Name: Sean Goh Ann Ray
Matric No: G2202190G

The raw data, y, consists of the number of users connected to the internet through a server, and has 100 observations collected at intervals of 1 minute. The data is plotted below in Figure 1, with the ACF and PACF shown in Figures 2 and 3.

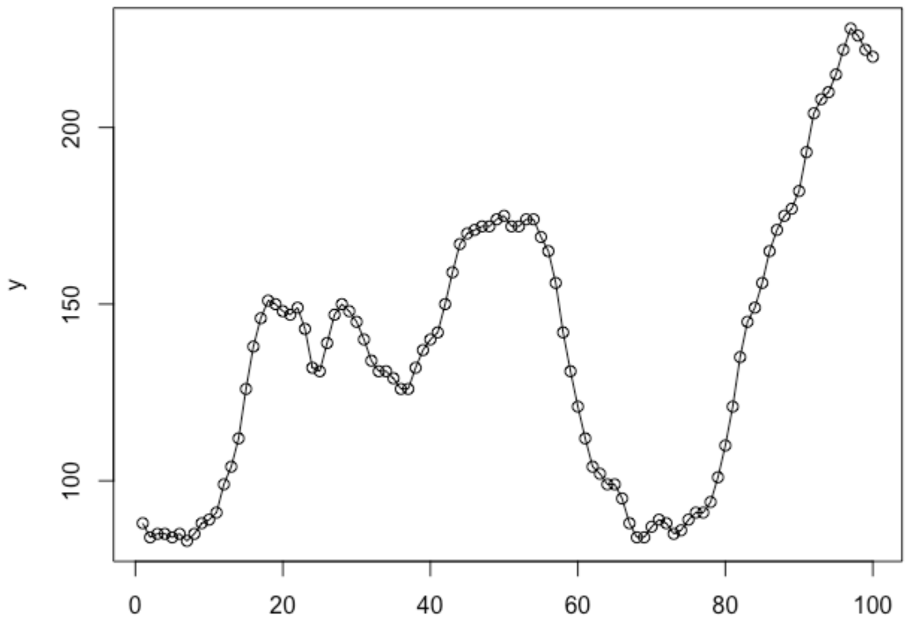

Figure 1. Time-Series of Raw Data y

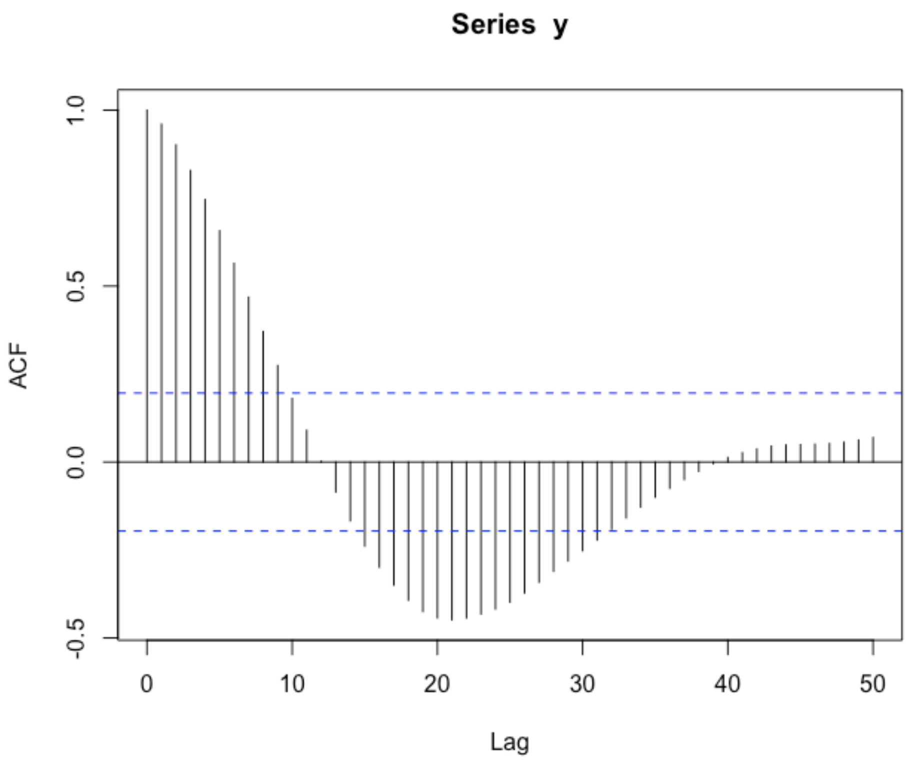

Figure 2. ACF of Raw Data y

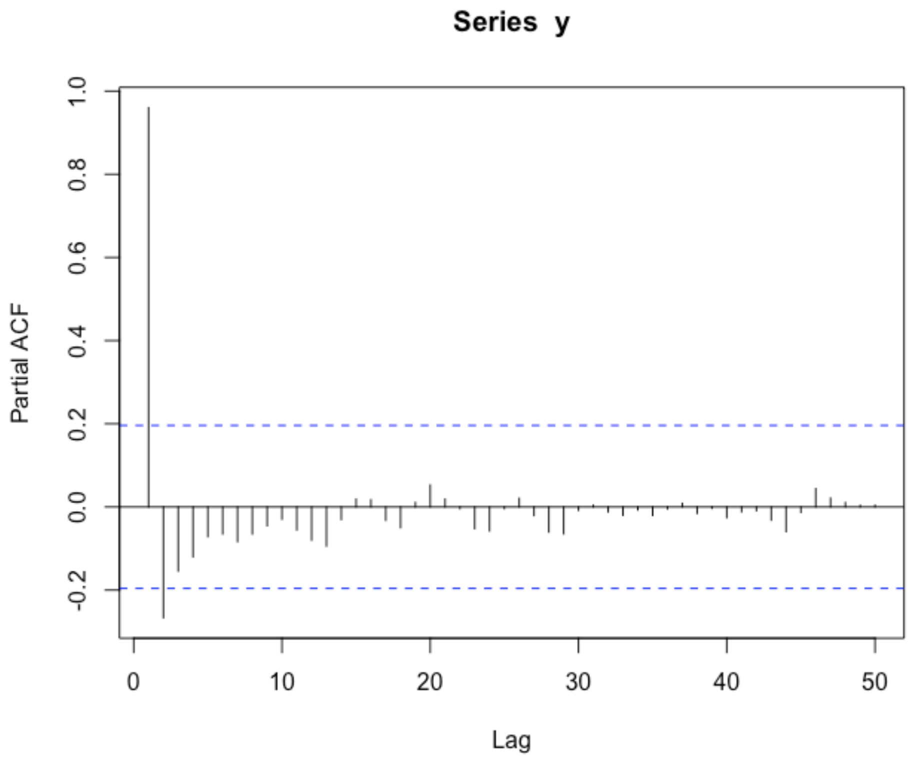

Figure 3. PACF of Raw Data y

As observed in Figure 1, the raw data shows a small upwards trend but no clear pattern in seasonality and variance. Since the mean does not seem to be constant, the raw data can be considered a non-stationary time-series data. This is further supported by the slow convergence of the ACF, which cuts off only at lag 32, as shown in Figure 2.

Therefore, differencing is applied once to the raw data, and is plotted as yd1 in Figure 4, as well as the ACF and PACF in Figures 5 and 6 respectively.

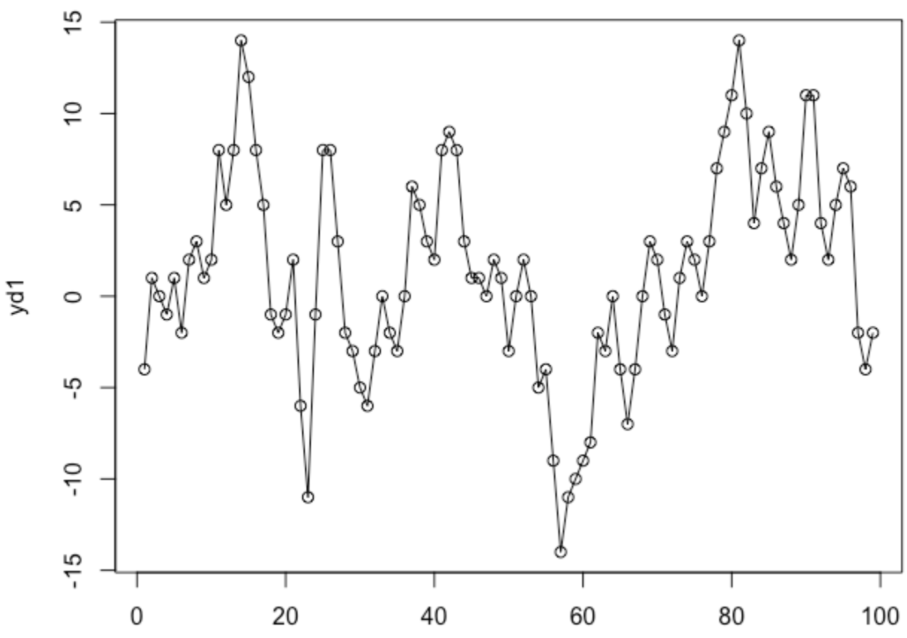

Figure 4. Time-Series of One-time Differenced Data yd1

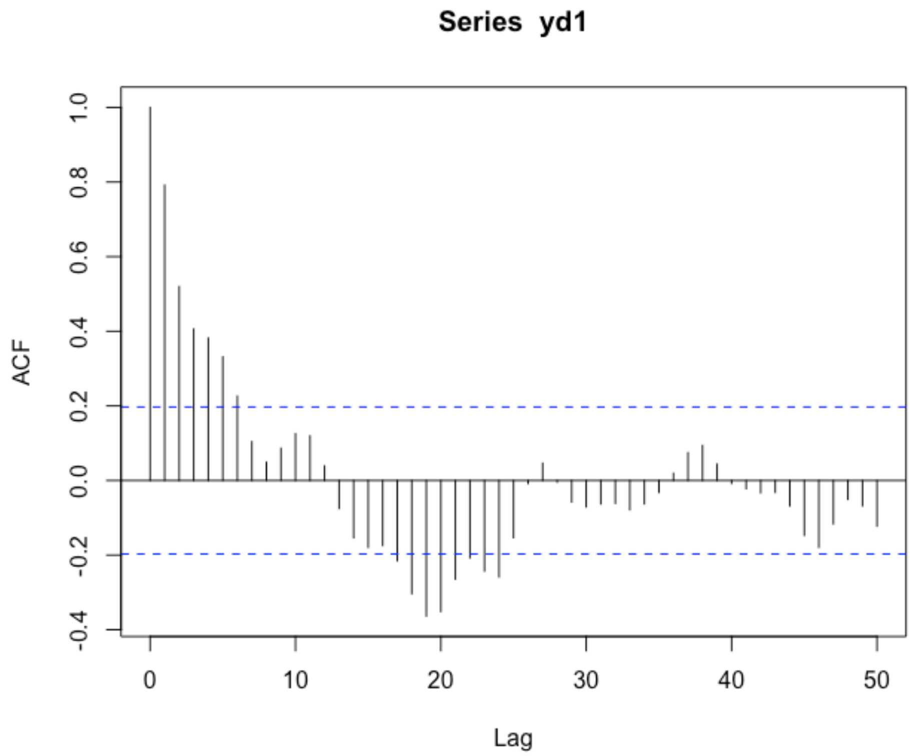

Figure 5. ACF of One-time Differenced Data yd1

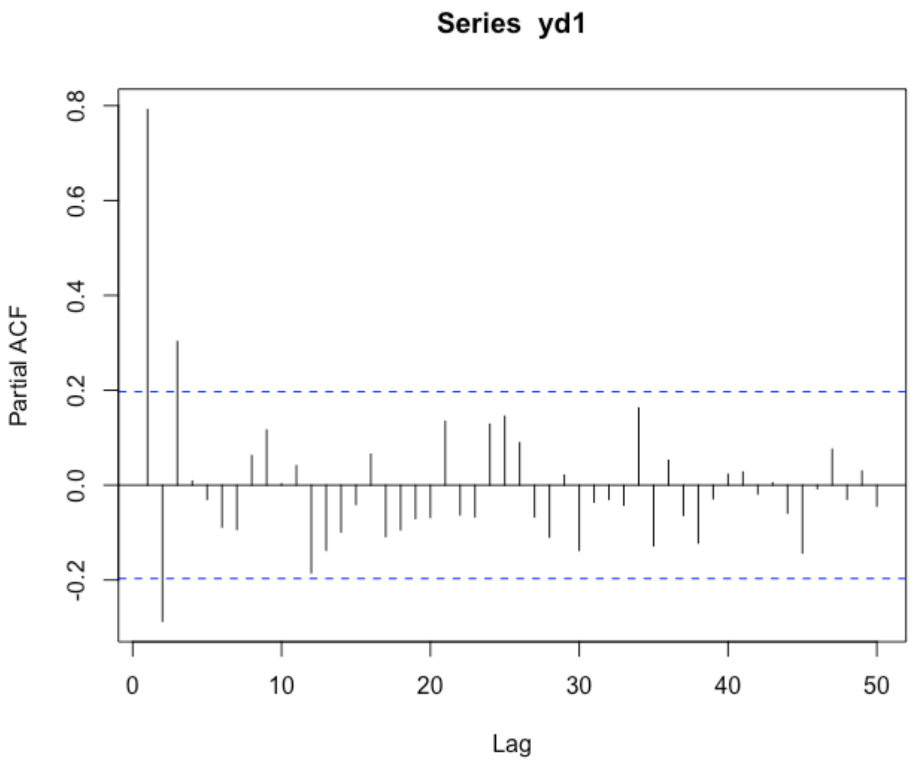

Figure 6. PACF of One-time Differenced Data yd1

As observed in Figure 4, the data can be considered a stationary time-series data as the mean is relatively constant without any patterns in the trend, seasonality, or variance. Based on the ACF plot in Figure 5, the cut-off does seem to occur faster at lag 24, thus ARIMA(0,1,24) can be used to fit the data. Based on the PACF plot in Figure 6, the cut-off occurs at lag 3, thus ARIMA(3,1,0) is proposed. Their fits and diagnostics are shown in Figures 7 to 10.

```
Call:
arima(x = y, order = c(0, 1, 24))

Coefficients:
         ma1     ma2     ma3     ma4     ma5     ma6     ma7     ma8     ma9    ma10    ma11
      1.1338  0.7857  0.5764  0.7222  0.5672  0.3592  0.2494  0.1425  0.2110  0.2715  0.5149
s.e.  0.1310  0.1941  0.2067  0.2201  0.2197  0.2133  0.2476  0.2843  0.3159  0.3507  0.3505
        ma12    ma13    ma14    ma15     ma16     ma17     ma18     ma19     ma20     ma21    ma22
      0.6829  0.4156  0.2350  -0.198  -0.3015  -0.1165  -0.3565  -0.5369  -0.3688  -0.0833  0.2816
s.e.  0.3216  0.3086  0.3002   0.308   0.3098   0.2993   0.2903   0.2967   0.2693   0.2390  0.2082
        ma23    ma24
      0.5568  0.2661
s.e.  0.1901  0.1334

sigma^2 estimated as 5.688:  log likelihood = -238.89,  aic = 527.78
```

Figure 7. Diagnostics of ARIMA(0,1,24) Fitted Model

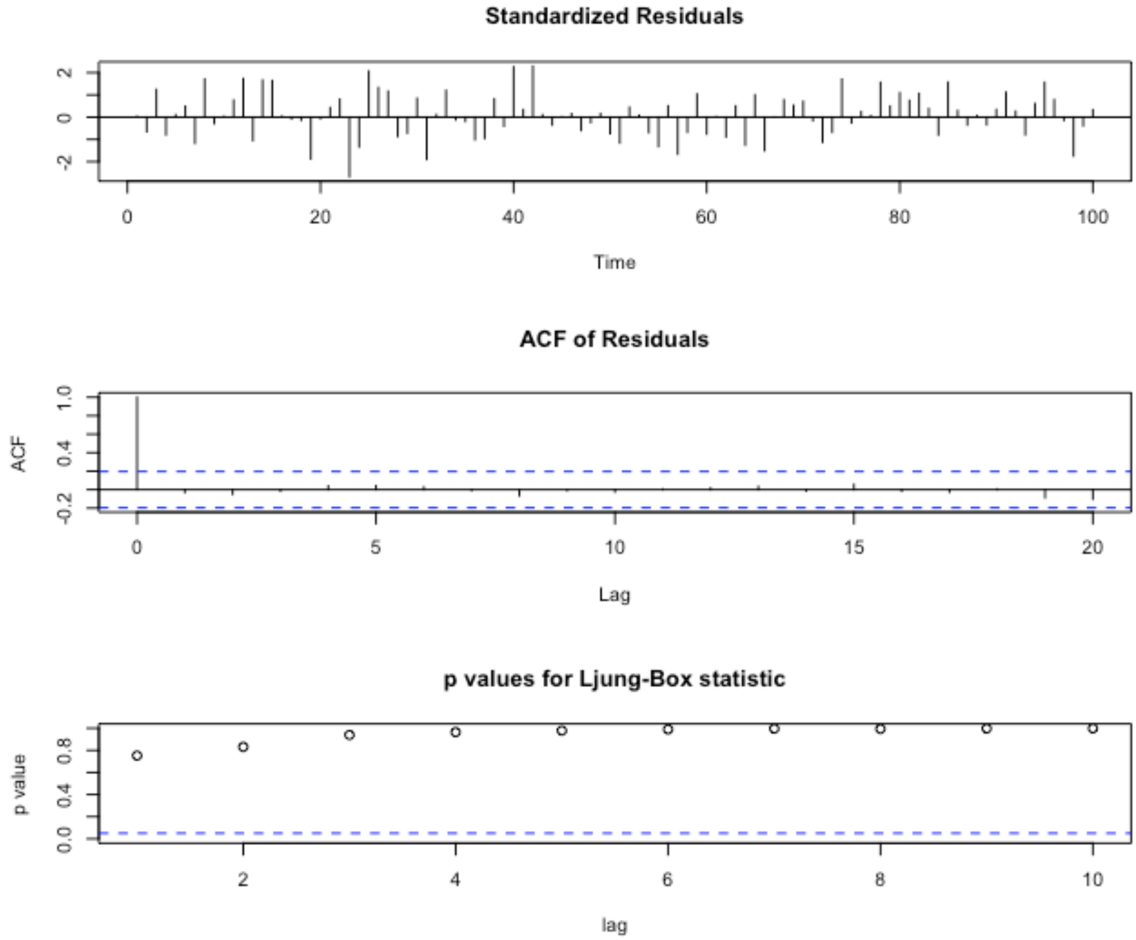

Figure 8. Plotted Diagnostics of ARIMA(0,1,24) Fitted Model

```
Call:
arima(x = y, order = c(3, 1, 0))

Coefficients:
         ar1      ar2     ar3
      1.1513  -0.6612  0.3407
s.e.  0.0950   0.1353  0.0941

sigma^2 estimated as 9.363:  log likelihood = -252,  aic = 511.99
```

Figure 9. Diagnostics of ARIMA(3,1,0) Fitted Model

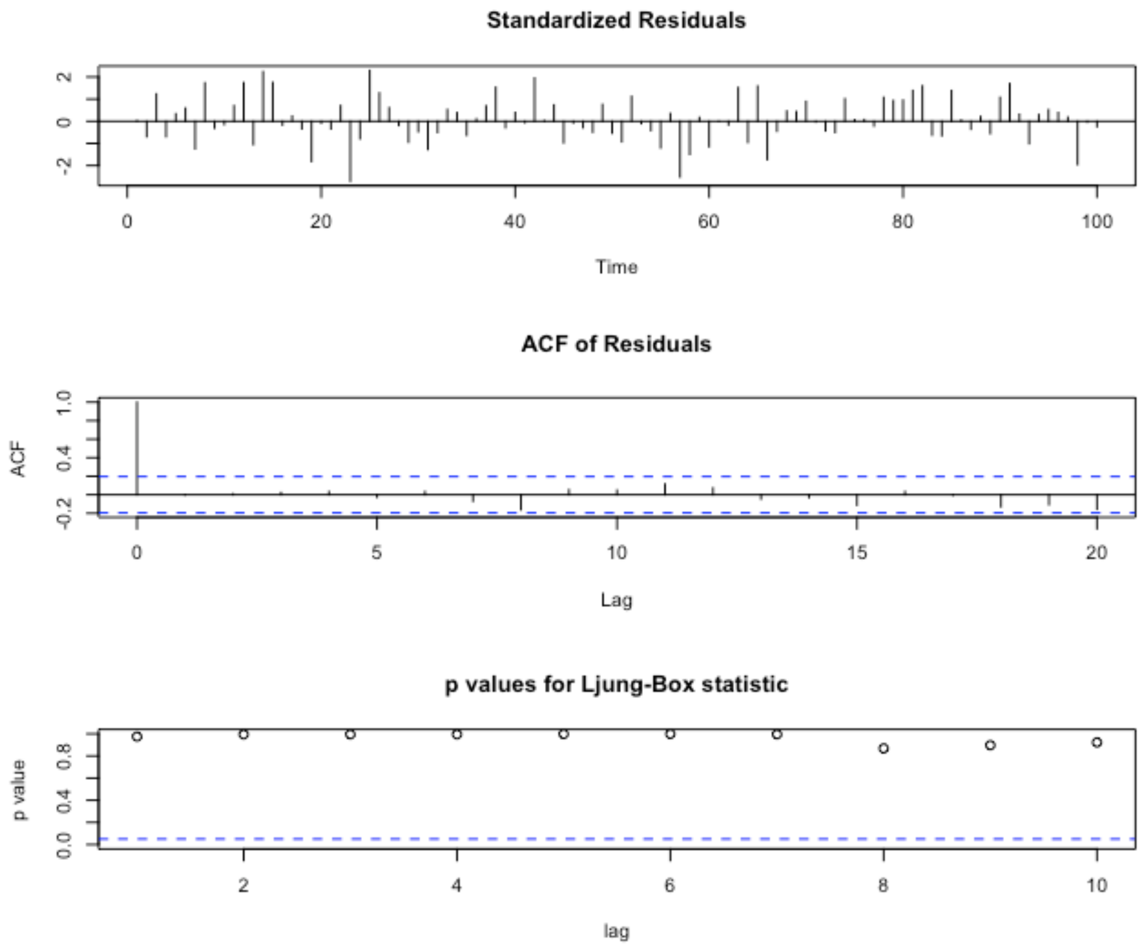

Figure 10. Plotted Diagnostics of ARIMA(3,1,0) Fitted Model

From Figures 8 and 10, both fitted models are adequate in a sense that their residuals seem to be random and thus can be considered white noise, the ACF of the residuals cut-off at lag 0, and the p values for the Ljung-Box statistic are above 0.05. However, the ARIMA(3,1,0) fitted model has a better AIC score of 511.99 compared to the ARIMA(0,1,24) fitted model's AIC score of 527.78, which is likely due to the penalty from the large number of parameters.

The auto.arima function was also utilized here to find the best combination of the p and q parameters of the ARIMA model based on the AIC, which turns out to be ARIMA(1,1,1), with the diagnostics shown in Figures 11 and 12.

```
Series: y
ARIMA(1,1,1)

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

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

Figure 11. Diagnostics of ARIMA(1,1,1) Fitted Model

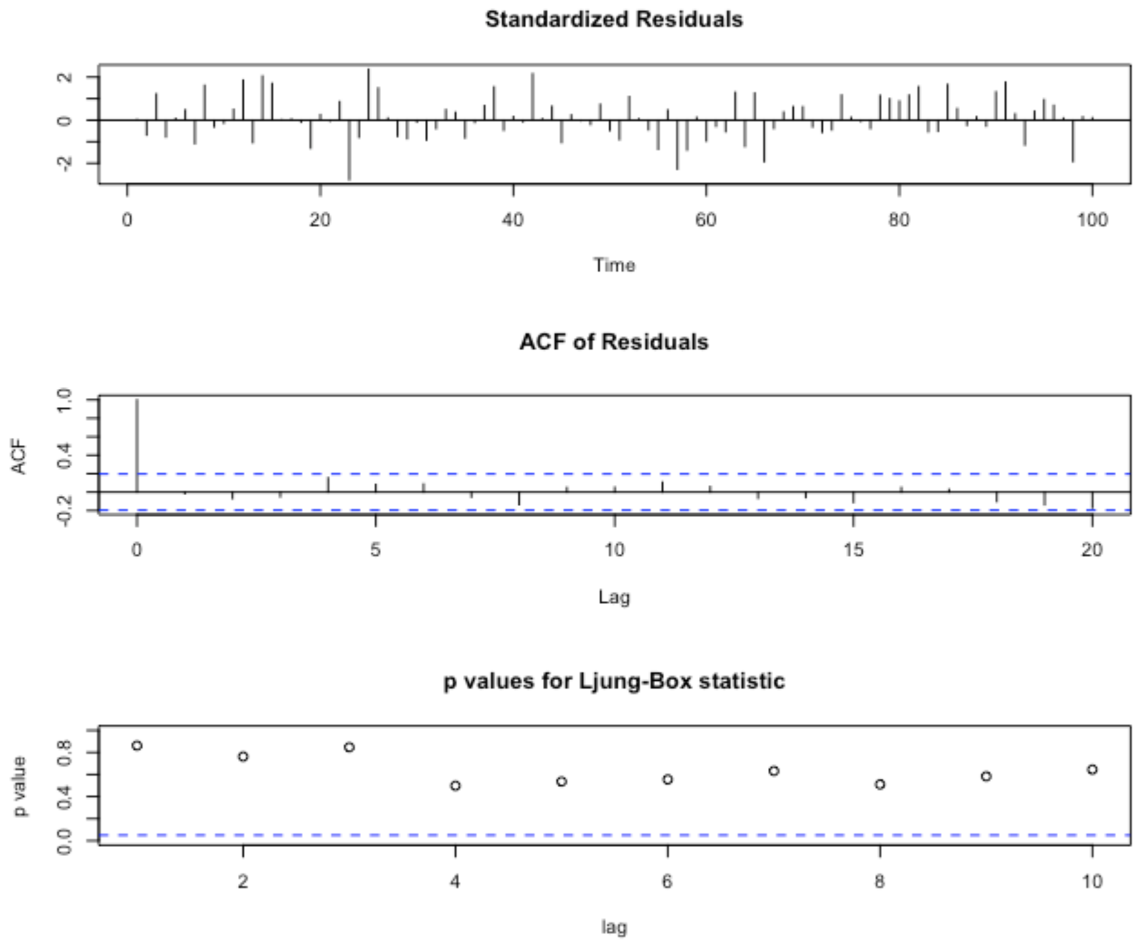

Figure 12. Plotted Diagnostics of ARIMA(1,1,1) Fitted Model

As seen in Figure 12, the ARIMA(1,1,1) fitted model is also adequate as the residuals seem to be random and thus can be considered white noise, the ACF of the residuals cut-off at lag 0, and the p values for the Ljung-Box statistic are above 0.05. However, it still has a higher AIC score of 514.3, higher than the AIC score of the ARIMA(3,1,0) fitted model of 511.99. Therefore, with one-time differencing, it can be said that the ARIMA(3,1,0) fitted model performs the best.

Although the time-series of one-time differencing appears to be stationary, the ACF still has a slow convergence and cuts off only at lag 24. Since one way to interpret a stationary time-series is a fast convergence and cut-off, it can be said that the one-time differenced data is still not a stationary time-series. Therefore, differencing is applied once more to get a two-time differenced time-series data, yd2. The time-series plot is shown in Figure 13, along with the ACF in Figure 14 and PCF in Figure 15.

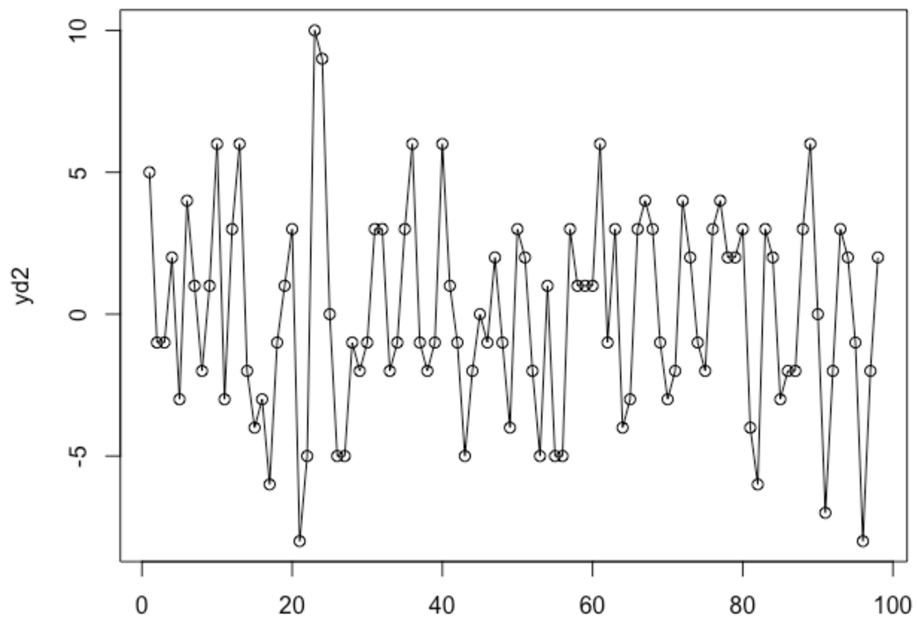

Figure 13. Time-Series of Two-time Differenced Data yd2

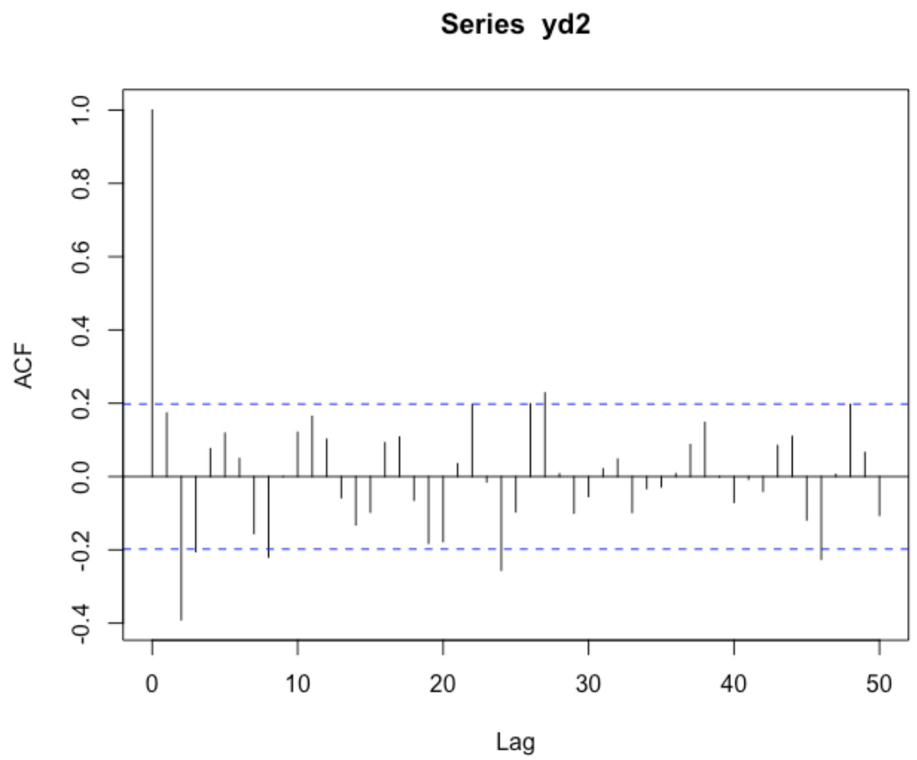

Figure 14. ACF of Two-time Differenced Data yd2

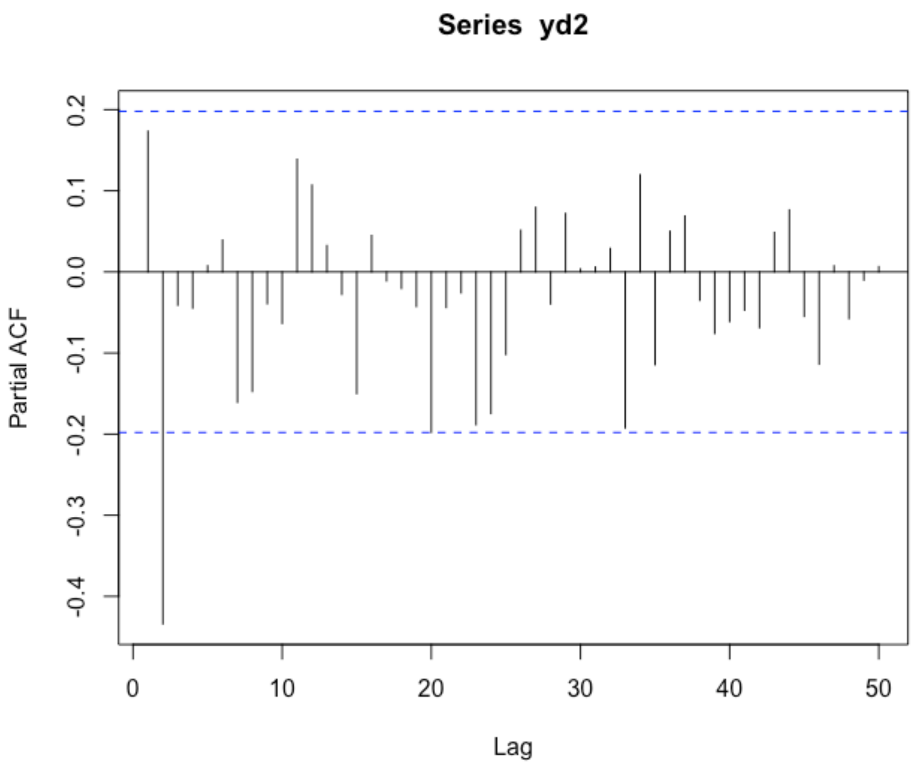

Figure 15. PACF of Two-time Differenced Data yd2

As observed in Figure 13, the data can be considered a stationary time-series data as the mean is relatively constant without any patterns in the trend, seasonality, or variance. Based on the ACF plot in Figure 14, the convergence occurs much more quickly and seems to cut off at lag 3, with the majority of lags after that are below the cut-off. Therefore, the ARIMA(0,2,3) model can be used to fit the data. Based on the PACF plot in Figure 15, the cut-off occurs at lag 2, thus ARIMA(2,2,0) is proposed. Their fits and diagnostics are shown in Figures 16 to 19.

```
Call:
arima(x = y, order = c(0, 2, 3))

Coefficients:
         ma1      ma2      ma3
      0.2588  -0.3949  -0.3182
s.e.  0.1048   0.0968   0.1232

sigma^2 estimated as 9.994:  log likelihood = -252.17,  aic = 512.33
```
Figure 16. Diagnostics of ARIMA(0,2,3) Fitted Model

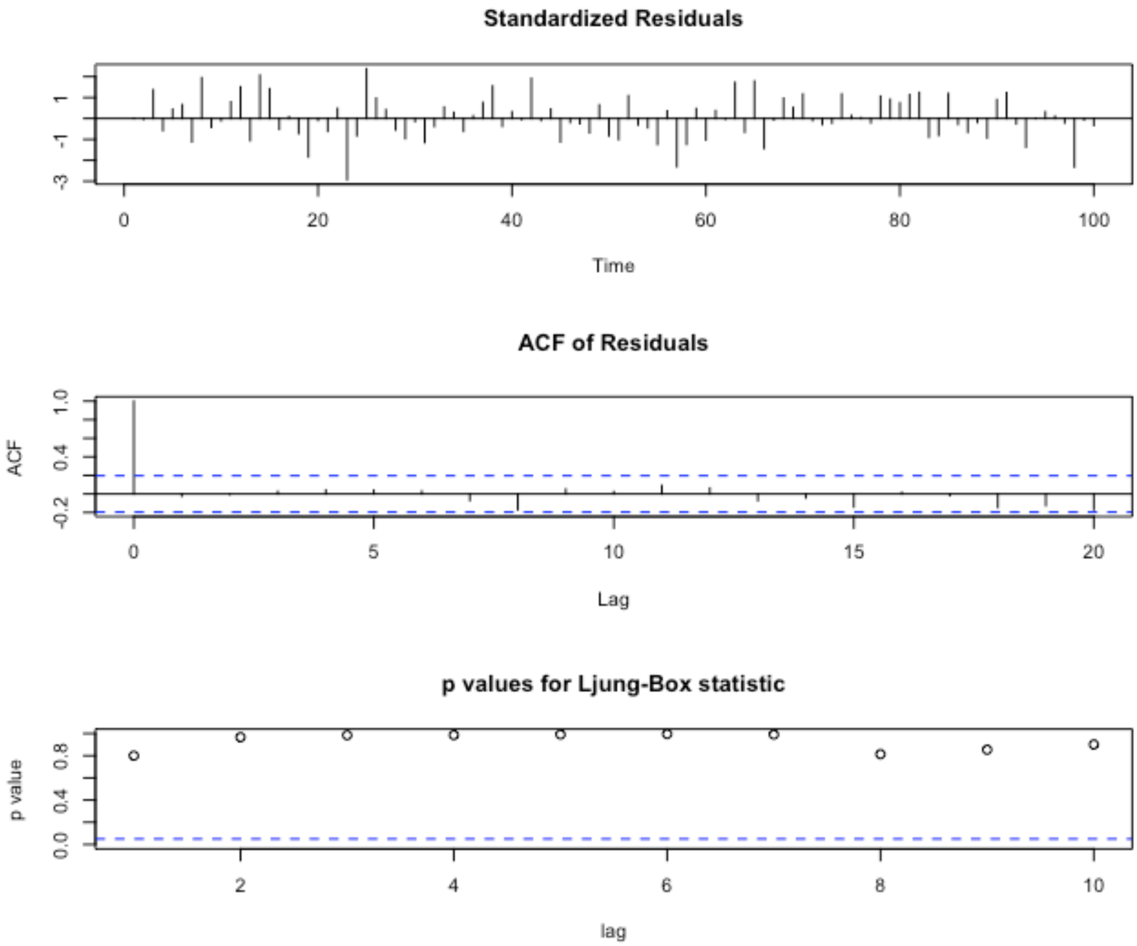

Figure 17. Plotted Diagnostics of ARIMA(0,2,3) Fitted Model

```
Call:
arima(x = y, order = c(2, 2, 0))

Coefficients:
         ar1      ar2
      0.2579  -0.4407
s.e.  0.0915   0.0906

sigma^2 estimated as 10.13:  log likelihood = -252.73,  aic = 511.46
```

Figure 18. Diagnostics of ARIMA(2,2,0) Fitted Model

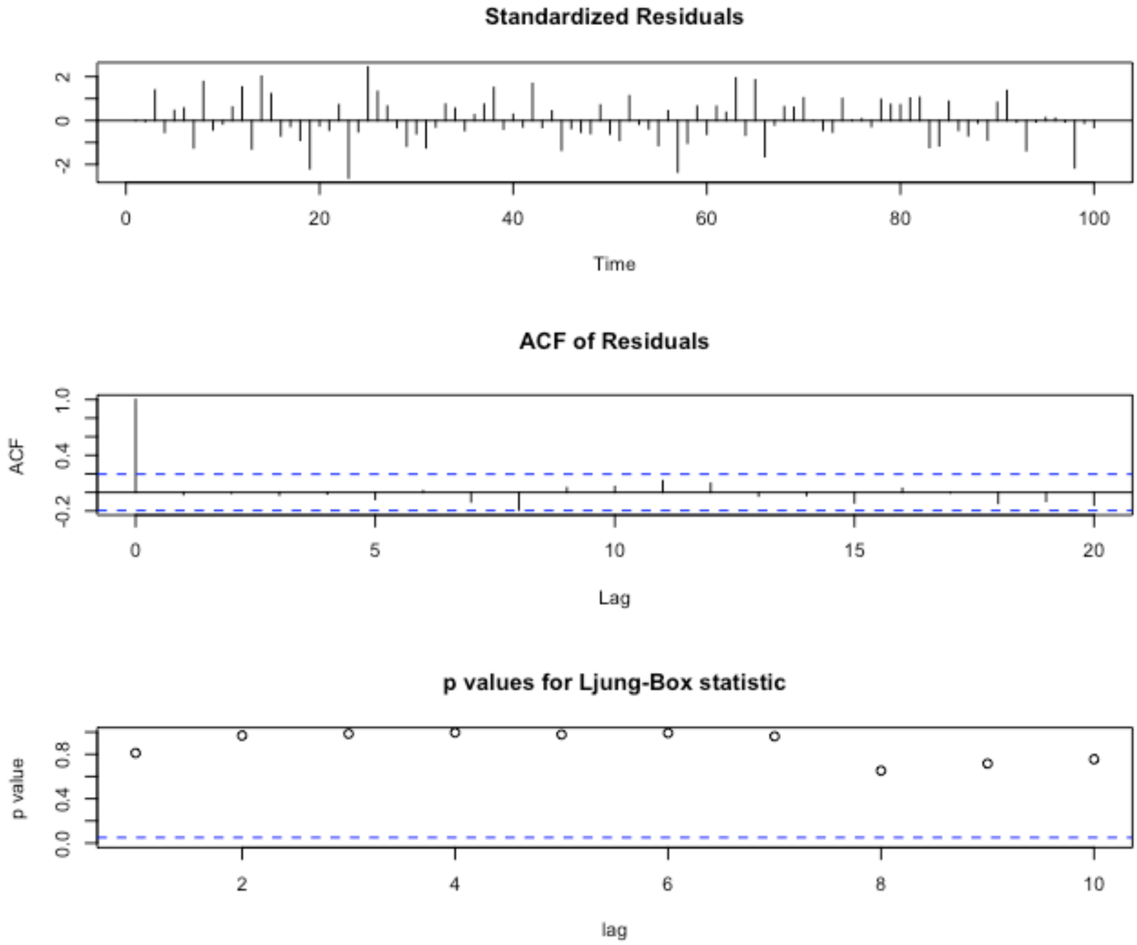

Figure 19. Plotted Diagnostics of ARIMA(2,2,0) Fitted Model

From Figures 17 and 19, both fitted models are adequate in a sense that their residuals seem to be random and thus can be considered white noise, the ACF of the residuals cut-off at lag 0, and the p values for the Ljung-Box statistic are above 0.05. However, the ARIMA(2,2,0) fitted model has a slightly better AIC score of 511.46 compared to the ARIMA(0,2,3) fitted model's score of 512.33, which is likely due to the penalty from an additional parameter.

The auto.arima function was also utilized again to find the best combination of the p and q parameters of the ARIMA model after performing two-time differencing, which turns out to be ARIMA(2,2,0), the same as the one proposed when looking at the PACF plot in Figure 15.

The time plots of the 5 ARIMA fitted models are shown in Figure 20, whereby the black circles represent the original data and the colored lines represent the fitted ARIMA model's prediction.

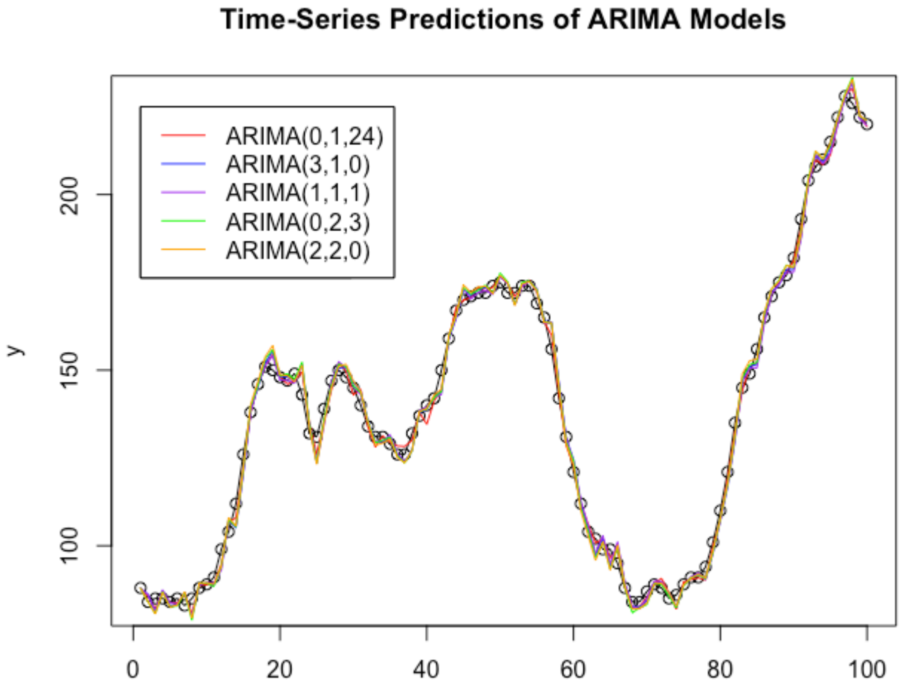

Figure 20. Time-Series Predictions of ARIMA Models

Looking at Figure 20, it is difficult to see which model performs best, but based on the AIC score alone, the best model appears to be the ARIMA(2,2,0).

R Code

```
y = read.delim("Project_1_Data.txt", header = TRUE)
y = y$x
x = length(y)
plot(1:x,y)
lines(1:x,y, type="l")
acf(y, lag.max=50)
pacf(y, lag.max=50)

# Differencing once, plot time series, acf and pacf
yd1 = diff(y)
plot(1:(x-1),yd1)
lines(1:(x-1),yd1, type="l")
acf(yd1, lag.max=50)
pacf(yd1, lag.max=50)

# Fit MA model
fit1ma = arima(y, order=c(0,1,24))
tsdiag(fit1ma)
fit1ma

# Fit AR model
fit1ar = arima(y, order=c(3,1,0))
tsdiag(fit1ar)
fit1ar

# Use auto.ARIMA
fit1auto = auto.arima(y, d=1, max.p=30, max.q=30, ic="aic")
tsdiag(fit1auto) # Produces ARIMA(1,1,1)
fit1auto

# Differencing twice, plot time series, acf and pacf
yd2 = diff(yd1)
plot(1:(x-2),yd2)
lines(1:(x-2),yd2, type="l")
acf(yd2, lag.max=50)
pacf(yd2, lag.max=50)

# Fit MA model
fit2ma = arima(y, order=c(0,2,3))
tsdiag(fit2ma)
fit2ma

# Fit AR model
fit2ar = arima(y, order=c(2,2,0))
tsdiag(fit2ar)
```

```
fit2ar

# Use auto.ARIMA
fit2auto = auto.arima(y, d=2, max.p=30, max.q=30, ic="aic")
tsdiag(fit2auto) # Produces ARIMA(2,2,0)
fit2auto

# Plot Time-Series Predictions of ARIMA Models
plot(1:x, y, main="Time-Series Predictions of ARIMA Models")
lines(1:x, y, type="l")
lines(1:x, y-fit1ma$residuals, type="l", col="red")
lines(1:x, y-fit1ar$residuals, type="l", col="blue")
lines(1:x, y-fit1auto$residuals, type="l", col="purple")
lines(1:x, y-fit2ma$residuals, type="l", col="green")
lines(1:x, y-fit2ar$residuals, type="l", col="orange")
legend(1, 225, legend=c("ARIMA(0,1,24)", "ARIMA(3,1,0)", "ARIMA(1,1,1)", "ARIMA(0,2,3)",
"ARIMA(2,2,0)"), col=c("red", "blue", "purple", "green", "orange"), lty=1:1)
```