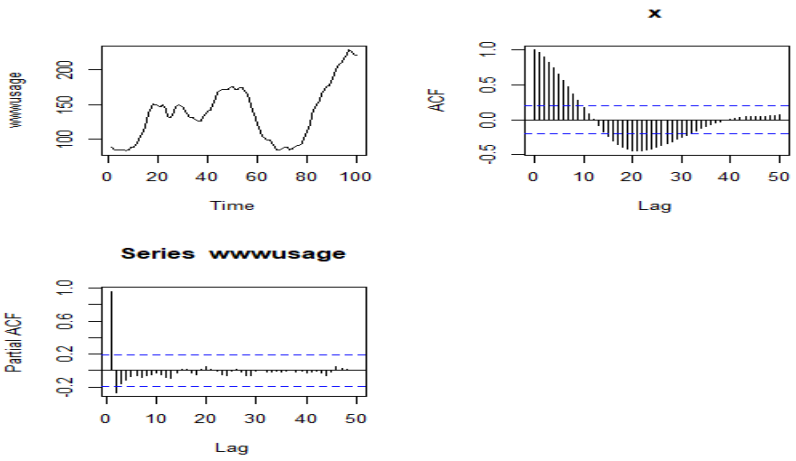
Data Set 1 (wwwusage)

```
library(forecast)
library(tseries)
wwwusage = read.csv("C:/Users/karsh/Desktop/Uni/Math Mods/MH4500 Time Series Analysis/Lab/Lab 3/wwwusage.txt", sep="")
```

1. Time series, SACF and SPACF Plots

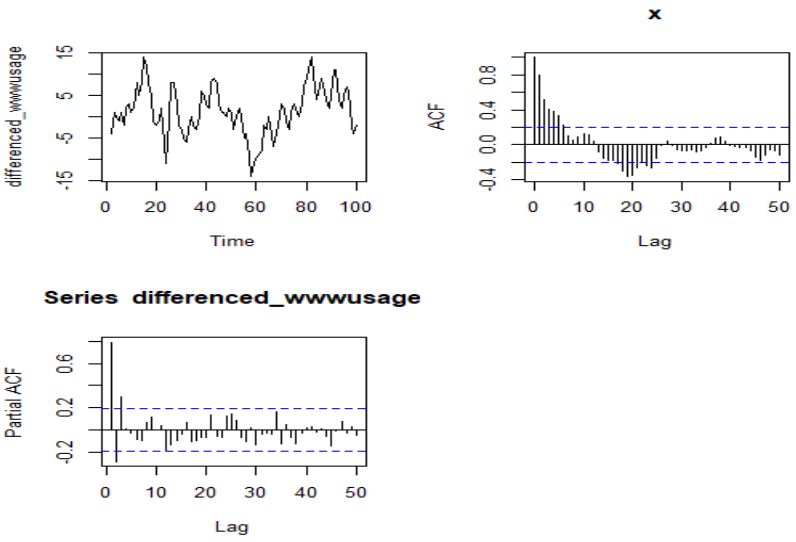


From the time series plot, the data appears to be non-stationary as the mean of the time series plot appears to be dependent on time. In addition, the SACF appears to die down very slowly. Hence, we need to apply the differencing operator. To double check, we use the Augmented Dickey -Fuller Test.

```
> adf.test(ts(wwwusage)) #check for stationarity using augmented dickey-fulle
r test

        Augmented Dickey-Fuller Test

data:  ts(wwwusage)
Dickey-Fuller = -2.6421, Lag order = 4, p-value = 0.3107
alternative hypothesis: stationary
```

From the Augmented Dickey-Fuller Test, the null hypothesis is – not stationary. The p-value = 0.3107 > 0.05, we fail to reject the null hypothesis and claim that the data is not stationary. Hence, we will need to difference the dataset using diff() function in R.

2.1 Differenced Time series, SACF and SPACF Plots

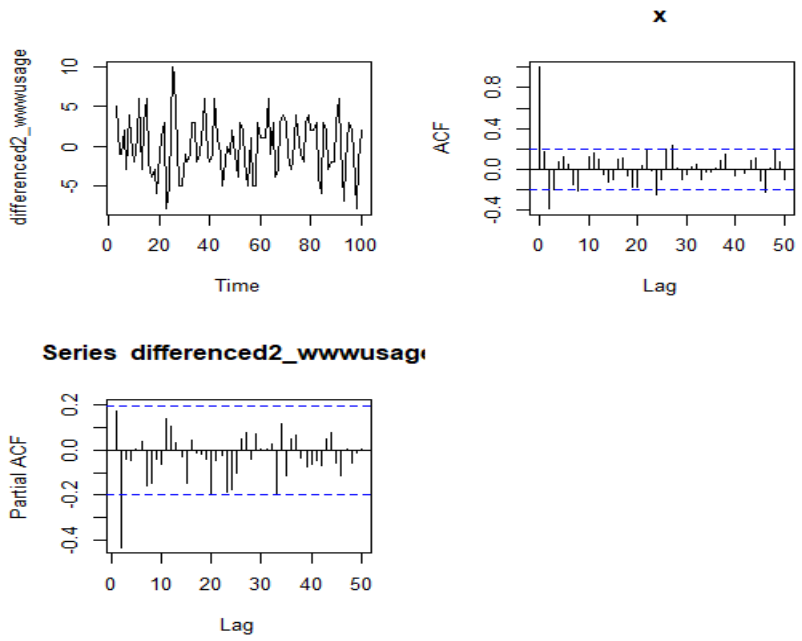


The mean of the time series plot appears to be dependent on time, hence it is not stationary. As a result, we apply the differencing operator again.

```
> adf.test(differenced_wwwusage)

        Augmented Dickey-Fuller Test

data:  differenced_wwwusage
Dickey-Fuller = -2.5459, Lag order = 4, p-value = 0.3506
alternative hypothesis: stationary
```

In addition, from the Augmented Dickey-Fuller Test. The null hypothesis is – not stationary. The p-value = 0.3506 > 0.05, we fail to reject the null hypothesis and claim that the data is not stationary. Hence, we will need to difference the dataset using diff() function in R again.

2.2 Differenced 2x Time series, SACF and SPACF Plots



The time series plot appears to be random without any seasonal component or trend.

```
> adf.test((differenced2_wwwusage))

        Augmented Dickey-Fuller Test

data:  (differenced2_wwwusage)
Dickey-Fuller = -4.828, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test((differenced2_wwwusage)) : p-value smaller than printed p-value
```

In addition, from the Augmented Dickey-Fuller test. The null hypothesis is – not stationary. The p-value = 0.01 < 0.05, we reject the null hypothesis and claim that the data is stationary. The SACF cuts off at lag 2 and the SPACF cuts off at lag 2. Therefore, we can suggest the MA(2) and AR(2) model respectively. We need to check both the ARIMA(0,2,2) and ARIMA(2,2,0) models.

3.1 ARIMA(0,2,2) – wwwusage_fit1

```
> wwwusage_fit1 = arima(wwwusage, order=c(0,2,2))
> wwwusage_fit1

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

Coefficients:
          ma1      ma2
      0.1318  -0.3590
s.e.  0.1075   0.1049

sigma^2 estimated as 10.75:  log likelihood = -255.61,  aic = 517.21
```

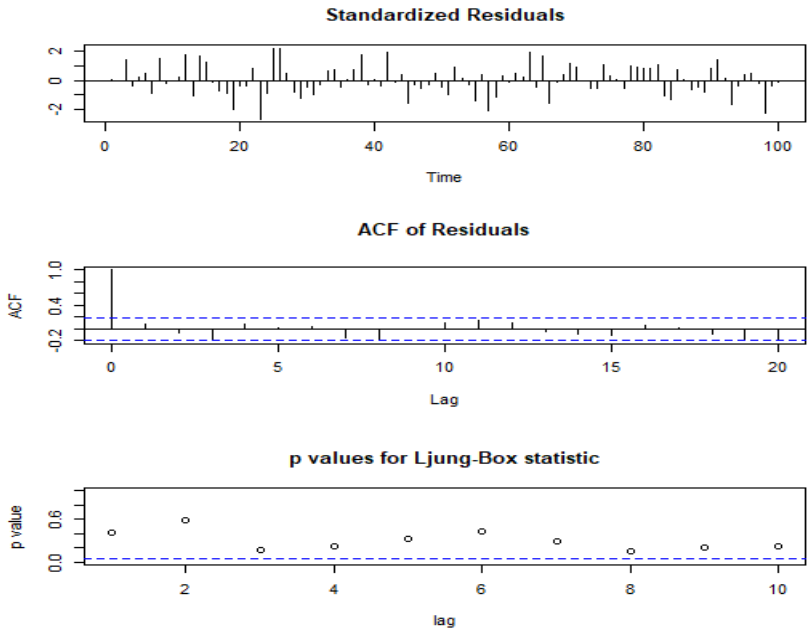
3.2 ARIMA(2,2,0) – wwwusage_fit2

```
> wwwusage_fit2 = arima(wwwusage, order=c(2,2,0))
> wwwusage_fit2

Call:
arima(x = wwwusage, 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
```

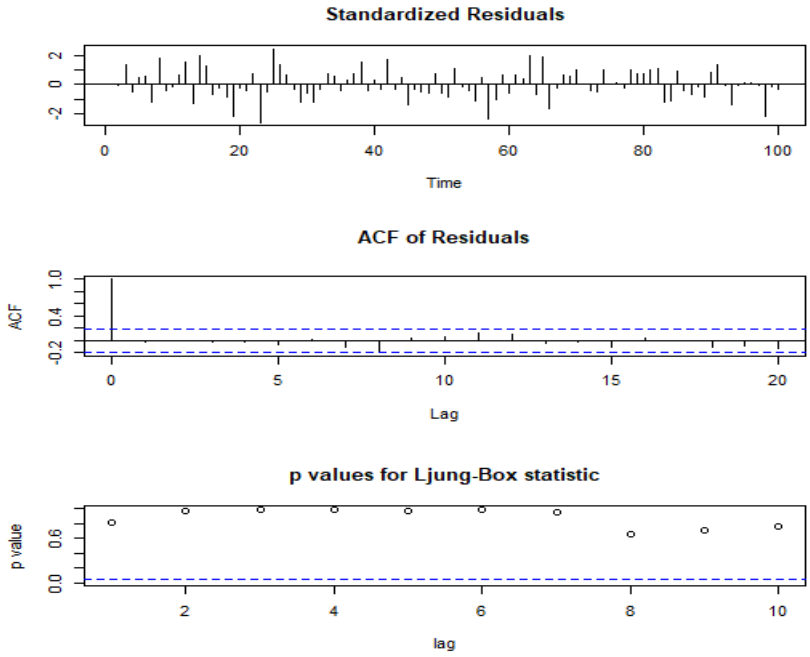
4. Diagnostic Checking

We now do the diagnostic checking to check that the model is adequate.

ARIMA(0,2,2)



ARIMA(2,2,0)



The p-values for both plots are all above 0.05, hence both models are adequate. We will now compare the AIC to determine which model is a better fit. AIC of ARIMA(0,2,2) = 517.21 and AIC of ARIMA(2,2,0) = 511.46. Therefore, we choose the ARIMA(2,2,0) model.

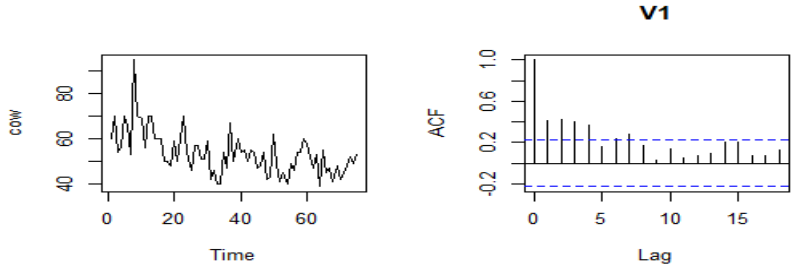
```
> auto.arima(wwwusage)
Series: wwwusage
ARIMA(1,1,1)

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
```

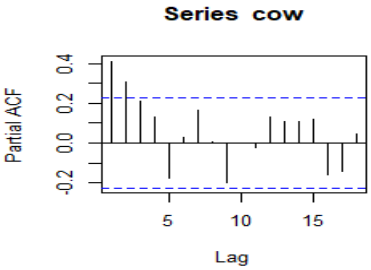
When cross-checking with the auto.arima() function in R, it returns a different result. However, as mentioned in the lecture, the auto.arima() function does not always return the best fit model. In addition, looking at the AIC value of ARIMA(1,1,1) = 514.3 and ARIMA(2,2,0) = 511.46, the ARIMA(2,2,0) model has a lower value and is thus more appropriate for the data.

Data Set 2 (cow)

cow = read.table("C:/Users/karsh/Desktop/Uni/Math Mods/MH4500 Time Series Analysis/Lab/Lab 3/cow.dat", quote="\"", comment.char="")

1. Time series, SACF and SPACF Plots

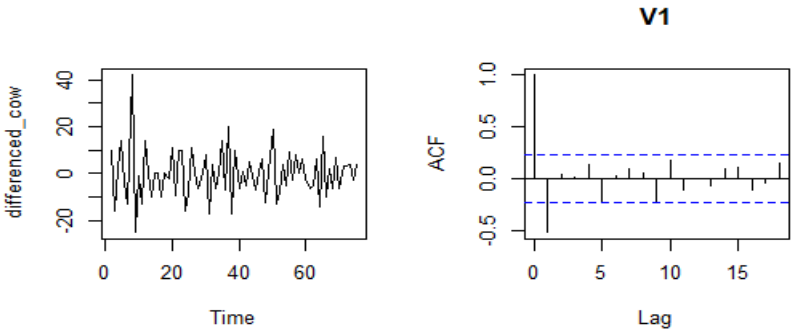


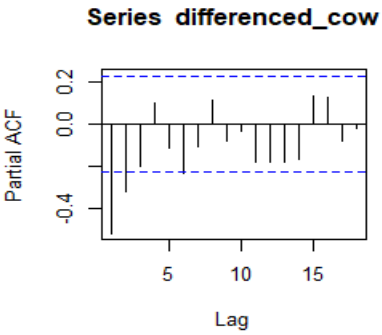



```
> adf.test(ts(cow))

        Augmented Dickey-Fuller Test

data:  ts(cow)
Dickey-Fuller = -3.4838, Lag order = 4, p-value = 0.04933
alternative hypothesis: stationary
```

The mean of the time series appears to be dependent on time, hence it is not stationary. Therefore, we apply the differencing operator onto the dataset.

2. Differenced Time series, SACF and SPACF Plots





The time series now appears to be random without any trend or seasonal components.

```
> adf.test(differenced_cow)

        Augmented Dickey-Fuller Test

data:  differenced_cow
Dickey-Fuller = -4.7324, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test(differenced_cow) : p-value smaller than printed p-value
```

In addition, from the Augmented Dickey-Fuller Test. The null hypothesis is – not stationary. The p-value = 0.01 < 0.05, we reject the null hypothesis and claim that the data is stationary. The ACF cuts off at lag 2 and SPACF cuts off at lag 1. Therefore, we can suggest the MA(1) and AR(2) model respectively. We need to check both the ARIMA(0,1,1) and ARIMA(2,1,0) models.

3.1 ARIMA(0,1,1) - cow_fit1

```
> cow_fit1 = arima(cow, order=c(0,1,1))
> cow_fit1

Call:
arima(x = cow, order = c(0, 1, 1))

Coefficients:
          ma1
      -0.7773
s.e.   0.0958

sigma^2 estimated as 66.85:  log likelihood = -260.96,  aic = 525.92
```

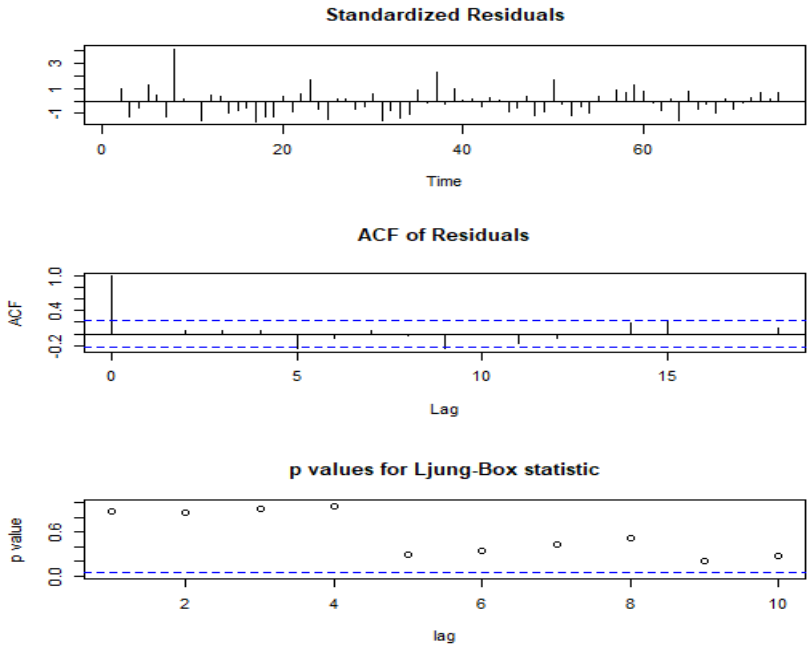
3.1 ARIMA(2,1,0) - cow_fit2

```
> cow_fit2 = arima(cow, order=c(2,1,0))
> cow_fit2

Call:
arima(x = cow, order = c(2, 1, 0))

Coefficients:
          ar1      ar2
      -0.6837  -0.3142
s.e.   0.1096   0.1104

sigma^2 estimated as 70.4:  log likelihood = -262.67,  aic = 531.34
```

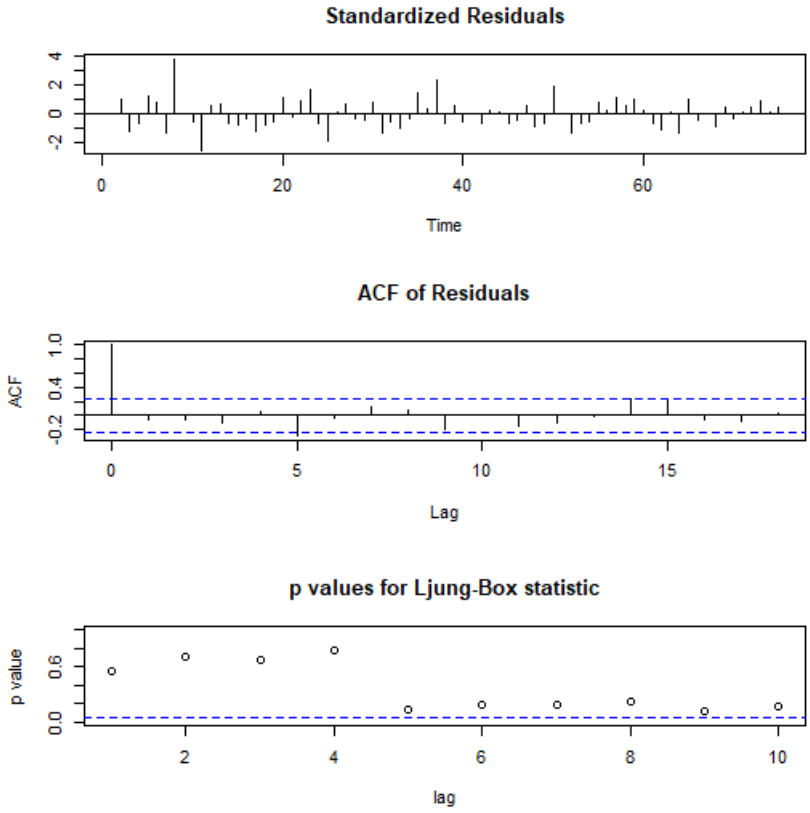
4. Diagnostic Checking

We now do the diagnostic checking to check that the model is adequate.

ARIMA(0,1,1)



ARIMA(2,1,0)



The p-values for both models are larger than 0.05 based on the plots above, hence both models are adequate. As such, we will compare the AIC of both models to determine which is a better fit. The

AIC of ARIMA(0,1,1) = 525.92 while the AIC of ARIMA(2,1,0) = 531.34. Therefore, we choose the ARIMA(0,1,1) model.
To cross check, we use the auto.arima() function in R,

```
> auto.arima(cow)
Series: cow
ARIMA(0,1,1)

Coefficients:
          ma1
      -0.7773
s.e.   0.0958

sigma^2 estimated as 67.77:  log likelihood=-260.96
AIC=525.92   AICc=526.08   BIC=530.52
```

It returns the same result as what we found above, where ARIMA(0,1,1) model is the best to fit the cow dataset.

R Code:

```
#dataset1
library(forecast)
library(tseries)
wwwusage=read.csv("C:/Users/karsh/Desktop/Uni/Math Mods/MH4500 Time Series Analysis/Lab/Lab 3/wwwusage.txt", sep="")
ts.plot(wwwusage)
acf(wwwusage,lag.max=50)
pacf(wwwusage, lag.max=50)
adf.test(ts(wwwusage)) #check for stationarity using augmented dickey-fuller test
differenced_wwwusage=diff(ts(wwwusage))
ts.plot(differenced_wwwusage)
acf(differenced_wwwusage, lag.max=50)
pacf(differenced_wwwusage,lag.max=50) #not stationary, difference again
adf.test(differenced_wwwusage)
differenced2_wwwusage=diff(ts(wwwusage),differences = 2)
ts.plot(differenced2_wwwusage)
acf(differenced2_wwwusage, lag.max=50)
pacf(differenced2_wwwusage, lag.max=50)
adf.test((differenced2_wwwusage))
wwwusage_fit1 = arima(wwwusage, order=c(0,2,2))
wwwusage_fit1
wwwusage_fit2 = arima(wwwusage, order=c(2,2,0))
wwwusage_fit2
tsdiag(wwwusage_fit1)
tsdiag(wwwusage_fit2)
auto.arima(wwwusage)
#dataset2
cow = read.table("C:/Users/karsh/Desktop/Uni/Math Mods/MH4500 Time Series Analysis/Lab/Lab 3/cow.dat", quote="\"", comment.char="")

ts.plot(cow)
acf(cow)
```

```
pacf(cow)
adf.test(ts(cow))
differenced_cow=diff(ts(cow))
ts.plot(differenced_cow)
acf(differenced_cow)
pacf(differenced_cow)
adf.test(differenced_cow)
cow_fit1 = arima(cow, order=c(0,1,1))
cow_fit1
cow_fit2 = arima(cow, order=c(2,1,0))
cow_fit2
tsdiag(cow_fit1)
tsdiag(cow_fit2)
auto.arima(cow)
```