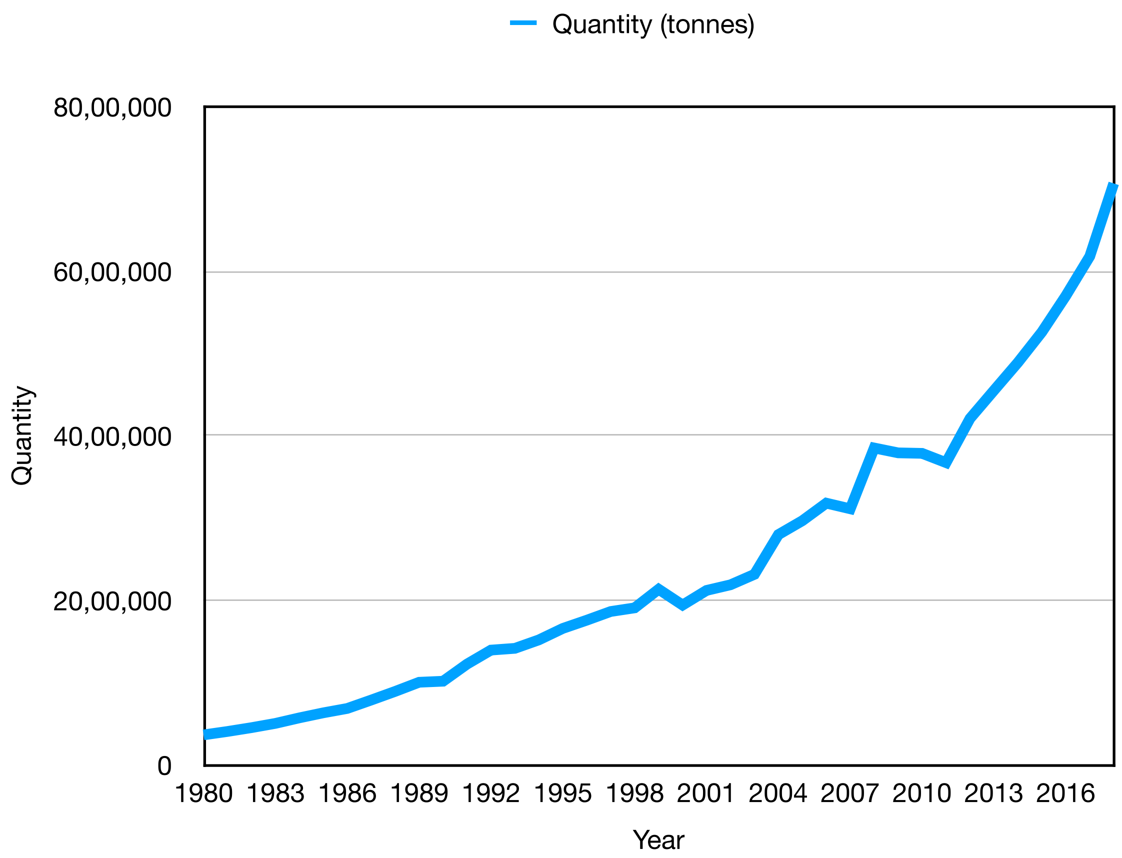
Time Series Analysis of Aquaculture Production in India

Dataset Information-

Dataset gives information about aquaculture production per year in India from Year 1980 till 2018.

Time Series plot of Aquaculture Production in India



As we can observe from the plot this time series has trend but does not have seasonality associated with it.

R code

Reading csv file and preprocessing quantity column by removing commas and converting strings to int.

```
> aquaculture_production <- read.csv("~/Desktop/Capstone/aqua_production.csv")
> aquaculture_production$Quantity..tonnes.=as.numeric(gsub(",", "", aquaculture_production$Quantity..tonnes, fixed = TRUE))
> aquaculture_production
   Year Quantity..tonnes.
1  1980            365180
2  1981            406622
3  1982            452939
4  1983            504733
5  1984            572000
6  1985            633250
7  1986            686260
8  1987            788310
9  1988            893330
10 1989           1004500
11 1990           1017136
12 1991           1225261
13 1992           1395444
14 1993           1416702
15 1994           1519528
16 1995           1658807
17 1996           1758739
18 1997           1864322
19 1998           1908485
20 1999           2134814
21 2000           1942531
22 2001           2120466
23 2002           2188789
24 2003           2315771
25 2004           2798686
26 2005           2967378
27 2006           3180863
28 2007           3112240
29 2008           3851057
30 2009           3791920
31 2010           3785779
32 2011           3673082
33 2012           4209478
34 2013           4550707
35 2014           4890000
36 2015           5260000
37 2016           5700000
38 2017           6180000
39 2018           7066000
```

Check for Time Series Stationary

We will use augmented dickey fuller test to check for stationarity of the time series.

```
> adf.test(aquaculture_production$Quantity..tonnes.)

	Augmented Dickey-Fuller Test

data:  aquaculture_production$Quantity..tonnes.
Dickey-Fuller = 2.7764, Lag order = 3, p-value = 0.99
alternative hypothesis: stationary
```

P-value is much higher than 0.05 so we can conclude that the given time series is not stationary.

We can see that p-value for time series generated after second order differencing is less than 0.05 hence it is stationary and value of d is 2.

```
> aquaculture_d1 <- diff(aquaculture_production$Quantity..tonnes., differences = 1)
> adf.test(aquaculture_d1)

	Augmented Dickey-Fuller Test

data:  aquaculture_d1
Dickey-Fuller = -1.0256, Lag order = 3, p-value = 0.921
alternative hypothesis: stationary

> aquaculture_d2 <- diff(aquaculture_production$Quantity..tonnes., differences = 2)
> adf.test(aquaculture_d2)

	Augmented Dickey-Fuller Test

data:  aquaculture_d2
Dickey-Fuller = -4.7634, Lag order = 3, p-value = 0.01
alternative hypothesis: stationary
```

Time Series Models-

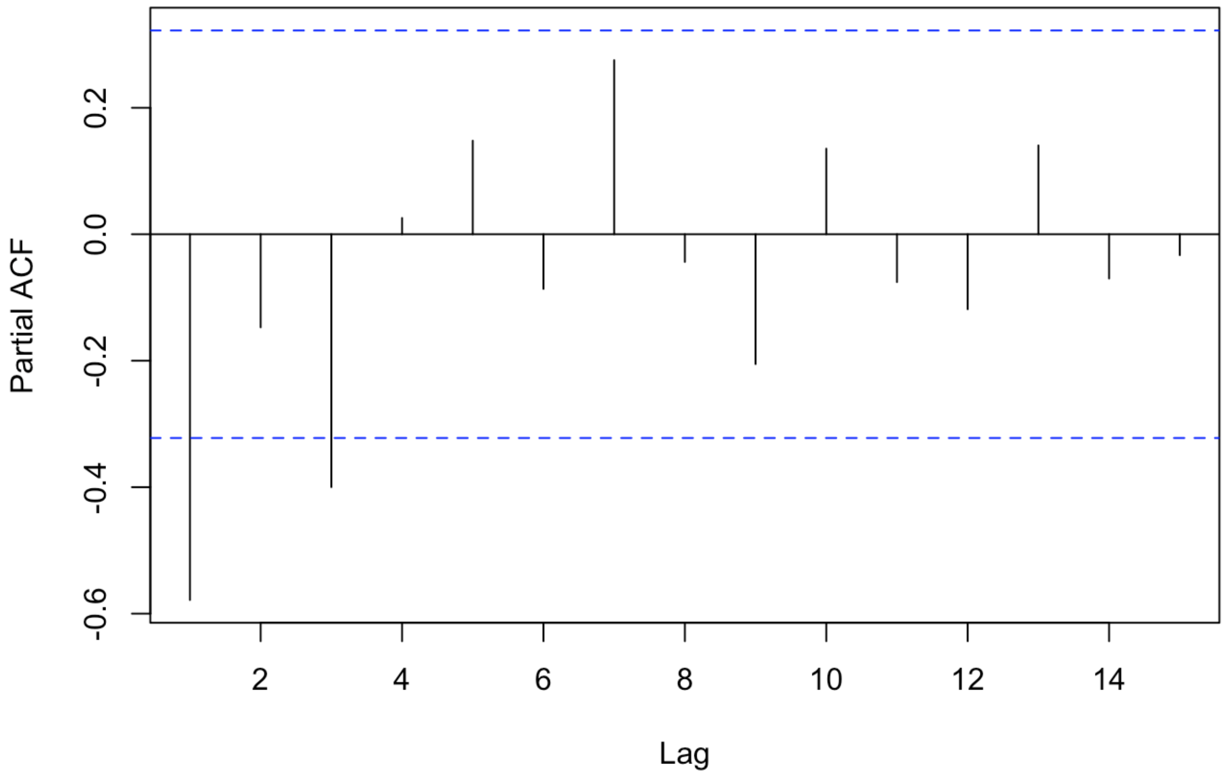
- ## Naive Model

```
> time_series<-ts(aquaculture_production$Quantity..tonnes.,start=1980,end=2018)
> model1<-naive(time_series)
> summary(model1)

Forecast method: Naive method

Model Information:
Call: naive(y = time_series)

Residual sd: 283643.5694

Error measures:
                   ME     RMSE      MAE      MPE     MAPE MASE      ACF1
Training set 176337.4 283643.6 199436.4 7.319256 8.208386    1 0.1437445

Forecasts:
     Point Forecast   Lo 80   Hi 80   Lo 95   Hi 95
2019        7066000 6702496 7429504 6510069 7621931
2020        7066000 6551928 7580072 6279795 7852205
2021        7066000 6436393 7695607 6103099 8028901
2022        7066000 6338992 7793008 5954138 8177862
2023        7066000 6253181 7878819 5822900 8309100
2024        7066000 6175601 7956399 5704252 8427748
2025        7066000 6104259 8027741 5595144 8536856
2026        7066000 6037856 8094144 5493589 8638411
2027        7066000 5975488 8156512 5398206 8733794
2028        7066000 5916500 8215500 5307991 8824009
```

- ## AR Model

Pacf plot for finding p value

```
plot(pacf(aquaculture_d2))
```



P value is 1 from the above graph.

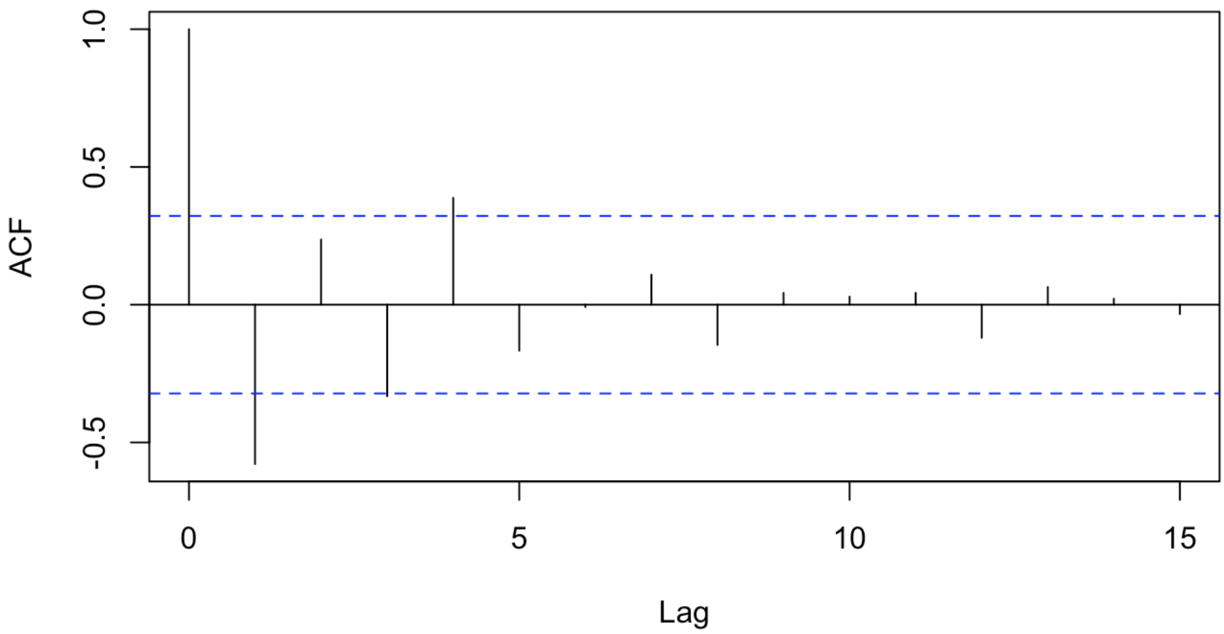
```
> model2<-ar(time_series)
> f2<-forecast(model2)
> summary(f2)

Forecast method: AR(1)

Model Information:

Call:
ar(x = time_series)

Coefficients:
     1
0.8784

Order selected 1  sigma^2 estimated as  7.28e+11

Error measures:
                   ME     RMSE      MAE       MPE     MAPE    MASE      ACF1
Training set 161587.8 402852.4 264638.1 -2.624657 12.51695 1.32693 0.6117881

Forecasts:
     Point Forecast   Lo 80   Hi 80     Lo 95   Hi 95
2019        6505518 5412052 7598984 4833206.0 8177830
2020        6013182 4557757 7468607 3787301.4 8239062
2021        5580706 3898402 7263010 3007843.4 8153568
2022        5200812 3362485 7039140 2389332.6 8012292
2023        4867108 2916904 6817313 1884527.7 7849688
2024        4573977 2541652 6606302 1465804.1 7682150
2025        4316486 2222996 6409976 1114769.8 7518202
2026        4090302 1950812 6229793  818233.8 7362370
2027        3891619 1717298 6065939  566282.8 7216954
2028        3717092 1516273 5917910  351230.4 7082953
```

Autoregressive model is clearly not suited for the given time series since it performs even worse than naive model.

- **MA Model**

Acf plot for finding q value for moving average model.

```
plot(acf(aquaculture_d2))
```



Q value from above graph is 1.

```
> model3<-ma(time_series,order = 1)
> f3<-forecast(model3)
> summary(f3)

Forecast method: ETS(M,A,N)

Model Information:
ETS(M,A,N)

Call:
 ets(y = object, lambda = lambda, biasadj = biasadj, allow.multiplicative.trend = allow.multiplicative.trend)


  Smoothing parameters:
    alpha = 0.7289
    beta  = 0.1641

  Initial states:
    l = 255493.7319
    b = 72112.9345

  sigma:  0.0716

     AIC     AICc      BIC
1067.004 1068.822 1075.322

Error measures:
                   ME     RMSE      MAE      MPE     MAPE      MASE       ACF1
Training set 58189.46 202562.9 138318.1 1.267555 5.270078 0.6935449 0.09772498

Forecasts:
     Point Forecast   Lo 80    Hi 80   Lo 95    Hi 95
2019        7346848 6672704  8020993 6315833  8377863
2020        7791293 6855697  8726889 6360423  9222164
2021        8235738 7014033  9457443 6367302 10104174
2022        8680183 7146234 10214132 6334210 11026156
2023        9124628 7251594 10997662 6260069 11989186
2024        9569073 7329651 11808495 6144173 12993973
2025       10013518 7380010 12647025 5985915 14041120
2026       10457962 7402277 13513648 5784694 15131231
2027       10902407 7396031 14408784 5539868 16264947
2028       11346852 7360817 15332888 5250737 17442967
```

Model performance is better compared to naive and AR(1) model.

- **ARIMA Model**

p=1, d=2, q=1

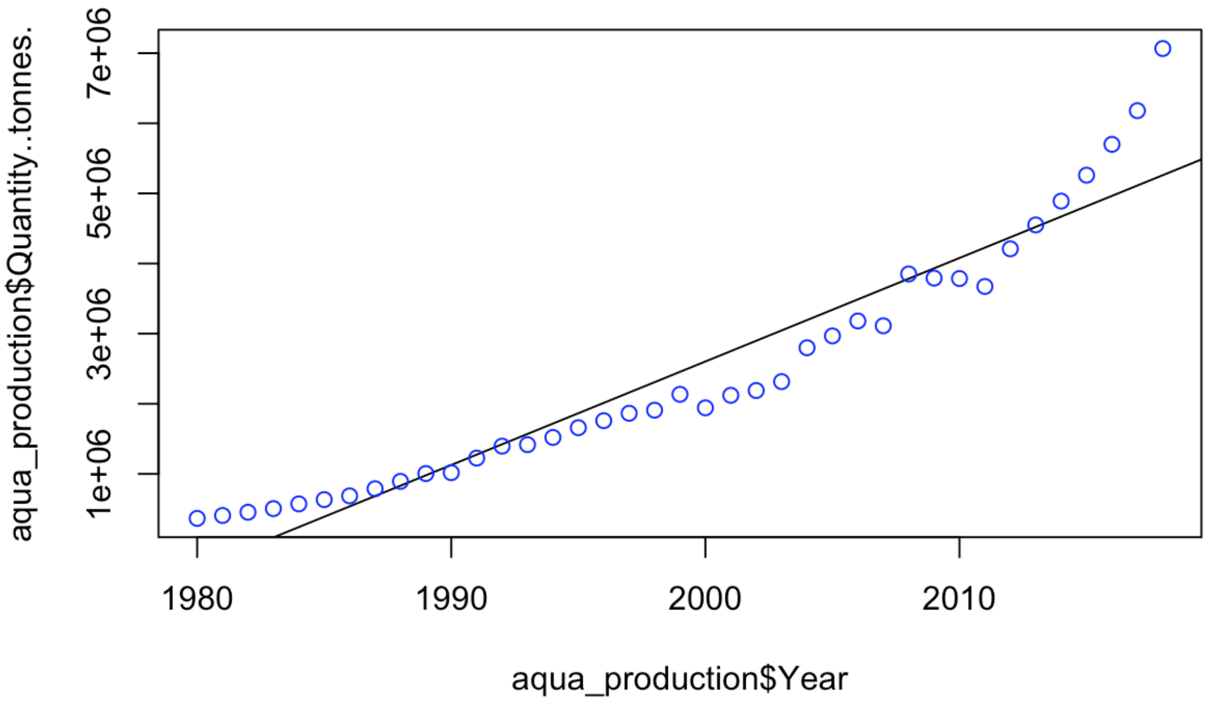
```
> model4<-arima(time_series,order=c(1,2,1))
> f4<-forecast(model4)
> summary(f4)

Forecast method: ARIMA(1,2,1)

Model Information:

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

Coefficients:
          ar1      ma1
      -0.2565  -0.6303
s.e.   0.2086   0.1766

sigma^2 estimated as 4.122e+10:  log likelihood = -505.12,  aic = 1016.23

Error measures:
                   ME     RMSE    MAE      MPE     MAPE      MASE        ACF1
Training set 43866.01 197748.1 130387 1.266394 4.696972 0.6537773 -0.05995047

Forecasts:
     Point Forecast    Lo 80    Hi 80   Lo 95    Hi 95
2019        7527638  7267455  7787822 7129722  7925554
2020        8098139  7708809  8487470 7502709  8693570
2021        8640714  8097888  9183539 7810533  9470894
2022        9190452  8484263  9896641 8110430 10270474
2023        9738353  8855968 10620738 8388861 11087844
2024       10286725  9216624 11356825 8650147 11923302
2025       10834976  9566021 12103931 8894277 12775675
2026       11383258  9904869 12861646 9122258 13644258
2027       11931532 10233587 13629477 9334749 14528315
2028       12479808 10552604 14407012 9532404 15427212
```

Performance of ARIMA(1,2,1) model is better than all the models seen so far.

- **Linear Regression Model**

```
> model6<-lm(aqua_production$Quantity..tonnes.~aqua_production$Year)
> summary(model6)

Call:
lm(formula = aqua_production$Quantity..tonnes. ~ aqua_production$Year)

Residuals:
    Min      1Q  Median      3Q     Max 
-731106 -315202 -109982  231606 1804016 

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)    
(Intercept)          -292743742   14942954  -19.59   <2e-16 ***
aqua_production$Year     147674       7475   19.75   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 525400 on 37 degrees of freedom
Multiple R-squared:  0.9134,	Adjusted R-squared:  0.9111 
F-statistic: 390.3 on 1 and 37 DF,  p-value: < 2.2e-16
```

```
> plot(aqua_production$Year,aqua_production$Quantity..tonnes.,col="blue",abline(lm(aqua_production$Quantit
y..tonnes.~aqua_production$Year)))
```



- **Exponential Smoothing Model**

```
> model7<-ets(time_series)
> f7<-forecast(model7)
> summary(f7)

Forecast method: ETS(M,A,N)

Model Information:
ETS(M,A,N)

Call:
 ets(y = time_series)

  Smoothing parameters:
    alpha = 0.7289
    beta  = 0.1641

  Initial states:
    l = 255493.7319
    b = 72112.9345

  sigma:  0.0716

     AIC     AICc      BIC
1067.004 1068.822 1075.322

Error measures:
                   ME     RMSE      MAE      MPE     MAPE      MASE       ACF1
Training set 58189.46 202562.9 138318.1 1.267555 5.270078 0.6935449 0.09772498

Forecasts:
     Point Forecast    Lo 80    Hi 80   Lo 95    Hi 95
2019        7346848 6672704  8020993 6315833  8377863
2020        7791293 6855697  8726889 6360423  9222164
2021        8235738 7014033  9457443 6367302 10104174
2022        8680183 7146234 10214132 6334210 11026156
2023        9124628 7251594 10997662 6260069 11989186
2024        9569073 7329651 11808495 6144173 12993973
2025       10013518 7380010 12647025 5985915 14041120
2026       10457962 7402277 13513648 5784694 15131231
2027       10902407 7396031 14408784 5539868 16264947
2028       11346852 7360817 15332888 5250737 17442967
```

- **Holt Winters Model**

```
> model8<-HoltWinters(time_series,beta=FALSE,gamma=FALSE)
> f8<-forecast(model8)
> summary(f8)

Forecast method: HoltWinters

Model Information:
Holt-Winters exponential smoothing without trend and without seasonal component.

Call:
HoltWinters(x = time_series, beta = FALSE, gamma = FALSE)

Smoothing parameters:
 alpha: 0.99992
 beta : FALSE
 gamma: FALSE

Coefficients:
     [,1]
a 7065929

Error measures:
                   ME     RMSE      MAE      MPE     MAPE     MASE      ACF1
Training set 176349.6 283653.4 199443.9 7.319779 8.208757 1.000038 0.1438206

Forecasts:
     Point Forecast   Lo 80   Hi 80   Lo 95   Hi 95
2019        7065929 6777383 7354475 6624636 7507222
2020        7065929 6657880 7473978 6441872 7689986
2021        7065929 6566180 7565678 6301628 7830230
2022        7065929 6488872 7642986 6183397 7948462
2023        7065929 6420762 7711096 6079232 8052627
2024        7065929 6359186 7772672 5985059 8146799
2025        7065929 6302561 7829297 5898458 8233400
2026        7065929 6249855 7882003 5817852 8314006
2027        7065929 6200353 7931505 5742145 8389713
2028        7065929 6153533 7978325 5670539 8461319
```

- **Holt Winters model with Trend Smoothing**

```
> model9<-HoltWinters(time_series,gamma=FALSE)
> f9<-forecast(model9)
> summary(f9)

Forecast method: HoltWinters

Model Information:
Holt-Winters exponential smoothing with trend and without seasonal component.

Call:
HoltWinters(x = time_series, gamma = FALSE)

Smoothing parameters:
 alpha: 0.7270029
 beta : 0.4338481
 gamma: FALSE

Coefficients:
     [,1]
a 6929770
b  580922

Error measures:
                   ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set 46227.43 202294.6 139213.8 1.371001 5.010079 0.6980362 -0.01494452

Forecasts:
     Point Forecast    Lo 80    Hi 80   Lo 95    Hi 95
2019        7510692  7254819  7766564 7119368  7902015
2020        8091614  7722002  8461225 7526341  8656886
2021        8672536  8165269  9179803 7896738  9448334
2022        9253458  8589667  9917249 8238277 10268638
2023        9834380  8997998 10670762 8555244 11113515
2024       10415302  9392025 11438578 8850335 11980268
2025       10996224  9772980 12219468 9125433 12867014
2026       11577146 10141787 13012505 9381954 13772338
2027       12158068 10499181 13816955 9621019 14695117
2028       12738990 10845763 14632217 9843549 15634431
```

- **Non Linear Regression Model**

```
> y<-aquaculture_production$Quantity..tonnes.
> x<-aquaculture_production$Year
> model11<-lm(y ~ poly(x,3,raw = TRUE))
> f11<-forecast(model11,newdata = data.frame(x=c(2019,2020,2021,2022)))
> summary(f11)

Forecast method: Linear regression model

Model Information:

Call:
lm(formula = y ~ poly(x, 3, raw = TRUE))

Coefficients:
            (Intercept)  poly(x, 3, raw = TRUE)1  poly(x, 3, raw = TRUE)2  poly(x, 3, raw = TRUE)3
             -1.232e+12                1.857e+09               -9.329e+05                1.562e+02


Error measures:
                       ME     RMSE      MAE       MPE     MAPE       MASE
Training set 3.357697e-11 174807.6 136841.7 0.1030052 8.187335 0.09471035

Forecasts:
  Point Forecast   Lo 80   Hi 80   Lo 95   Hi 95
1        7030200 6734959 7325441 6571338 7489062
2        7501657 7185046 7818267 7009582 7993732
3        8000718 7657014 8344421 7466535 8534900
4        8528320 8151432 8905208 7942563 9114078
```

Conclusion

We can observe from the RSME values of each model that Non linear regression model performs the best for the given time series followed by ARIMA model.

Forecasts Using The best model

Forecasted values for the best performing model ie - Non linear regression model from year 2019 till 2028.

```
> f11<-forecast(model11,newdata = data.frame(x=c(2019,2020,2021,2022,2023,2024,2025,2026,2027,2028)))
> f11
   Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
1         7030200  6734959  7325441  6571338  7489062
2         7501657  7185046  7818267  7009582  7993732
3         8000718  7657014  8344421  7466535  8534900
4         8528320  8151432  8905208  7942563  9114078
5         9085402  8669006  9501798  8438242  9732562
6         9672900  9210537 10135262  8954298 10391501
7        10291751  9776885 10806617  9491549 11091953
8        10942893 10368934 11516853 10050848 11834938
9        11627263 10987569 12266958 10633054 12621473
10       12345799 11633668 13057930 11239008 13452590
```