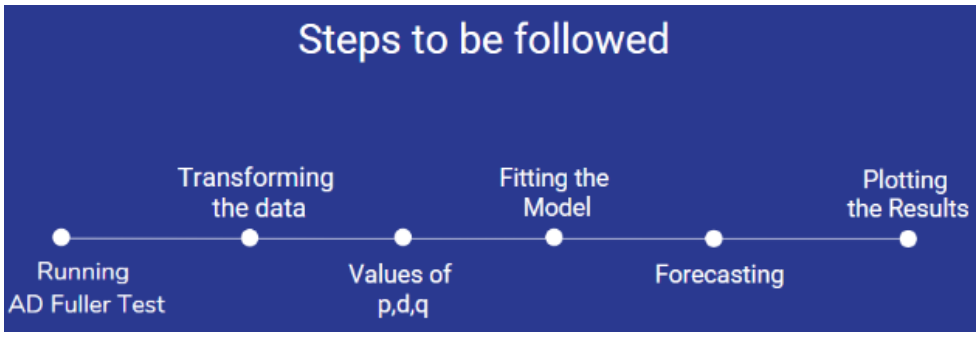
Time Series Forecasting using ARIMA model

A Brief about the ARIMA Model

ARIMA, short for '**Auto Regressive Integrated Moving Average'** is actually a class of models that 'explains' a given time series based on its own past values, that is, its own lags and the lagged forecast errors, so that equation can be used to forecast future values. Any 'non-seasonal' time series that exhibits patterns and is not a random white noise can be modeled with ARIMA models.

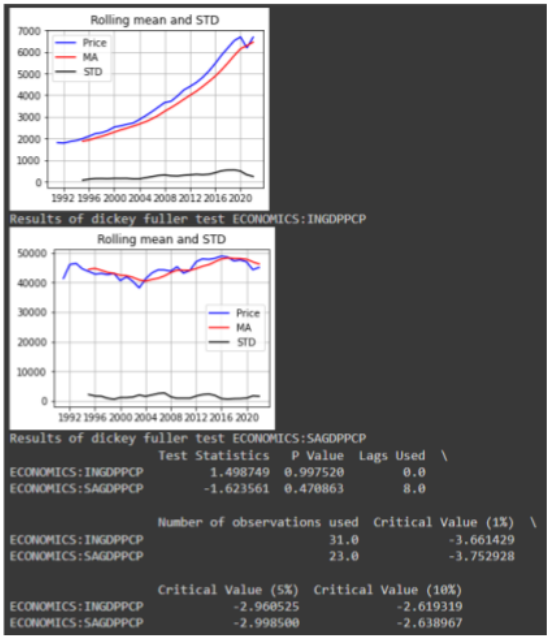


Running Ad Fuller Test

Ad Fuller Test

Augmented Dickey Fuller test (ADF Test) is a common statistical test used to test whether a given Time series is stationary or not.

```
#Perform Dickey fuller test
print('Results of dickey fuller test {name}'.format
(name = ticker[i]))
dftest = adfuller(hm, autolag = 'AIC')
dfoutput = pd.Series(dftest[0:4], index =
['Test Statistics', 'P Value', 'Lags Used',
'Number of observations used'])
for key,value in dftest[4].items():
dfoutput['Critical Value (%s)'%key] = value
```

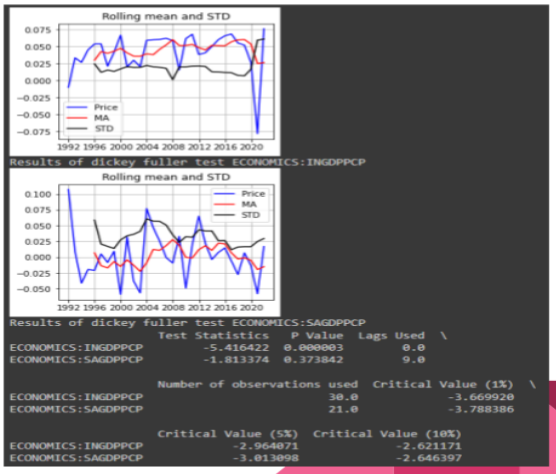


Transforming the data

As we can see in the slide before the P value is significantly high which denotes that the data is not stationary.

To make the data stationary we have to transform the data in such a way that the moving average and standard deviation becomes linear or close to linear. For this we have taken the logarithm of the data set.

By doing this the P value decreases drastically and the values comes closer to 0, this means that our data is now stationary and now the model can be fitted.



Fitting the model

```
from statsmodels.tsa.arima_model import ARIMA
for i in range(len(ticker)):
    p = int(Final_ACF[i])
    q = int(Final_PACF[i])
    d = int(lag)
    isd = dflogscale.iloc[:,i].to_frame()
    model = ARIMA(isd, order = (p, d, q))
    results_AR = model.fit()
    e = int(Forecast_Years)
    tf = results_AR.forecast(e)
    tf = np.exp(tf[0])
    hsd.append(tf)
```

```
                             ARIMA Model Results
==============================================================================
Dep. Variable:     D.ECONOMICS:INGDPPCP   No. Observations:                   31
Model:                  ARIMA(1, 1, 1)   Log Likelihood                66.181
Method:                        css-mle   S.D. of innovations              0.028
Date:                 Wed, 12 Oct 2022   AIC                           -124.362
Time:                         15:20:54   BIC                           -118.626
Sample:                     12-31-1991   HQIC                          -122.492
                          - 12-31-2021
==============================================================================
                                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const                          0.0441      0.002     17.823      0.000       0.039       0.049
ar.L1.D.ECONOMICS:INGDPPCP     0.8226      0.149      5.537      0.000       0.531       1.114
ma.L1.D.ECONOMICS:INGDPPCP    -1.0000      0.087    -11.446      0.000      -1.171      -0.829
                                    Roots
=============================================================================
                  Real          Imaginary           Modulus         Frequency
-----------------------------------------------------------------------------
AR.1            1.2156           +0.0000j            1.2156            0.0000
MA.1            1.0000           +0.0000j            1.0000            0.0000
-----------------------------------------------------------------------------
```

The model summary reveals a lot of information. The table in the middle is the coefficients table where the values under 'coef' are the weights of the respective terms.

```
print(results_AR.summary())
```

Forecasting and Plotting