STOCK PRICE FORECASTING

By:

Kalaivani. T

INTRODUCTION:

- Stock market forecasting is a major exertion in the field of finance and establishing business.
- Stock market forecasters focus on developing a successful approach for forecasting or predicting index value of stock prices.
- Forecasting stock indices is very difficult because the market indices are highly fluctuating as result of increase or decrease that characterize the stock price.
- Autoregressive integrated moving average (ARIMA) model has remained the mostly widely used time series model for forecasting stock market .
- To build the stock price forecasting model, we will using real time TCS dataset.

PROBLEM STAEMENT:

In this project, we will use time series forecasting to predict the price of TCS stock for the next few years. We will use historical data of TCS stock prices, and a variety of time series forecasting models. We will then evaluate the accuracy of our forecasts, and discuss the implications of our findings for investors depends on.

The overall economic conditions

The company's performance

Exploratory Data Analysis:

The output of the **df.info()** method can be helpful for understanding the structure of the Data Frame and the data types of the columns.

This information can be useful for performing data analysis and visualization.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2942 entries, 0 to 2941
Data columns (total 14 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Date            2942 non-null   object 
 1   series          2942 non-null   object 
 2   OPEN            2942 non-null   float64
 3   HIGH            2942 non-null   float64
 4   LOW             2942 non-null   float64
 5   PREV. CLOSE     2942 non-null   float64
 6   ltp             2942 non-null   float64
 7   close           2942 non-null   float64
 8   vwap            2942 non-null   float64
 9   52W H           2942 non-null   float64
 10  52W L           2942 non-null   float64
 11  VOLUME          2942 non-null   float64
 12  VALUE           2942 non-null   float64
 13  No of trades    2942 non-null   float64
dtypes: float64(12), object(2)
memory usage: 321.9+ KB
```

Extract Required column:

- **The close prices variable will now contain a Series object with the close prices.**

- **The Series object will have the same index as the Data Frame.**

```
#extract close column for prediction and analysis

df1 = df['close ']
df1 = pd.DataFrame(df1)
```

```
df1
```

Date	close
2011-06-01	1175.40
2011-06-02	1167.75
2011-06-03	1152.10
2011-06-06	1164.35
2011-06-07	1179.30

Line Graph Analysis

- Identify trend
- Identify patterns

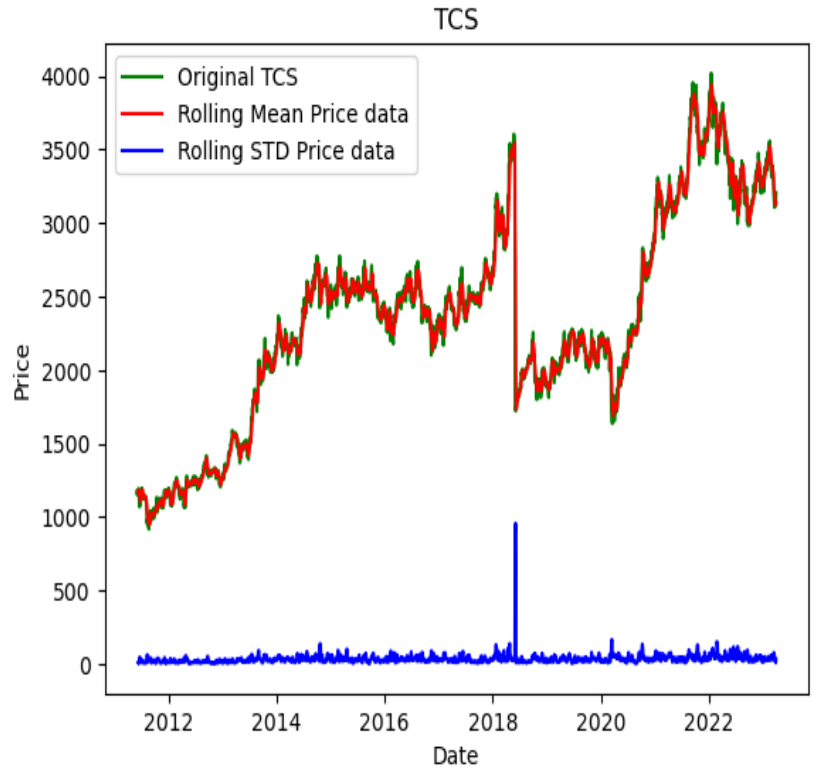
```
#plot line graph of price vs Date

plt.plot(df1,color='green',label='PRICE')
plt.title('TCS')
plt.legend()
plt.xlabel('Date')
plt.ylabel('Price')
plt.show()
```

TREND ANALYSIS:

The stability of the model over time. A common time-series model assumption is that the coefficients are constant with respect to time. Checking for instability amounts to examining whether the coefficients are time-invariant.

The forecast accuracy of the model .

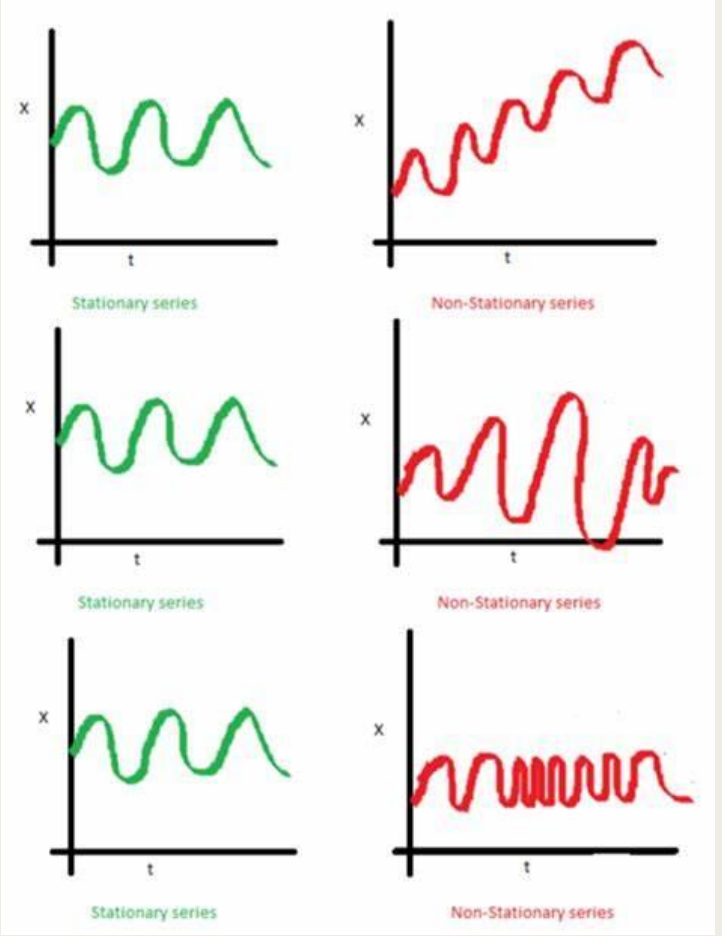
```
#CREATE THE LINE GRAPH CONTANING Original price, Rolling Mean Price data,Rolling STD Price data


plt.plot(df1,color='green',label='Original TCS')

plt.plot(r_mean,color='red',label='Rolling Mean Price data')

plt.plot(r_std,color='blue',label='Rolling STD Price data')

plt.title('TCS')
plt.legend(loc='best')
plt.xlabel('Date')
plt.ylabel('Price')
plt.show()
```



ARIMA MODEL:

Automated Regressive Integrated Moving Average.

- It 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.
- An ARIMA model is characterized by 3 terms: p, d, q.
- p is the order of the AR term
- q is the order of the MA term
- d is the number of differencing required to make the time series stationary



ADF:

```
from statsmodels.tsa.stattools import adfuller

ad_ft=ad fuller(df1,autolag='AIC’)
```

The provided code appears to be performing an

Augmented Dickey Fuller (ADF) test on a time series data.

The ADF test is commonly used to

Determine if a time series is stationary or not.

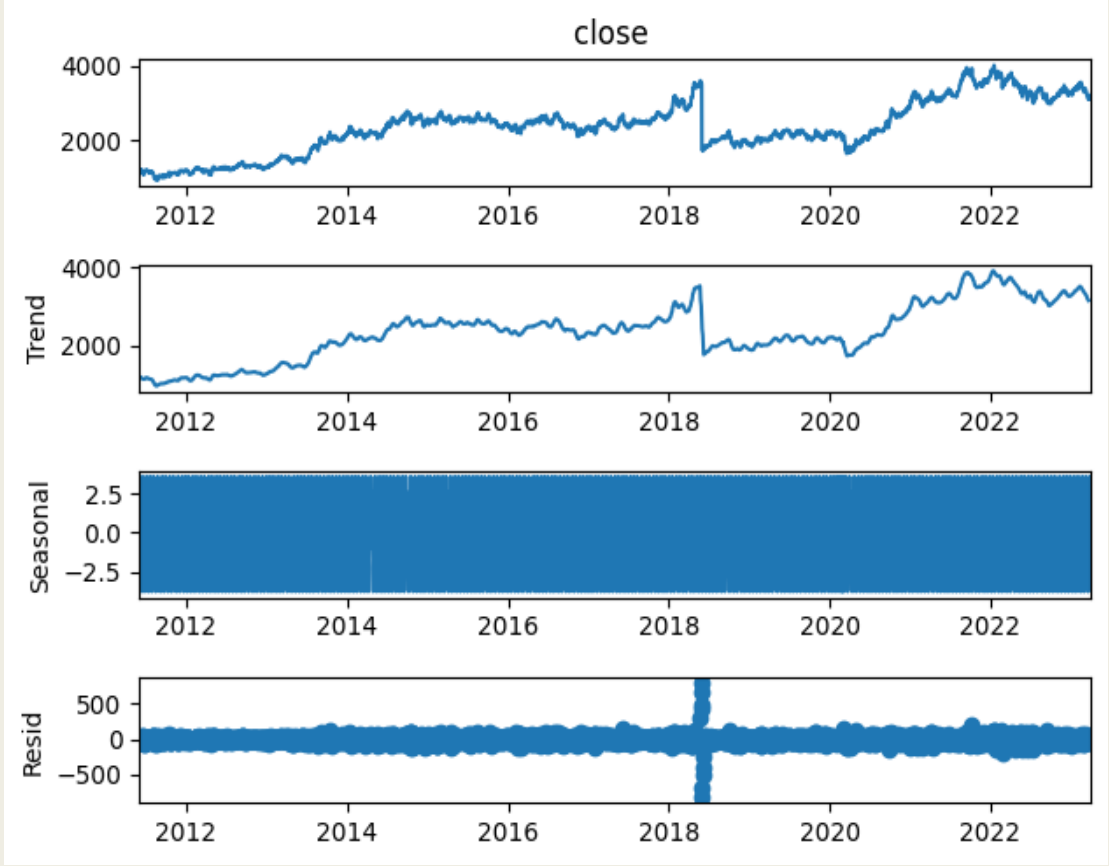
s.no	values	metrics
0	-2.058209	test statistics
1	0.261648	P Values
2	0.000000	No. of Lags Used
3	2941.000000	No. of Observation used
4	-3.432575	Critical Value 1%
5	-2.862523	Critical Value 5%
6	-2.567293	Critical Value 10%

Seasonal decompose:

- `from statsmodels.tsa.seasonal import seasonal_decompose`
- `decompose_df=seasonal_decompose(df1['close'],model='additive',period=12)`
- `decompose_df.plot()`
- `plt.show()`

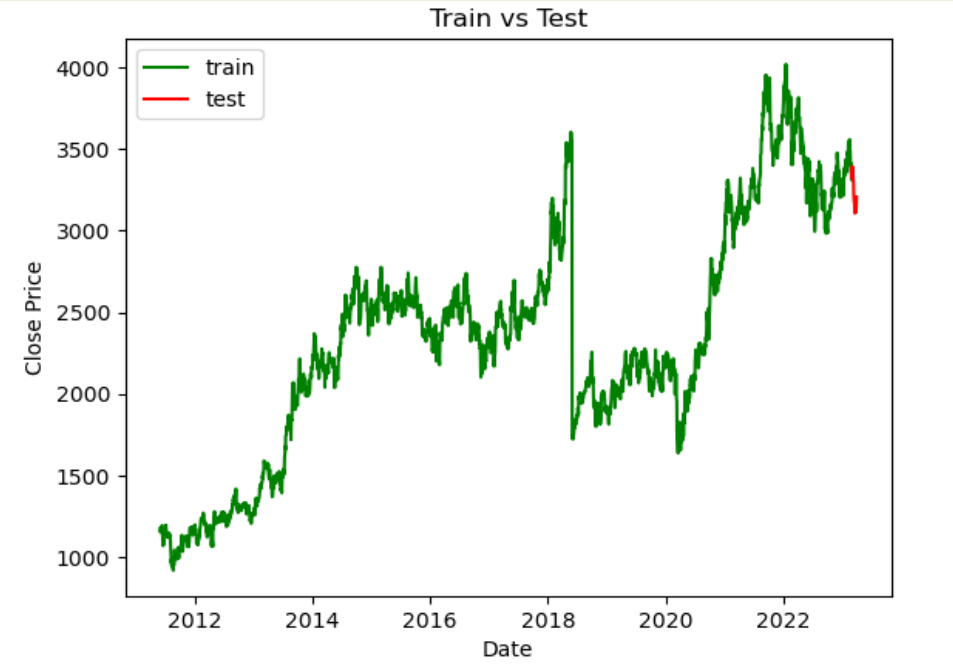
POLYNOMIAL REGRESSION:

- Polynomial regression is a technique we cause to fit a regression model when the relationship between the predictor variable(s) and the response variable is nonlinear.

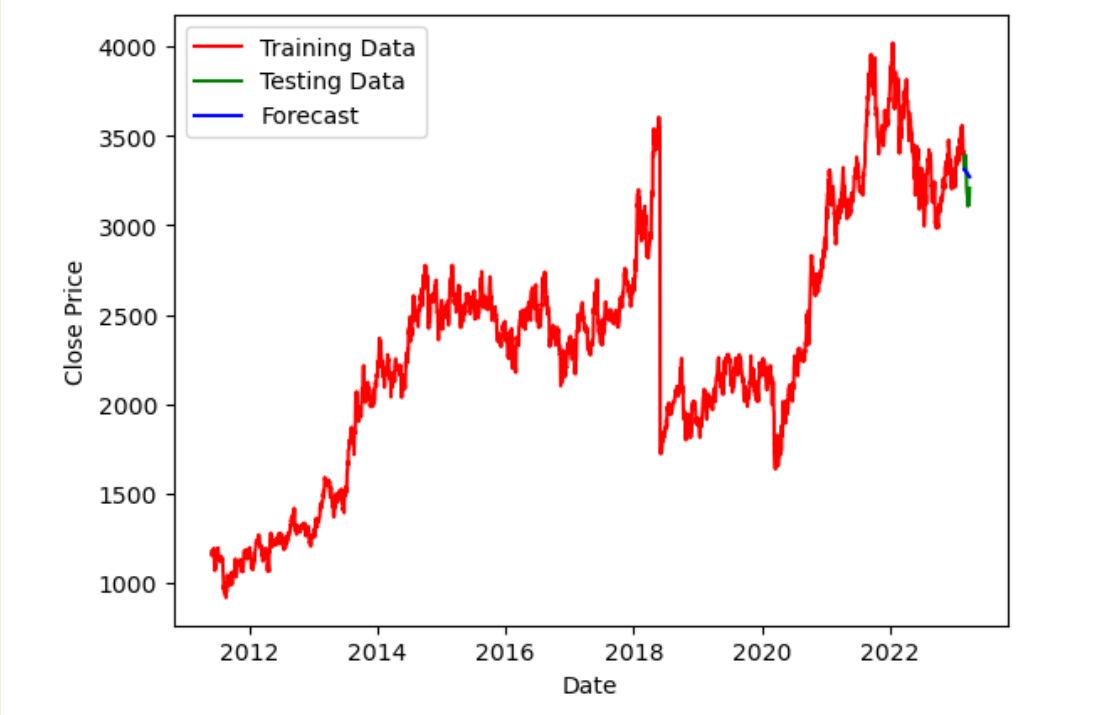


GRAPHS ANALYSIS:

- TRAIN VS TEST



- FORECASTING



INFERENCES:

- **We have drawn some inferences from the data frame. Here is a summary of a few of them:**

Today we are all connected with digital and every In all seasons, supermarkets should be aware of customer demand and avoid stockouts. This project focus on predicting the future demand of sales using various machine learning algorithms.

- **Analyzing the past sales of the store**
- **Understanding the factors that affect the sales of a store**
- **Deriving inferences related to those sales**
- **Predicting the future sales from the inferences derived**
- **Help the businesses stock up/stock down products accordingly**

THANK YOU