Midterm II R Appendix

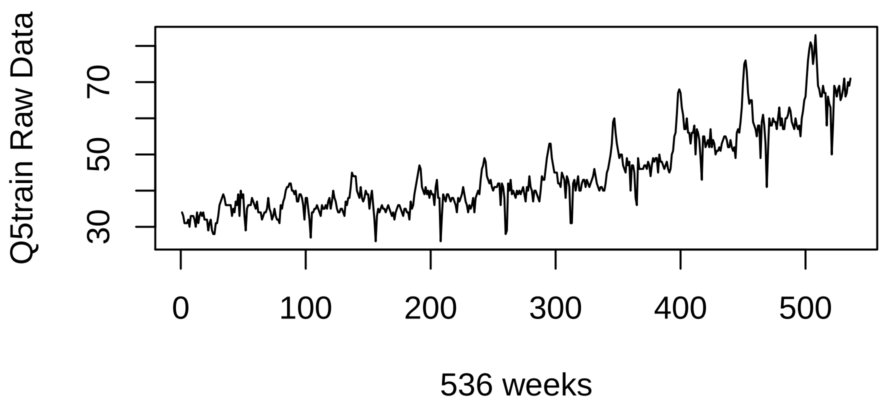
Huidi Wang

April 19, 2016

R Appendix for Q5 Dataset

(1) Exploratory data

```
#Extract data Q5train from Q5train.csv
Q5train.raw=read.delim(file="q5train.csv")
Q5train_data=Q5train.raw[1:536,1]
len5=length(Q5train_data)
Q5train=rep(0,len5)
for (i in 1:len5){
  Q5train[i]=as.numeric(unlist(strsplit(as.character(Q5train_data[i]),","))[2])
}
plot(Q5train, type="l", xlab="536 weeks", ylab="Q5train Raw Data")
```




```
summary(Q5train)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   26.00   36.00   41.00   44.71   52.00   83.00
```

```
#We have total 536 weekly data in Q2train with min=26, max=83, and mean=44.71
```

(2) Data Transformation

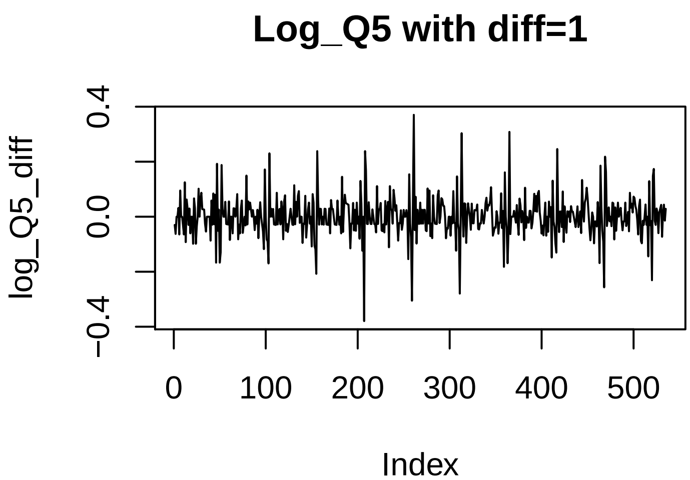
Here attached the ACF and pACF of Q5train after taking log.

```
log_Q5train=log(Q5train)
#acf(log_Q5train, lag.max=100)
#pacf(log_Q5train, lag.max=100)
```

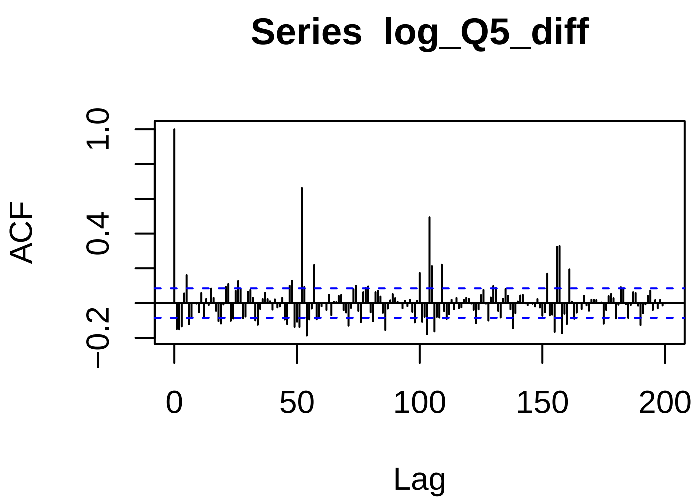
(3) Deal with Trend and Seasonality

diff=1

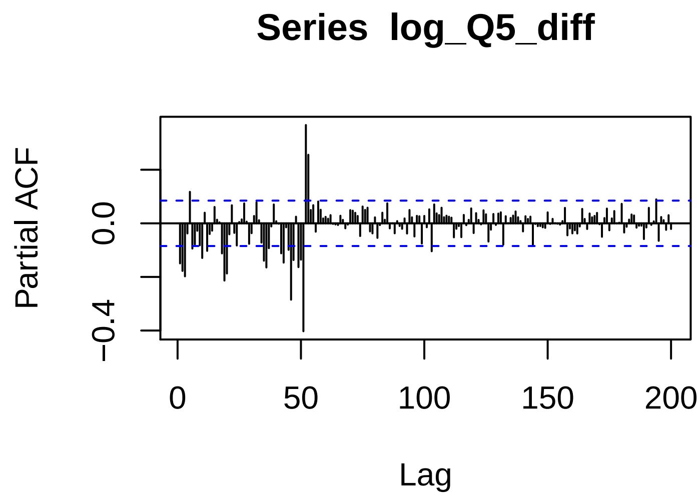
```
log_Q5_diff=diff(log_Q5train)
plot(log_Q5_diff, type="l", main="Log_Q5 with diff=1")
```




```
acf(log_Q5_diff, lag.max=200)
```



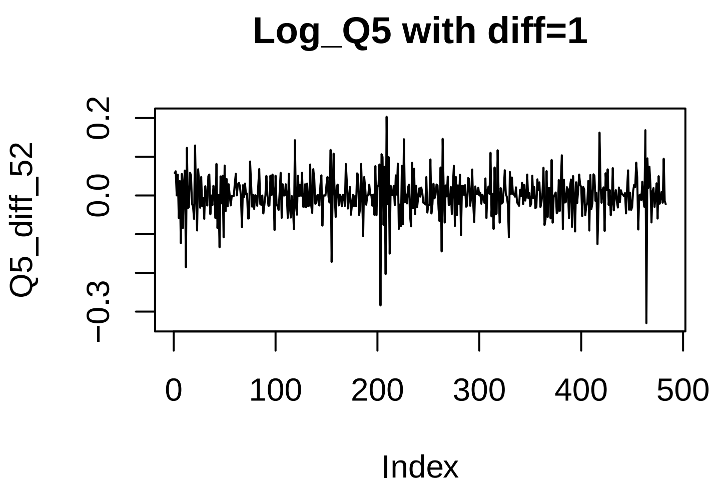

```
#$acf: [53,]  0.6616793396 [105,]  0.4937017279  [157,]  0.3233697281
pacf(log_Q5_diff, lag.max=200)
```



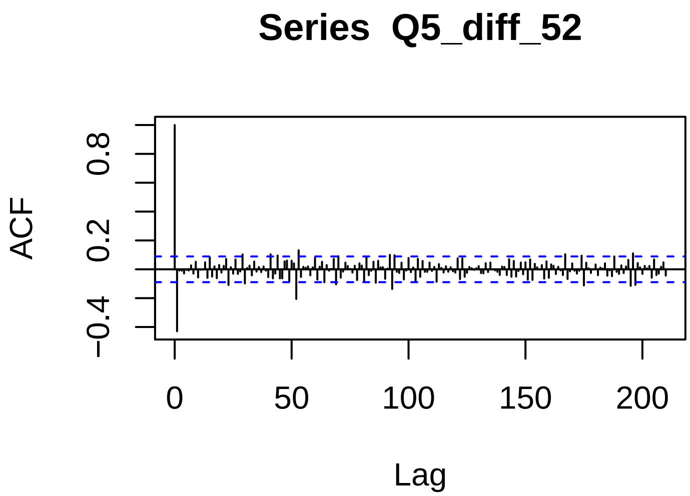

```
#In Partial ACF, only the spikes around lag=52 have higher values comparing to other lags' pacfs.
```

diff=52

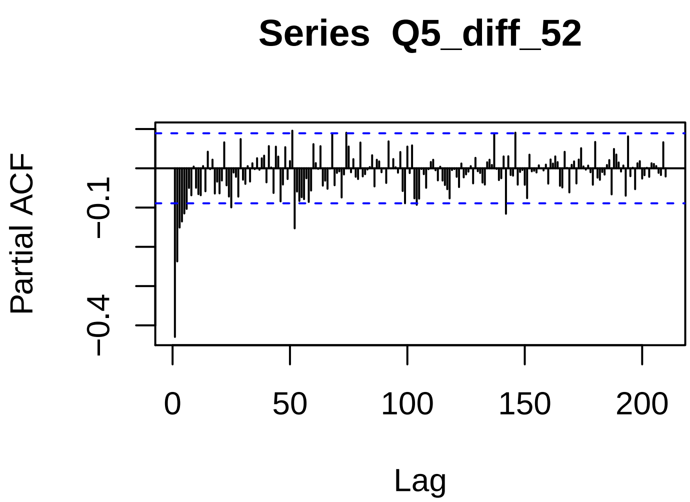
```
Q5_diff_52=diff(log_Q5_diff,52)     #diff=52 to remove seasonality
plot(Q5_diff_52, type="l",main="Log_Q5 with diff=1")
```




```
#Seems seasonality and trend have been moved. Data Q5_diff_52 is stationary now after diff=1 and diff=5
acf(Q5_diff_52, lag.max=210)
```




```
#pacf: [52,]  0.0414221978 [53,] -0.2060435605 [54,]  0.1317868379 [55,] -0.0532097436
pacf(Q5_diff_52, lag.max=210)
```



(4) Fit an ARMA model to the residuals after removing trend and sesonality

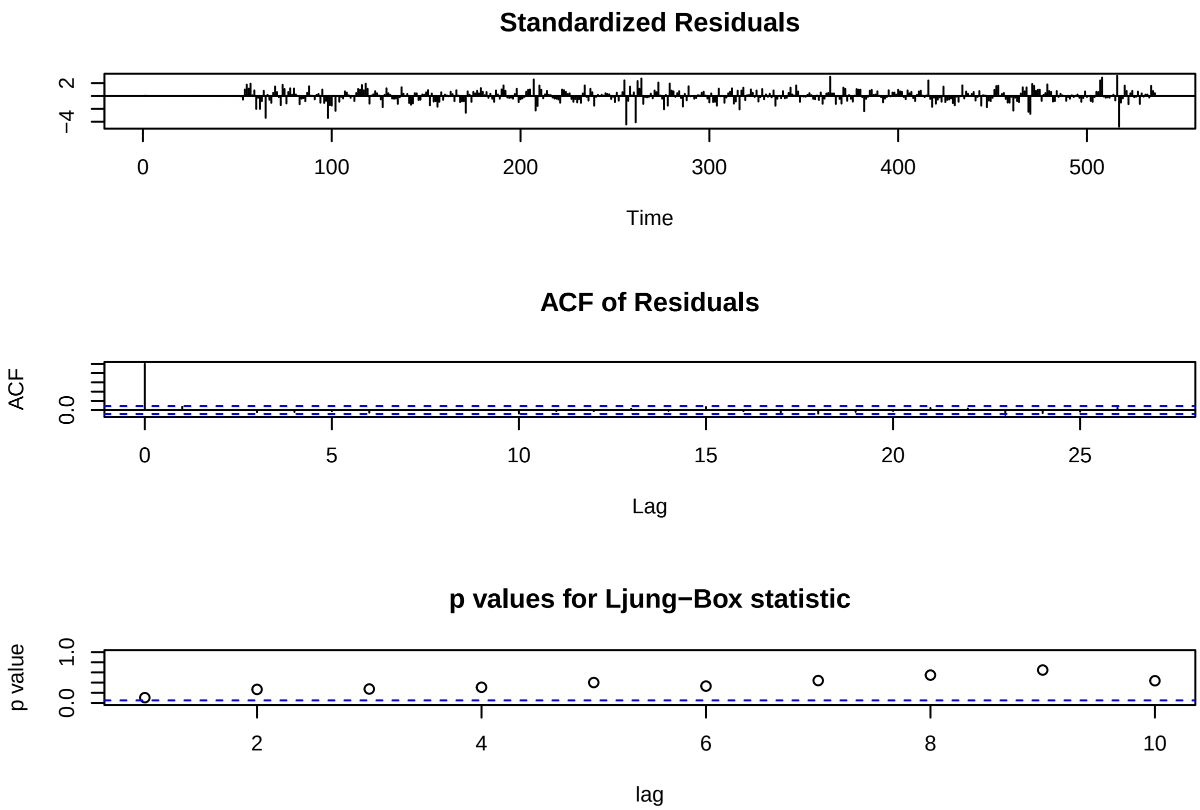
(5) Model Diagnostics (check if the fitted ARMA is adequate.)

1)Residuals and P-value   Let's show fitted model MA(1) $\times$ AR(1)6 through arima function as follow.

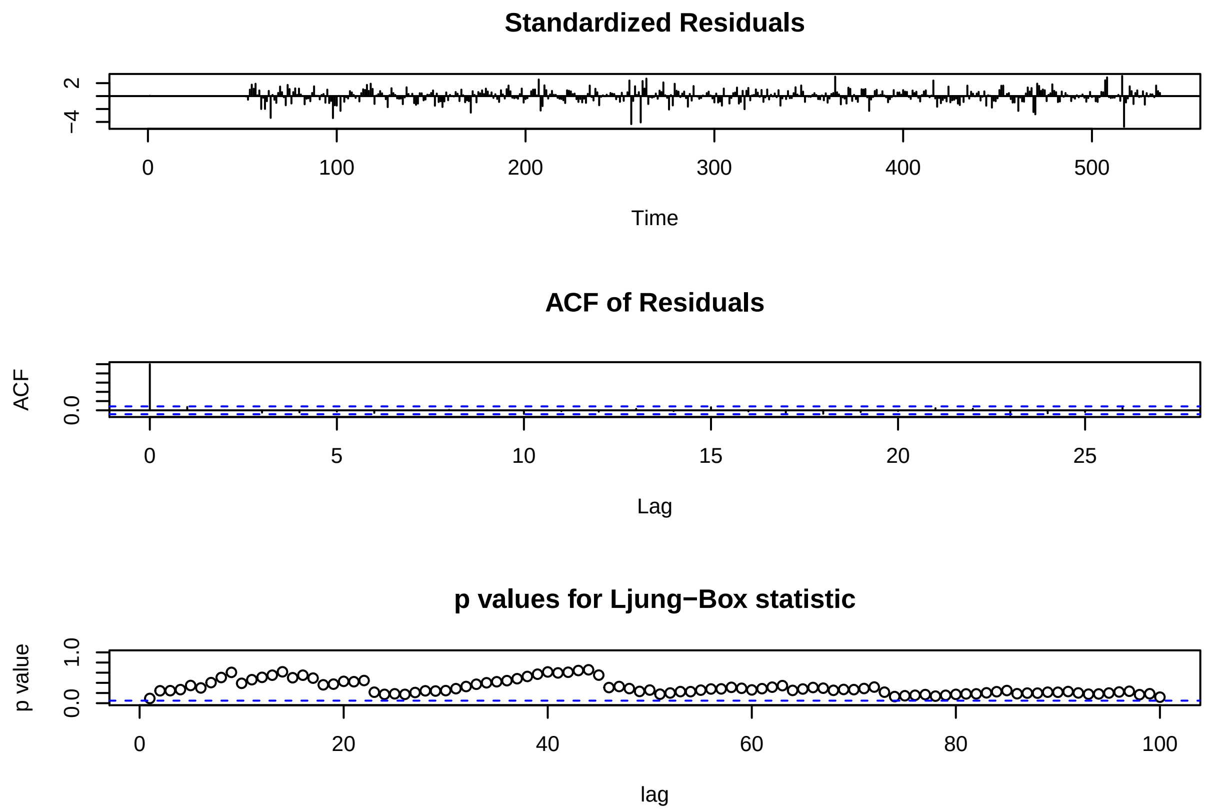
```
Q5_0 <- arima(log_Q5train, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1), period = 52))
Q5_0
```

```
##
## Call:
## arima(x = log_Q5train, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1),
##     period = 52))
##
## Coefficients:
##           ma1     sma1
##       -0.7262  -0.2807
## s.e.   0.0380   0.0468
##
## sigma^2 estimated as 0.002021:  log likelihood = 810.46,  aic = -1614.93
```

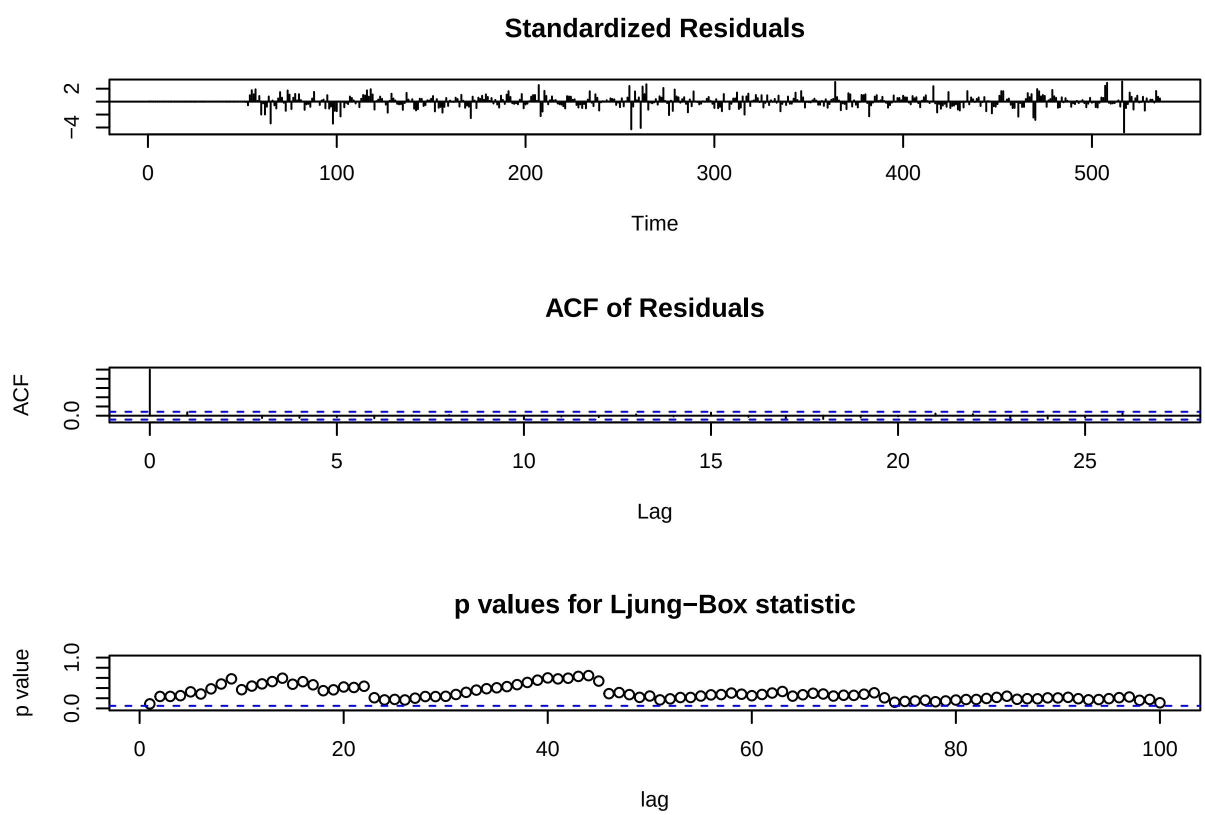
```r
tsdiag(Q5_0) #aic = -1614.93 trying to minimum aic
```



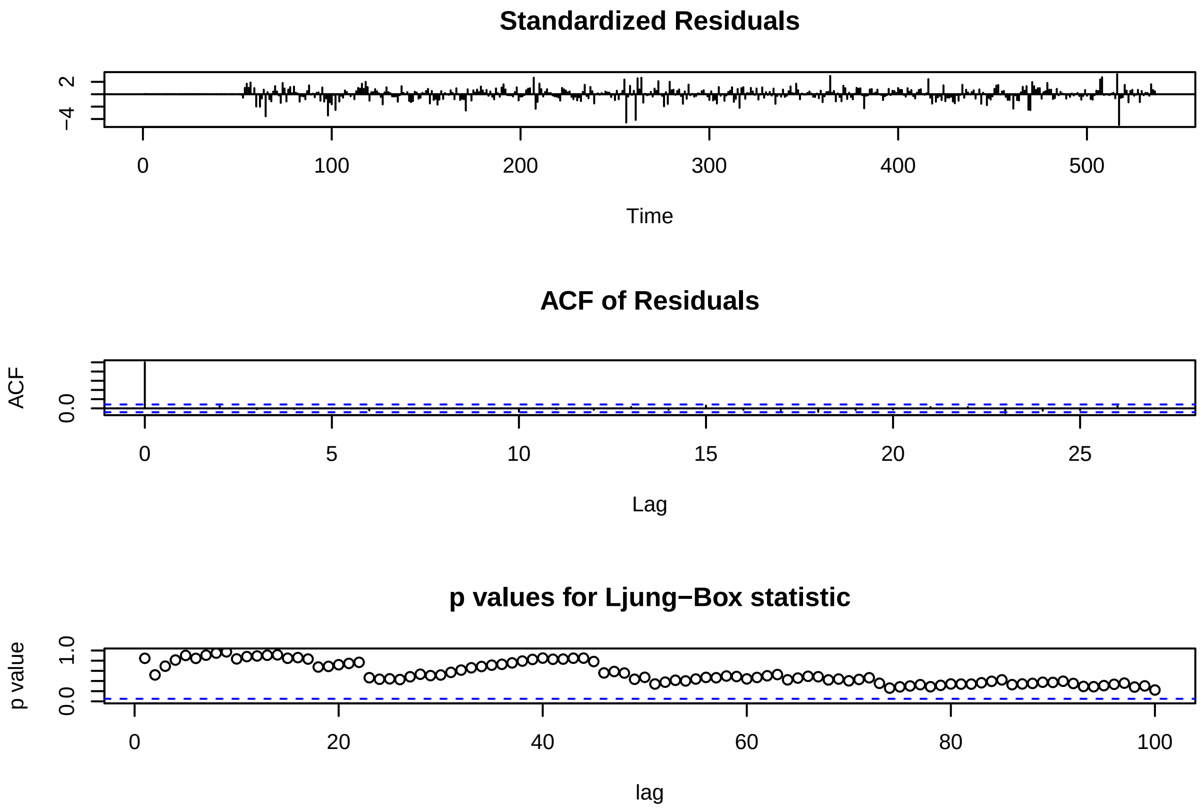

```r
Q5_1 <- arima(log_Q5train, order = c(0, 1, 1), seasonal = list(order = c(1, 1, 1), period = 52))
tsdiag(Q5_1,gof.lag=100) # aic = -1614.1
```



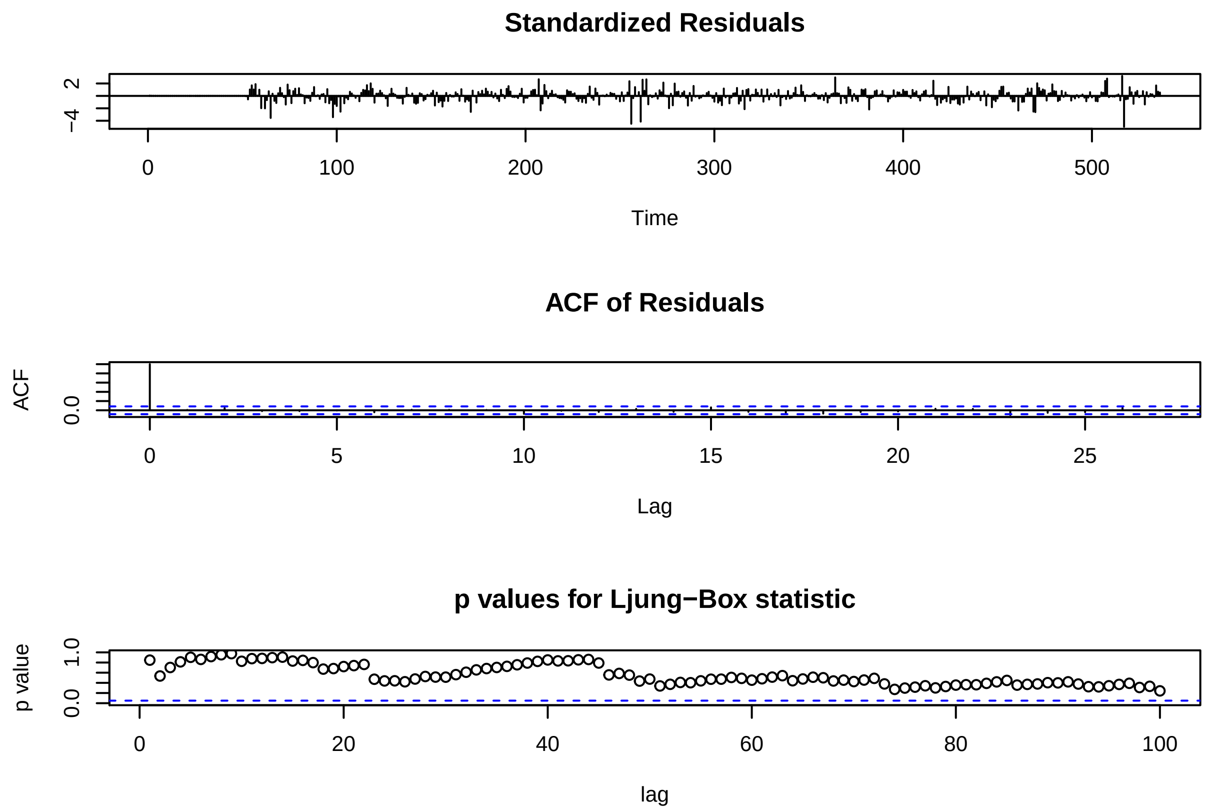

```r
Q5_2<- arima(log_Q5train, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 2), period = 52))
tsdiag(Q5_2,gof.lag=100)#aic = -1614.07
```



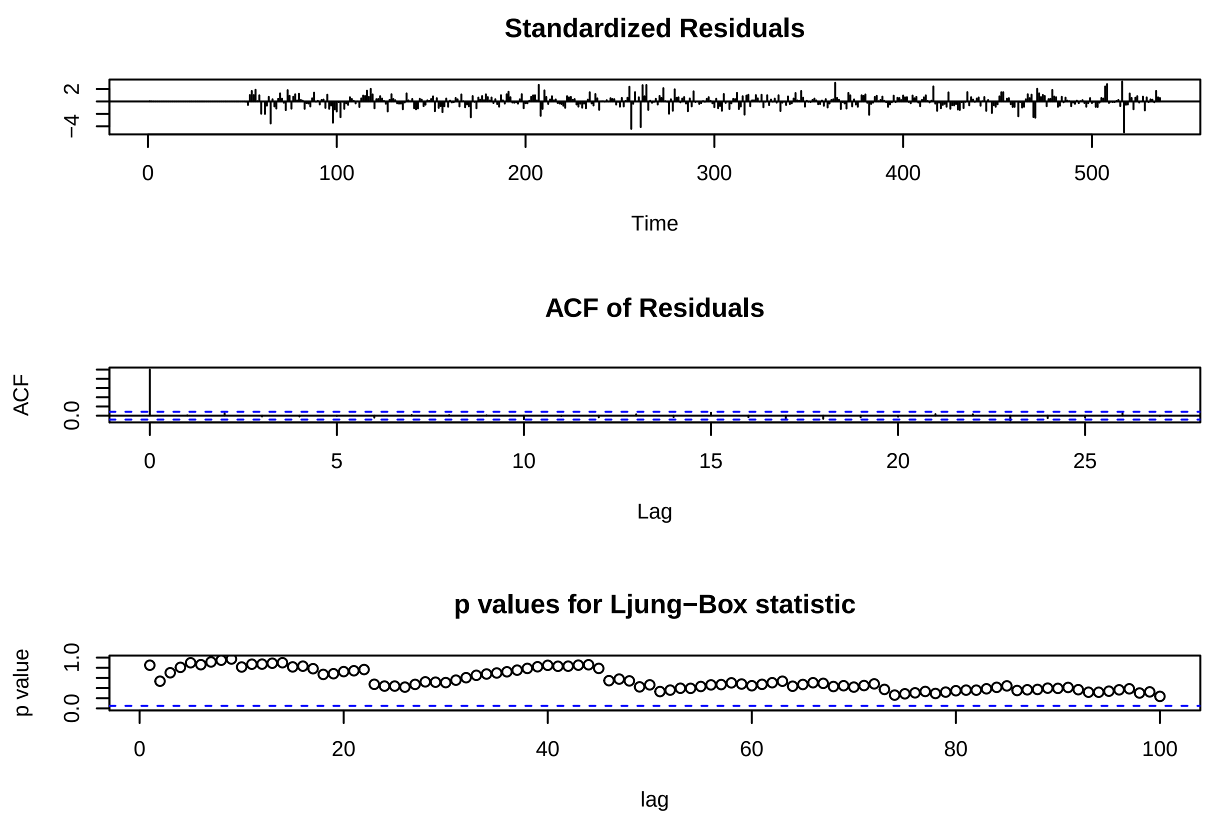
```
Q5_3<- arima(log_Q5train, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 1), period = 52))
tsdiag(Q5_3,gof.lag=100, main="c(0, 1, 2)c(0, 1, 1)52") #aic = -1616.93 better
```




```
Q5_4<- arima(log_Q5train, order = c(0, 1, 2), seasonal = list(order = c(1, 1, 1), period = 52))
tsdiag(Q5_4,gof.lag=100)#aic = -1616.4
```




```
Q5_5<- arima(log_Q5train, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 2), period = 52))
tsdiag(Q5_5, gof.lag=100) #aic = -1617.09 better
```



2)AIC and BIC

Compare all the models' AICs and BICs from above. ALways choose the one with smallest AIC or smallest BIC.

```
BIC(Q5_0) #chose by smallesdt BIC
```

```
## [1] -1602.389
```

```
BIC(Q5_1)
```

```
## [1] -1597.38
```

```
BIC(Q5_2)
```

```
## [1] -1597.945
```

```
BIC(Q5_3)
```

```
## [1] -1600.211
```

```
BIC(Q5_4)
```

```
## [1] -1595.496
```

```
BIC(Q5_5)
```

```
## [1] -1596.188
```

3)Cross Validation

Overfitting Models by CV

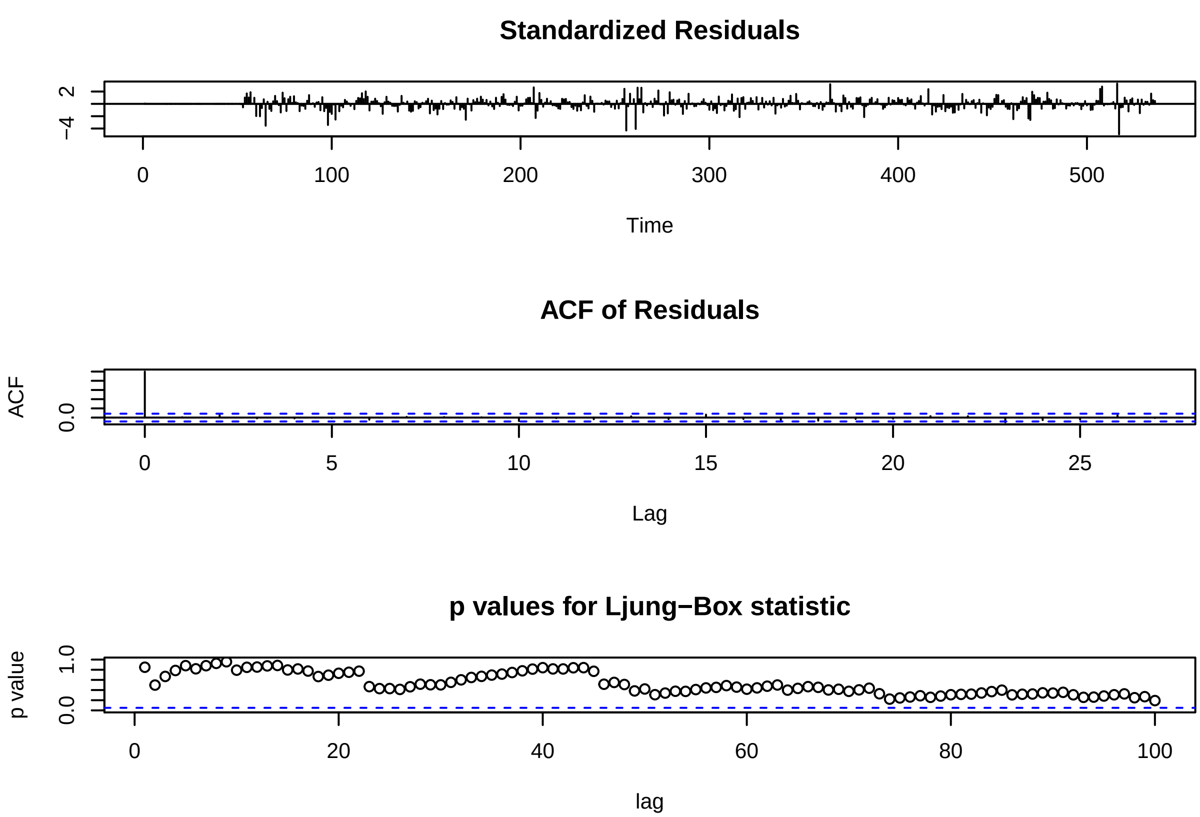
```
computeCVmse <- function(order.totry, seasorder.totry){
  MSE <- numeric()
  for(k in 5:9)
  {
    train.dt = log_Q5train[1:(k*52+16)]
    test.dt = log_Q5train[((k*52+16)+1):((k*52+16) + 52)]
    fm1 = arima(train.dt, order = order.totry, seasonal = list(order = seasorder.totry, period = 52))
    fcast.m1 = predict(fm1, n.ahead = 52)
    MSE[k-4] = mean((exp(fcast.m1$pred) - exp(test.dt))^2)
  }
  return (MSE)
}
```

Start from my prediction as before MA(1) $\times$ AR(1)52 ==>(c(0, 1, 1), c(1,1,0)_52)

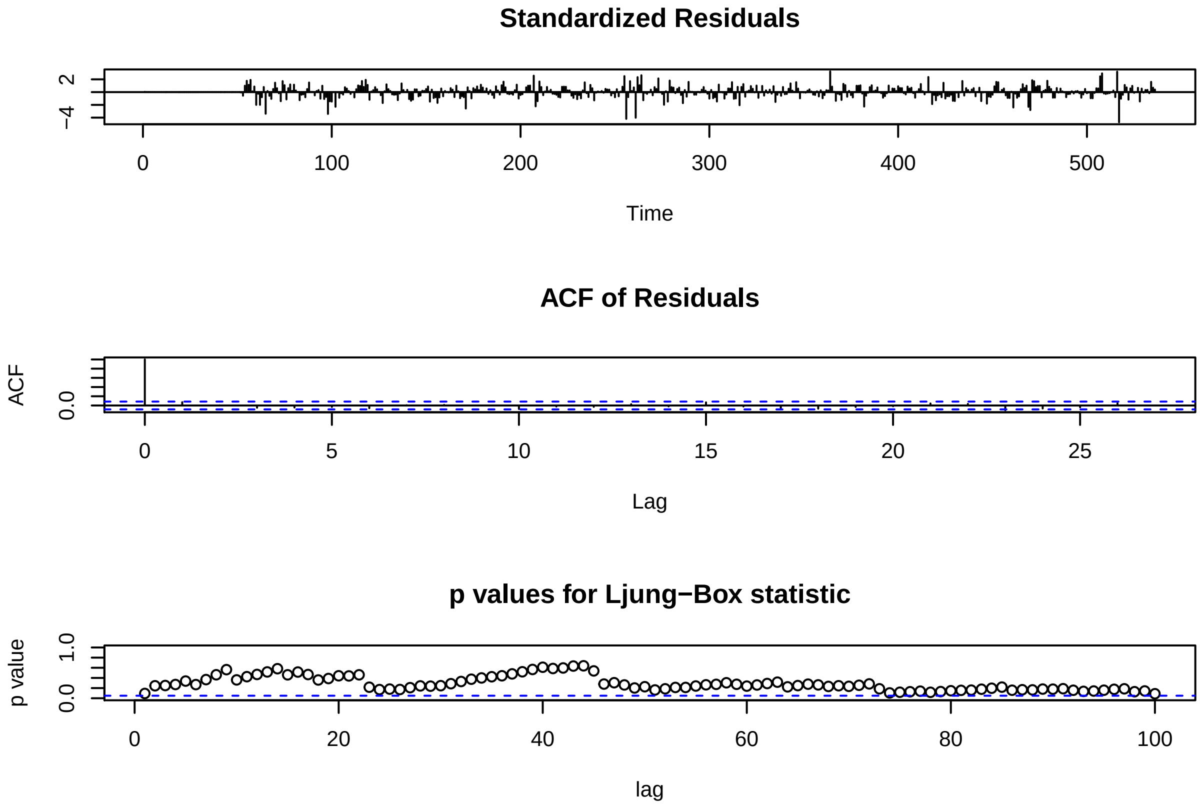
```r
#MSE5_0 = computeCVmse(c(0, 1, 1), c(0,1,1))  #CV0=6.501581\par
#This looks not bad.To compare it with other simple ARMA:\par
#MSE5_1 = computeCVmse(c(1, 1, 0), c(0,1,1))  #7.74596\par
#MSE5_2 = computeCVmse(c(1, 1, 0), c(1,1,0))  #7.613845\par
#MSE5_3 = computeCVmse(c(0, 1, 1), c(1,1,0))  #6.516456\par

#CV0 is the smallest, so we start from MSE5_0 to overfit models.\par
#MSE5_4 = computeCVmse(c(0, 1, 1), c(1,1,1))  #6.50228 \par
#MSE5_5 = computeCVmse(c(0, 1, 1), c(0,1,2))  #6.418859 \par
#MSE5_6 = computeCVmse(c(0, 1, 2), c(0,1,1))  #6.507531 \par
#MSE5_7 = computeCVmse(c(0, 1, 2), c(0,1,3))  #6.284072 \par

#MSE5_8 = computeCVmse(c(0, 1, 1), c(0,1,3))  #6.277156 \par
#MSE5_9 = computeCVmse(c(1, 1, 1), c(0,1,3))  #6.289948 \par
#MSE5_10 = computeCVmse(c(0, 1, 1), c(2,1,1))  #6.490148 \par
```

So far, by playing around with various combinations, we can find out MSE5_7 and MSE5_8 have smallest CV. Please see all CV calculations in R Appendix. Then I'm curious if their residuals make sense too.

```r
Q5_6<- arima(log_Q5train, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 3), period = 52))
Q5_7<- arima(log_Q5train, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 3), period = 52))
tsdiag(Q5_6,gof.lag=100)
```

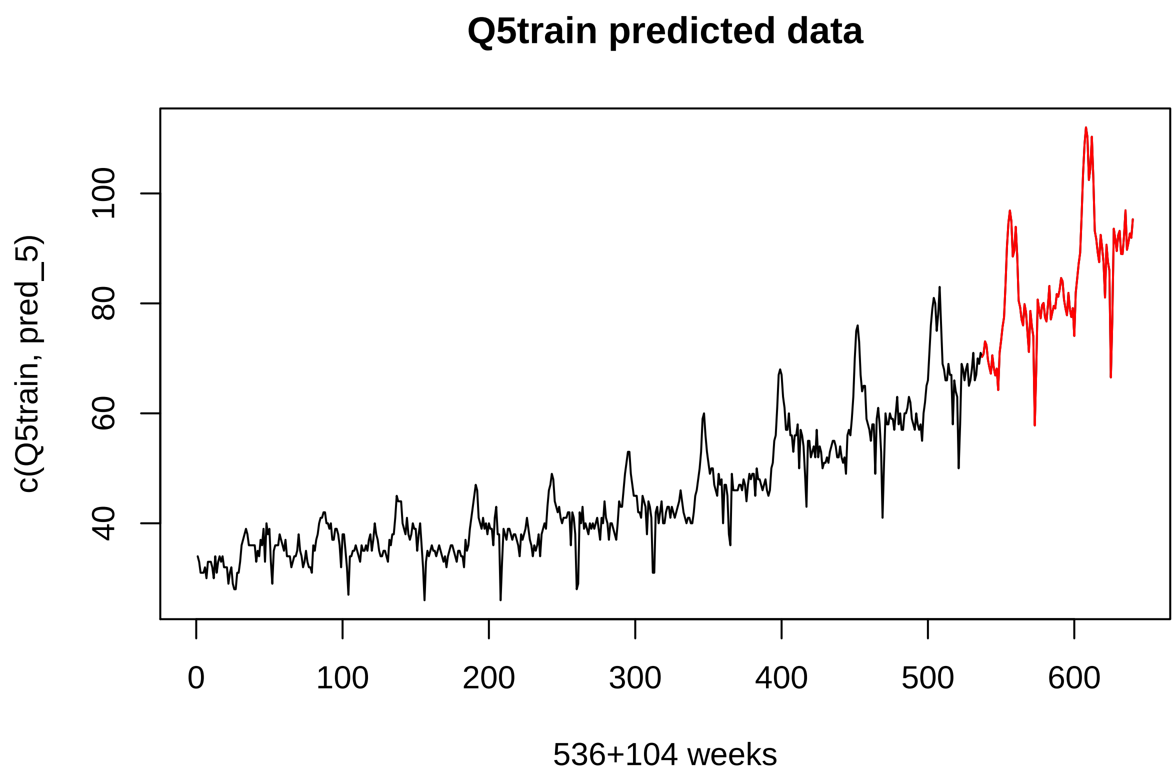



```r
tsdiag(Q5_7,gof.lag=100)
```



(6) Forecast and Conclusion

```
pred_5 <- exp(predict(Q5_6, n.ahead = 104)$pred)
plot(1:(len5 + length(pred_5)), c(Q5train, pred_5), type = 'l', col = 1, xlab="536+104 weeks", main="Q5
points((len5 + 1) : (len5 + length(pred_5)), pred_5, type = 'l', col = 2)
```

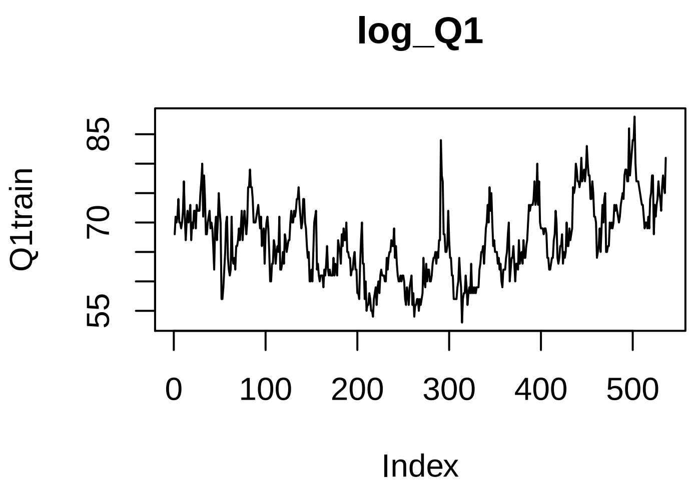



```
#Create the file:
write.table(pred_5,
            sep = ",",
            col.names = FALSE,
            row.names = FALSE,
            file = "Q5_Huidi_Wang_25157840.txt")
```

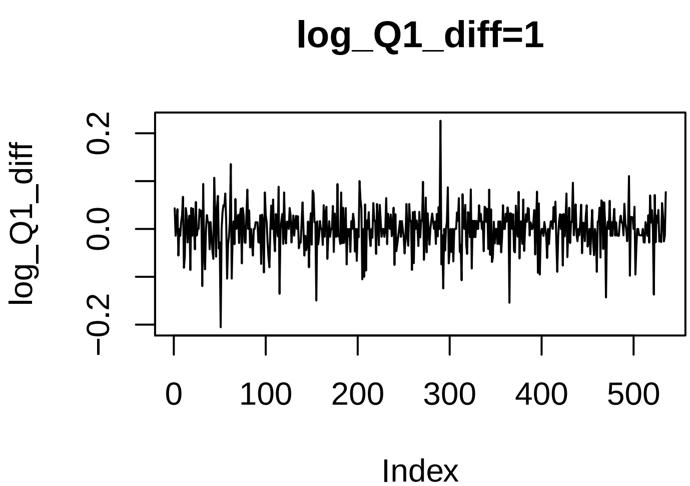
Q1train Analysis

```
#Extract dataset from original csv.file
Q1train.raw=read.delim(file="q1train.csv")
Q1train_data=Q1train.raw[1:536,1]
len1=length(Q1train_data)
Q1train=rep(0,len1)
for (i in 1:len1){
  Q1train[i]=as.numeric(unlist(strsplit(as.character(Q1train_data[i]),","))[2])
}
```

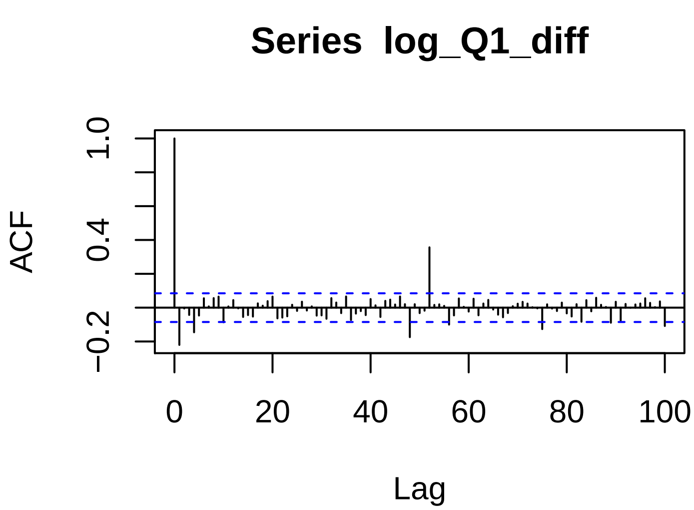
```
#Plot raw data, log data with corresponding acf and pacf
plot(Q1train, type="l", main="log_Q1")
```



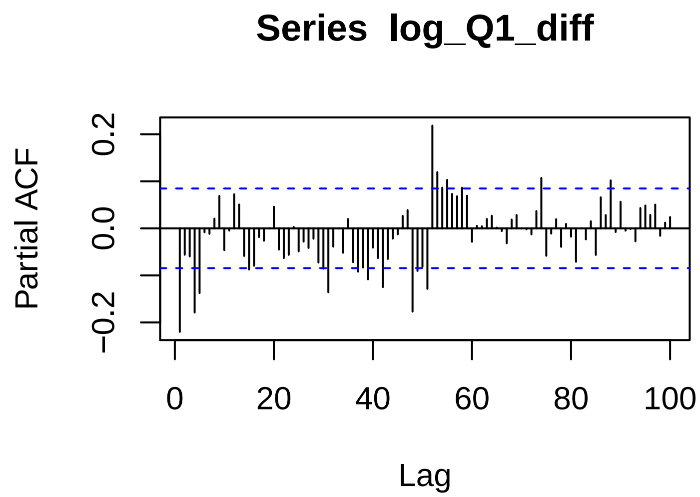

```
log_Q1train=log(Q1train)
#plot(log_Q1train, type="l")
log_Q1_diff=diff(log_Q1train)
plot(log_Q1_diff, type="l", main="log_Q1_diff=1")
```



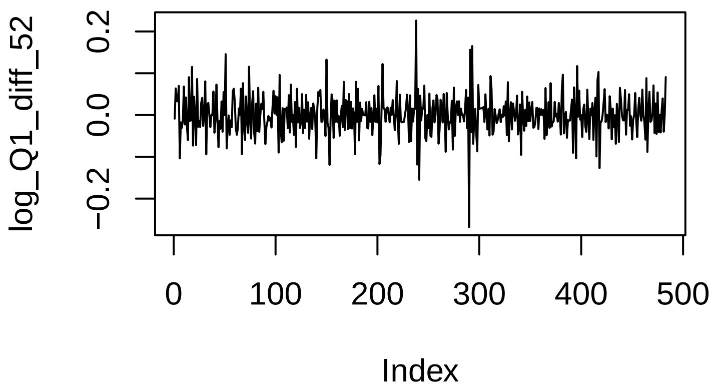

```
acf(log_Q1_diff, lag.max = 100)
```



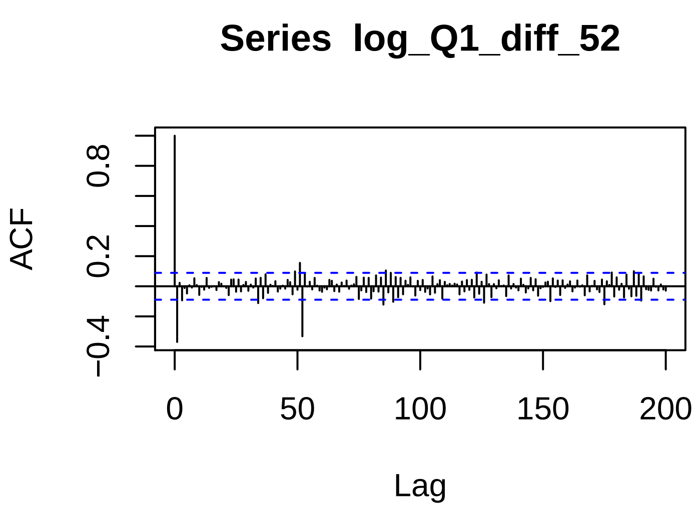
```r
#lag=52 is very big 0.3564543093
pacf(log_Q1_diff, lag.max=100)
```



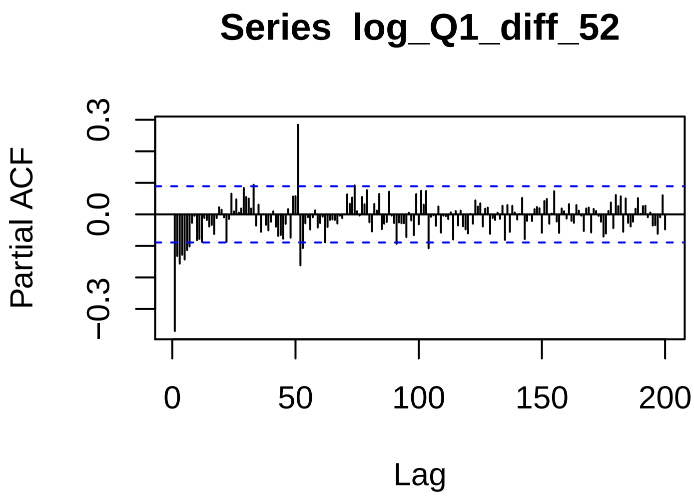

```r
log_Q1_diff_52=diff(log_Q1_diff,52)
plot(log_Q1_diff_52, type="l")
```




```r
acf(log_Q1_diff_52, lag.max = 200)
```

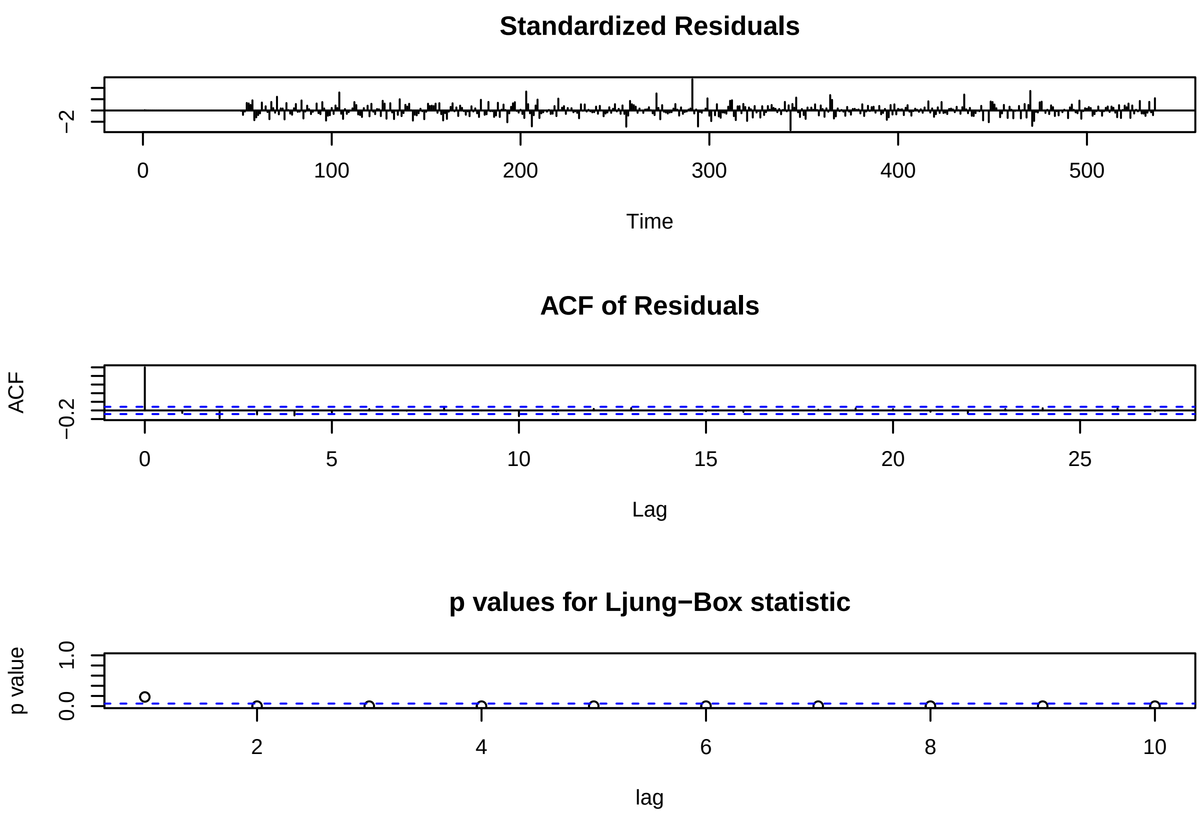


```
pacf(log_Q1_diff_52, lag.max=200)
```

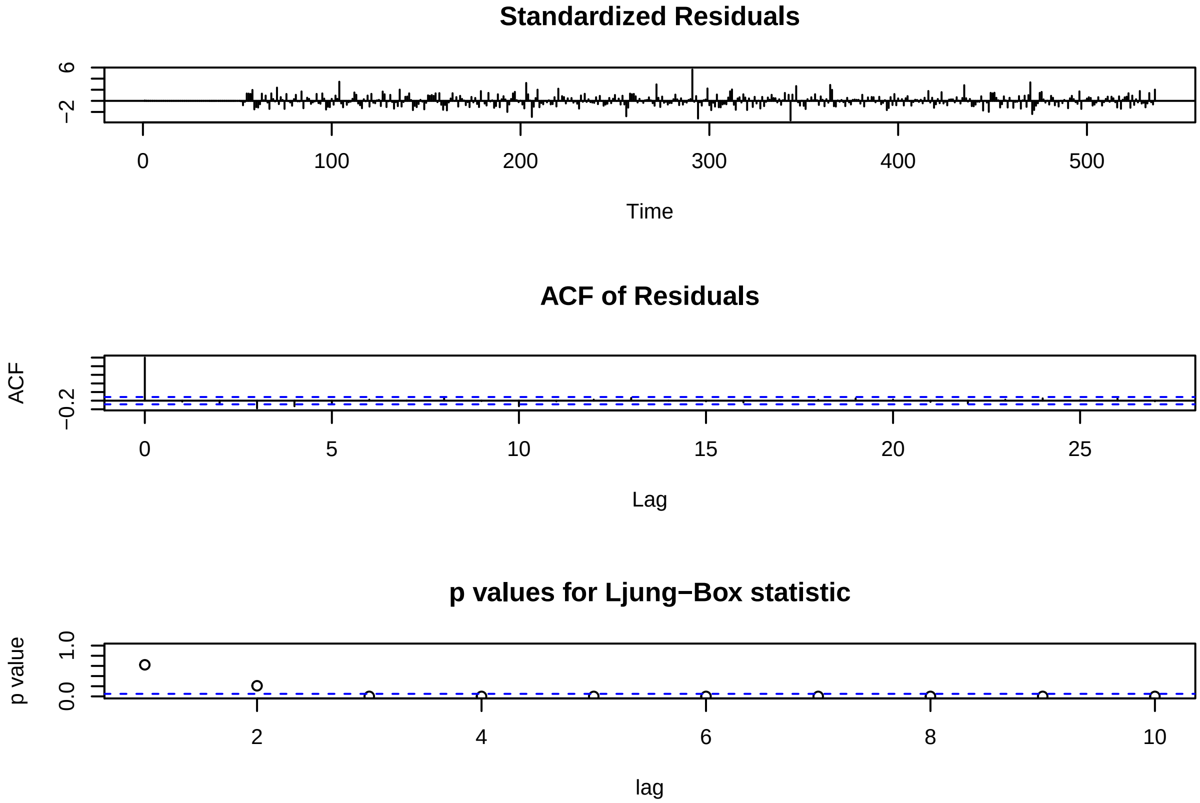



```
# Maybe an Arma(1,0) x (0,1)_52 model?
```

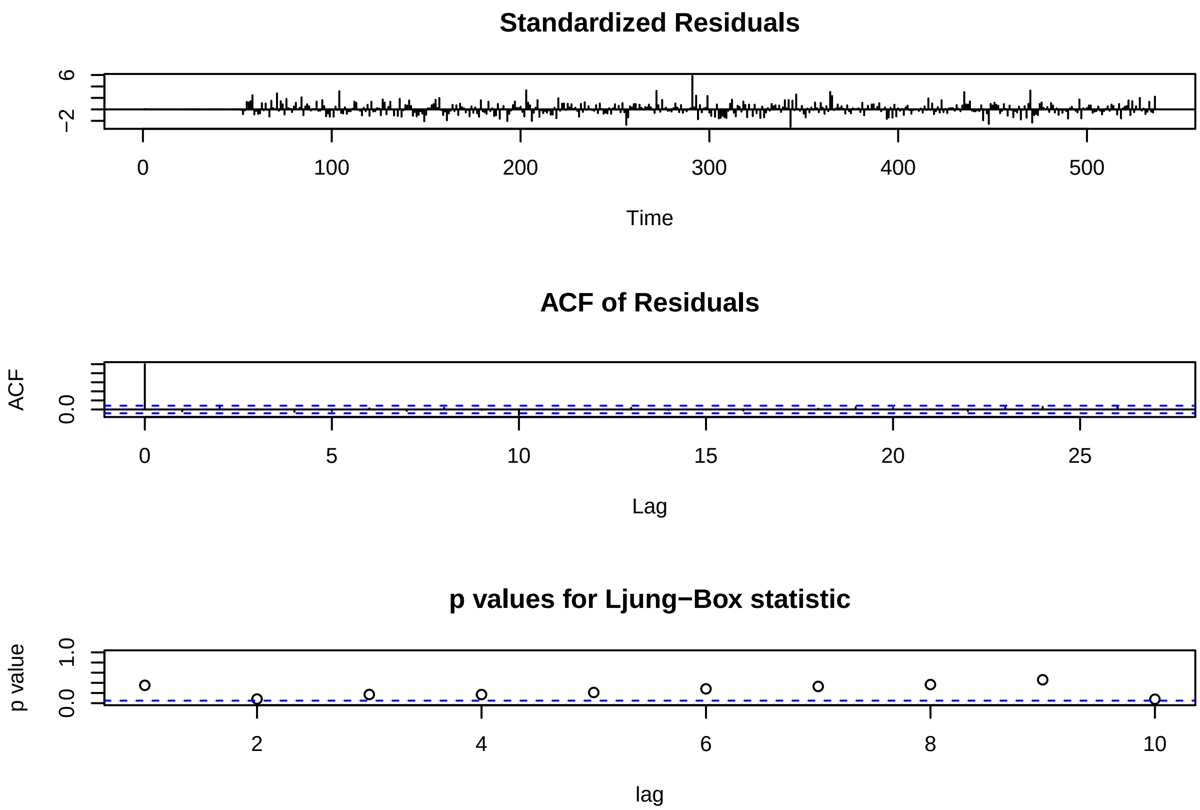
```
#Model Diagnostic
#Residuals and P-value
q1_1 <- arima(log_Q1train, order = c(1, 1, 0), seasonal = list(order = c(0, 1, 1), period = 52))
tsdiag(q1_1)  #aic = -1697.61
```



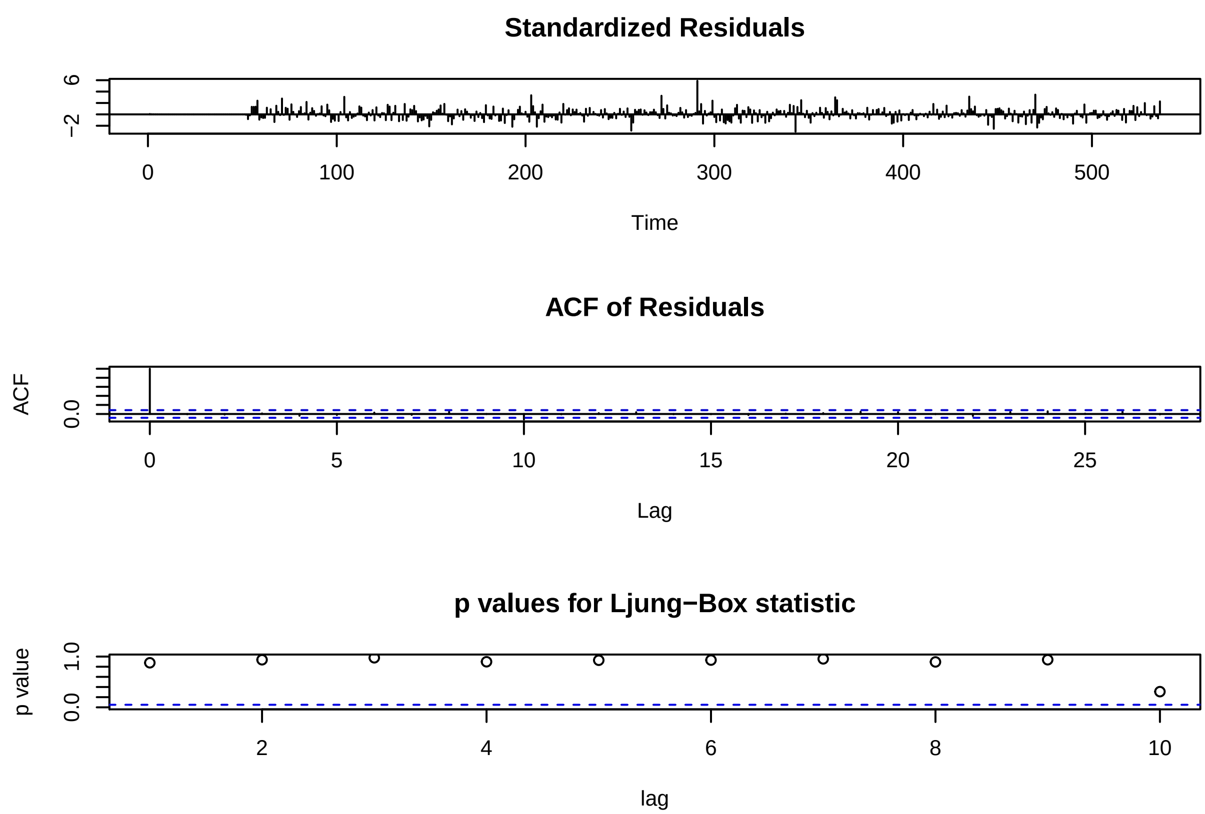
```
# Ljung-Box test seems to suggest that I did not do a good job in finding the right model
# Overfitting for diagnostics, we want to minimize the AIC AIC(m1) = -1697.61
#q1_2 <- arima(log_Q1train, order = c(1, 1, 0), seasonal = list(order = c(0, 1, 2), period = 52))
#tsdiag(q1_2)  #aic = -1696.14
q1_3 <- arima(log_Q1train, order = c(2, 1, 0), seasonal = list(order = c(0, 1, 1), period = 52))
tsdiag(q1_3)  #aic = -1707.07
```




```
q1_4 <- arima(log_Q1train, order = c(1, 1, 1), seasonal = list(order = c(0, 1, 1), period = 52))
tsdiag(q1_4)  #aic = -1750.72 better, and Ljung_Box shows this model's p_value relatively higher than o
```



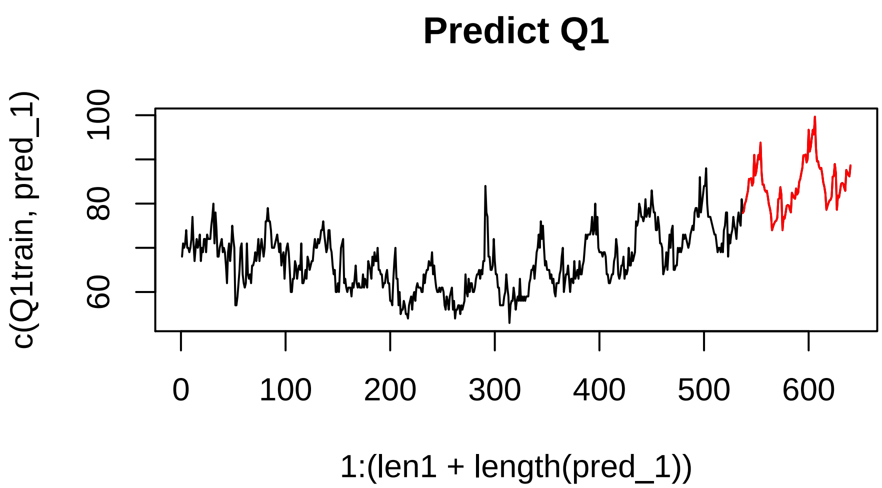

```
####Based on q1_4, predict more models.
#q1_4 <- arima(log_Q1train, order = c(1, 1, 2), seasonal = list(order = c(0, 1, 1), period = 52))
#tsdiag(q1_4)  #aic = -1752.18
q1_5 <- arima(log_Q1train, order = c(1, 1, 3), seasonal = list(order = c(0, 1, 1), period = 52))
tsdiag(q1_5)  #aic = -1752.75 Ljung_Box shows this model's p_value relatively higher than other models.
```




```
#I will chose c(1, 1, 3)c(0, 1, 1) under this method.

#AIC,BIC method
#BIC(q1_1) #-1685.072
#BIC(q1_3) #-1690.348
#BIC(q1_4) #-1731.278    #chose by smallest BIC
#BIC(q1_5) #-1727.671    #chose by smallest AIC

###Cross_Validation

computeCVmse <- function(order.totry, seasorder.totry){
  MSE <- numeric()
  for(k in 5:9)
  {
    train.dt = log_Q1train[1:(k*52+16)]
    test.dt = log_Q1train[((k*52+16)+1):((k*52+16) + 52)]
    fm1 = arima(train.dt, order = order.totry, seasonal = list(order = seasorder.totry, period = 52))
    fcast.m1 = predict(fm1, n.ahead = 52)
    MSE[k-4] = mean((exp(fcast.m1$pred) - exp(test.dt))^2)
  }
  return (MSE)
}

###Start from my prediction
#MSE1_0<- computeCVmse(c(1,1,0), c(0,1,1)) #13.07271

#MSE1_1 <- computeCVmse(c(0,1,1), c(0,1,1))  #13.29131
#MSE1_2<- computeCVmse(c(0,1,1), c(1,1,0))  #14.02561
```

```r
#MSE1_3<- computeCVmse(c(1,1,0), c(1,1,0))  #13.8719

###Based on c(1,1,0)c(0,1,1)52 to overfit more models
#MSE1_4<- computeCVmse(c(1,1,0), c(0,1,2)) #13.30125
#MSE1_5<- computeCVmse(c(2,1,0), c(0,1,1)) #13.22346
#MSE1_6<- computeCVmse(c(1,1,1), c(0,1,1)) #12.98248
#MSE1_7<- computeCVmse(c(1,1,0), c(1,1,1)) #13.28773
#MSE1_8<- computeCVmse(c(1,1,2), c(0,1,1)) #13.08729
###Based on c(0,1,1)c(0,1,1)52 to overfit more models
#MSE1_9<- computeCVmse(c(0,1,1), c(0,1,2)) #13.58906
#MSE1_10<- computeCVmse(c(1,1,1), c(0,1,1)) #12.98248
#MSE1_11<- computeCVmse(c(0,1,1), c(1,1,1)) #13.52111
#MSE1_12<- computeCVmse(c(0,1,2), c(0,1,1)) #12.86737
#MSE1_13<- computeCVmse(c(0,1,3), c(0,1,1)) #12.67822    be chosen to predict next 104
#MSE1_14<- computeCVmse(c(1,1,3), c(0,1,1)) #12.90245
```

```r
#Forecast
q1=arima(log_Q1train, order = c(0, 1, 3), seasonal = list(order = c(0, 1, 1), period = 52))
pred_1 <- exp(predict(q1, n.ahead = 104)$pred)
plot(1:(len1 + length(pred_1)), c(Q1train, pred_1), type = 'l', col = 1, main="Predict Q1")
points((len1 + 1) : (len1 + length(pred_1)), pred_1, type = 'l', col = 2)
```




```r
#Create the file:
write.table(pred_1,
            sep = ",",
            col.names = FALSE,
            row.names = FALSE,
            file = "Q1_Huidi_Wang_25157840.txt")
```

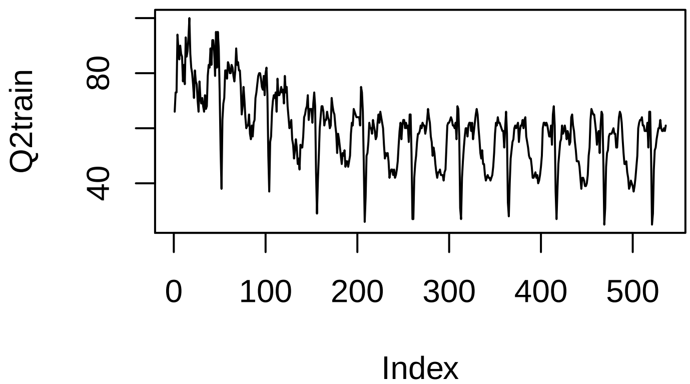
*

*Q2train Analysis**

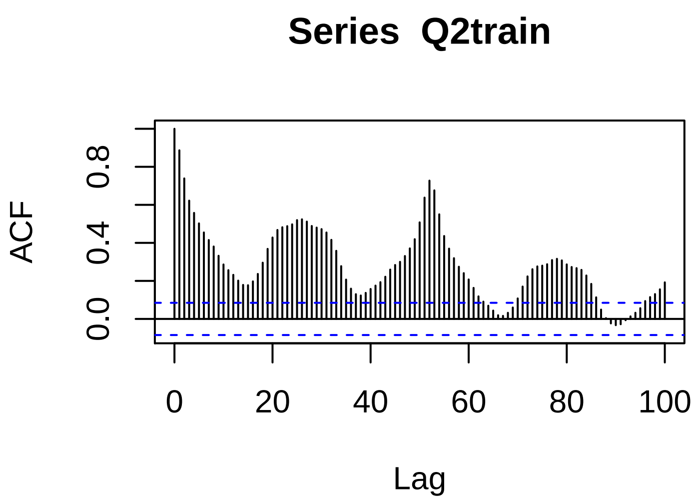
```r
#Extract dataset from original csv.file
Q2train.raw=read.delim(file="q2train.csv")
Q2train_data=Q2train.raw[1:536,1]
len2=length(Q2train_data)
```

```
Q2train=rep(0,len2)
for (i in 1:len2){
  Q2train[i]=as.numeric(unlist(strsplit(as.character(Q2train_data[i]),","))[2])
}
```

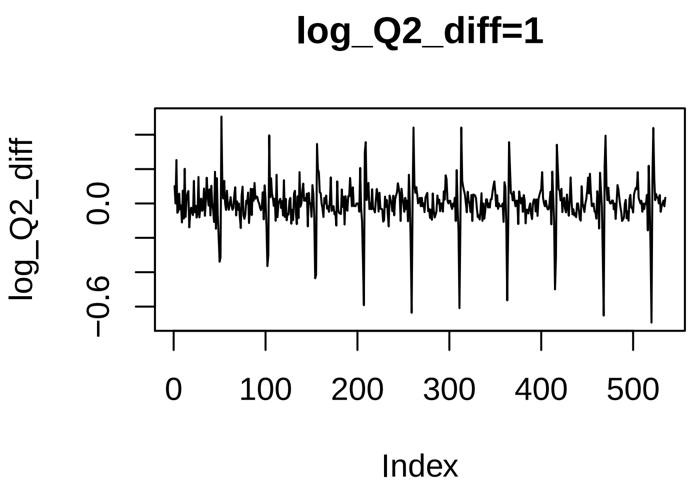
```
# Plots of raw data, log data, acf and pacf
plot(Q2train, type="l")
```



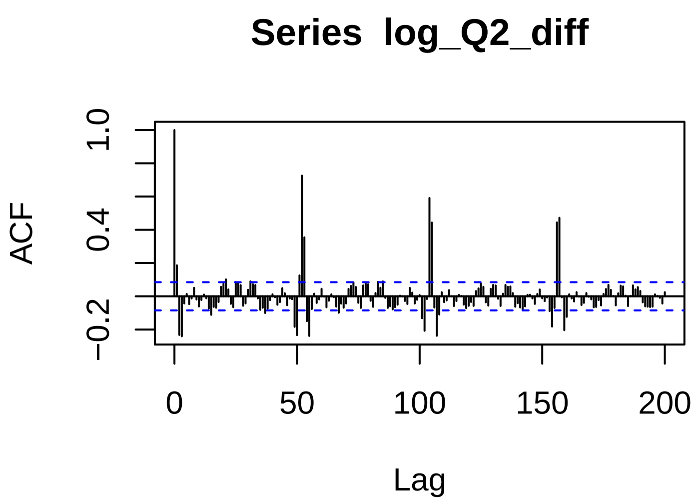

```
acf(Q2train, lag.max=100)
```



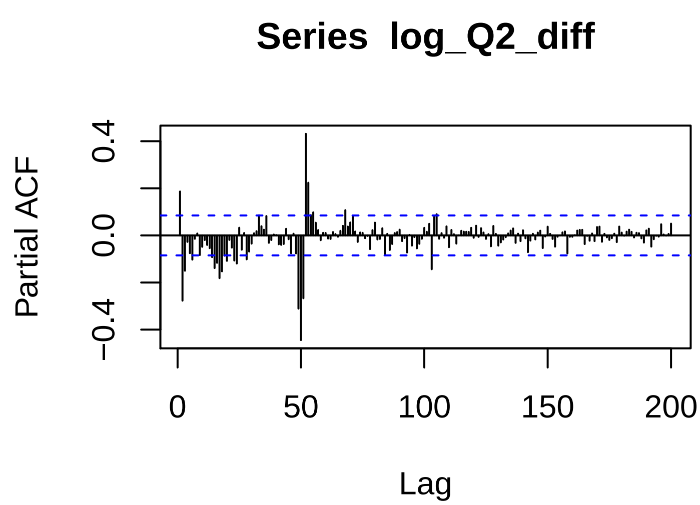

```
log_Q2train=log(Q2train)
#plot(log_Q2train, type="l")

log_Q2_diff=diff(log_Q2train)
plot(log_Q2_diff, type="l", main="log_Q2_diff=1")
```



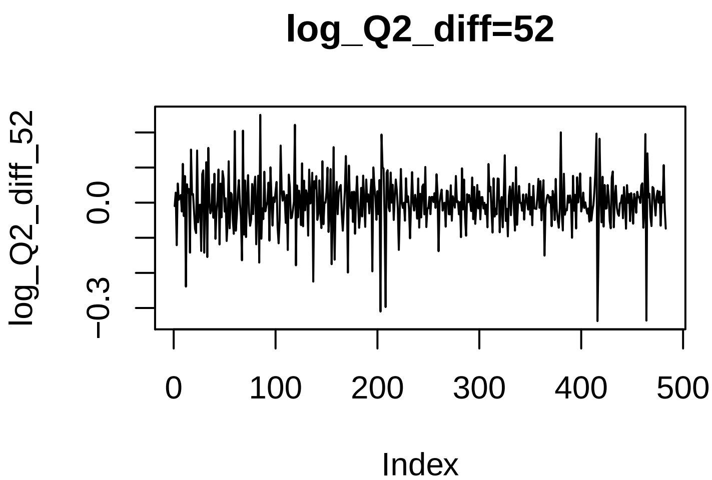

```
acf(log_Q2_diff, lag.max=200)
```



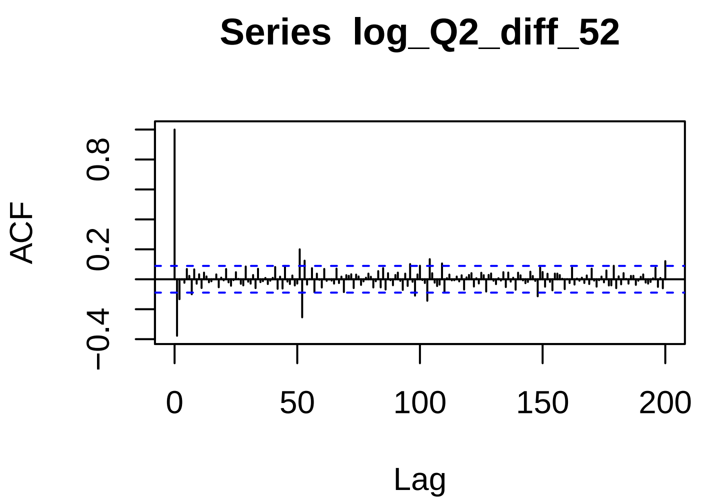

```
#[53,]  0.6154586266
#[105,]  0.5185120649
#[157,]  0.3953714618
pacf(log_Q2_diff, lag.max=200)
```



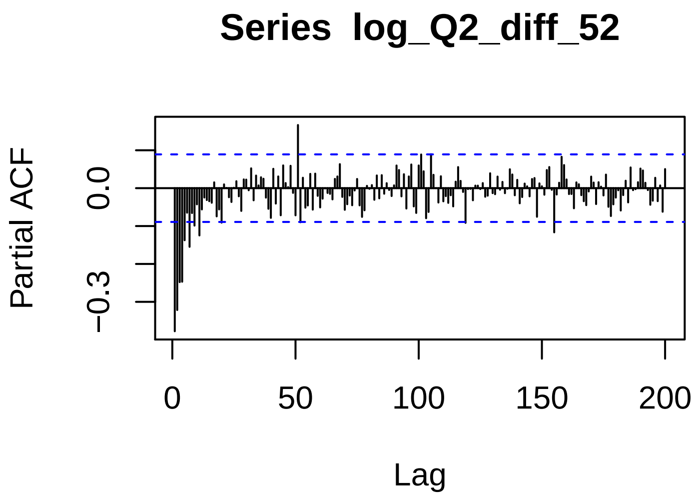
```r
log_Q2_diff_52=diff(log_Q2_diff,52)
plot(log_Q2_diff_52, type="l",main="log_Q2_diff=52")
```




```r
acf(log_Q2_diff_52, lag.max=200)
```

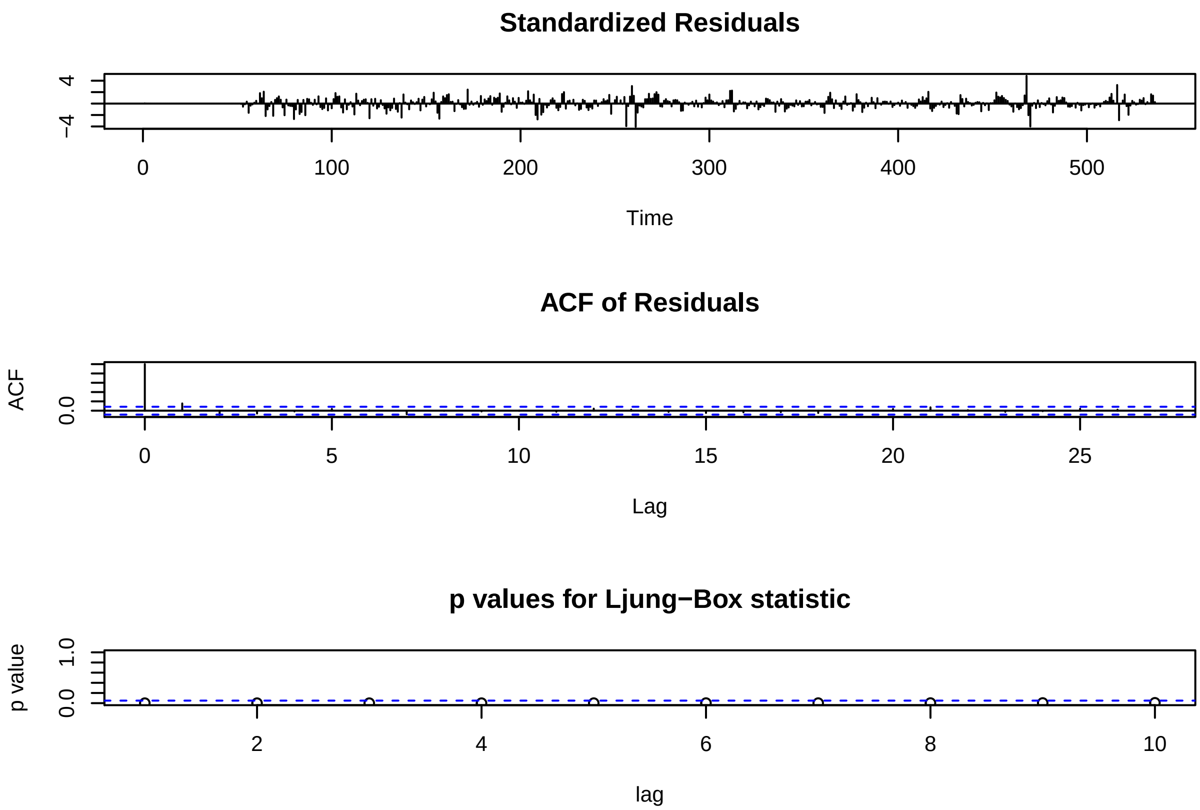



```r
#[52,]  0.2604153481
#[53,] -0.3322802641
pacf(log_Q2_diff_52, lag.max=200)
```



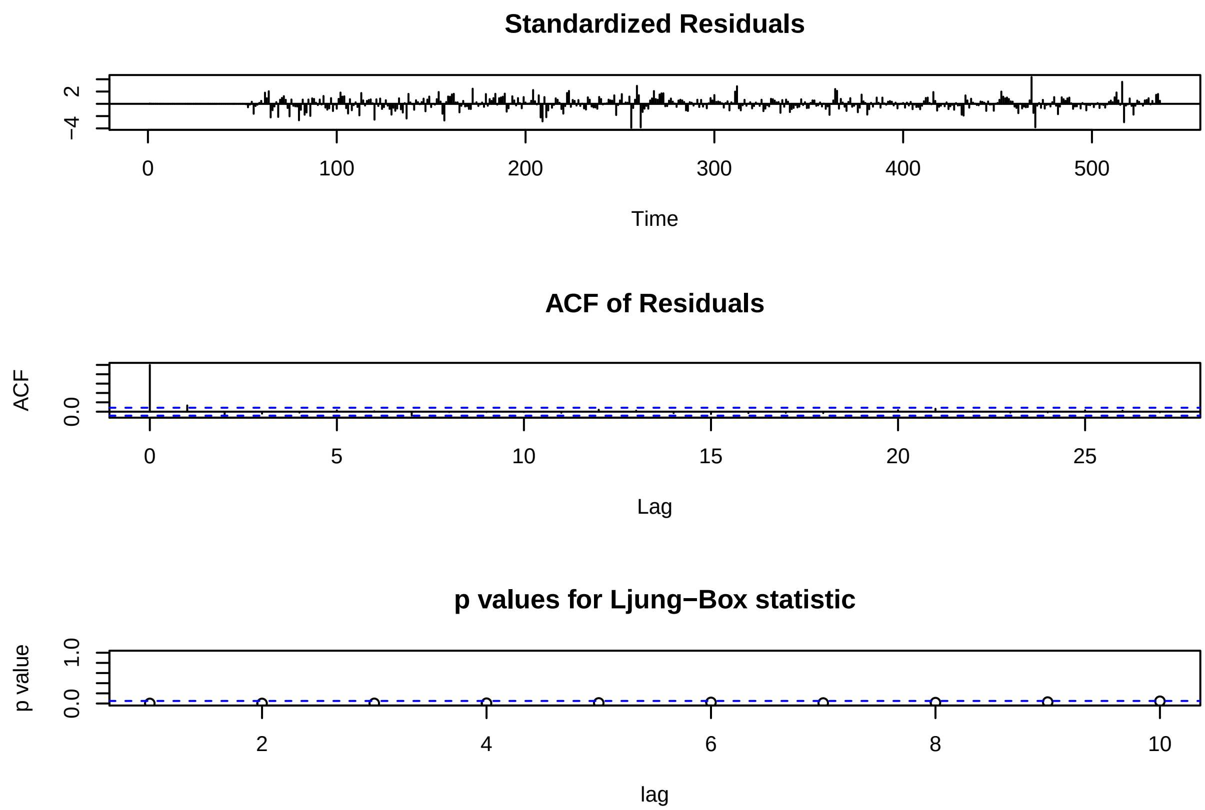

```
#Guess c(0,1,1)c(1,1,0)52 model
```

```
#___________Model Diagnostic________
# ###Residuals and P-value
Q2_1 <- arima(log_Q2train, order = c(0, 1, 1), seasonal = list(order = c(1, 1, 0), period = 52))
tsdiag(Q2_1) #aic = -1362.59
```

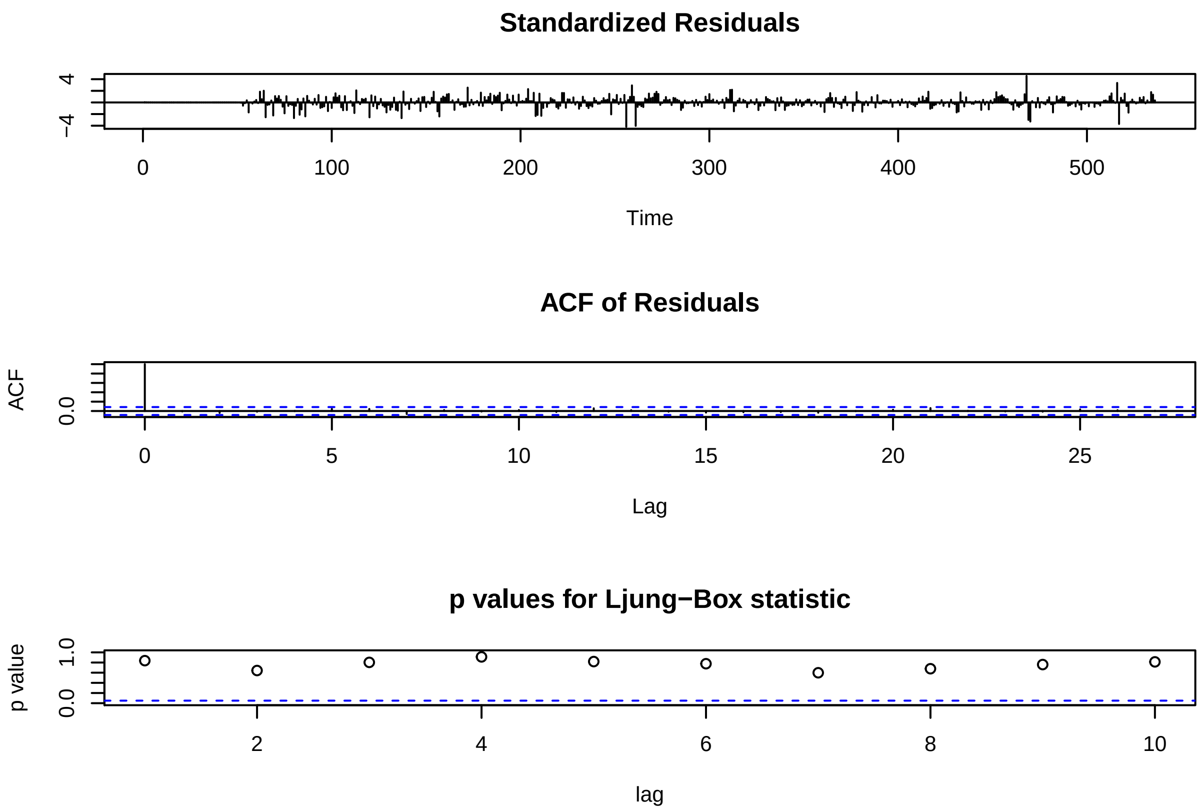



```
#### Ljung-Box test seems to suggest that I did not do a good job in finding the right model
#### Overfitting for diagnostics, we want to minimize the AIC AIC(m1) =-1362.59
#Q2_2 <- arima(log_Q2train, order = c(0, 1, 2), seasonal = list(order = c(1, 1, 0), period = 52))
#tsdiag(Q2_2) #aic = -1379.23 better, Ljung_Box shows this model's p_value relatively higher than other
```

```
#Q2_3 <- arima(log_Q2train, order = c(0, 1, 1), seasonal = list(order = c(2, 1, 0), period = 52))
#tsdiag(Q2_3) #aic = -1364.07
#Q2_4 <- arima(log_Q2train, order = c(0, 1, 1), seasonal = list(order = c(1, 1, 1), period = 52))
#tsdiag(Q2_4) #aic = -1362.08
Q2_9 <- arima(log_Q2train, order = c(0, 1, 1), seasonal = list(order = c(1, 1, 2), period = 52))
tsdiag(Q2_9) # aic = -1381  better, but Ljung_Box shows this model's p_value are relatively lower.
```




```
####Based on Q2_2: c(0, 1, 2)c(1, 1, 0)52, make more prediction
#Q2_5 <- arima(log_Q2train, order = c(0, 1, 2), seasonal = list(order = c(1, 1, 1), period = 52))
#tsdiag(Q2_5) #aic = -1378.76
#Q2_6 <- arima(log_Q2train, order = c(0, 1, 3), seasonal = list(order = c(1, 1, 0), period = 52))
#tsdiag(Q2_6) #aic = -1378.12
#Q2_7 <- arima(log_Q2train, order = c(1, 1, 2), seasonal = list(order = c(1, 1, 0), period = 52))
#tsdiag(Q2_7) #aic = -1377.67
Q2_8 <- arima(log_Q2train, order = c(0, 1, 2), seasonal = list(order = c(2, 1, 0), period = 52))
tsdiag(Q2_8) #aic = -1380.2 better, Ljung_Box shows this model's p_value relatively higher than other m
```

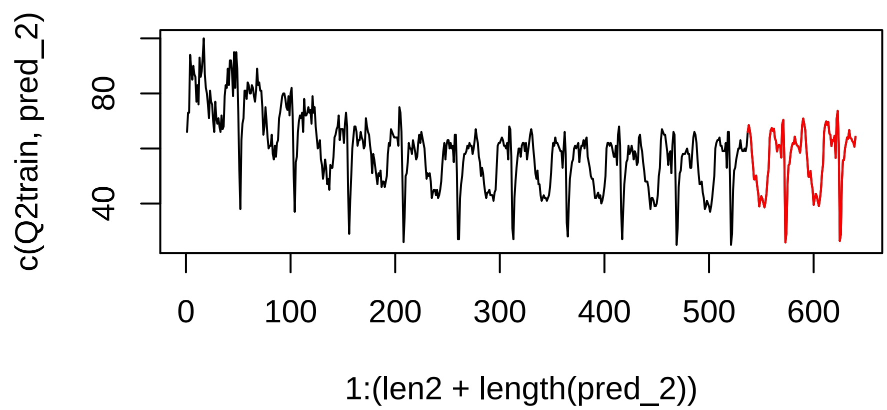



```
# ###AIC and BIC method
#BIC(Q2_1)
#BIC(Q2_2)          #chose by smallest BIC=-1362.507
#BIC(Q2_3)
#BIC(Q2_4)
#BIC(Q2_5)
#BIC(Q2_6)
#BIC(Q2_7)
#BIC(Q2_8)
#BIC(Q2_9)           #chose by smallest AIC=-1381
```

```
###Cross_Validation
computeCVmse <- function(order.totry, seasorder.totry){
  MSE <- numeric()
  for(k in 5:9)
  {
    train.dt = log_Q2train[1:(k*52+16)]
    test.dt = log_Q2train[((k*52+16)+1):((k*52+16) + 52)]
    fm1 = arima(train.dt, order = order.totry, seasonal = list(order = seasorder.totry, period = 52))
    fcast.m1 = predict(fm1, n.ahead = 52)
    MSE[k-4] = mean((exp(test.dt) - exp(fcast.m1$pred))^2)
  }
  return (MSE)
}


#Start from my prediction MA(1)AR(1)52
#MSE9 <- computeCVmse(c(0, 1, 1), c(1,1,0))   #CV=8.337722 #good prediction!
```

```r
#MSE6 <- computeCVmse(c(1, 1, 0), c(0,1,1))   #15.99587
#MSE7 <- computeCVmse(c(1, 1, 0), c(1,1,0))   #15.58562
#MSE8 <- computeCVmse(c(0, 1, 1), c(0,1,1))   #8.528534

#Based on c(0, 1, 1)c(1,1,0)52 to overfit more models
#MSE10 <- computeCVmse(c(0, 1, 1), c(1,1,1)) #8.426476
#MSE11 <- computeCVmse(c(0, 1, 1), c(2,1,0)) #8.476351
#MSE10 <- computeCVmse(c(0, 1, 2), c(1,1,0)) #8.425827 also good since its p-value are high and AIC ver
#MSE11 <- computeCVmse(c(0, 1, 2), c(2,1,0)) #8.549324

#MSE12 <- computeCVmse(c(0, 1, 1), c(0,1,2)) #8.413814
#MSE13 <- computeCVmse(c(0, 1, 1), c(1,1,2)) #8.230702 better to choose with smallest AIC
#MSE14 <- computeCVmse(c(0, 1, 1), c(0,1,3)) #8.769546
```

```r
#Forecast
pred_2 <- exp(predict(Q2_9, n.ahead = 104)$pred)
plot(1:(len2 + length(pred_2)), c(Q2train, pred_2), type = 'l', col = 1)
points((len2 + 1) : (len2 + length(pred_2)), pred_2, type = 'l', col = 2)
```



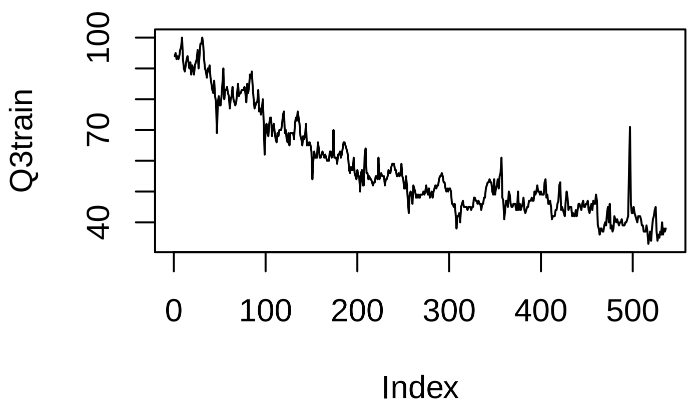

```r
#Create the file:
write.table(pred_2,
            sep = ",",
            col.names = FALSE,
            row.names = FALSE,
            file = "Q2_Huidi_Wang_25157840.txt")
```

*

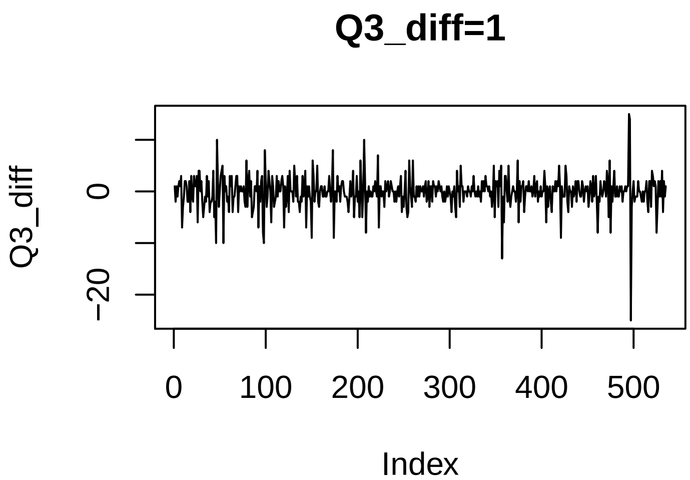
*Q3train Analysis**

```r
#Extract dataset from original csv.file
Q3train.raw=read.delim(file="q3train.csv")
Q3train_data=Q3train.raw[1:536,1]
len3=length(Q3train_data)
Q3train=rep(0,len3)
for (i in 1:len3){
  Q3train[i]=as.numeric(unlist(strsplit(as.character(Q3train_data[i]),","))[2])
}
```

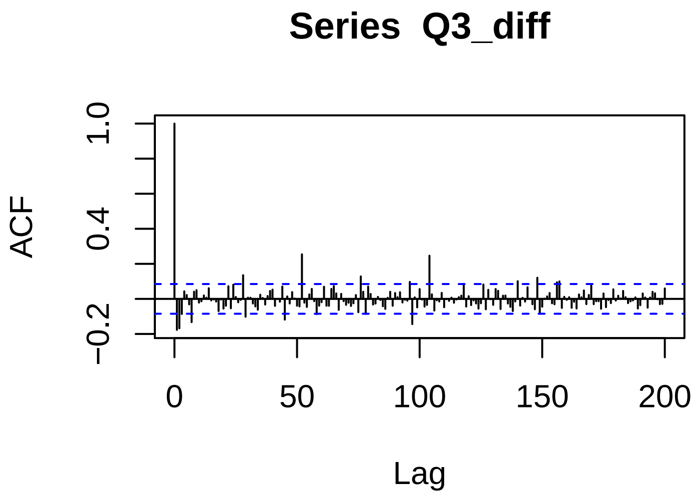
```r
# Plots of raw data, log data, acf and pacf
plot(Q3train, type="l")
```



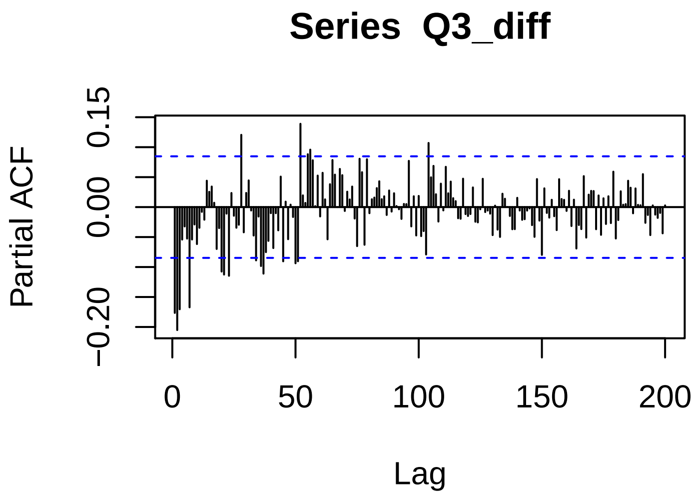

```r
Q3_diff=diff(Q3train)
plot(Q3_diff, type="l",main="Q3_diff=1")
```



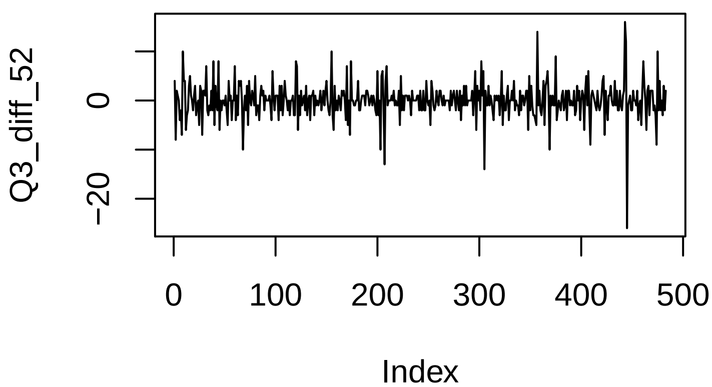

```r
acf(Q3_diff, lag.max=200)
```



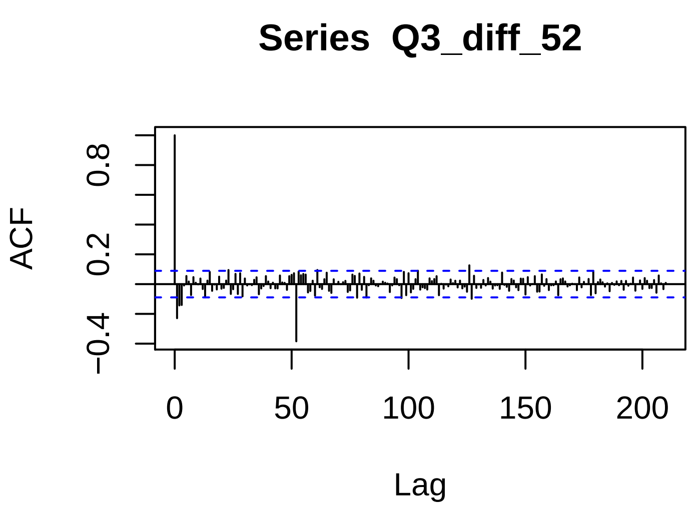
```r
pacf(Q3_diff, lag.max=200)
```



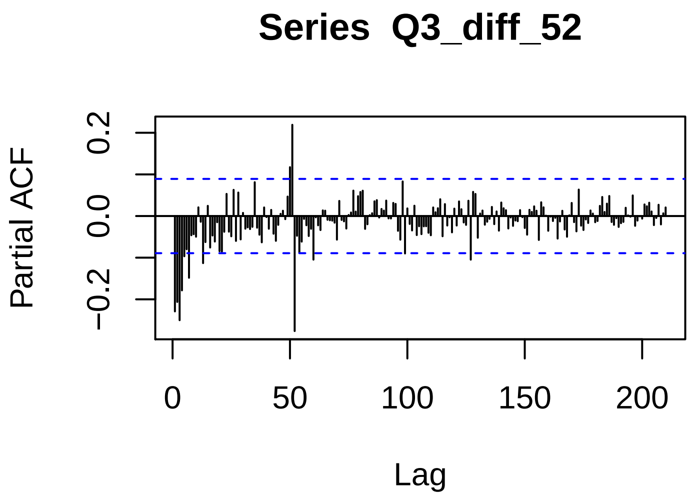

```r
Q3_diff_52=diff(Q3_diff,52)
plot(Q3_diff_52, type="l")
```




```r
acf(Q3_diff_52, lag.max=210)
```

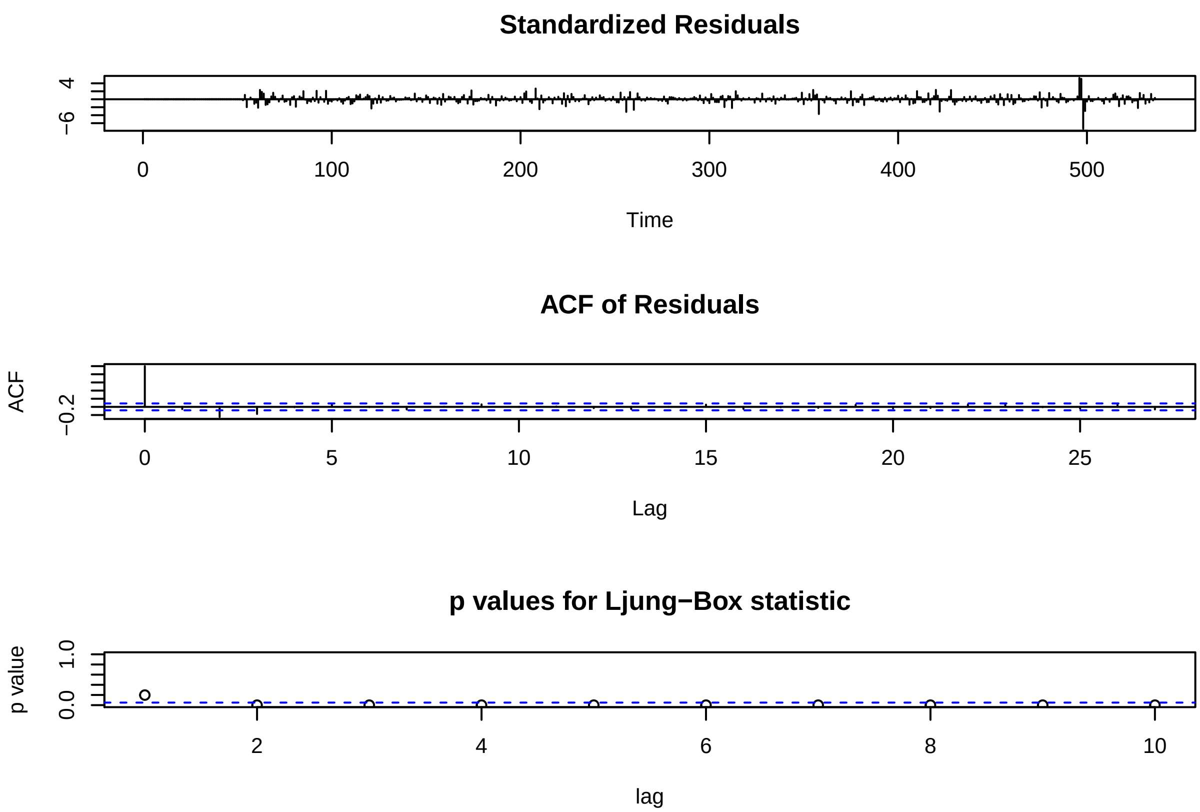


```
pacf(Q3_diff_52, lag.max=210)
```

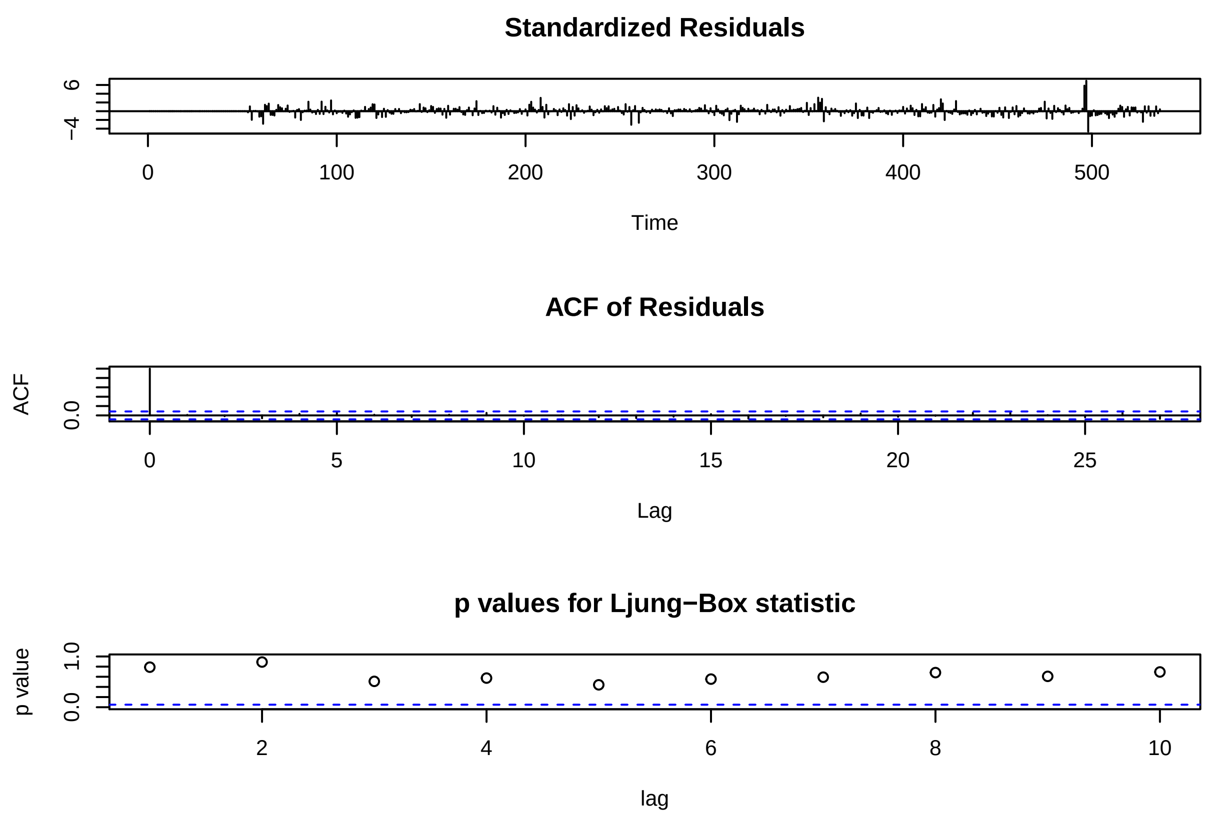



```
#Guess AR(1)AR(1)_52
```

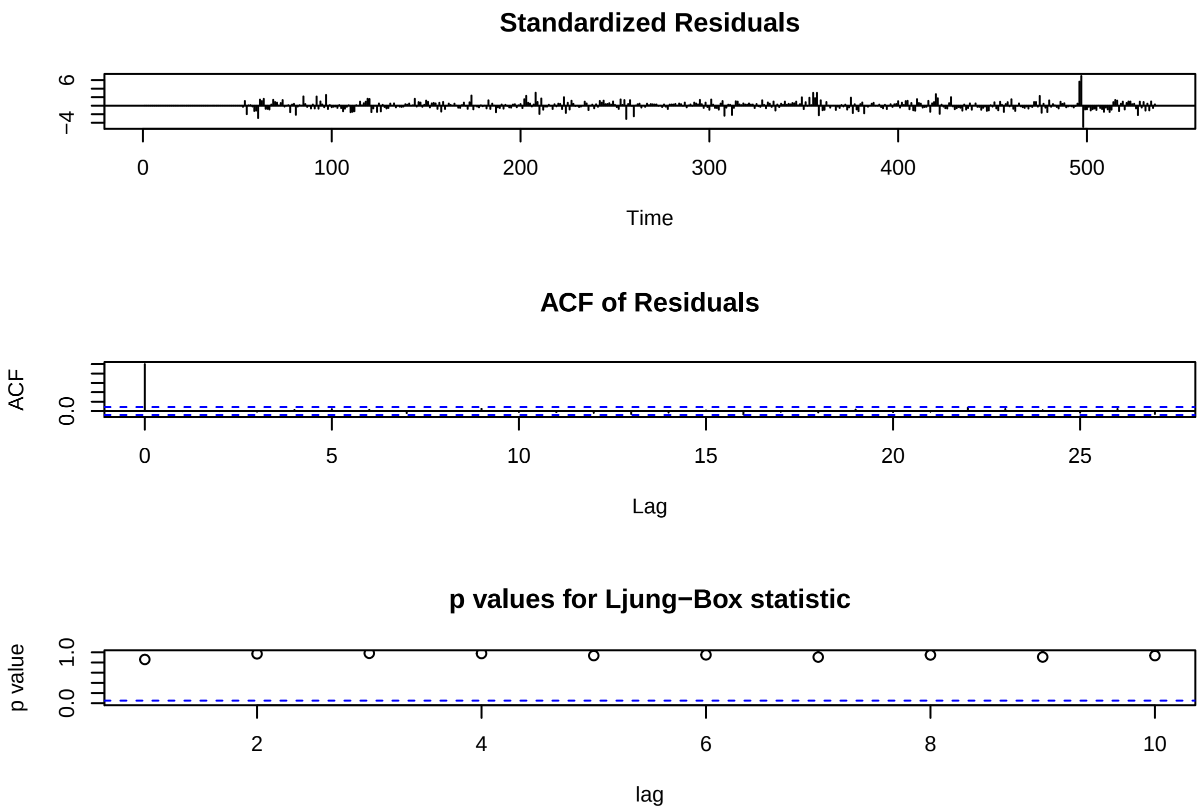
```
#___________Model Diagnostic________
# ###Residuals and P-value
Q3_1 <- arima(Q3train, order = c(1, 1, 0), seasonal = list(order = c(1, 1, 0), period = 52)) #better
tsdiag(Q3_1)  #aic = 2450.82
```



```
#Ljung-Box test seems to suggest that I did not do a good job in finding the right model
# Overfitting for diagnostics, we want to minimize the AIC AIC(m1) =2450.82
#Q3_2<- arima(Q3train, order = c(1, 1, 0), seasonal = list(order = c(2, 1, 0), period = 52),method="CSS
#tsdiag(Q3_2) #aic = -2(-1219.73)+2*(2+1+2)=2449
#Q3_3 <- arima(Q3train, order = c(2, 1, 0), seasonal = list(order = c(1, 1, 0), period = 52))
#tsdiag(Q3_3) #aic = 2431.49
Q3_4 <- arima(Q3train, order = c(1, 1, 1), seasonal = list(order = c(1, 1, 0), period = 52))
tsdiag(Q3_4) #aic = 2372.21 much better, with Ljung_Box shows this model's p_value
```



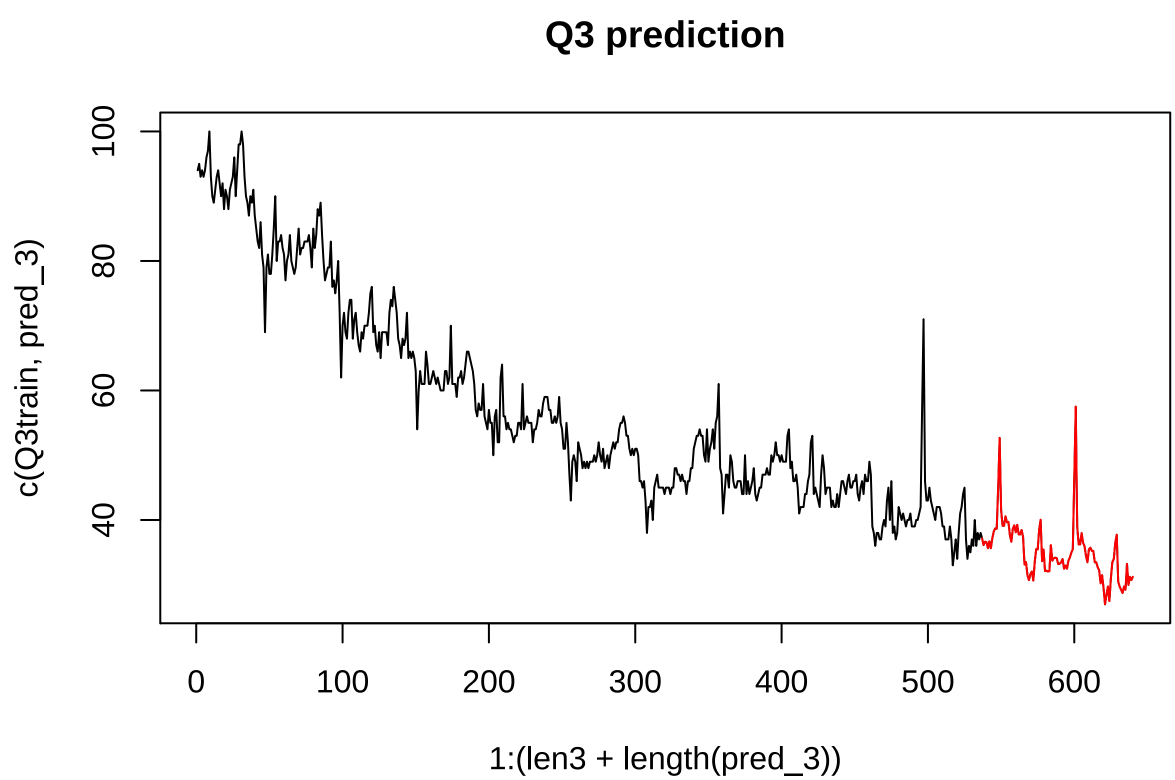

```
#relatively higher than other models.

#Based on c(1, 1, 1)c(1, 1, 0)52 to overfit more models
#Q3_5 <- arima(Q3train, order = c(1, 1, 2), seasonal = list(order = c(1, 1, 0), period = 52))
#tsdiag(Q3_5) #aic =2373.62
#Q3_6 <- arima(Q3train, order = c(1, 1, 1), seasonal = list(order = c(1, 1, 1), period = 52))
#tsdiag(Q3_6) #aic = 2363.62
#Q3_7 <- arima(Q3train, order = c(0, 1, 2), seasonal = list(order = c(1, 1, 1), period = 52))
#tsdiag(Q3_7) #aic = aic = 2369.38
Q3_8 <- arima(Q3train, order = c(0, 1, 3), seasonal = list(order = c(1, 1, 1), period = 52))
tsdiag(Q3_8)  #aic = aic = 2362.79
```




```
#Seems all of them have very close small AIC, Q3_8 has highest Ljung_Box p-value and smallest AIC.
#Personally, I will choose it as fitted model.


# ###AIC and BIC
#BIC(Q3_1)
#BIC(Q3_2)
#BIC(Q3_3)
#BIC(Q3_4)
#BIC(Q3_5)
#BIC(Q3_6) #chose for smallest BIC=2384.518
#BIC(Q3_7)
#BIC(Q3_8) #chose for smallest AIC=2362.79

####Cross_Validation

computeCVmse <- function(order.totry, seasorder.totry){
  MSE <- numeric()
for(k in 4:9)
{
  train.dt = Q3train[1:(k*52+16)]
  test.dt = Q3train[((k*52+16)+1):((k*52+16) + 52)]
  fm1 = arima(train.dt, order = order.totry, seasonal = list(order = seasorder.totry, period = 52))
  fcast.m1 = predict(fm1, n.ahead = 52)
  MSE[k-4] = mean(((fcast.m1$pred) - (test.dt))^2)
}
   return (MSE)
}
```

```r
#MSE3_1<- computeCVmse(c(1, 1, 0), c(0,1,1)) #18.77068
#MSE3_2<- computeCVmse(c(1, 1, 0), c(1,1,0)) #16.58912
#MSE3_3 <- computeCVmse(c(0, 1, 1), c(0,1,1)) #21.32714
#MSE3_4<- computeCVmse(c(0, 1, 1), c(1,1,0)) #19.76607
####Based on c(1, 1, 0)c(1,1,0)52 to overfit other models.
#MSE3_5<- computeCVmse(c(1, 1, 1), c(1,1,0)) #21.76579
#MSE3_6<- computeCVmse(c(1, 1, 0), c(1,1,1)) #18.01934
#MSE3_7<- computeCVmse(c(2, 1, 0), c(1,1,0)) #17.42202
#MSE3_8<- computeCVmse(c(1, 1, 0), c(1,1,1)) #18.01934

#Seems like after change different formations, the simple c(1, 1, 0)c(1,1,0)52 still looks good to fit
```

```r
####Forecast
pred_3 <- predict(Q3_1, n.ahead = 104)$pred
plot(1:(len3 + length(pred_3)), c(Q3train, pred_3), type = 'l', col = 1, main="Q3 prediction")
points((len3 + 1) : (len3 + length(pred_3)), pred_3, type = 'l', col = 2)
```



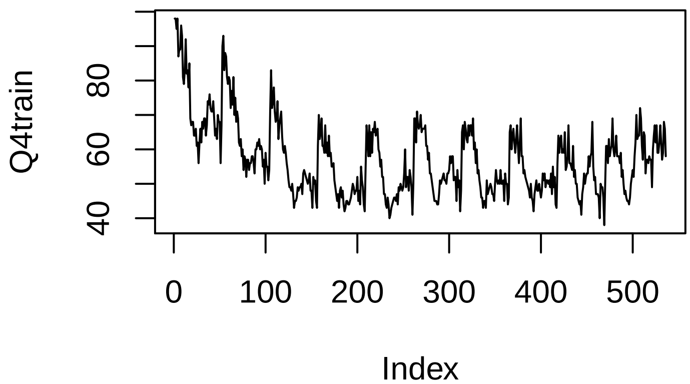

```r
#Create the file:
write.table(pred_3,
            sep = ",",
            col.names = FALSE,
            row.names = FALSE,
            file = "Q3_Huidi_Wang_25157840.txt")
```

*

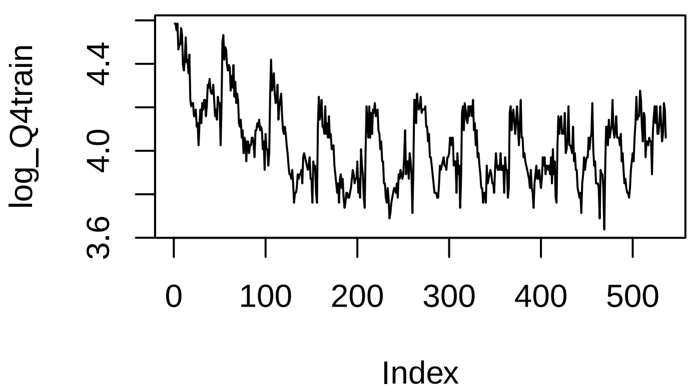
*Q4train Analysis**

```r
#Extract dataset from original csv.file
Q4train.raw=read.delim(file="q4train.csv")
Q4train_data=Q4train.raw[1:536,1]
len4=length(Q4train_data)
Q4train=rep(0,len4)
for (i in 1:len4){
  Q4train[i]=as.numeric(unlist(strsplit(as.character(Q4train_data[i]),","))[2])
}
```

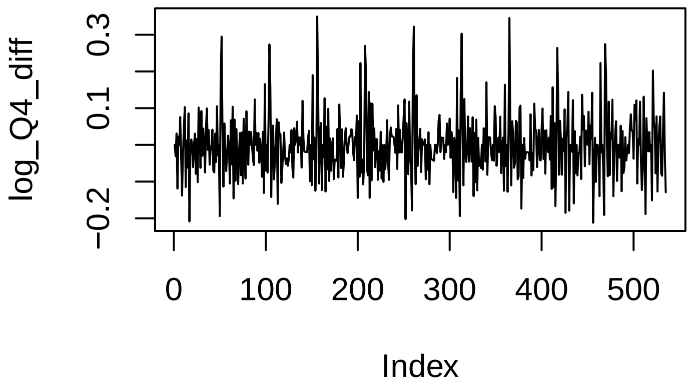
```r
# Plots of raw data, log data, acf and pacf
plot(Q4train, type="l")
```



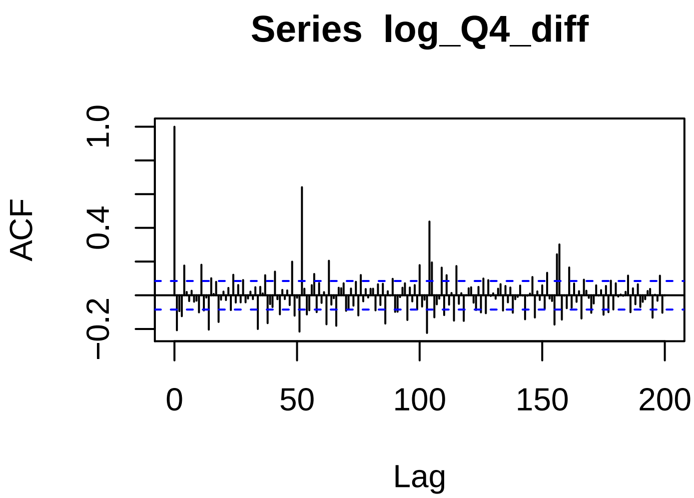

```r
log_Q4train=log(Q4train)
plot(log_Q4train, type="l")
```



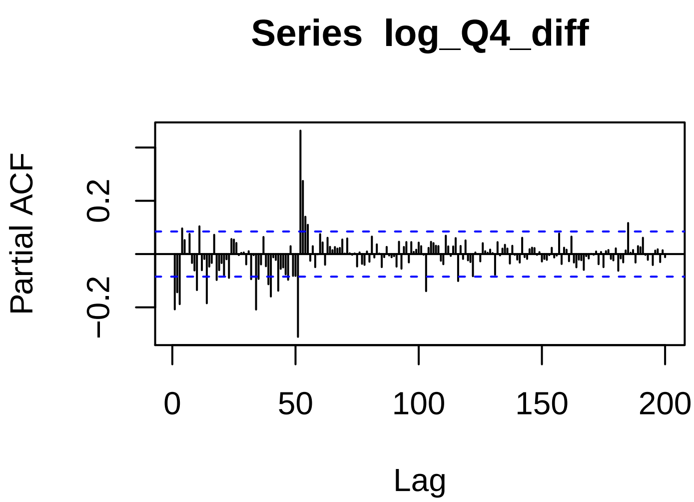

```r
log_Q4_diff=diff(log_Q4train)
plot(log_Q4_diff, type="l")
```



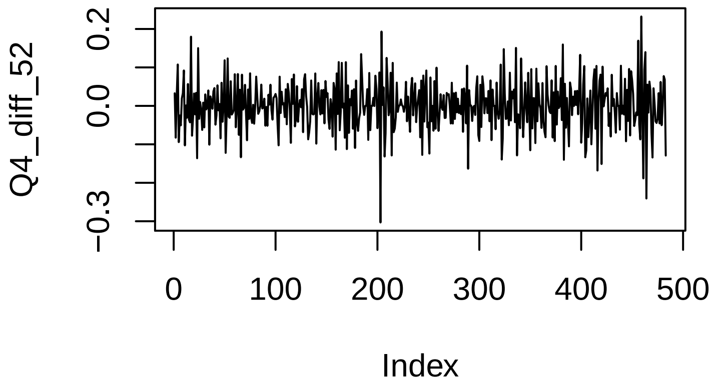

```
acf(log_Q4_diff, lag.max=200)
```



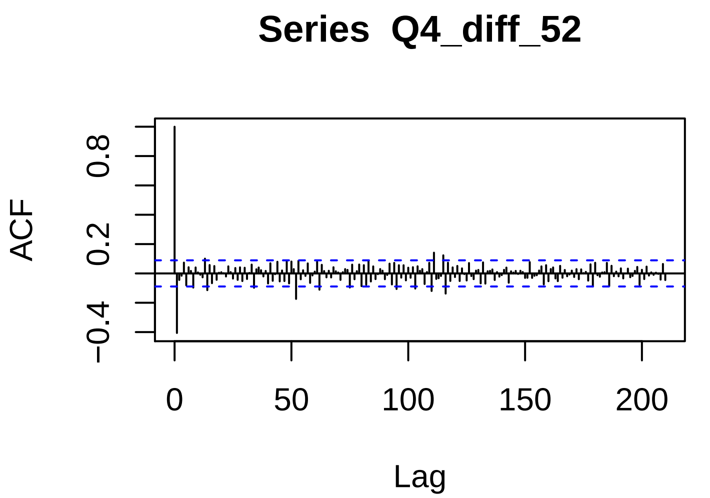

```
#[53,]  0.641294044
pacf(log_Q4_diff, lag.max=200)
```



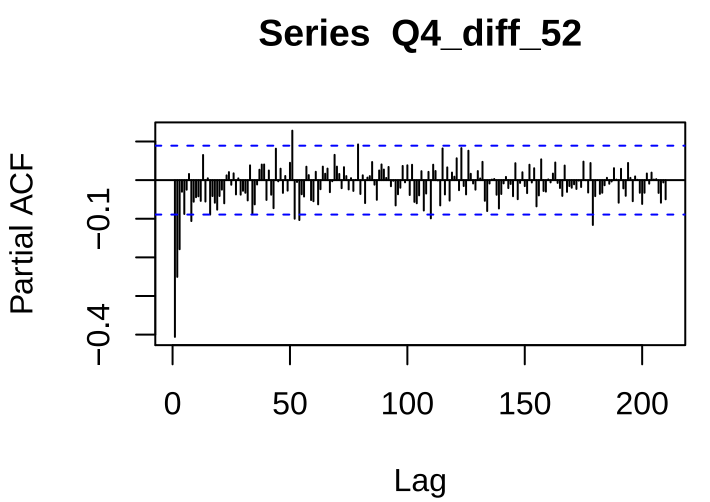

```
Q4_diff_52=diff(log_Q4_diff,52)
plot(Q4_diff_52, type="l")
```




```r
acf(Q4_diff_52, lag.max=210)
```

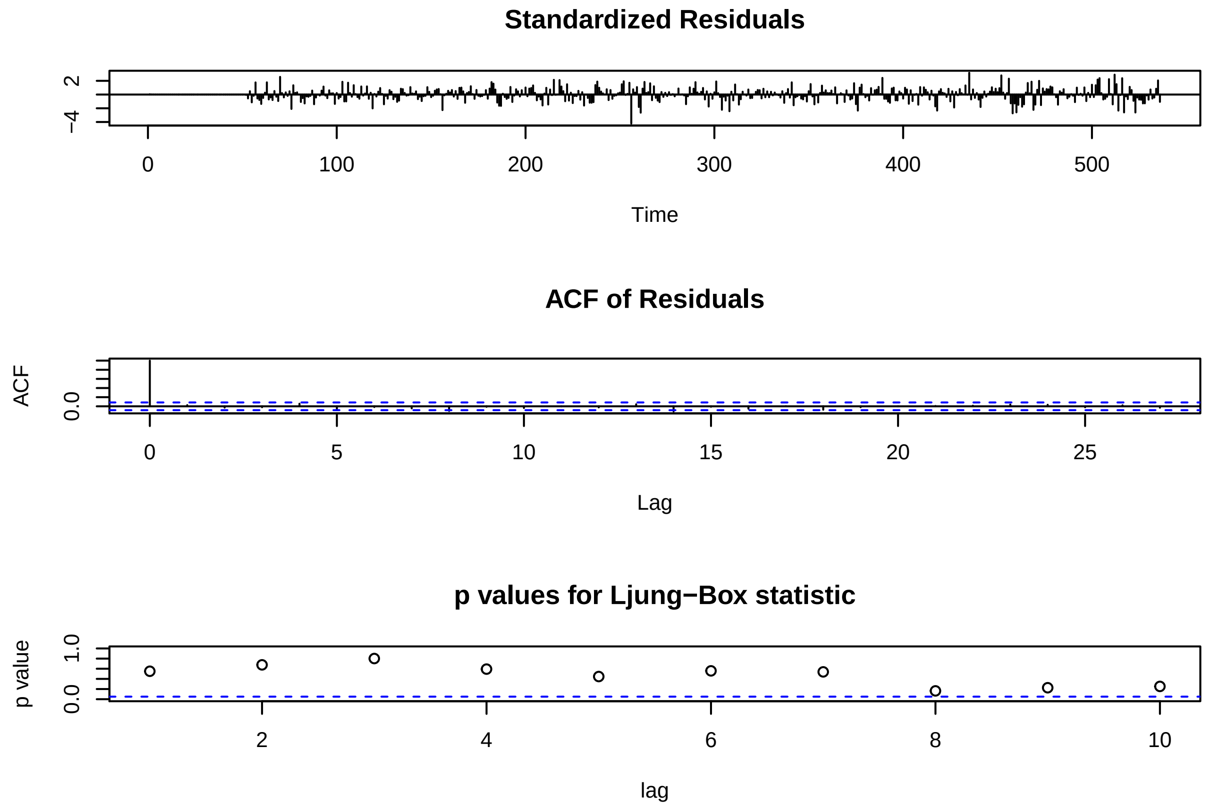



```r
#[53,] -0.1738370170
pacf(Q4_diff_52, lag.max=210)
```

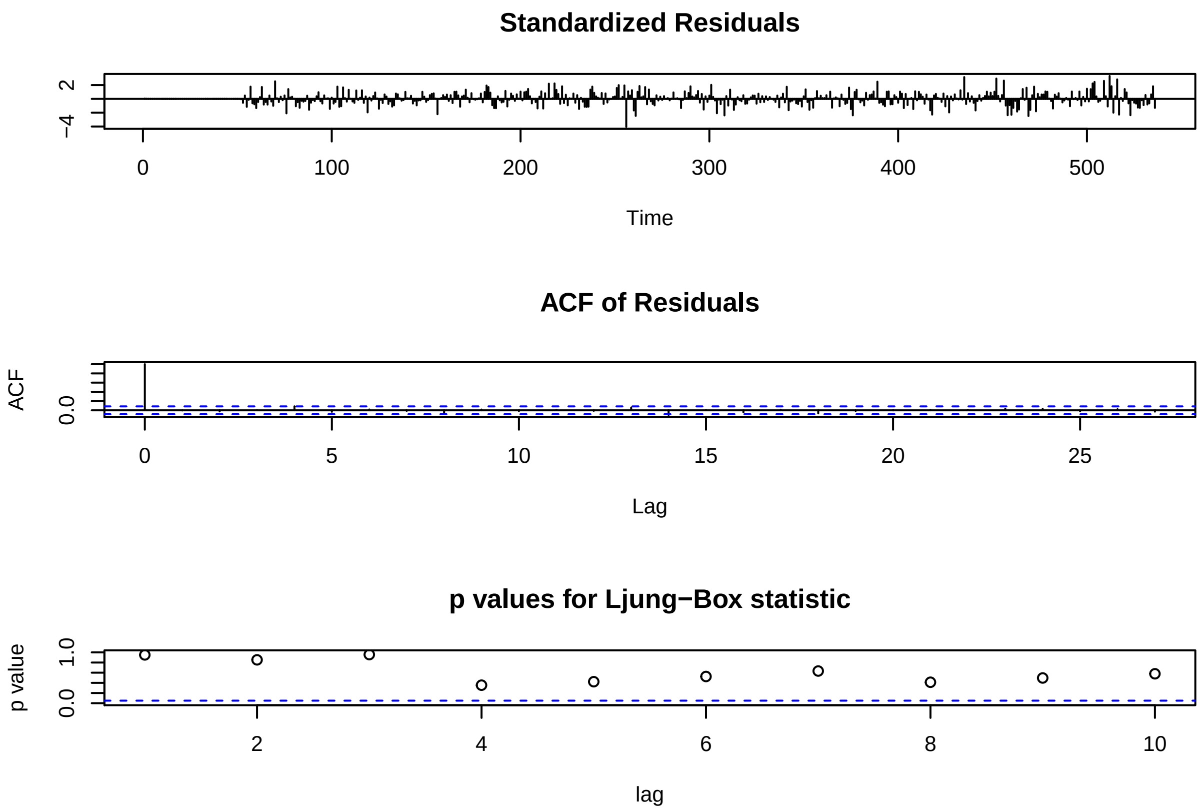



```r
#Little bit tricky this dataset. Probably like MA(1)MA(1)_52
```

```r
#___________Model Diagnostic________
# ###Residuals and P-value
Q4_1 <- arima(log_Q4train, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1), period = 52))
tsdiag(Q4_1) #aic=-1408.11
```




```r
#Seems not bad if we see the p-values, so try to minimize aic
#Q4_2 <- arima(log_Q4train, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 2), period = 52))
#tsdiag(Q4_2) #aic=-1407.12
#Q4_3 <- arima(log_Q4train, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 1), period = 52))
#tsdiag(Q4_3) #aic=-1407.23
Q4_4 <- arima(log_Q4train, order = c(1, 1, 1), seasonal = list(order = c(0, 1, 1), period = 52))
#tsdiag(Q4_4) #aic = -1407.28
Q4_5 <- arima(log_Q4train, order = c(1, 1, 2), seasonal = list(order = c(0, 1, 1), period = 52))
tsdiag(Q4_5) #aic = -1413.64 better with higher P-value and smaller AIC==> chosen
```



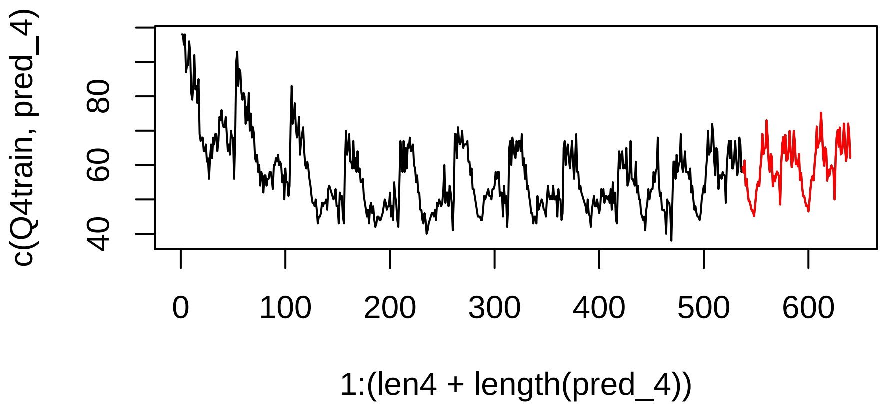

```
#Q4_6<- arima(log_Q4train, order = c(1, 1, 2), seasonal = list(order = c(0, 1, 2), period = 52))
#tsdiag(Q4_6) #aic = -1412.63
```

```
#AIC and BIC method
#BIC(Q4_1) #chosen by smallest BIC=-1395.573
#BIC(Q4_2)
#BIC(Q4_3)
#BIC(Q4_4)
#BIC(Q4_5) #chosen by smallest AIC=-1413.64
#BIC(Q4_6)
```

```
### Cross Validation
computeCVmse <- function(order.totry, seasorder.totry){
  MSE <- numeric()
  for(k in 5:9)
  {
    train.dt = log_Q4train[1:(k*52+16)]
    test.dt = log_Q4train[((k*52+16)+1):((k*52+16) + 52)]
    fm1 = arima(train.dt, order = order.totry, seasonal = list(order = seasorder.totry, period = 52))
    fcast.m1 = predict(fm1, n.ahead = 52)
    MSE[k-4] = mean((exp(fcast.m1$pred) - exp(test.dt))^2)
  }
  return (MSE)
}

#MSE4_1 <- computeCVmse(c(0, 1, 1), c(0,1,1))  #20.70567
#MSE4_2 <- computeCVmse(c(0, 1, 1), c(1,1,0))  #21.02253
```

```
#MSE4_3 <- computeCVmse(c(1, 1, 0), c(1,1,0))  #20.43283
#MSE4_4 <- computeCVmse(c(1, 1, 0), c(0,1,1))  #20.11962
#####Looks like their MSE are not too much different.
#MSE4_5 <- computeCVmse(c(0, 1, 2), c(0,1,1))  #20.66754 better
#MSE4_6 <- computeCVmse(c(1, 1, 1), c(0,1,1))  #20.61996 better
#MSE4_7 <- computeCVmse(c(0, 1, 1), c(1,1,1))  #21.16821
#MSE4_9 <- computeCVmse(c(0, 1, 2), c(1,1,1))  #21.19115
#MSE4_10 <- computeCVmse(c(0, 1, 3), c(0,1,1))  #20.58602
#MSE4_11 <- computeCVmse(c(0, 1, 3), c(1,1,1)) #21.12911
#MSE4_12 <- computeCVmse(c(1, 1, 2), c(0,1,1)) #21.12911
####Finally, I will choose c(1, 1, 1)c(0,1,1)_52
```

```
pred_4 <- exp(predict(Q4_4, n.ahead = 104)$pred)
plot(1:(len4 + length(pred_4)), c(Q4train, pred_4), type = 'l', col = 1)
points((len4 + 1) : (len4 + length(pred_4)), pred_4, type = 'l', col = 2)
```




```
## Let's create the file:
write.table(pred_4,
            sep = ",",
            col.names = FALSE,
            row.names = FALSE,
            file = "Q4_Huidi_Wang_25157840.txt")
```