228          0.3189199
## 10  0.1982996          0.043783849          0.3528153
```

By observing the prediction interval dataframe for the next 10 values, we observe that the predictions are following a seasonal pattern with slow decays.

```
flu.per = mvspec(flu, log = "no")
```



```
p1 <- flu.per$details[order(flu.per$details[,3],decreasing = TRUE),]
p1[1,1];p1[2,1];p1[3,1]
```

```
## frequency
##    0.9778

## frequency
##    1.9556

## frequency
##    1.6889
```

```
cat("cycles are occuring at", 1/p1[1,1],1/p1[2,1],1/p1[3,1])
```

```
## cycles are occuring at 1.022704 0.511352 0.5921014
```

```
library(MASS)
flu.u1 = 2*p1[1,3]/qchisq(.025,2)
flu.l1 = 2*p1[1,3]/qchisq(.975,2)
flu.u2 = 2*p1[2,3]/qchisq(.025,2)
flu.l2 = 2*p1[2,3]/qchisq(.975,2)
flu.u3 = 2*p1[3,3]/qchisq(.025,2)
flu.l3 = 2*p1[3,3]/qchisq(.975,2)
```

```
Res <- data.frame(Series=c(rep("flu",3)),
Dominant.Freq=c(p1[1,1],p1[2,1],p1[3,1]),Spec=c(p1[1,3],p1[2,3],p1[3,3]),
lower = c(flu.l1,flu.l2,flu.l3),
Upper = c(flu.u1,flu.u2,flu.u3))
Res
```

```
##   Series Dominant.Freq   Spec        lower      Upper
## 1    flu        0.9778 0.0374 0.0101385801 1.47722109
## 2    flu        1.9556 0.0078 0.0021144632 0.30808354
## 3    flu        1.6889 0.0023 0.0006234956 0.09084515
```

We can't establish the significance of the first peak since the first periodogram ordinate is 0.0374, which lies in the confidence intervals of the second peak. We can't establish the significance of the second peak since the second periodogram ordinate is 0.0078, which lies in the confidence intervals of the third peak. We can't establish the significance of the third peak since the third periodogram ordinate is 0.0023, which lies in the confidence intervals of the second peak.