Assignment III

(Applied Econometrics – Time Series Data)

Data – Air Quality Index of Delhi (data.gov.in)
Monthly data from 2003-2014

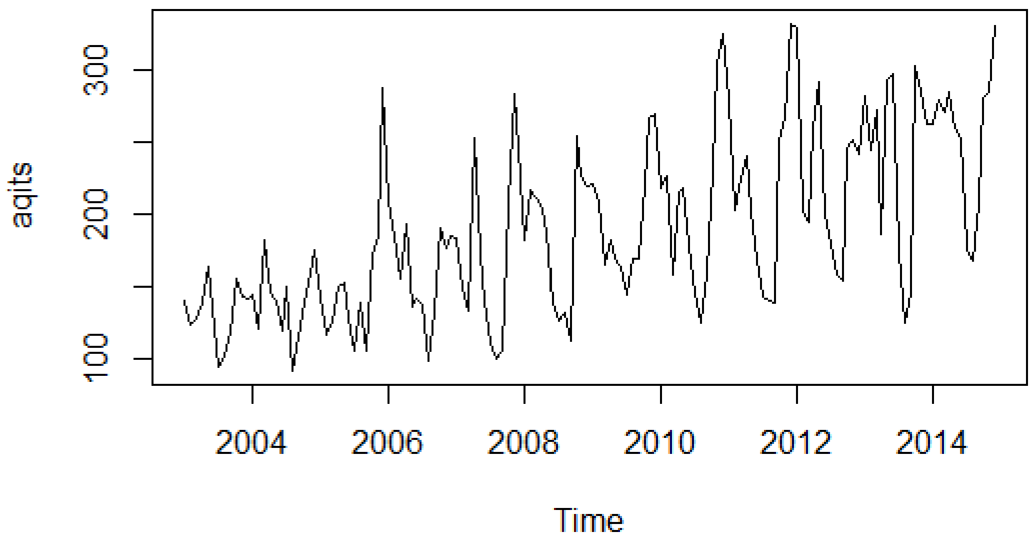
Submitted By – Adesh Kumar Pradhan 2017B3A70960H

Table of Contents

Air Pollution has been an alarming concern for the entire nation, while Delhi becoming the most hazardous place to live in recent times. So here we have collected monthly **Air Quality Index** data from biggest open source data centre in India – data.gov.in

Out of all the components of AQI being concentrations of Nitrogen Dioxide (NO2), Sulphur Dioxide (SO2), Carbon Monoxide (CO), Ozone (O3), PM10 and PM2.5 , we were able to clean and extract data for **PM2.5** or Particulate Matter (size less than 2.5 μm) which is directly inhaled by us and constitute for most of the Air Pollution. The **monthly data** was collected for the years 2003-2014.

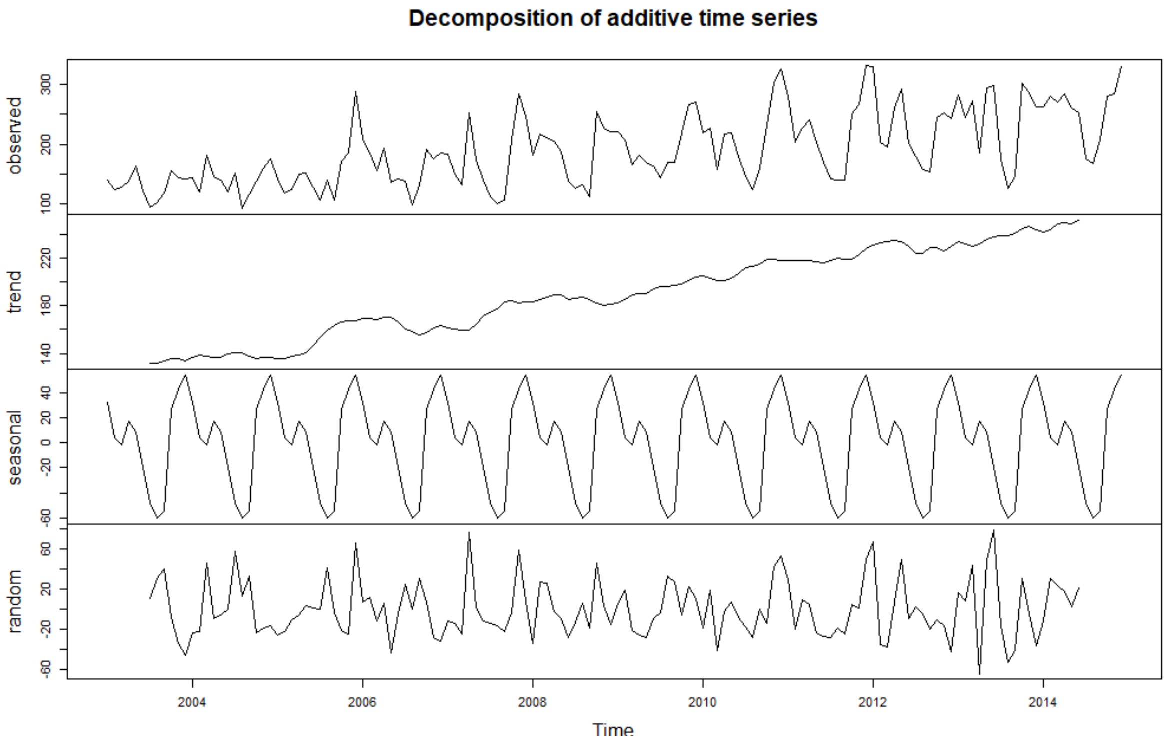
1.1 Plotting the time series data



We can see from the time plot that this time series could probably be described using an **additive model**, because the random fluctuations in the data are roughly constant in size over time so there is no need of log transformation.

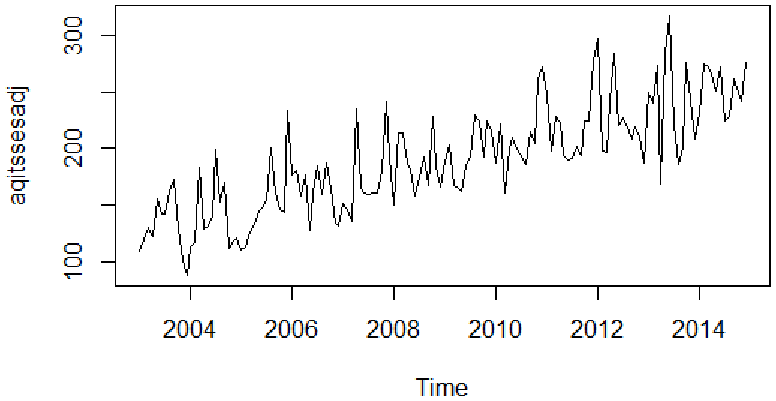
We can also see from this time series that there seems to be **seasonal variation** in AQI per month, there is a peak every Winter and a trough every summer.

1.2 Decomposing seasonal time series data



The seasonal component shows that AQI in Delhi is high during winters (Nov, Dec and Jan). Crop residue burning in other northern states for next cropping season and vehicular pollution are among the major reasons behind such high levels of air pollution during winters in Delhi. Air Pollution drops again during (July, August and September).

1.3 Seasonally Adjusting

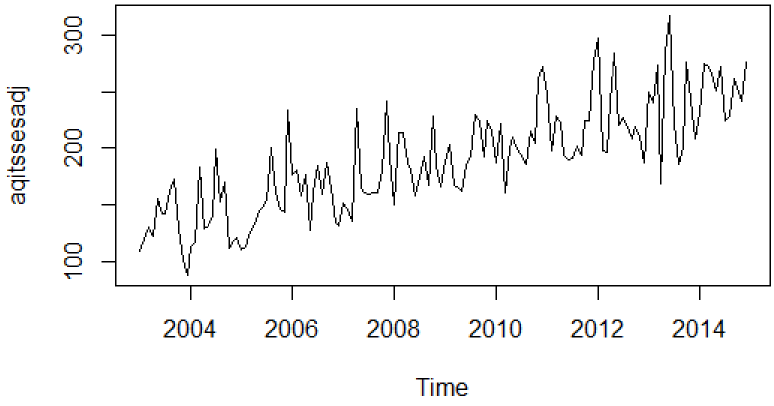


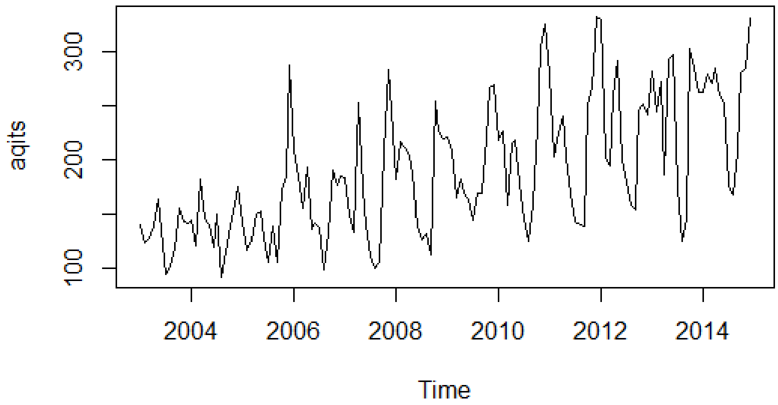
The trend component shows, in general Air Quality is degrading each year drastically. The seasonally adjusted time series now just contains the trend component and an irregular component.

2.1 Checking for non-stationarity

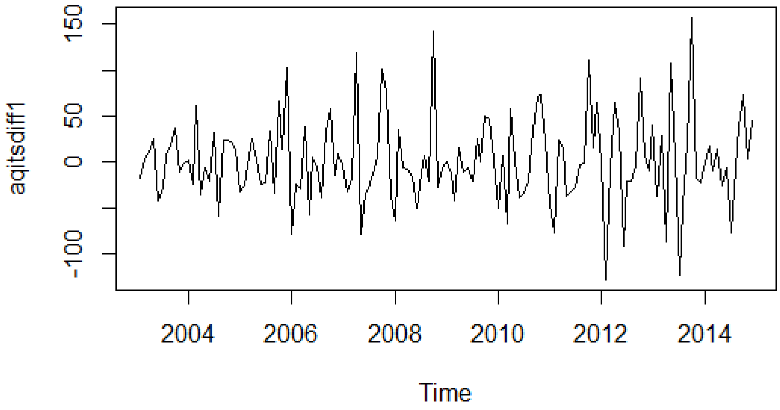
Method-1: From Inspection

After seasonally adjusting the time series, we can say it is non-stationary in mean, as the level changes a lot over time.





Hence, we need to 'difference' the time series until we obtain a stationary time series. Differencing once (d=1) and plotting gives



The time series of first differences appears to be stationary in mean and variance.

Method-2: Formal Tests for Stationarity

a.

```
> adf.test(aqits)
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag    ADF p.value
[1,]   0 -1.006   0.318
[2,]   1 -0.922   0.348
[3,]   2 -0.675   0.437
[4,]   3 -0.339   0.546
[5,]   4 -0.244   0.573
Type 2: with drift no trend
     lag   ADF p.value
[1,]   0 -4.74  0.0100
[2,]   1 -5.00  0.0100
[3,]   2 -4.54  0.0100
[4,]   3 -3.51  0.0100
[5,]   4 -3.15  0.0257
Type 3: with drift and trend
     lag   ADF p.value
[1,]   0 -6.73    0.01
[2,]   1 -7.70    0.01
[3,]   2 -7.87    0.01
[4,]   3 -6.82    0.01
[5,]   4 -7.03    0.01
----
Note: in fact, p.value = 0.01 means p.value <= 0.01
```

ADF test shows series is non-stationary since p-value>0.05

b.

```
> pp.test(aqits)
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag Z_rho p.value
   4 -1.17   0.473
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   4 -38.9    0.01
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   4 -63.2    0.01
---------------
Note: p-value = 0.01 means p.value <= 0.01
```

PP test shows series is non-stationary since p-value>0.05

c.

```
> kpss.test(aqits)
KPSS Unit Root Test
alternative: nonstationary

Type 1: no drift no trend
 lag  stat p.value
   2 0.461     0.1
-----
 Type 2: with drift no trend
 lag  stat p.value
   2 0.994    0.01
-----
 Type 1: with drift and trend
 lag   stat p.value
   2 0.0103     0.1
-----------
Note: p.value = 0.01 means p.value <= 0.01
    : p.value = 0.10 means p.value >= 0.10
```

KPSS test shows series is non-stationary since p-value<0.05

It is clear from the above tests that our time series is non-stationary, so let's run the above tests on first differences.

a.

```
> adf.test(aqitsdiff1)
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag    ADF p.value
[1,]   0 -12.33    0.01
[2,]   1  -9.97    0.01
[3,]   2  -9.99    0.01
[4,]   3  -8.45    0.01
[5,]   4  -7.92    0.01
Type 2: with drift no trend
     lag    ADF p.value
[1,]   0 -12.30    0.01
[2,]   1  -9.95    0.01
[3,]   2  -9.98    0.01
[4,]   3  -8.44    0.01
[5,]   4  -7.91    0.01
Type 3: with drift and trend
     lag    ADF p.value
[1,]   0 -12.26    0.01
[2,]   1  -9.92    0.01
[3,]   2  -9.94    0.01
[4,]   3  -8.41    0.01
[5,]   4  -7.88    0.01
----
Note: in fact, p.value = 0.01 means p.value <= 0.01
```

ADF test shows series is stationary since p-value<0.05

b.

```
> pp.test(aqitsdiff1)
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag Z_rho p.value
   4  -116    0.01
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   4  -116    0.01
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   4  -116    0.01
---------------
Note: p-value = 0.01 means p.value <= 0.01
```

PP test shows series is stationary since p-value<0.05

c.

```
> kpss.test(aqitsdiff1)
KPSS Unit Root Test
alternative: nonstationary

Type 1: no drift no trend
 lag   stat p.value
   2 0.0315     0.1
-----
 Type 2: with drift no trend
 lag   stat p.value
   2 0.0153     0.1
-----
 Type 1: with drift and trend
 lag   stat p.value
   2 0.0112     0.1
-----------
Note: p.value = 0.01 means p.value <= 0.01
    : p.value = 0.10 means p.value >= 0.10
```

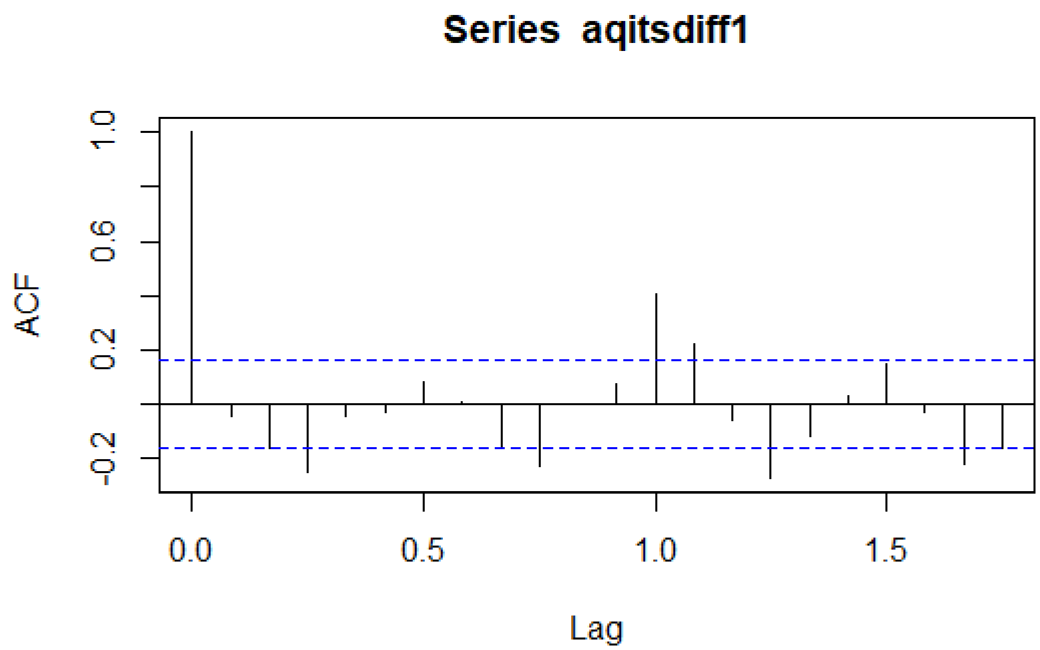
KPSS test shows series is stationary since p-value>0.05

After analysing from both the methods we conclude that the time series of first differences appears to be stationary in mean and variance, and so an ARIMA(p,1,q) model is probably appropriate for the time series of the AQIs of Delhi. Thus, it appears that we need to difference the time series of the AQIs of Delhi once in order to achieve a stationary series.

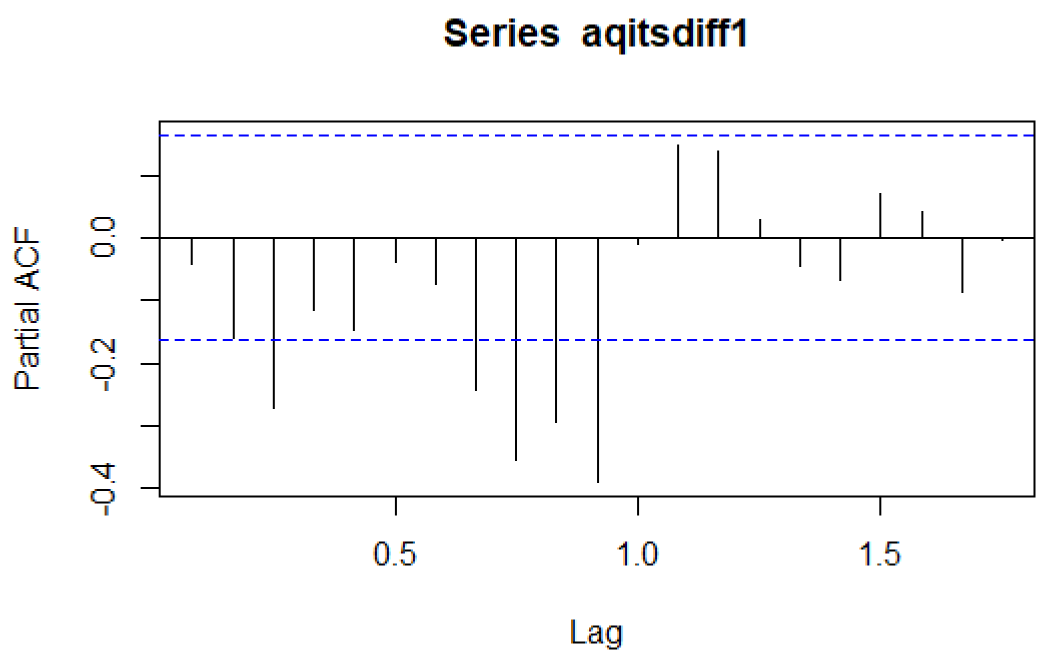
2.2 Selecting a Candidate ARIMA Model

We can now examine whether there are correlations between successive terms of this irregular component which could help us to make a predictive model for the AQI of Delhi.

Method-1: Plotting a Correlogram(acf) and Partial Correlogram(pacf)



We see from the correlogram of ACF that the autocorrelations are significant at many lags from 1-20, hence we can not rely on ACF plot to deduce the ARMA model.



From the PACF correlogram, we see that the partial autocorrelations are significant at many lags from 1-20, hence we cannot rely on PACF plot to deduce the ARMA model.

The possible ARMA model will be an ARMA(p,q) model with differences, d = 1, that is, a mixed model with p and q greater than 0, since the auto correlogram and partial correlogram tail off to zero. From ACF and PACF plots, it seems p and q will constitute of many lags, hence not reliable as per the principal of parsimony.

Method-2: The auto.arima() Function

```
> auto.arima(aqits)
Series: aqits
ARIMA(2,0,2)(0,1,1)[12] with drift

Coefficients:
          ar1      ar2     ma1     ma2     sma1   drift
      -0.9433  -0.7659  1.1496  0.9432  -0.7813  0.9195
s.e.   0.1154   0.0877  0.0747  0.0623   0.0971  0.0819

sigma^2 estimated as 908:  log likelihood=-639.78
AIC=1293.56   AICc=1294.46   BIC=1313.73
```

It gives seasonal ARIMA(2,0,2) with (0,1,1) as seasonal part. [12] stands for number of periods in season, i.e. months in year in this case.

If we use the “bic” criterion, which penalises the number of parameters, we get

```
> auto.arima(aqits, ic = "bic")
Series: aqits
ARIMA(0,0,1)(0,1,1)[12] with drift

Coefficients:
         ma1     sma1   drift
      0.2170  -0.7823  0.9208
s.e.  0.0903   0.0954  0.0904

sigma^2 estimated as 958.5:  log likelihood=-644.58
AIC=1297.17   AICc=1297.48   BIC=1308.7
```

It gives seasonal ARIMA(0,0,1) with (0,1,1) as seasonal part.

Possible ARIMA models –

a. ARIMA(2,0,2) – has 4 parameters
b. ARIMA(0,0,1) – has 1 parameter
c. ARIMA(p,1,q) – has at least 3 parameters

From principle of parsimony we choose ARIMA(0,0,1) model or MA(1) model with least parameters with seasonal part (0,1,1)

$X_t = \mu + \varepsilon_t + \theta\, \varepsilon_{t-1}$

where X_t is the stationary time series we are studying (the monthly AQIs of Delhi), μ is the mean of time series X_t, ε_t is white noise with mean zero and constant variance, and θ is a parameter which is equal to 0.271 as can be seen from the outcome of auto.arima().

A MA (moving average) model is usually used to model a time series that shows short-term dependencies between successive observations.

Intuitively, it makes good sense that a MA model can be used to describe the time series of the monthly AQIs of Delhi, as we might expect the AQI of a particular month to have some effect on the AQI of next month or two, but not much effect on the AQIs much longer after that.

2.3 Forecasting

We discussed above that an appropriate ARIMA model for the time series of AQIs of Delhi is an ARIMA(0,0,1) model with seasonal part (0,1,1) and period = 12(months), so to fit an ARIMA (0,0,1) model to this time series we have

```
> aqitsarima

Call:
arima(x = aqits, order = c(0, 0, 1), seasonal = list(order = c(0, 1, 1), period = 12))

Coefficients:
         ma1     sma1
      0.3170  -0.4304
s.e.  0.0802   0.0765
```

An ARIMA(0,0,1) can be written as

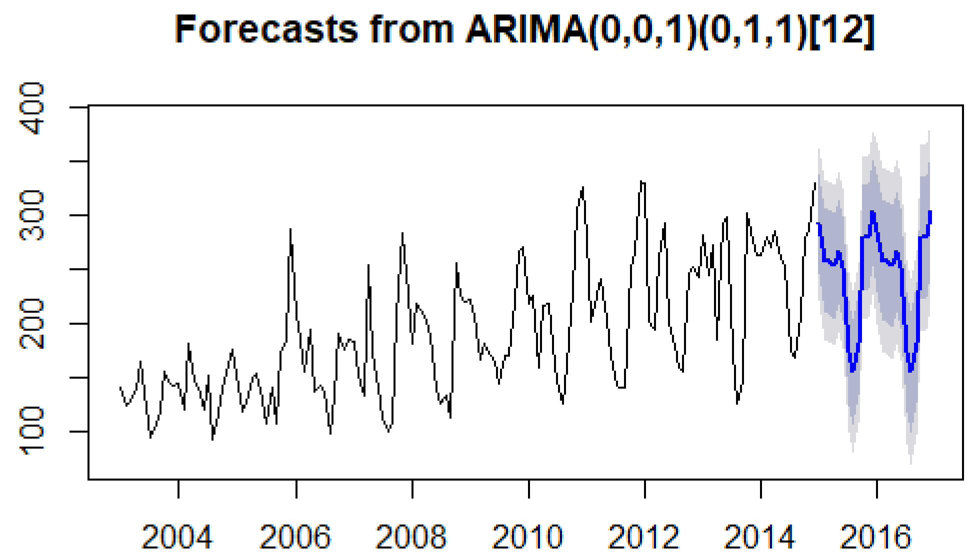
$X_t = \mu + \varepsilon_t + \theta\, \varepsilon_{t-1}$

From the output of the “arima()” R function (above), the estimated value of θ(given as ‘ma1’ in the R output) is 0.3170

After forecasting for the next 24 months (2 years) we get estimates as

	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
Jan 2015	291.8938	245.52121	338.2665	220.97303	362.8147
Feb 2015	256.6778	208.03153	305.3240	182.27977	331.0758
Mar 2015	257.0060	208.35978	305.6523	182.60802	331.4040
Apr 2015	252.7733	204.12708	301.4196	178.37532	327.1713
May 2015	266.2228	217.57653	314.8690	191.82477	340.6208
Jun 2015	251.5193	202.87310	300.1656	177.12133	325.9173
Jul 2015	173.2311	124.58488	221.8774	98.83312	247.6291
Aug 2015	154.3447	105.69846	202.9909	79.94670	228.7427
Sep 2015	181.2374	132.59116	229.8836	106.83940	255.6354
Oct 2015	278.5567	229.91044	327.2029	204.15868	352.9547
Nov 2015	279.7984	231.15211	328.4446	205.40035	354.1964
Dec 2015	302.5903	253.94408	351.2366	228.19231	376.9883
Jan 2016	283.9122	228.55648	339.2678	199.25296	368.5714
Feb 2016	256.6778	200.69248	312.6631	171.05565	342.2999
Mar 2016	257.0060	201.02073	312.9913	171.38391	342.6281
Apr 2016	252.7733	196.78802	308.7586	167.15120	338.3954
May 2016	266.2228	210.23747	322.2081	180.60065	351.8449
Jun 2016	251.5193	195.53404	307.5046	165.89722	337.1415
Jul 2016	173.2311	117.24583	229.2164	87.60901	258.8532
Aug 2016	154.3447	98.35941	210.3300	68.72258	239.9668
Sep 2016	181.2374	125.25210	237.2227	95.61528	266.8595
Oct 2016	278.5567	222.57139	334.5420	192.93456	364.1788
Nov 2016	279.7984	223.81306	335.7836	194.17623	365.4205
Dec 2016	302.5903	246.60502	358.5756	216.96820	388.2124

And the plot as –

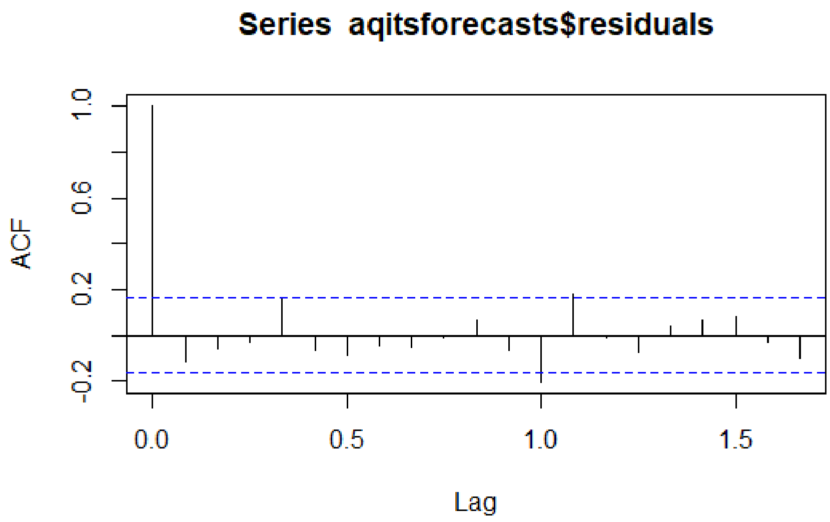


The point forecasts for the next 24months are quite appropriate for the seasonal component as the AQIs in Nov, Dec, Jan and Feb is high and July, Aug and Sept low, which is the actual seasonal nature of AQIs in Delhi. But there is no increasing trend in AQIs over the years which was expectable from our previous analysis.

2.4 Is the given predictive model good?

If the predictive model cannot be improved upon, there should be no correlations between forecast errors for successive predictions. Therefore, it is a good idea to investigate –

a. Whether correlations between successive forecast errors are there
To figure out whether there are correlations between successive forecast errors, we can obtain a correlogram of the in-sample forecast errors for lags 1-20.



We can see from the correlogram that none of the sample autocorrelations for lags 1-20 exceed the significance bounds. The autocorrelation for lags 12 and 13 exceed the significance bounds too, but it is likely that this is due to chance, since they just exceed the significance bounds (especially for lag 13), the autocorrelations for lags 1-11 do not exceed the significance bounds, and we would expect 1 in 20 lags to exceed the 95% significance bounds by chance alone.

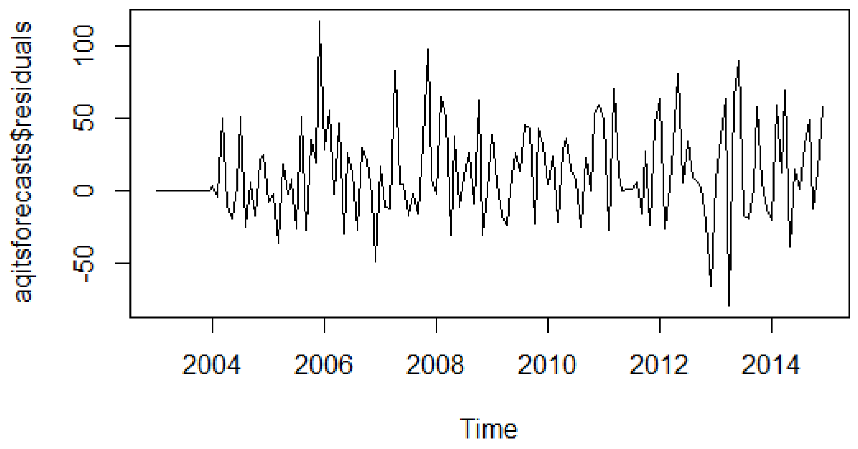
To test whether there is significant evidence for non-zero correlations at lags 1-20, we can carry out a Ljung Box test.

```
> Box.test(aqitsforecasts$residuals, lag=20, type="Ljung-Box")

        Box-Ljung test

data:  aqitsforecasts$residuals
X-squared = 26.993, df = 20, p-value = 0.1355
```

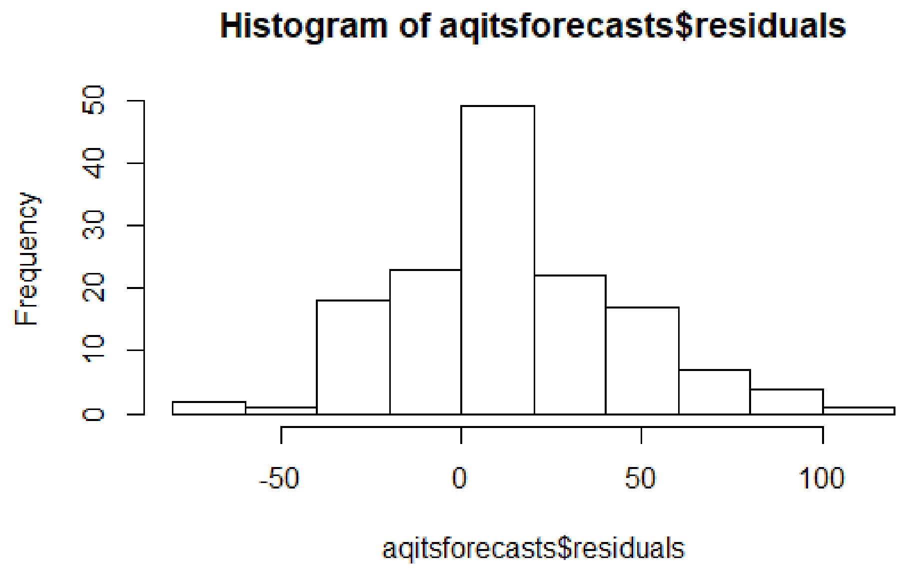
From ACF & Box-Ljung test, we conclude that since the correlogram shows that none of the sample autocorrelations for lags 1-20 exceed the significance bounds, and the p-value for the Ljung-Box test is 0.1355, we can conclude that there is very little evidence for non-zero autocorrelations in the forecast errors at lags 1-20.

b. <u>Whether the forecast errors of an ARIMA model are normally distributed with mean zero and constant variance</u>

To investigate whether the forecast errors are normally distributed with mean zero and constant variance, we can make a time plot and histogram (with overlaid normal curve) of the forecast errors



The time series plot of the in-sample forecast errors shows that the variance of the forecast errors seems to be roughly constant over time(though perhaps there is slightly higher variance for the second half of the time series).



The histogram of the time series shows that the forecast errors are roughly normally distributed and the mean seems to be close to zero.

Therefore, it is plausible that the forecast errors are normally distributed with mean zero and constant variance

Since successive forecast errors do not seem to be correlated, and the forecast errors seem to be normally distributed with mean zero and constant variance, the ARIMA(0,0,1) does seem to provide an adequate predictive model for the monthly AQIs of Delhi.