Assignment 1

Ong Jia Hui JONG119@e.ntu.edu.sg
G1903467L

1. Problem

The wwwusage time series data consists of the number of users connected to the internet through a server. The data are collected at a time interval of one minute and there are 100 observations. Please fit an appropriate ARIMA model for it and submit a short report including R codes, the fitted model, the diagnostic checking, AIC, etc.

1.1 Data

The wwwusage.txt data file is as follows:

```
"x"
88
84
85
85
84
85
83
85
88
89
91
99
104
112
126
138
146
151
150
148
147
149
143
132
131
139
147
150
148
145
140
134
131
131
129
126
126
132
137
140
142
150
159
167
170
171
172
172
174
175
172
172
174
174
169
165
156
142
131
121
112
104
102
99
99
95
88
84
84
87
89
88
85
86
89
91
91
94
101
110
121
135
145
149
156
165
171
175
177
182
193
204
208
210
215
222
228
226
222
220
```

2. Initial Step

This section describes the inital steps taken to find out the appropriate ARIMA model for the time series data from Section 1.1.

2.1 Original Time Plot

A time series is said to be weakly stationary if the following two conditions are satisfied:

1. Mean is constant throughout time
2. Covariance is independent of time lag

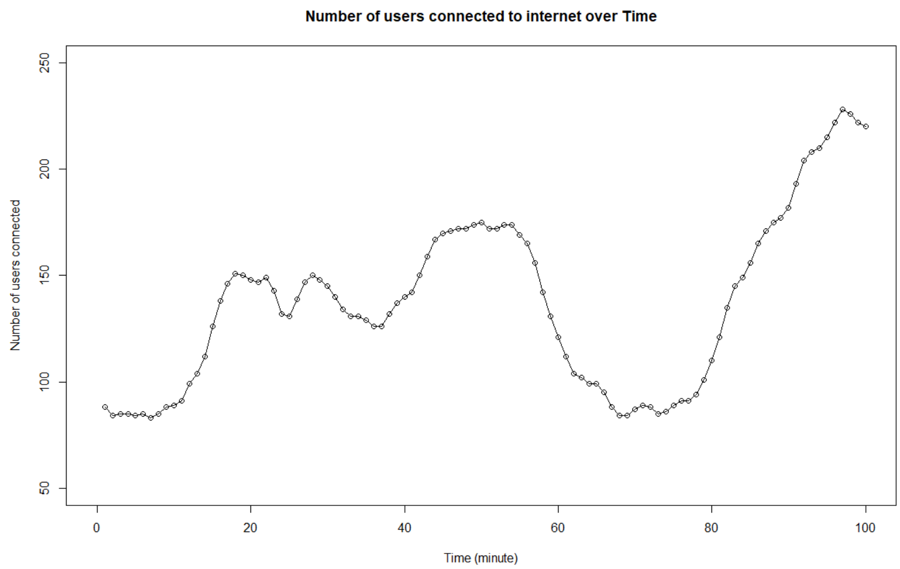
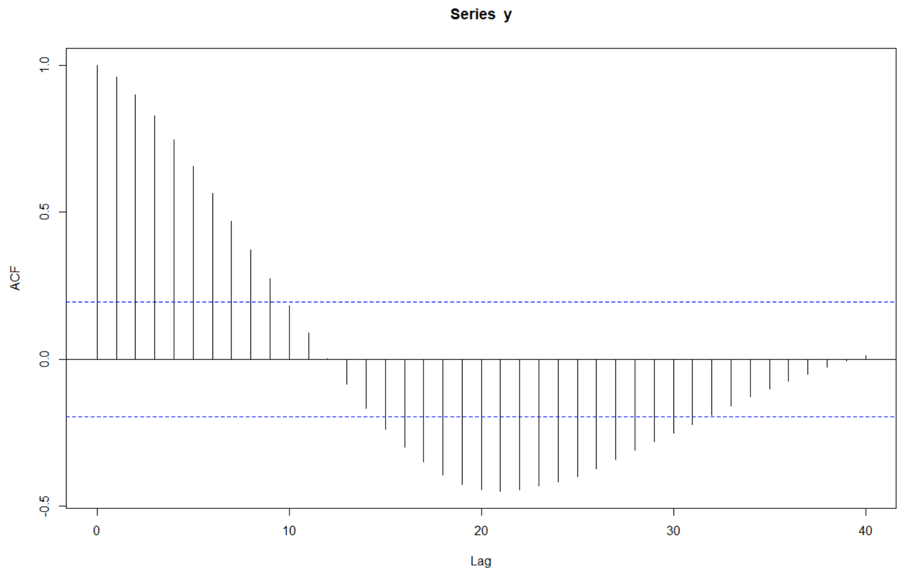


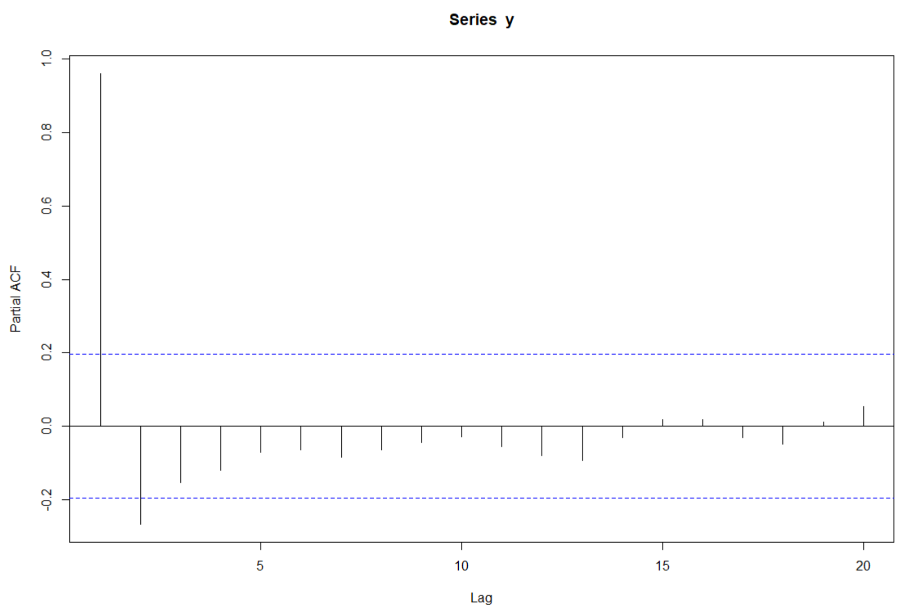
Figure 1: Time Series Plot of Original Data

However, the original time plot in Figure 1 appears to have a upward trending component, i.e. its mean is not constant throughout time, therefore it implies that the data is non-stationary.

2.2 ACF and PACF of Original Data



(i) ACF Plot of Original Data



(ii) PACF Plot of Original Data

Figure 2: ACF and PACF Plots of Original Data

The ACF plot in Figure 2i also shows that the ACF does not cut off until lag 32. The PACF plot in Figure 2ii cuts off at lag 2.

3. Models after One-Time Differencing (d=1)

3.1 Difference Transform

As the data appears to have a trending component, we apply one time differencing to remove the trending component. $\mathbf{Z_t} = \mathbf{X_t} - \mathbf{X_{t-1}}$ using R function *diff(y)*.

3.2 Time Plot After One-Time Differencing

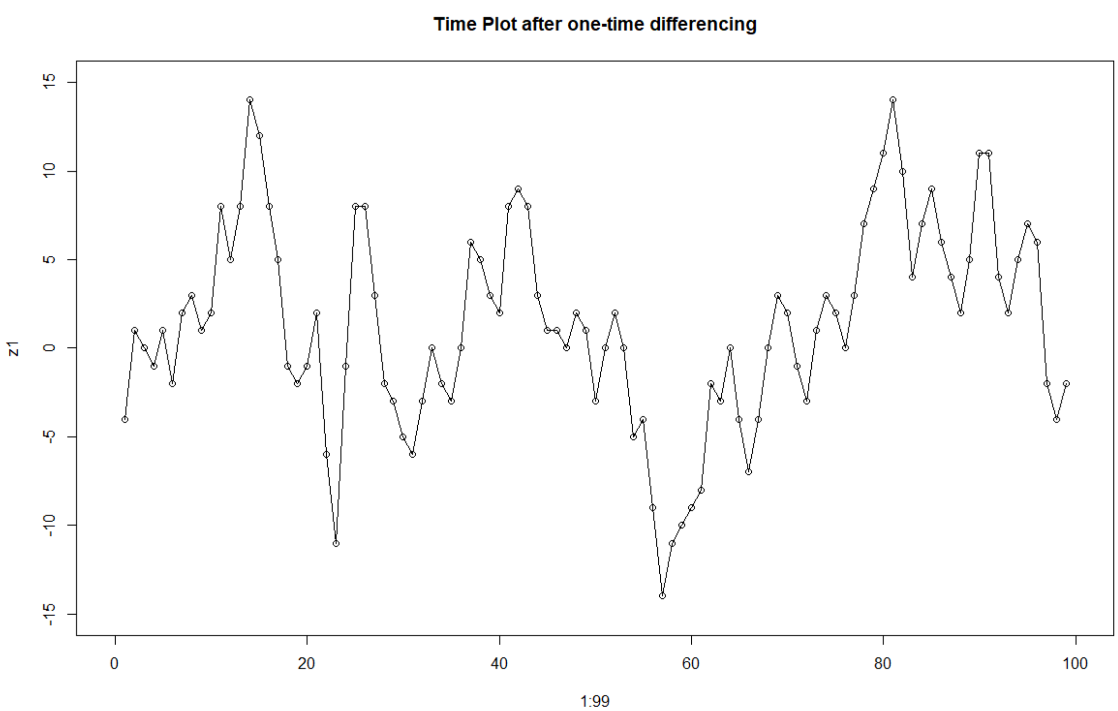
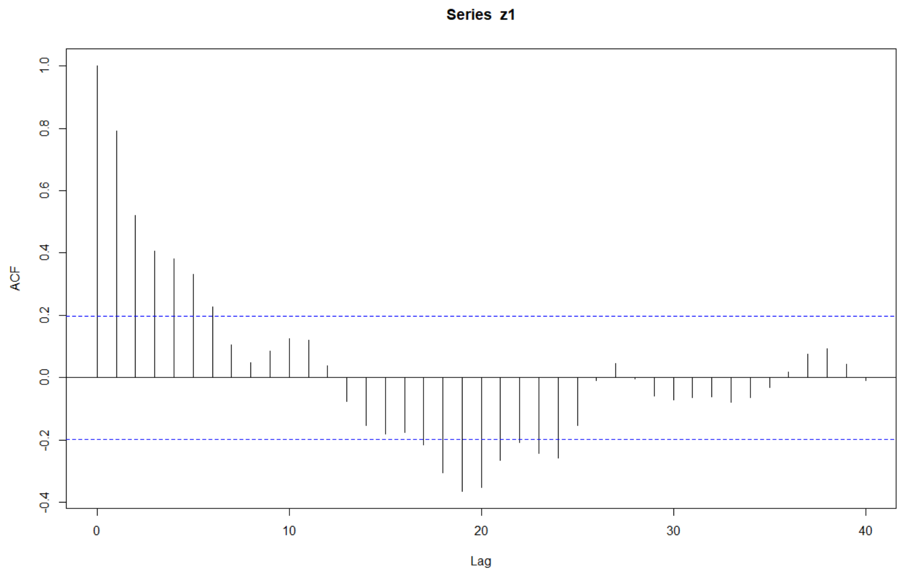
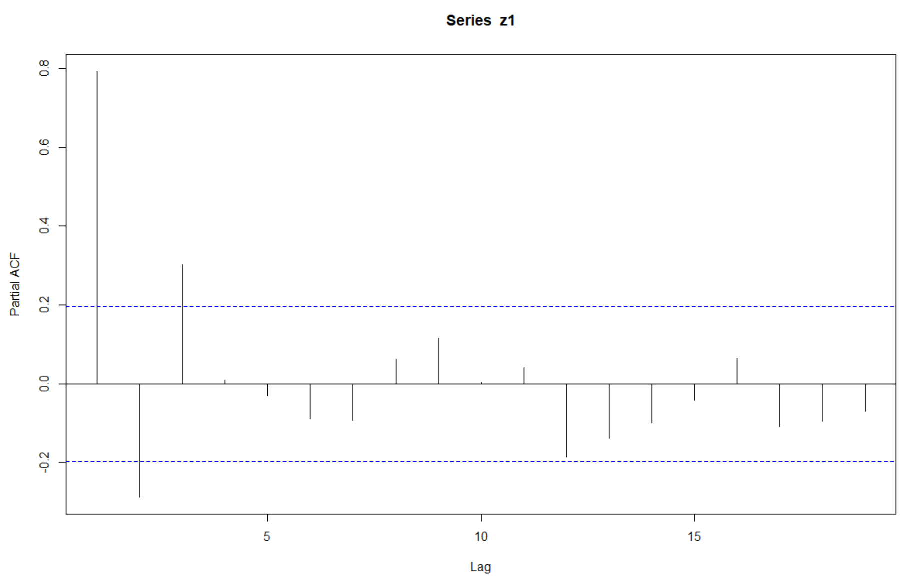


Figure 3: Time Series Plot After One-Time Differencing

After one time differencing is applied, the trending component is removed as seen in Figure 3. The time plot now has 99 observations and looks more stationary.



(i) ACF Plot After One-Time Differencing



(ii) PACF Plot After One-Time Differencing

Figure 4: ACF and PACF Plots After One-Time Differencing

The ACF plot in Figure 4i shows that the ACF does not cut off until lag 24 and PACF plot in Figure 4ii shows that it cuts off at lag 3. This suggests a possible model of ARIMA(3,1,0) for the time series data. Furthermore, the *ar.yw()* Yule Walker function also suggested order 3 on the differenced data when used to estimate the the AR coefficient.

```
ar.yw.default(x = z, order.max = 5)
```

```
Coefficients:
      1        2        3
 1.1060  -0.5957   0.3029

Order selected 3  sigma^2 estimated as  10.32
```

3.3 ARIMA(3,1,0) Model Diagnostics

The ARIMA(3,1,0) is used to fit the original time series data. A diagnostic check was conducted on the fitted model using *tsdiag(fit)*.

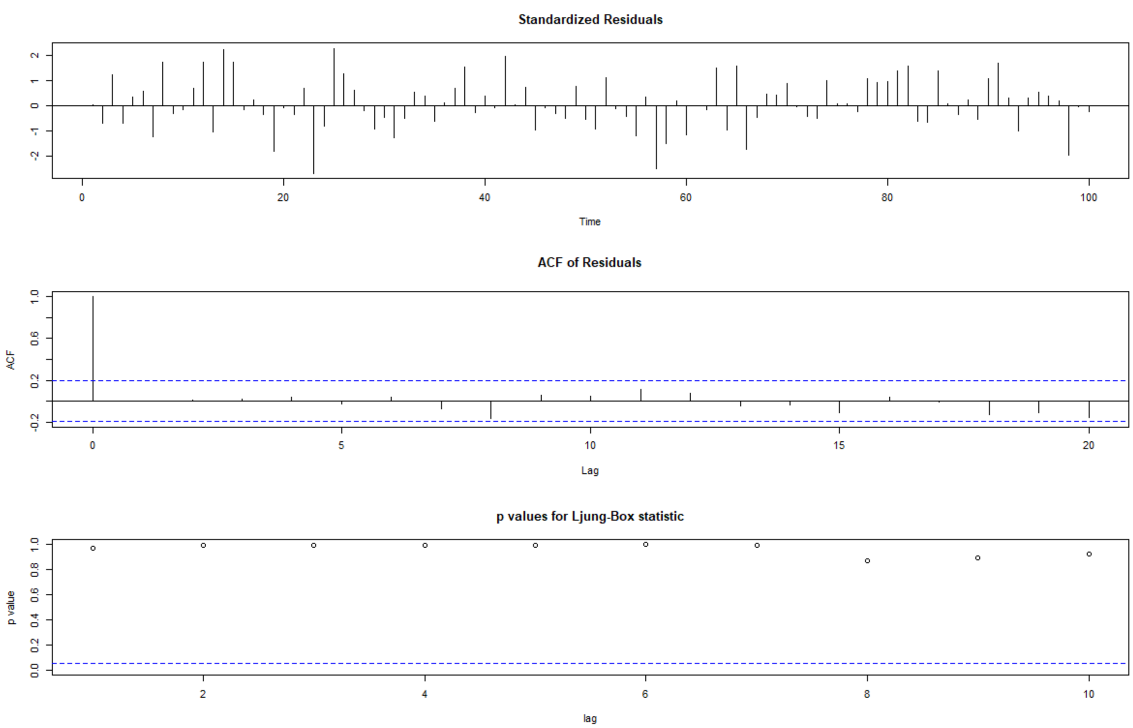


Figure 5: Diagnostic Check on ARIMA(3,1,0)

The diagnostic check on ARIMA(3,1,0) as shown in Figure 5 justifies that the fitted model is OK (adequate) as it summarizes the following:

- The residuals looks to be random, which means it resembles white noise.
- The ACF of the residuals cuts off after lag 0.
- The p-values of Ljung-Box statistics are all above 0.05, therefore significant.

The AIC and BIC values of the fitted ARIMA(3,1,0) model are 511.994 and 522.3745 respectively.

3.4 ARIMA(1,1,1) Model Diagnostics

A model diagnostics was also carried out on ARIMA(1,1,1) model, which was suggested by the *auto.arima* function.

```
> fitauto <- auto.arima(y,max.p = 5,max.q = 5,max.P = 5,max.Q = 5,
                        max.d = 3,seasonal = FALSE,ic = 'aicc')
> fitauto
Series: y
ARIMA(1,1,1)

Coefficients:
         ar1     ma1
      0.6504  0.5256
s.e.  0.0842  0.0896

sigma^2 estimated as 9.995:  log likelihood=-254.15
AIC=514.3   AICc=514.55   BIC=522.08
```

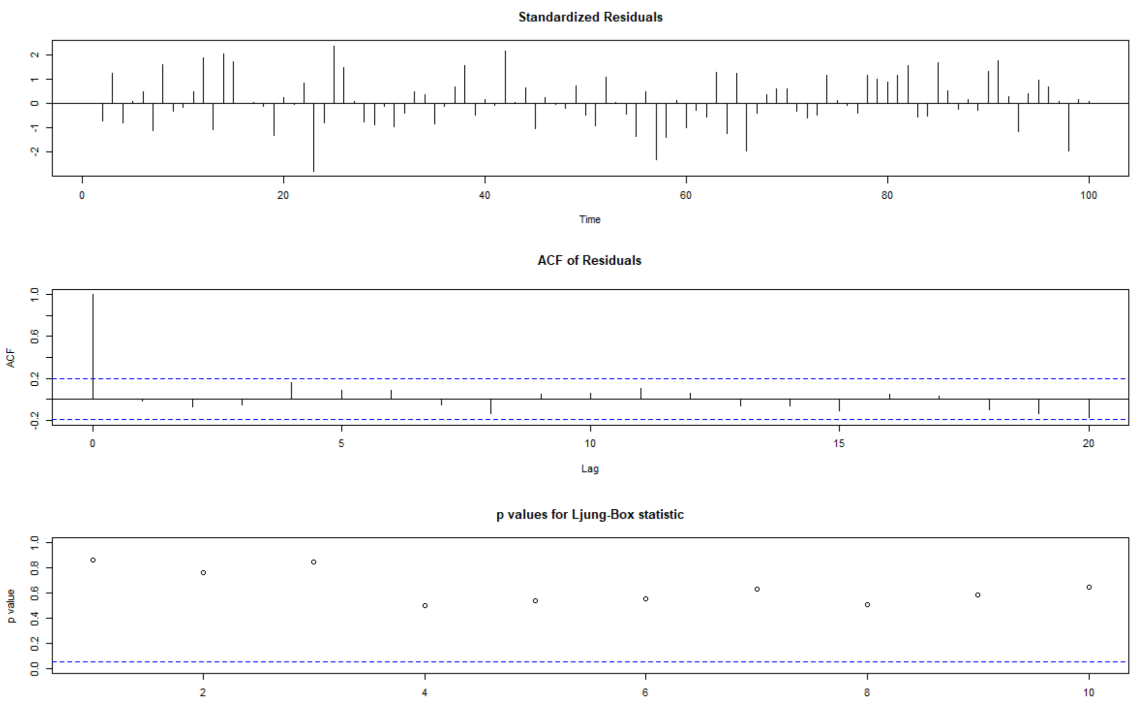


Figure 6: Diagnostic Check on ARIMA(1,1,1)

The diagnostic check on ARIMA(1,1,1) as shown in Figure 6 justifies that the fitted model is OK (adequate) as it summarizes the following:

- The residuals looks to be random, which means it resembles white noise.
- The ACF of the residuals cuts off after lag 0.
- The p-values of Ljung-Box statistics are all above 0.05, therefore significant.

The AIC and BIC values of the fitted ARIMA(1,1,1) model are 514.2995 and 522.0848 respectively. It has a higher AIC than ARIMA(3,1,0), but a lower BIC than ARIMA(3,1,0).

4. Models after Two-Time Differencing (d=2)

4.1 Difference Transform

After trying out two d=1 models, another round of differencing was applied to find some adequate d=2 models.

4.2 Time Plot After Two-Time Differencing

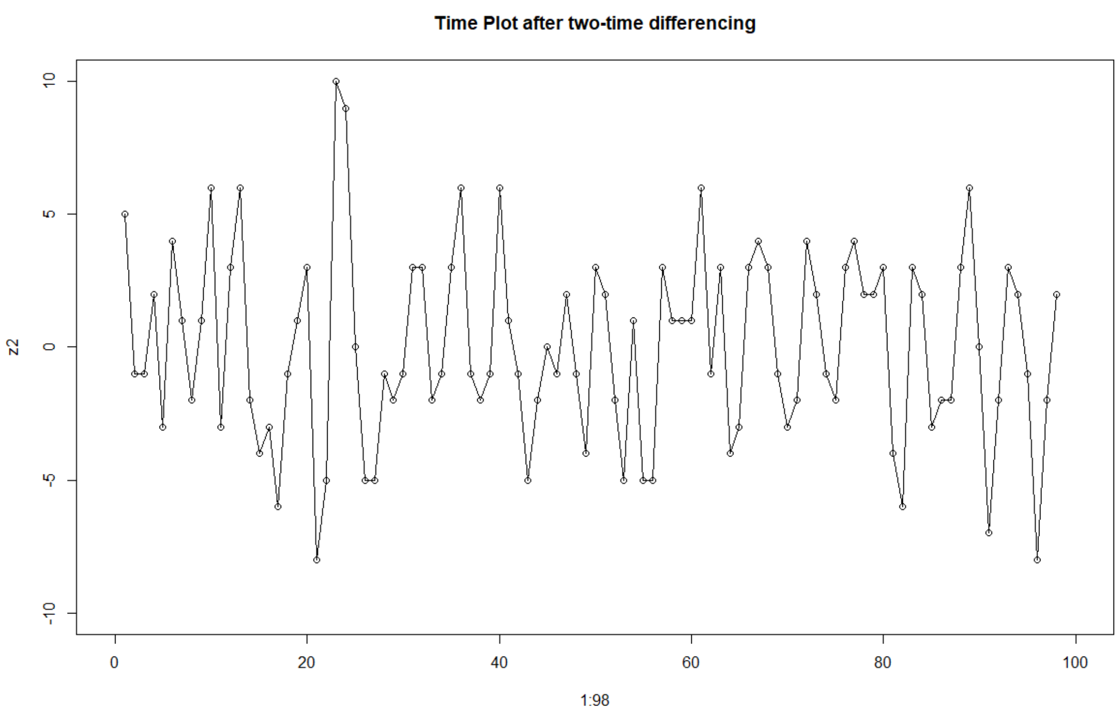
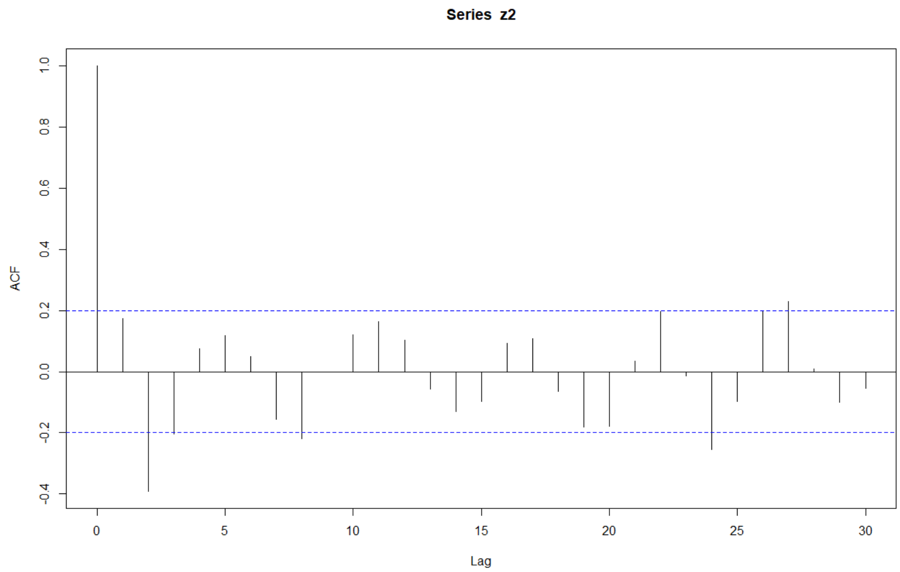
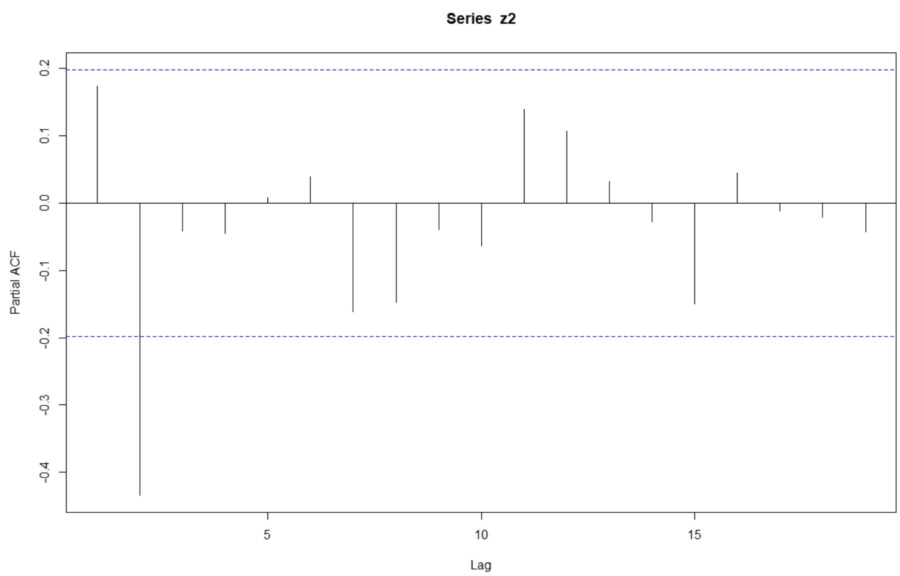


Figure 7: Time Series Plot After Two-Time Differencing

After one more time of differencing, the time plot can be seen as Figure 7. It now has 98 observations.



(i) ACF Plot After Two-Time Differencing



(ii) PACF Plot After Two-Time Differencing

Figure 8: ACF and PACF Plots After Two-Time Differencing

The ACF plot in Figure 8i shows that the ACF does not cut off until lag 27 and PACF plot in Figure 8ii cuts off at lag 2. This suggests a possible model of ARIMA(2,2,0) for the time series data. Furthermore, using *ar.yw()* Yule Walker function on the differenced data to estimate the the AR coefficient also suggested order 2.

4.3 ARIMA(2,2,0) Model Diagnostics

The ARIMA(2,2,0) is used to fit the original time series data. A diagnostic check was conducted on the fitted model using *tsdiag(fit)*.

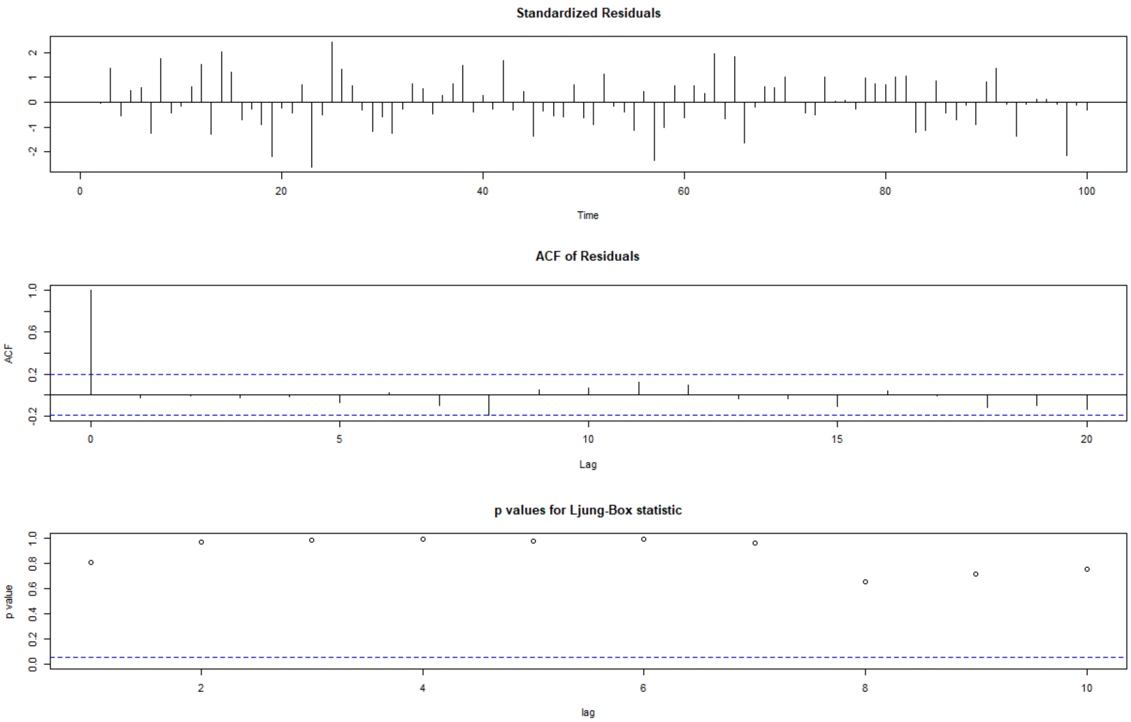


Figure 9: Diagnostic Check on ARIMA(2,2,0)

The diagnostic check on ARIMA(2,2,0) as shown in Figure 9 justifies that the fitted model is OK (adequate) as it summarizes the following:

- The residuals looks to be random, which means it resembles white noise.
- The ACF of the residuals cuts off after lag 0.
- The p-values of Ljung-Box statistics are all above 0.05, therefore significant.

The AIC and BIC values of the fitted ARIMA(2,2,0) model were found to be 511.4645 and 519.2194 respectively. The AIC and BIC values are both lower than the one-time differencing models.

4.4 ARIMA(5,2,5) Model Diagnostics

Using a trial and error approach, ARIMA(5,2,5) model was found to have the lowest AIC value. Hence, it is used to evaluate if the lowest AIC model means the best fitting model. The codes for finding the model with the best AIC and BIC can be found in Appendix B.

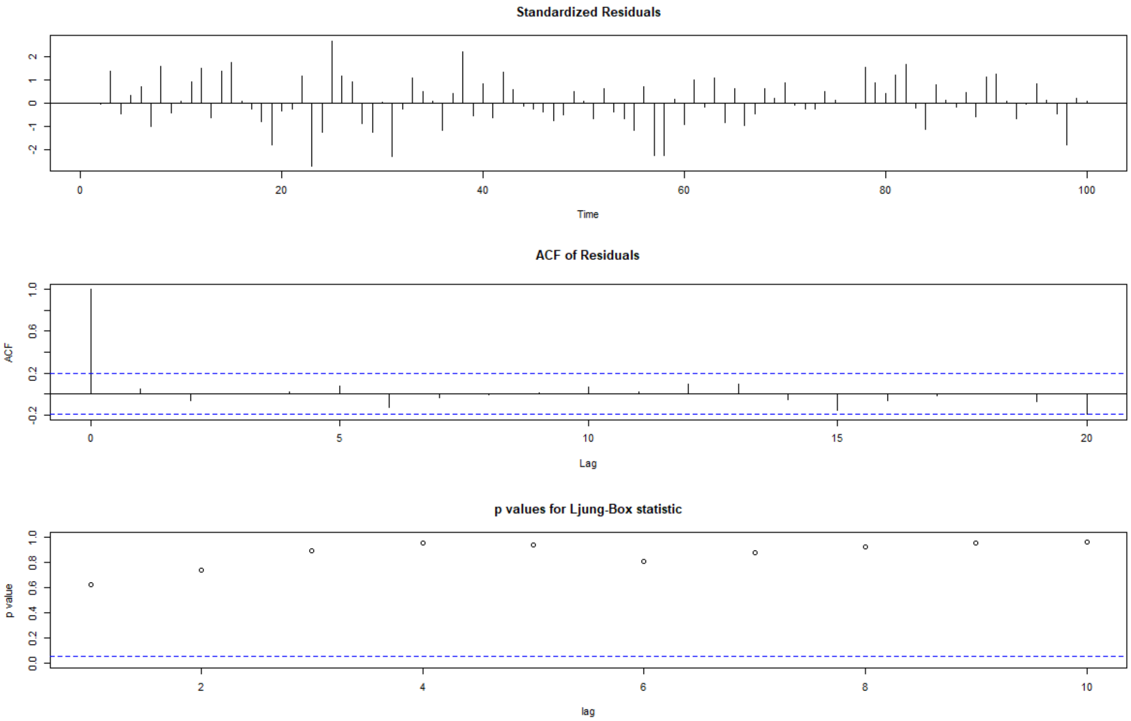


Figure 10: Diagnostic Check on ARIMA(5,2,5)

The diagnostic check on ARIMA(5,2,5) as shown in Figure 10 justifies that the fitted model is OK (adequate) as it summarizes the following:

- The residuals looks to be random, which means it resembles white noise.
- The ACF of the residuals cuts off after lag 0.
- The p-values of Ljung-Box statistics are all above 0.05, therefore significant.

The AIC and BIC values of the fitted ARIMA(5,2,5) model are 509.8135 and 538.2481 respectively. It has the lowest AIC values out of all four models fitted.

5. AIC, Fitted Values and Forecast Values Analysis

AIC and BIC measure the error and penalize adding of parameters. Table 1 shows the AIC and BIC values of all the models test.The lowest AIC value was achieved by the ARIMA(5,2,5) model.

Table 1: AIC and BIC of the fitted models

ARIMA	*AIC*	*BIC*
(3,1,0)	511.994	522.3745
(1,1,1)	514.2995	522.0848
(2,2,0)	511.4645	519.2194
(5,2,5)	**509.8135**	538.2481

Since all four models, namely ARIMA(3,1,0), ARIMA(1,1,1), ARIMA(2,2,0) and ARIMA(5,2,5) are deemed to be adequate using *tsdiag*, the fitted and forecast values are plotted to give a better overview of the models' performance.

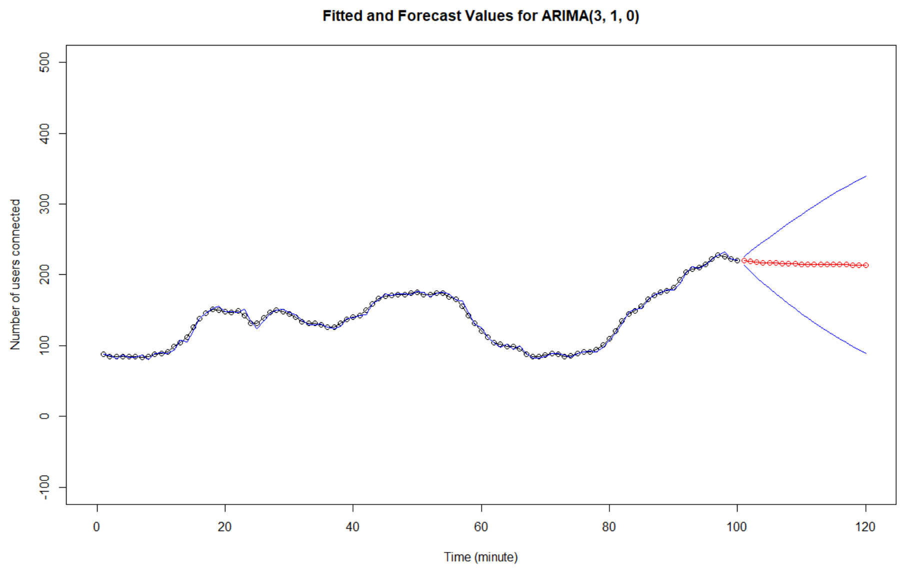


Figure 11: Fitted and Forecast values on ARIMA(3,1,0)

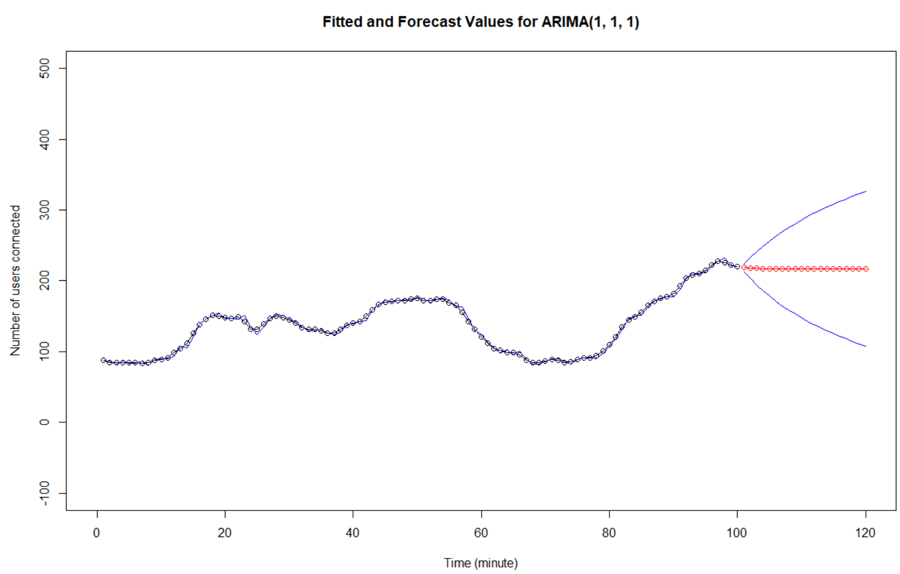


Figure 12: Fitted and Forecast values on ARIMA(1,1,1)

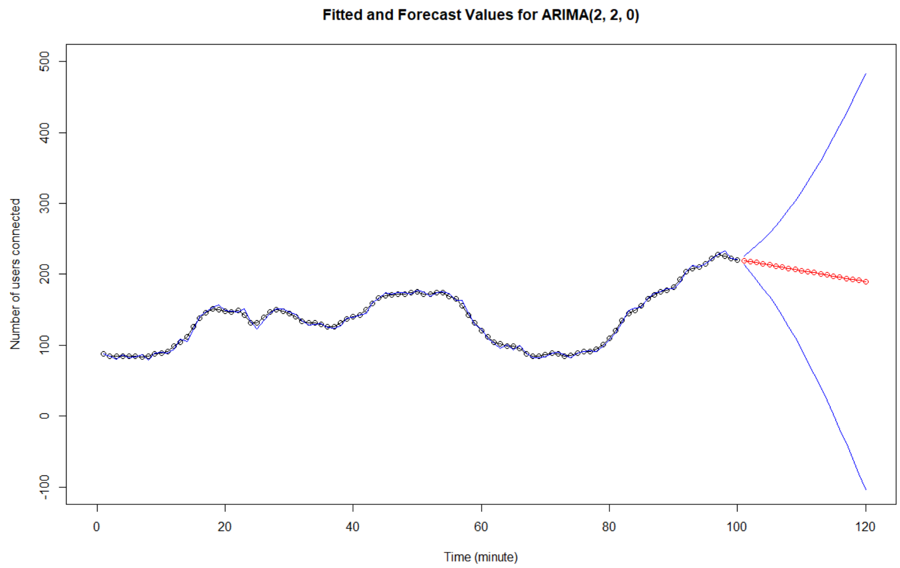


Figure 13: Fitted and Forecast values on ARIMA(2,2,0)

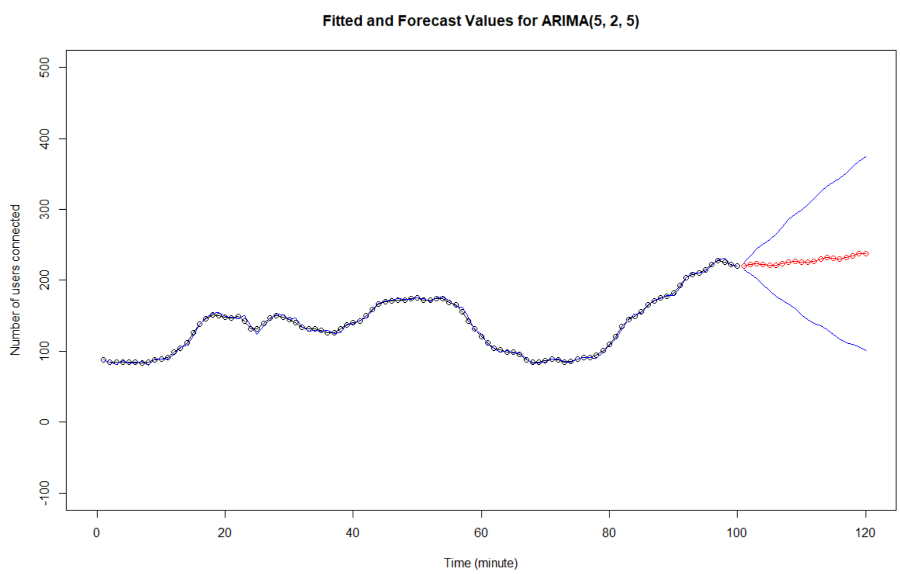


Figure 14: Fitted and Forecast values on ARIMA(5,2,5)

The range between upper and lower confidence of the prediction values for ARIMA(2,2,0) in Figure 13 is the largest among the four models. This aligns with the forecast accuracy test ran on the four models using 10 observations as test set as shown in Table 2. ARIMA(2,2,0) holds the highest values for RSME, MAE and MASE, which suggest that out of the four models, it is not an appropriate model for forecasting.

Table 2: Accuracy of fitted models with test set of 10 observations

$ARIMA$	Set	$RMSE$	MAE	MPE	$MAPE$	$MASE$
(3,1,0)	Training	3.044632	2.367157	0.2748377	1.890528	0.5230995
(3,1,0)	Test	12.071916	10.086919	-1.2910728	4.816833	2.2290289
(1,1,1)	Training	3.113754	2.405275	0.2805566	1.917463	0.5315228
(1,1,1)	Test	11.351388	9.265086	-1.4666403	4.437330	2.0474185
(2,2,0)	Training	3.150308	2.511921	0.2062350	1.994727	0.5550897
(2,2,0)	Test	14.750798	13.118737	0.7787425	6.155561	2.8990065
(5,2,5)	Training	2.713488	2.098567	0.2169541	1.642098	0.4637459
(5,2,5)	Test	12.528193	9.415103	-4.1207123	4.581967	2.0805697

6. Residual Analysis

The *sarima(x,p,d,q,P,D,Q,S)* function was used to analyze the residuals of the models with seasonality component turned off.

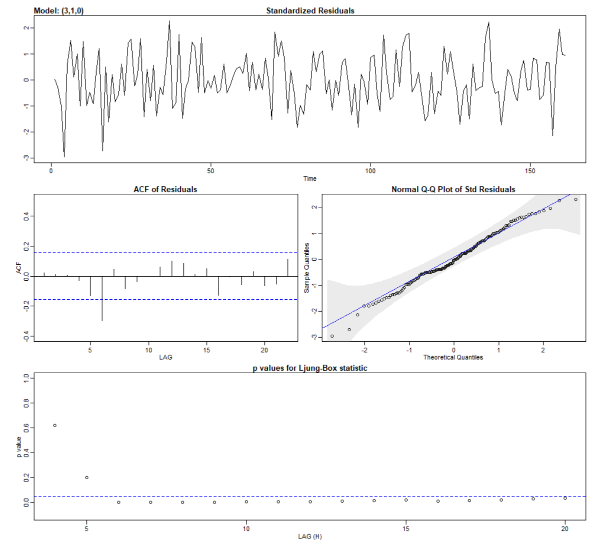


Figure 15: Residual Analysis using SARIMA(3,1,0)

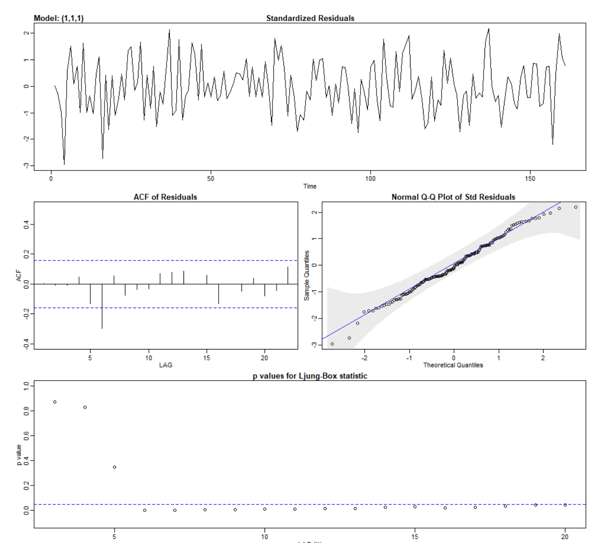


Figure 16: Residual Analysis using SARIMA(1,1,1)

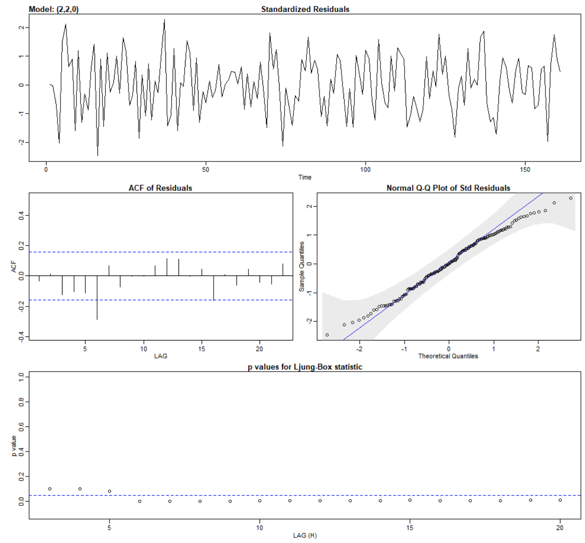


Figure 17: Residual Analysis using SARIMA(2,2,0)

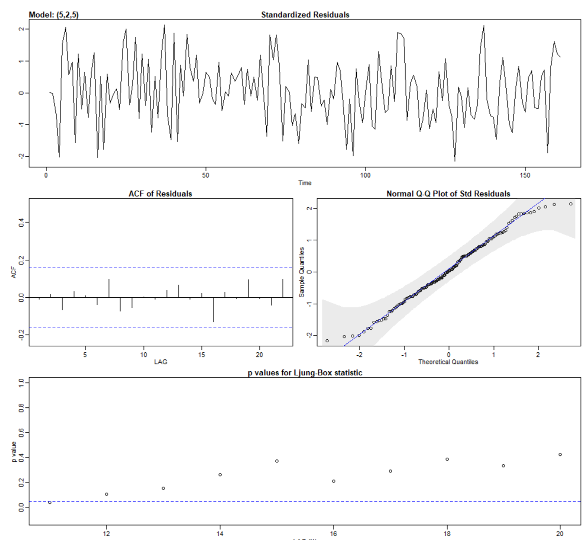


Figure 18: Residual Analysis using SARIMA(5,2,5)

The residual analysis of Figure 18 seems to indicate that the p-values of SARIMA(5,2,5) are mostly $\rangle$ 0.05 and the ACF of its residuals all lies in the boundary.

7. Conclusion

The time series data was first plotted in a time plot to have a overview of the data. It was noticed that the data was non-stationary, hence an one-time differencing was applied to transform it into a stationary time series. The ACF and PACF plots indicates that ARIMA(3,1,0) could be an possible fit for the time series data. A diagnostic check was done on the fitted model and it shows that it is an adequate fit.

The general steps are as follows:

1. Step 1 - Check stationarity: If a time series has a trend or seasonality component, it must be made stationary before we can use ARIMA to forecast.
2. Step 2 - Difference: If the time series is not stationary, it needs to be stationarized through differencing. Take the first difference, then check for stationarity.
3. Step 3 - Select AR and MA terms: Use ACF, PACF plots.
4. Step 4 - Diagnostics: tsdiag, AIC, BIC values to evaluate if fitted models is adequate.

Auto arima function suggested ARIMA(1,1,1) may be a better fit of a lower AIC value than ARIMA(3,1,0), hence a diagnostic check was also performed and it shows that it is also an adequate fit.

Subsequently, two-times differencing was performed on the original data to test models with d=2. The ACF and PACF plots indicates that ARIMA(2,2,0) could be an possible fit for the time series data. A diagnostic check was done on the fitted model and it shows that it is an adequate fit. An iterative method was performed to obtain the model with the lowest AIC value and ARIMA(5,2,5) was suggested. A diagnostic check was done on the fitted model and it shows that it is an adequate fit.

Following fitted models and forecast accuracy tests, it seems that ARIMA(2,2,0) may not be a good model for forecasting. Generally, the lower the AIC values, the better the fitted model is. Even though ARIMA(5,2,5) provides the lowest AIC, usually lesser parameters are preferred, which is possibly the reason why auto.arima suggested ARIMA(1,1,1) over ARIMA(3,1,0) and ARIMA(5,2,5). However, based on residual analysis, the residuals for the ARIMA(5,2,5) model appear to be Gaussian white noise, which may indicate a better fitted model.

Appendix A. ARIMA models R Codes

```
# Load data
y = scan("./data/wwwusage.txt", skip = 1)
min(y) # find min y
max(y) # find max y
mean(y)
plot(1:100, y, xlim=c(0,100),ylim=c(50,250),
     main="Number of users connected to internet over Time",
     xlab="Time (minute)", ylab="Number of users connected")
lines(1:100,y,type="l") # plot line
acf(y, lag.max = 40)  # does not cut off until lag 32 (dies down quickly)
pacf(y) # cut off after lag 2

# Apply one time differencing (z1)
z1=diff(y)
length(z1)
min(z1)
max(z1)
plot(1:99,z1,xlim=c(0,100),ylim=c(-15,15), main="Time Plot after one-time
    differencing") # Plot time plot after differencing
lines(1:99,z1,type="l")
acf(z1, lag.max = 40)  # does not cut off until lag  24 (dies down quickly)
pacf(z1) # cut off after lag 3

# Yule Walker to estimate AR coefficient (z1)
ts.yw <- ar.yw(z1, order.max = 5)
ts.yw
summary(ts.yw) # also suggested 3

# Try arima(3, 1, 0) from yule walker est on differenced data
fit310 = arima(x = y, order=c(3,1,0))
fit310
tsdiag(fit310)
AIC(fit310) # 511.994
BIC(fit310) # 522.3745

# Forecast for arima(3, 1, 0)
plot(1:100, y, xlim=c(0,120), ylim=c(-100,500), main="Fitted and Forecast
    Values for ARIMA(3, 1, 0)",
     xlab="Time (minute)", ylab="Number of users connected")
lines(1:100,y,type="l")
lines(fitted(fit310),col="blue")
forecast310 = predict(fit310, n.ahead=20)
lines(101:120, forecast310$pred, type="o", col="red")
lines(101:120, forecast310$pred-1.96*forecast310$se, col="blue")
lines(101:120, forecast310$pred+1.96*forecast310$se, col="blue")

# Try arima(1, 1, 1) suggested by auto.arima
fitauto <- auto.arima(y,max.p = 5,max.q = 5,max.P = 5,max.Q = 5,max.d = 3,
    seasonal = FALSE,ic = 'aicc')
fitauto
fit111 = arima(x = y, order=c(1,1,1))
fit111
tsdiag(fit111)
```

```
AIC(fit111) # 514.2995
BIC(fit111) # 522.0848 (lower BIC than arima(3,1,0))

plot(1:100, y, xlim=c(0,120), ylim=c(-100,500), main="Fitted and Forecast Values for ARIMA(1, 1, 1)",
     xlab="Time (minute)", ylab="Number of users connected")
lines(1:100,y,type="l")
lines(fitted(fit111),col="blue")
forecast111 = predict(fit111, n.ahead=20)
lines(101:120, forecast111$pred, type="o", col="red")
lines(101:120, forecast111$pred-1.96*forecast111$se, col="blue")
lines(101:120, forecast111$pred+1.96*forecast111$se, col="blue")

# Apply two time differencing
z2=diff(z1)
length(z2)
min(z2)
max(z2)
plot(1:98,z2,xlim=c(0,100),ylim=c(-10,10), main="Time Plot after two-time differencing") # Plot time plot after differencing
lines(1:98,z2,type="l")
acf(z2, lag.max = 30)  # does not cut off until lag  27
pacf(z2) # cut off after lag 2

# Yule Walker to estimate AR coefficient (z2)
ts.yw <- ar.yw(z2, order.max = 5)
ts.yw
summary(ts.yw) # also suggested 2

# Try arima(2, 2, 0) based on Yule Walker est on z2
fit220 = arima(x = y, order=c(2,2,0))
fit220
tsdiag(fit220)
AIC(fit220) # 511.4645 (lower than arima(1,1,1))
BIC(fit220) # 519.2194 (lower than arima(1,1,1))

plot(1:100, y, xlim=c(0,120), ylim=c(-100,500), main="Fitted and Forecast Values for ARIMA(2, 2, 0)",
     xlab="Time (minute)", ylab="Number of users connected")
lines(1:100,y,type="l")
lines(fitted(fit220),col="blue")
forecast220 = predict(fit220, n.ahead=20)
lines(101:120, forecast220$pred, type="o", col="red")
lines(101:120, forecast220$pred-1.96*forecast220$se, col="blue")
lines(101:120, forecast220$pred+1.96*forecast220$se, col="blue")

# Try arima(5, 2, 5) with lowest AIC via brute force testing
fit525 = arima(x = y, order=c(5,2,5))
fit525
tsdiag(fit525)
AIC(fit525) # 509.8135 (lowest AIC)
BIC(fit525) # 538.2481

plot(1:100, y, xlim=c(0,120), ylim=c(-100,500), main="Fitted and Forecast Values for ARIMA(5, 2, 5)",
```

```
     xlab="Time (minute)", ylab="Number of users connected")
lines(1:100,y,type="l")
lines(fitted(fit525),col="blue")
forecast525 = predict(fit525, n.ahead=20)
lines(101:120, forecast525$pred, type="o", col="red")
lines(101:120, forecast525$pred-1.96*forecast525$se, col="blue")
lines(101:120, forecast525$pred+1.96*forecast525$se, col="blue")

plot(forecast(fit310,h=20), ylim=c(-100,500))
plot(forecast(fit111,h=20), ylim=c(-100,500))
plot(forecast(fit220,h=20), ylim=c(-100,500))
plot(forecast(fit525,h=20), ylim=c(-100,500))

acctest <- window(y, start=91, end=100)
accuracy(forecast(fit310), acctest)
accuracy(forecast(fit111), acctest)
accuracy(forecast(fit220), acctest)
accuracy(forecast(fit525), acctest)

library(forecast)
checkresiduals(fit310)
checkresiduals(fit111)
checkresiduals(fit220)
checkresiduals(fit525)

library(sarima)
sfit310 <- sarima(y, p = 3, d = 1, q = 0) #4.676866
sfit310$ttable
sfit111 <- sarima(y, p = 1, d = 1, q = 1) #4.664191
sfit111$ttable
sfit220 <- sarima(y, p = 2, d = 2, q = 0) #4.844786
sfit220$ttable
sfit525 <- sarima(y, p = 5, d = 2, q = 5) #4.628987
sfit525$ttable
```

Listing 1: R Codes to find adequate models for wwwusage data

Appendix B. Best AIC and BIC Model R Codes

```r
# Use brute force to obtain min AIC and BIC models
pArr <- seq(0 , 9, by = 1)
dArr <- seq(0 , 3, by = 1) # at most 3 times differencing
qArr <- seq(0 , 9, by = 1)
corder <- NA
aic <- NA
bic <- NA
i = 1
for (p in pArr) {
  for (d in dArr){
    for (q in qArr) {
      aicc <- NA
      bicc <- NA
      message(sprintf("Trying order: c(%s)", paste(p,d,q, sep=",")))
      tryCatch({
        fit <-arima(x = y, order=c(p,d,q))
        aicc <- AIC(fit)
        bicc <- BIC(fit)
      },
      finally={
        corder[i] <- sprintf("c(%d,%d,%d)", p,d,q)
        aic[i] <- aicc
        bic[i] <- bicc
        i = i + 1
      })
    }
  }
}
aicInd <- which(aic == min(aic, na.rm = TRUE))
bicInd <- which(bic == min(bic, na.rm = TRUE))
message(sprintf("Best AIC: %s Order: %s", aic[aicInd], corder[aicInd]))
message(sprintf("Best BIC: %s Order: %s", bic[bicInd], corder[bicInd]))
```

Listing 2: R Codes to find model of lowest AIC and BIC