**NAME**: Penugonda Abhignanendra **REGNO**:19BCE1677
**FACULTY**: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

# Essential of Data Analytics
# Tasks for Week-2: Time-series Forecasting

Understand time-series operations/functions and forecast the annual gdp growth rate of India based on given instructions.

**Aim**: To develop a time series forecasting model for the given data using R programming and to predict the future data

## Algorithm:

- Set the working directory and the read the respective csv file using read.csv() function.
- Import the forecast and tseries libraries.
- Using ts() function we convert normal numerical data int R time series object. In ts the previous data which use for predicting is written first later the start will the start data of year, month or day and end will be the end data of year, month or day.
- By using class we can check the class of the object
- By plotting we can the variation in data
- By the acf() function if all the lines are bellow the blue line that means the data taken is stationary and also the data is less correlated with the lagged data points else the data is non stationary in that situation we have to convert the non-stationary data to stationary data.
- Pacf function can computes an estimate of the partial auto correlation function of time series.
- By adf.test() function we can get the p value if the produced p value is not less than critical p value that means 0.05 then the data is not stationary.
- By using the aut.arima() we can make the non-stationary data into stationary by using some models. the pdq values in arima means p value is the auto regression d value is integrated and q is the moving average.
- By using the best model, we use forecast to future data.
- By plotting the graph, we can see the range of change in data
- By using accuracy data, we can we can find how best our model is.

## Inference:

**NAME**: Penugonda Abhignanendra **REGNO**:19BCE1677
**FACULTY**: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

# Essential of Data Analytics
# Tasks for Week-2: Time-series Forecasting

## Case 1-gold:

The p-value is more than the critical p value that means the data is non stationary data. We convert it into stationary and the arima values are p=0, d=1, q=0.

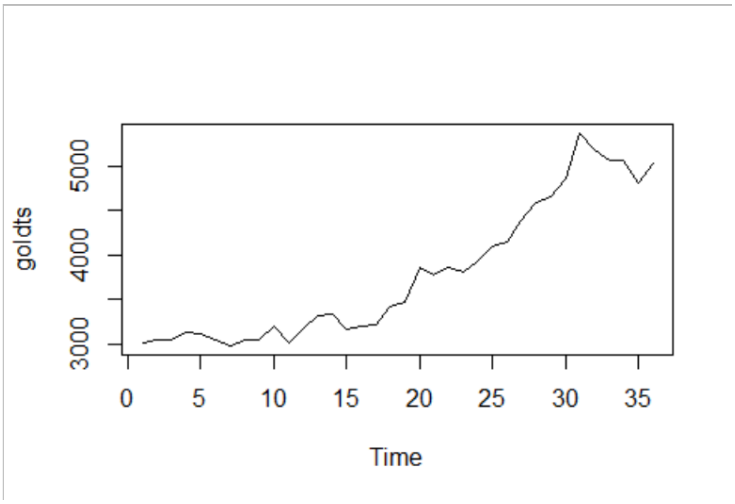
## Case 2-gdp:

The p-value is less than the critical p value that means the data is stationary data. We convert it into stationary and the arima values are p=0, d=1, q=1.

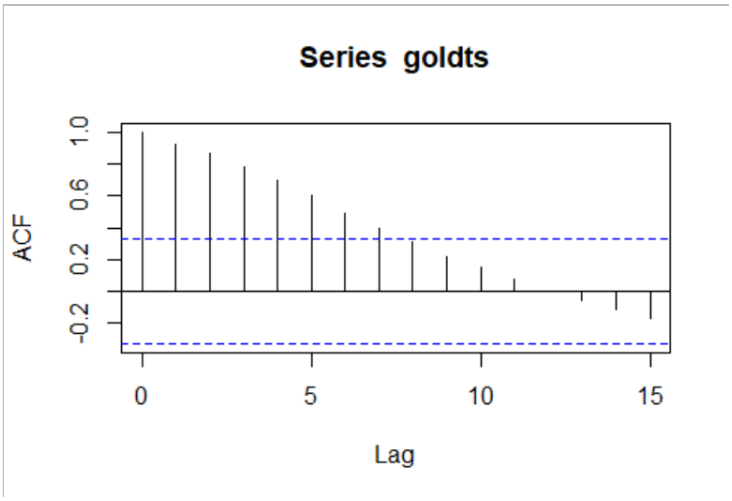
## Result:

## Case 1- Gold:

**Plot of gold data**



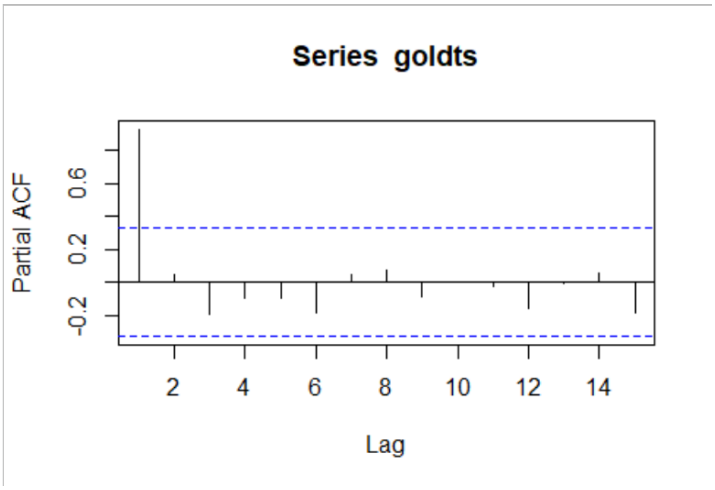

**ACF of gold data**



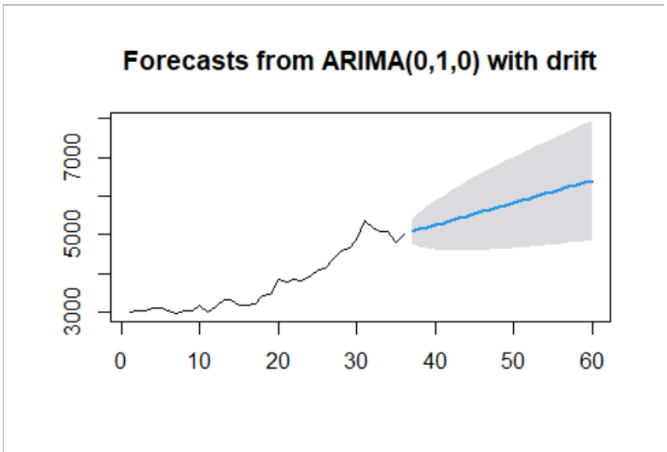
**NAME**: Penugonda Abhignanendra **REGNO**:19BCE1677
**FACULTY**: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

# Essential of Data Analytics
# Tasks for Week-2: Time-series Forecasting

## Partial ACF of gold data




## Forecast of gold data




## Forecasted value for gold data

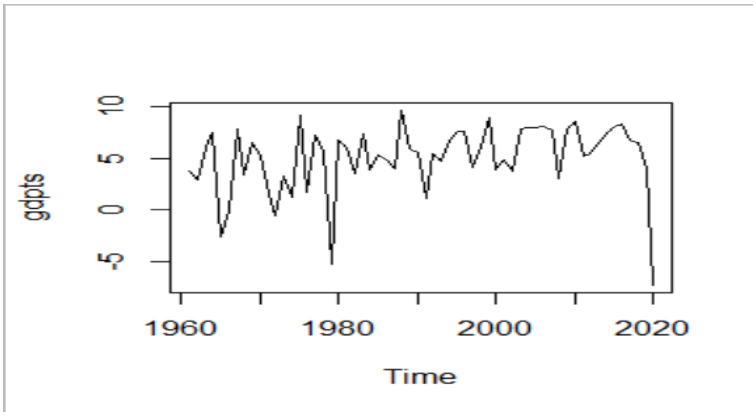
**NAME**: Penugonda Abhignanendra **REGNO**:19BCE1677
**FACULTY**: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

# Essential of Data Analytics
# Tasks for Week-2: Time-series Forecasting

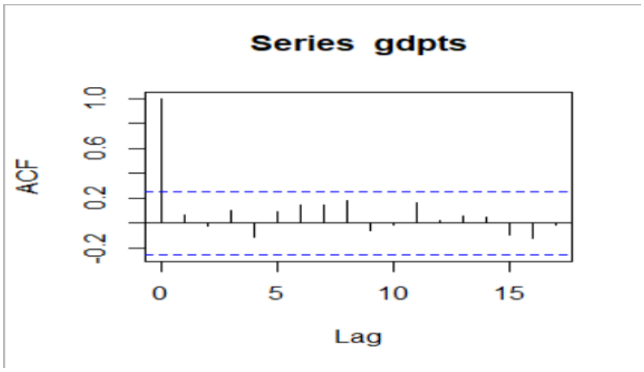
```
> goldf
   Point Forecast    Lo 95    Hi 95
37       5081.371 4767.741 5395.001
38       5138.743 4695.203 5582.283
39       5196.114 4652.891 5739.338
40       5253.486 4626.226 5880.746
41       5310.857 4609.559 6012.155
42       5368.229 4599.995 6136.462
43       5425.600 4595.813 6255.387
44       5482.971 4595.892 6370.051
45       5540.343 4599.453 6481.233
46       5597.714 4605.929 6589.500
47       5655.086 4614.892 6695.279
48       5712.457 4626.011 6798.904
49       5769.829 4639.019 6900.638
50       5827.200 4653.704 7000.696
51       5884.571 4669.887 7099.255
52       5941.943 4687.423 7196.463
53       5999.314 4706.184 7292.444
54       6056.686 4726.066 7387.305
55       6114.057 4746.975 7481.139
56       6171.429 4768.832 7574.025
57       6228.800 4791.566 7666.034
58       6286.171 4815.116 7757.227
59       6343.543 4839.426 7847.660
60       6400.914 4864.447 7937.382
> plot(goldf)
> accuracy(goldf)
                     ME     RMSE      MAE         MPE     MAPE      MASE
Training set 0.08218409 155.5098 116.6965 -0.1799051 2.960037 0.9286895
                    ACF1
Training set -0.07882193
```

## Case 2-GDP:

**Plot of gdp data**



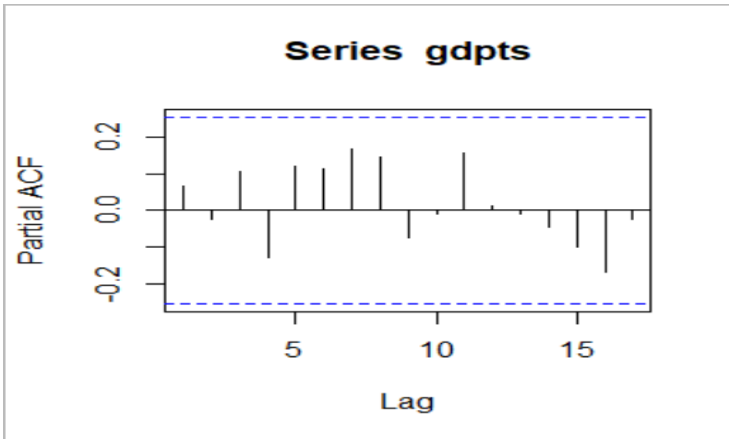

**ACF of gdp data**



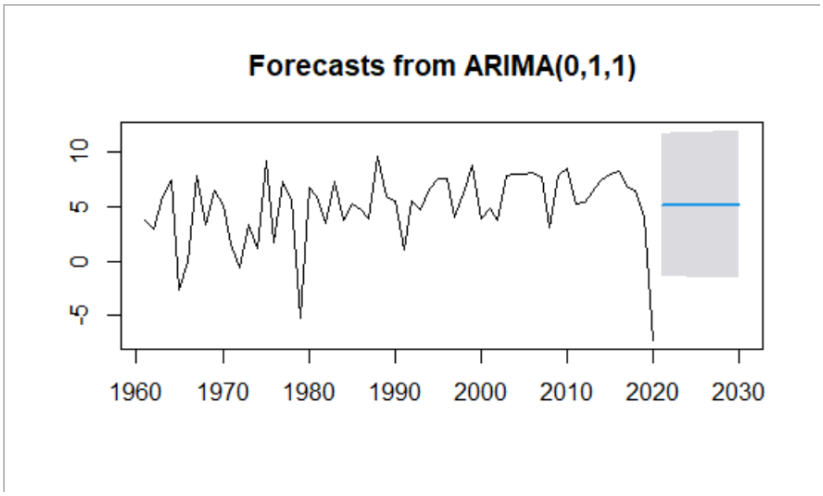
**NAME**: Penugonda Abhignanendra **REGNO**:19BCE1677
**FACULTY**: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

# Essential of Data Analytics
# Tasks for Week-2: Time-series Forecasting

## Partial ACF of gdp data




## Forecast of gdp data




## Forecasted value for gdp data

**NAME**: Penugonda Abhignanendra **REGNO**:19BCE1677
**FACULTY**: Dr. Lakshmi Pathi Jakkamputi **SLOT**: L21, L22

# Essential of Data Analytics
# Tasks for Week-2: Time-series Forecasting

```
> gdpf=forecast(gdpmodel,level=c(95),h=10)
> gdpf
     Point Forecast     Lo 95    Hi 95
2021       5.177274 -1.376684 11.73123
2022       5.177274 -1.401989 11.75654
2023       5.177274 -1.427197 11.78174
2024       5.177274 -1.452309 11.80686
2025       5.177274 -1.477327 11.83187
2026       5.177274 -1.502250 11.85680
2027       5.177274 -1.527082 11.88163
2028       5.177274 -1.551821 11.90637
2029       5.177274 -1.576470 11.93102
2030       5.177274 -1.601029 11.95558
> plot(gdpf)
> accuracy(gdpf)
                     ME     RMSE      MAE      MPE     MAPE      MASE
Training set 0.2704179 3.287709 2.345416 121.6616 161.0542 0.7720211
                     ACF1
Training set -0.02667223
>
```

## Program:

## Case 1 Gold:

```
#gold forecasting
setwd("C:/Abhi notes/class3-2/eda/lab")
gold<-read.csv("gold.csv")
library(forecast)
library(tseries)
View(gold)
goldts<-ts(gold$Price,start=min(gold$Month),end=max(gold$Month),frequency = 1)
class(goldts)
plot(goldts)
acf(goldts)
pacf(goldts)
adf.test(goldts)
goldmodel=auto.arima(goldts,ic="aic",trace=TRUE)
goldf=forecast(goldmodel,level=c(95),h=24)
goldf
plot(goldf)
```

# Essential of Data Analytics
# Tasks for Week-2: Time-series Forecasting

```
accuracy(goldf)
```

## case 2 GDP:

```
#gdp forecasting
setwd("C:/Abhi notes/class3-2/eda/lab")
gdp<-read.csv("gdp.csv")
library(forecast)
library(tseries)
gdpts<-ts(gdp$GDP_gr,start=min(gdp$Year),end=max(gdp$Year),frequency = 1)
class(gdpts)
plot(gdpts)
acf(gdpts)
pacf(gdpts)
adf.test(gdpts)
gdpmodel=auto.arima(gdpts,ic="aic",trace=TRUE)
gdpf=forecast(gdpmodel,level=c(95),h=10)
gdpf
plot(gdpf)
accuracy(gdpf)
```