FORECASTING CONSUMER PRICE INDEX OF UNITED STATES USING TIME SERIES MODEL

INDEX:

ACKNOWLEDGEMENT:

I would like to express my special thanks of gratitude to my mentor, Prof. Sujan Chandra Sir for his active guidance and support in completing my Project.

I would also like to extend my gratitude to our Head of the Department, Dr. Ajoy Kumar Biswas Sir and my other Professors for providing me with all the facility that was required for completing the project.

EXECUTIVE SUMMARY:

The objective of this project is to use the monthly Consumer Price Index (CPI) data of United States from January 2010 to May 2022 and forecast next 12 values using methods of time series analysis. The data is found to have the trend component in it, which is being analysed. From the visual representations of the data it is evident that the rate of rise in trend of the CPI values has a significant change from mid of 2020. Thus, a piece-wise timeseries model is considered for analysing the data. The model ARIMA(1,2,2) and Holts' Linear Trend Model gives a good fit for the data at two different time intervals. The ARIMA(1,,2,2) fits the efficiently from January ,2010 to mid of 2020 and Holts Linear Trend Model fits efficiently the second part of the data, from mid of 2020 to May, 2022. The forecast can be used can be used by the governing bodies to take initiatives about different policies, to maintain efficient working of government. The data suggests that CPI is highly influenced by the covid19 pandemic which is concerning and must be taken care of by the government. It can be concluded that, if the current rise in CPI due to the pandemic is not controlled, the CPI values would increase in a higher rate and the forecasts given by the Holts' Linear Trend Model will be an approximate match.

DATA:

The Consumer Price Index data of the United States from the January 2010 to May 2022 is:

DATE	DATA
1/1/2010	216.687
2/1/2010	216.741
3/1/2010	217.631
4/1/2010	218.009
5/1/2010	218.178
6/1/2010	217.965
7/1/2010	218.011
.	.
.	.
.	.
.	.
.	.
.	.
.	
11/1/2021	277.948
12/1/2021	278.802
1/1/2022	281.148
2/1/2022	283.716
3/1/2022	287.504
4/1/2022	289.109

*Here the date format is: month/date/year

There are 148 data point in total, which shall be used to forecast the next 12 values.

For the complete dataset, click on the following link:

https://www.kaggle.com/datasets

INTRODUCTION:

The economy of the United States is a highly developed mixed economy. It is the world's largest economy by nominal GDP, and the second-largest by purchasing power parity (PPP) behind China. It has the world's seventh-highest per capita GDP (nominal) and the eighth-highest per capita GDP (PPP) as of 2022.

The American economy is fuelled by high productivity, transportation infrastructure, and extensive natural resources. Americans have the highest average household and employee income among OECD member states. In 2021, they had the highest median household income. The U.S. has one of the world's highest income inequalities among the developed countries. The largest U.S. trading partners are Canada, Mexico, China, Japan, Germany, South Korea, the United Kingdom, Taiwan, India, and Vietnam. The U.S. is the world's largest importer and second-largest exporter. It has free trade agreements with several countries, including the USMCA, Australia, South Korea, Israel, and several others that are in effect or under negotiation.

CPI stands for Consumer Price Index, is defined as, a measure of the average change overtime in the prices paid by urban consumers for a market basket of consumer goods and services.

1. The CPI affects nearly all Americans because of the many ways it is used. Some examples of how it is used follow:
   - *As an economic indicator*. The CPI is the most widely used measure of inflation and is sometimes viewed as an indicator of the effectiveness of government economic policy. It provides information about price changes in the Nation's economy to government, business, labour, and private citizens and is used by them as a guide to making economic decisions. In addition, the President, Congress, and the Federal Reserve Board use trends in the CPI to aid in formulating fiscal and monetary policies.
   - *As a deflator of other economic series*. The CPI and its components are used to adjust other economic series for price changes and to translate these series into inflation-free dollars. Examples of series adjusted by the CPI include retail sales, hourly and weekly earnings, and components of the National Income and Product Accounts.

The CPI is also used as a deflator of the value of the consumer's dollar to find its purchasing power. The purchasing power of the consumer's dollar

measures the change in the value to the consumer of goods and services that a dollar will buy at different dates. In other words, as prices increase, the purchasing power of the consumer's dollar declines.

- *As a means of adjusting dollar values*. The CPI is often used to adjust consumers' income payments (for example, Social Security), to adjust income eligibility levels for government assistance, and to automatically provide cost-of-living wage adjustments to millions of American workers.

Another example of how dollar values may be adjusted is the use of the CPI to adjust the Federal income tax structure. These adjustments prevent inflation-induced increases in tax rates. In addition, eligibility criteria for millions of food stamp recipients, and children who eat lunch at school, are affected by changes in the CPI. Many collective bargaining agreements also tie wage increases to the CPI.

The CPI represents all goods and services purchased for consumption by the reference population (U or W). BLS has classified all expenditure items into more than 200 categories, arranged into eight major groups (**food and beverages, housing, apparel, transportation, medical care, recreation, education and communication,** and **other goods and services**). Included within these major groups are various government-charged user fees, such as water and sewerage charges, auto registration fees, and vehicle tolls.
In addition, the CPI includes taxes (such as sales and excise taxes) that are directly associated with the prices of specific goods and services. However, the CPI excludes taxes (such as income and Social Security taxes) not directly associated with the purchase of consumer goods and services. The CPI also does not include investment items, such as stocks, bonds, real estate, and life insurance because these items relate to savings, and not to day-to-day consumption expenses.

Objective:

In this project our objective is to use the past Consumer Price Index values and forecast the next 12 months value using different methods of time series analysis. This forecast can be used can be used by governing bodies to plan and take initiatives about different governing policies, to maintain efficient working of government. Besides that, CPI index is very useful for individuals who are interested in stock marketing and making of investment plans.

METHODOLOGY:

Time series analysis is a specific way of analysing a sequence of data points collected over an interval of time. In time series analysis, analysts record data points at consistent intervals over a set period of time rather than just recording the data points intermittently or randomly. However, this type of analysis is not merely the act of collecting data over time.

Time is a crucial variable because it shows how the data adjusts over the course of the data points as well as the final results. It provides an additional source of information and a set order of dependencies between the data.

Time series analysis typically requires a large number of data points to ensure consistency and reliability. Time series data can be used for forecasting—predicting future data based on historical data.

Models of time series analysis include:

- **Classification:** Identifies and assigns categories to the data.
- **Curve fitting:** Plots the data along a curve to study the relationships of variables within the data.
- **Descriptive analysis:** Identifies patterns in time series data, like trends, cycles, or seasonal variation.
- **Explanative analysis:** Attempts to understand the data and the relationships within it, as well as cause and effect.
- **Exploratory analysis:** Highlights the main characteristics of the time series data, usually in a visual format.
- **Forecasting:** Predicts future data. This type is based on historical trends. It uses the historical data as a model for future data, predicting scenarios that could happen along future plot points.
- **Intervention analysis:** Studies how an event can change the data.
- **Segmentation:** Splits the data into segments to show the underlying properties of the source information.

Under classification of data:

By Stationary time series it is meant that, the one whose properties do not depend on the time at which the series is observed. Thus, time series with trends, or with seasonality, are not stationary — the trend and seasonality will affect the value of the time series at different times. On the other hand, a white noise series is stationary — it does not matter when you observe it, it should look much the same at any point in time.

The stationarity of the data is checked by using **augmented Dickey–Fuller test** (**ADF**).

The hypotheses for the test:

- The null hypothesis for this test is that there is a unit root (non-Stationary).
- The alternate hypothesis differs slightly according to which equation you're using. The basic alternate is that the time series is stationary.

A key point to remember here is: Since the null hypothesis assumes the presence of unit root, that is $\alpha=1$, the p-value obtained should be less than the significance level (say 0.05) in order to reject the null hypothesis. Thereby inferring that the series is stationary.

The p-value obtained is greater than the significance level of 0.05, and the ADF statistic is higher than any of the critical values. Clearly, there is no reason to reject the null hypothesis. So, the data is non-stationary, otherwise, stationary.

If the data is found to be non-stationary, analyse all the variations present in the data.

In time series data, variations can occur sporadically throughout the data:

- **Functional analysis** can pick out the patterns and relationships within the data to identify notable events.
- **Trend analysis** means determining consistent movement in a certain direction. There are two types of trends: deterministic, where the underlying cause is found, and stochastic, which is random and unexplainable.
- **Seasonal variation** describes events that occur at specific and regular intervals during the course of a year. Serial dependence occurs when data points close together in time tend to be related.
- **Irregularities** describes the irregular components in the data.

Transforming the data to Stationary:

A stationary time series is one whose properties do not depend on the time at which the series is observed. Thus, time series with trends, or with seasonality, are not stationary — the trend and seasonality will affect the value of the time series at different times. On the other hand, a white noise series is stationary — it does not matter when you observe it, it should look much the same at any point in time.

Some cases can be confusing — a time series with cyclic behaviour (but with no trend or seasonality) is stationary. This is because the cycles are not of a fixed length, so before observing the series, it cannot be concluded where the peaks and troughs of the cycles will be.

In general, a stationary time series will have no predictable patterns in the long-term. Time plots will show the series to be roughly horizontal (although some cyclic behaviour is possible), with constant variance.

to make a non-stationary time series stationary — compute the differences between consecutive observations. This is known as **differencing**.

Transformations such as logarithms can help to stabilise the variance of a time series. Differencing can help stabilise the mean of a time series by removing changes in the level of a time series, and therefore eliminating (or reducing) trend and seasonality.

As well as looking at the time plot of the data, the ACF plot is also useful for identifying non-stationary time series. For a stationary time series, the ACF will drop to zero relatively quickly, while the ACF of non-stationary data decreases slowly.
The differenced series is the *change* between consecutive observations in the original series, and can be written as

$$y'_t = y_t - y_{t-1}$$

The differenced series will have only T−1 values, since it is not possible to calculate a difference y′1 for the first observation.

Occasionally the differenced data will not appear to be stationary and it may be necessary to difference the data a second time to obtain a stationary series:

$$y''_t = y'_t - y'_{t-1}$$

In this case, y''_t will have T−2 values. Then, modelling the "change in the changes" of the original data. In practice, it is almost never necessary to go beyond second-order differences.

Next the ACF and PACF are plotted:

Autocorrelation analysis is an important step in the Exploratory Data Analysis of time series forecasting. The autocorrelation analysis helps detect patterns and check for randomness.

Autocorrelation function (ACF) and **Partial Autocorrelation Function (PACF, also called Partial ACF)** are important functions in analysing a time series. They generally produce plots that are very important in finding the values **p**, **q** and **r** for Autoregressive (AR) and Moving Average (MA) models.

- An ACF measures and plots the average correlation between data points in time series and previous values of the series measured for different lag lengths.
- A PACF is similar to an ACF except that each partial correlation controls for any correlation between observations of a shorter lag length.

Identification of Model and Forecasting:

Model Used:

- **AUTO REGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) MODEL**
- ***HOLT'S LINEAR TREND MODEL***

➢ ARIMA

The ARIMA model predicts a given time series based on its own past values. It can be used for any nonseasonal series of numbers that exhibits patterns and is not a series of random events. For example, sales data from a clothing store would be a time series because it was collected over a period of time. One of the key characteristics is the data is collected over a series of constant, regular intervals. A modified version can be created to model predictions over multiple seasons.

For a period of multiple seasons, the data must be corrected to account for differences between the seasons. For example, holidays fall on different days of the year, causing a seasonal effect to the data. Sales may be artificially higher or lower depending on where the holiday falls in the calendar. The data scientist must be able to seasonally adjust the data to provide an accurate prediction for future sales.

The Autocorrelation Function (ACF) is used to determine the number of MA(q) terms in the model. It determines the correlation between the observations at the current point in time and all previous points in time. The Partial Autocorrelation Function (PACF) results determine the order of the model or the values for the MA portion of the model. The model order reflects how many times differencing must be used to transform a time series into a stationary series. The ACF and PACF plots are used to check residual time errors in the series.

➢ HOLT'S

Holt's linear trend method (also known as double exponential smoothing), which like its name suggests, adds a (linear) trend component to the simple exponential smoothing model.

The trend equation is computed from the step per step change in the level component. Additionally, from the overall equation, the trend component is now being multiplied by the time step, $\boldsymbol{h}$, therefore the forecasts are no longer flat but are a linear function of $\boldsymbol{h}$. Hence, the model's name is *Holt's linear trend method*.

Piecewise Trend Approximation:

Given a time series Y=$\{(y_1, t_1),(y_2, t_2),\ldots,(y_n, t_n)\}$, where y_i is a real numeric value and t_i is the timestamp, it can be represented as a PTA representation

$$T' = \{(R_1, R_{t_1}), \ldots, (R_m, R_{t_m})\}, m \leq n, n \in N,$$

where R_{t_i} is the right end point of the i th segment, R_i $(1 < i \leq m)$, is the ratio between $R_{t_{i-1}}$ and R_{t_i} in the i th segment, and R_i is the ratio between the first point t_i and R_{t_i}. The length of the i th segment can be calculated as $R_{t_i} - R_{t_{i-1}}$,

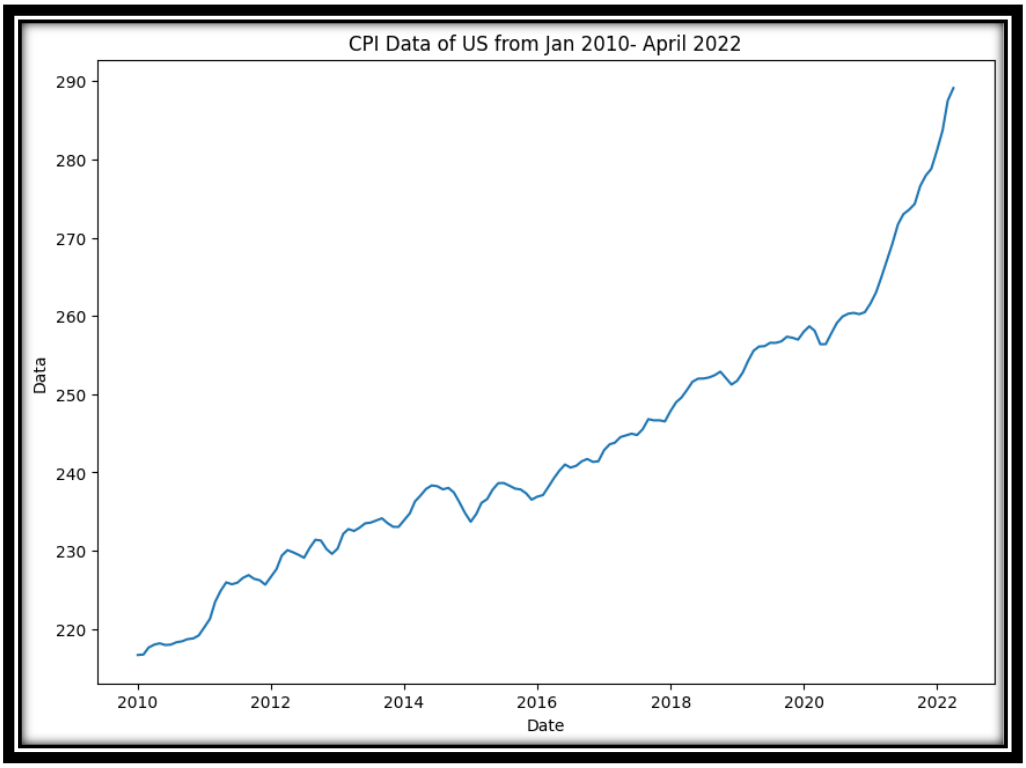
Piecewise Trend Approximation approximates a time series by applying a piecewise discontinuous function to reduce dimensionality. The algorithm of PTA consists of three main steps:

1. Local trend transformation: the original time series is transformed into a new series where the values of data points are ratios between any two consecutive data points in original series.
2. Segmentation: the transformed local trend series is divided into variable-length segments such that two conjunctive segments represent different trends.
3. Segment approximation: each segment is represented by the ratios between the first and last data points within the segment, which indicates the characteristic of trend.

RESULT ANALYSIS:

At first, the plot of the historical data is done, so that the data can be visualized.

The plot is shown in the following figure:



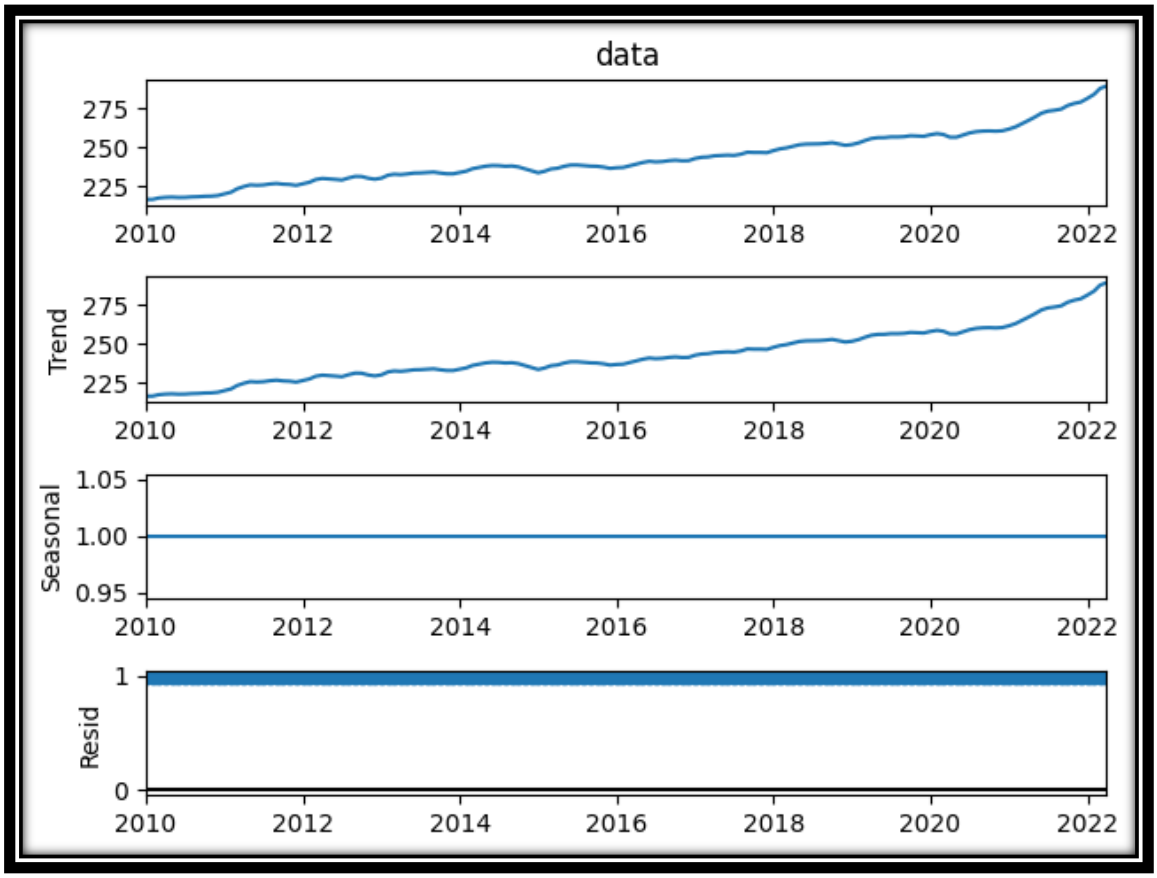
On visualizing the data, it can be said that the data might not be stationary and trend might be present in the data.

To confirm that, AD Fuller test is used and it is checked if the data is stationary or not.

ADF statistic	2.6548400104517 84
P-value	0.9990832963294357
Critical Values 1%,	-3.476273058920005
Critical Values 5%,	-2.881687616548444
Critical Values 10%,	-2.5775132580261593

The data is not Stationary, as here p-value > 0.05 which implies that the null hypothesis is accepted. Hence, it is inferred that the data is not stationary.

Hence, the data is further decomposed to get an idea of all the components present in the Data.



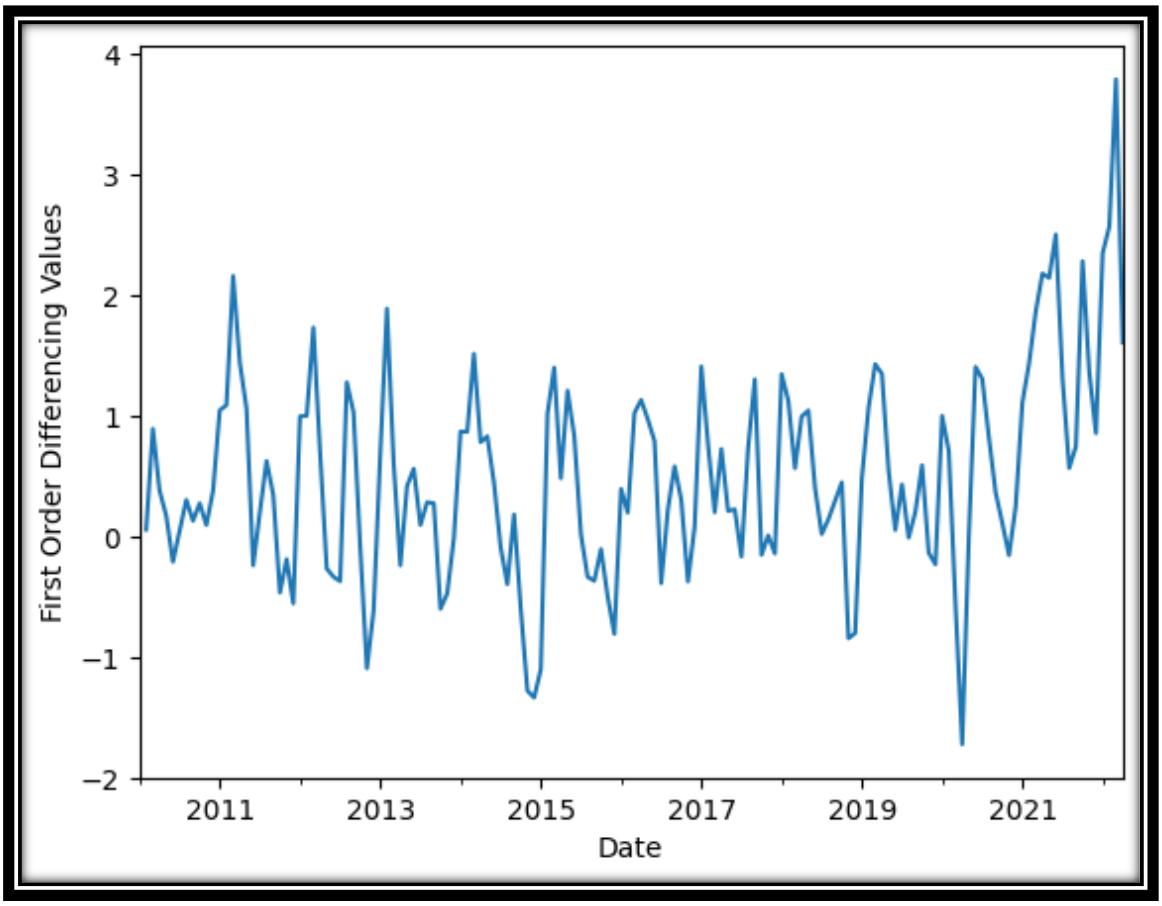
HENCE ONLY TREND COMPONENT IS PRESENT IN THE DATA

Now different transformations are made to transform the data into stationary.

➢ First, it is tried with first order differencing having lag 1.

Here differencing method is used, and try to make the data stationary.

Plotting the graph after differencing as,



By looking at the plot stationarity cannot be concluded, so test is done for the first order differencing data to check if it is stationary or not. Here AD Fuller test is used,

ADF statistic	0.12476873392700001
P-value	0.9677120466787718
Critical Values 1%,	-3.479742586699182
Critical Values 5%,	-2.88319822181578
Critical Values 10%,	-2.578319684499314

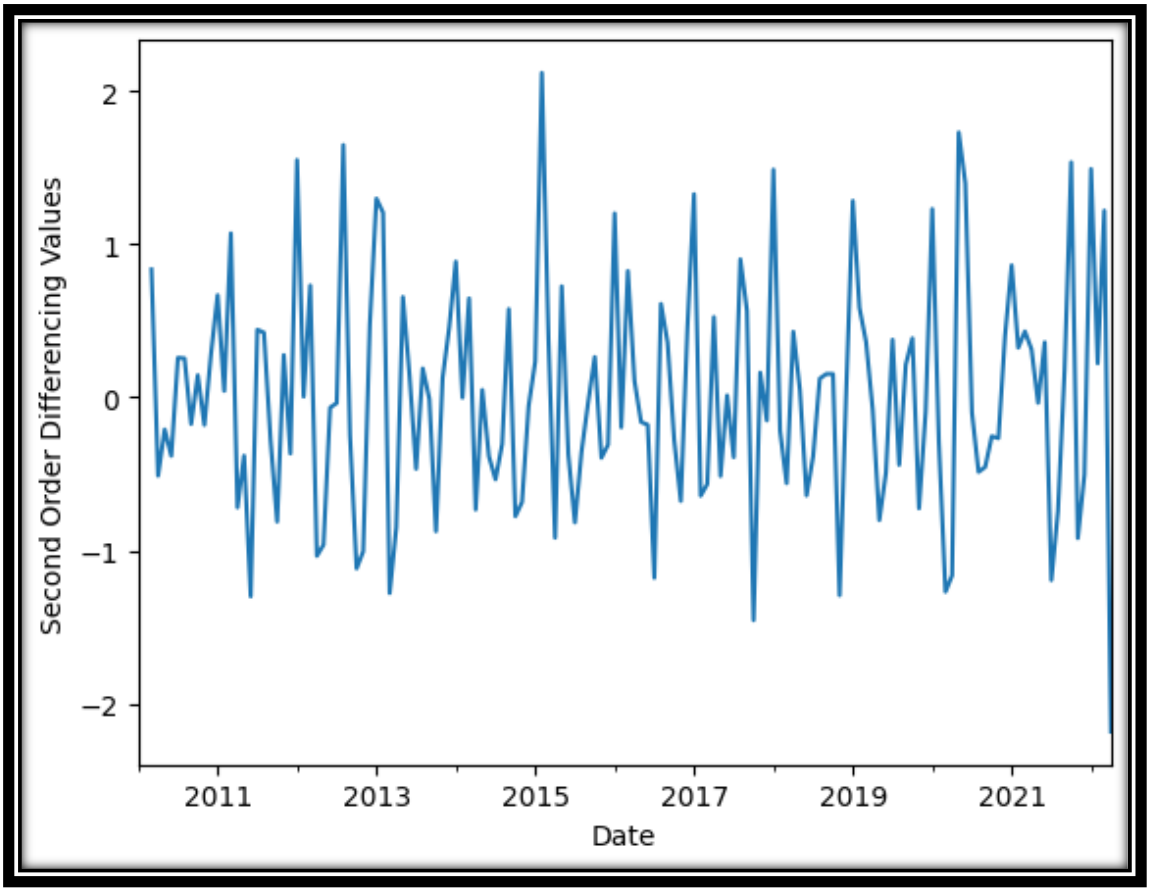
The First order differenced values are not Stationary.

Since the first order differencing is not stationary, Second order differencing with lag 1 is tried.
(After trying all other transformations, second order differencing is tried)

➢ Second Order Differencing with lag 1.

Here differencing method on the first order differenced values are used, and try to make the data stationary.

Plotting the second order differencing values as,



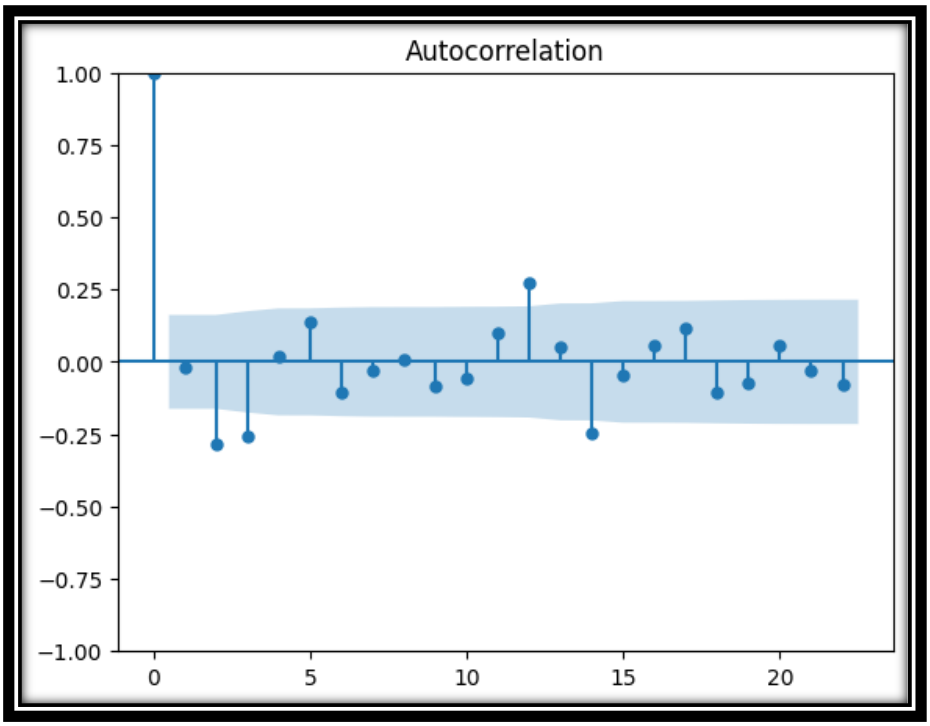
By looking at the plot, stationarity cannot be concluded, so testing is required to check if the second order differencing data is stationary or not. Here AD Fuller test is used,

ADF statistic	-7.472222353448878
P-value	5.019872371783548e-11
Critical Values 1%,	-3.479742586699182
Critical Values 5%,	-2.88319822181578
Critical Values 10%,	-2.578319684499314

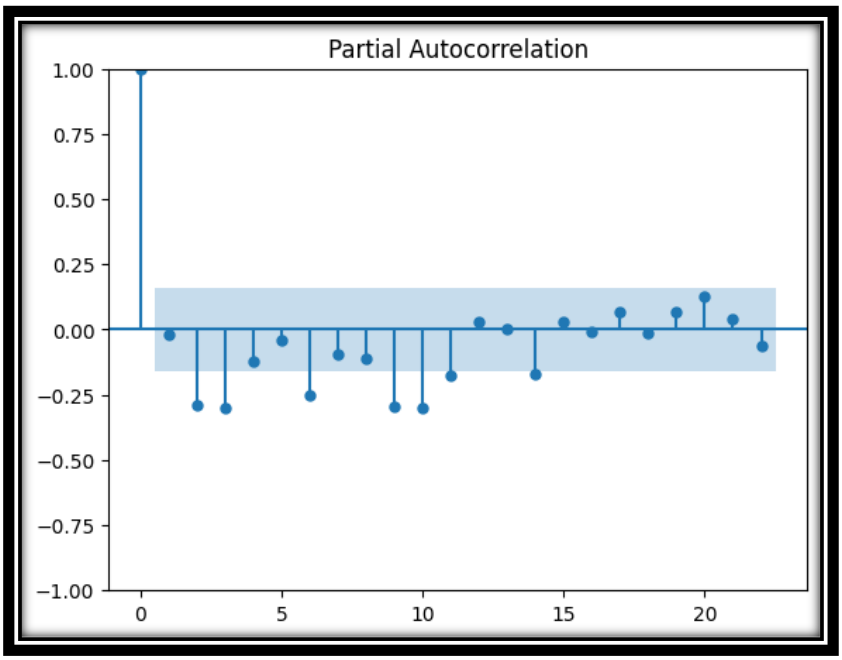
The Second order differenced values are Stationary.

Now, after making the data stationary, identification of the model is made. For doing so, plotting of the ACF and PACF is made.

➢ ACF plot:



➢ PACF plot:



From the ACF and PACF plot it can be seen that there are a number significant lag.

To come to a conclusion about model selection, find the AIC and BIC values for different ARIMA model.

The values are:

MODEL	AIC	BIC
ARIMA(1, 2, 1)	305.706027914469	314.65684777959405
ARIMA(1, 2, 2)	298.1966674620787	310.13109394891205
ARIMA(1, 2, 3)	297.53748411254446	312.45551722108615
ARIMA(2, 2, 1)	294.5728856837659	306.5073121705992
ARIMA(2, 2, 2)	296.4875226783099	311.4055557868516
ARIMA(2, 2, 3)	295.9401668619015	313.8418065921515
ARIMA(3, 2, 1)	296.4534080151691	311.3714411237108
ARIMA(3, 2, 2)	297.581930480473	315.483570210723
ARIMA(3, 2, 3)	296.0728060472721	316.95805239923044

Hence, from the values it is observed that, the minimum of AIC and BIC value is obtained in ARIMA(2,2,1) model.

Now, moving further to fit the model and predict values.

For fitting the model, split the data into two halves at first.
One for TRAINING and the other for TESTING.
Based on the training data the model is fitted and to check our model's efficiency the testing data is used.

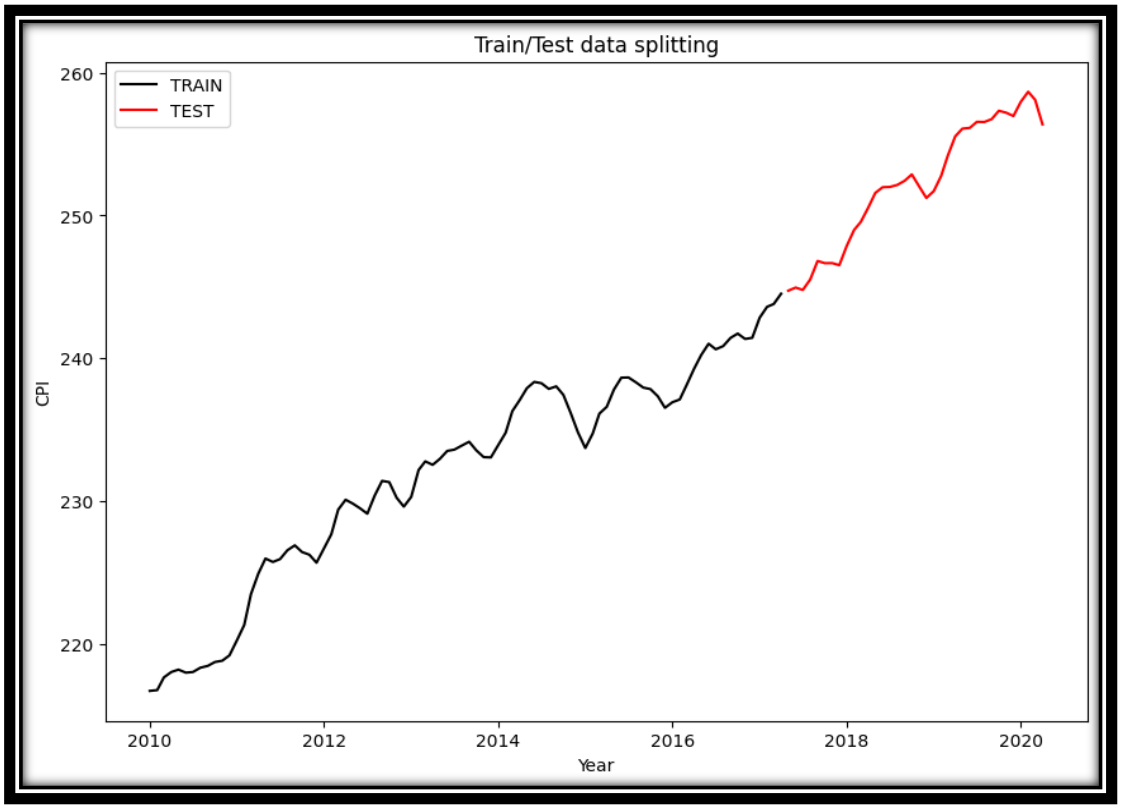
In this case, it is found that there is a sharp change in the slope of the trend values from mid of 2020. So, 'piece-wise time series' is preferred and at the very beginning the raw data is split into two halves:

- One half from January 2010 - April 2020
- Another half, from May 2020 - May 2022

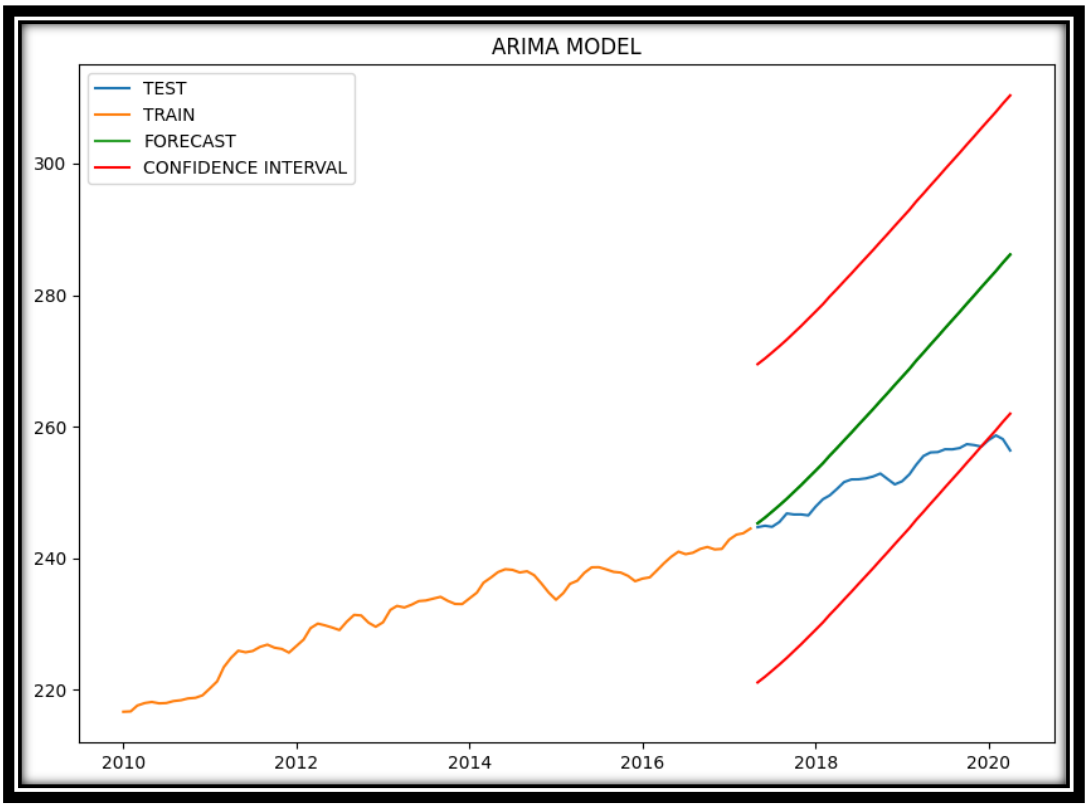
For the 2 different halves fitting of models is done differently.

➢ For the FIRST Half:

- ❖ Training and testing data is split as:



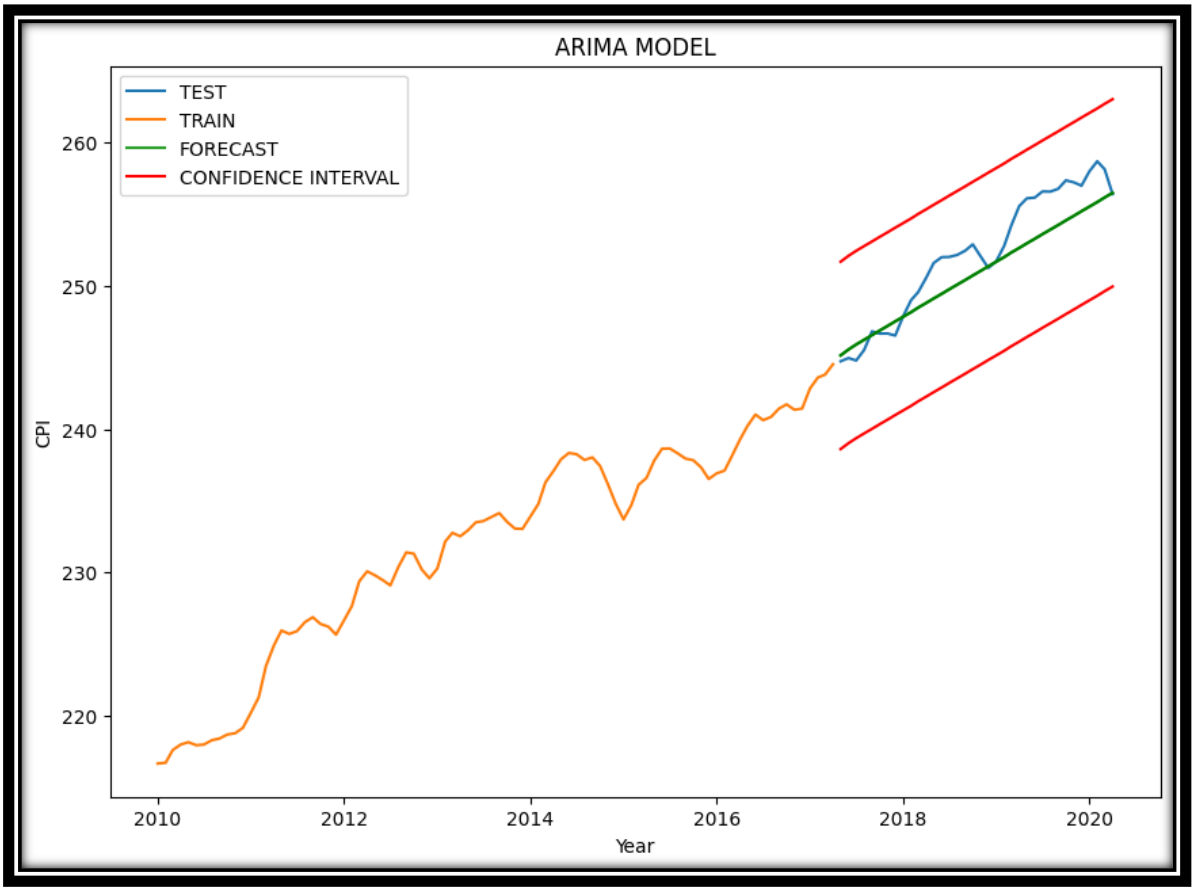
- ❖ Now as suggested by the AIC and BIC Values, fitting the APIMA(2,2,1) model to the data as:



But the error for this model is quite high,

Mean square Error- 15.099878470345644

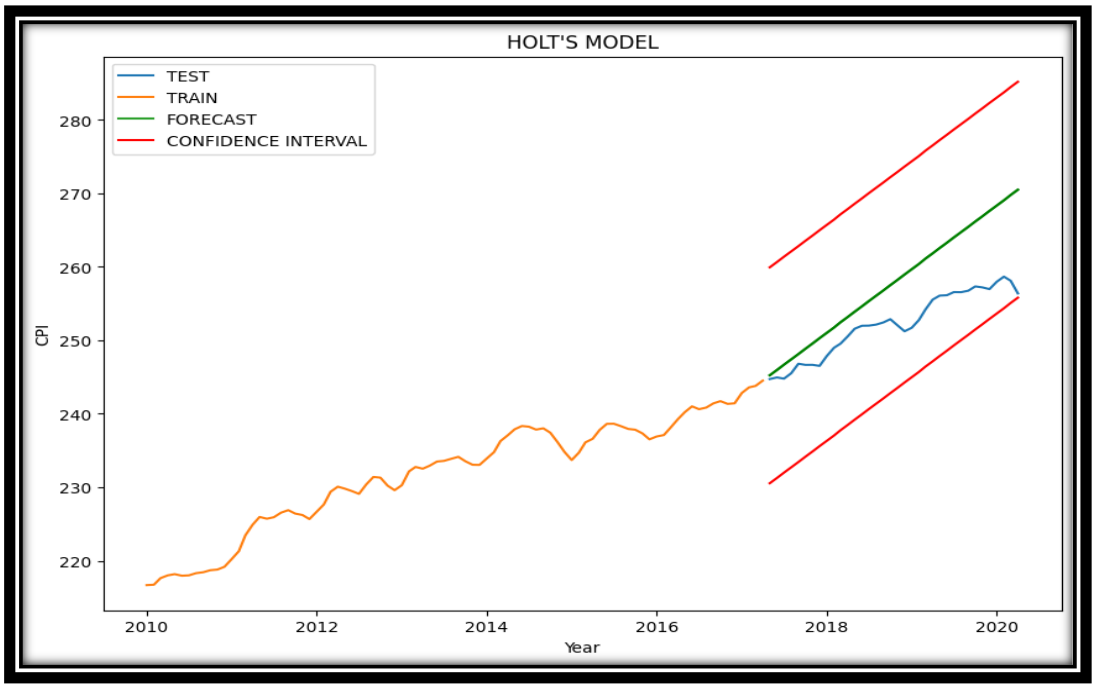
- So checking for next small BIC and AIC value, ARIMA(1,2,2) is found, and on fitting this model, following is observed:



Mean square error for ARIMA(1,2,2) model is 1.885805442705394

Now, the error is quite low, and thus this model is a good fit for this portion of data.

- Again, trying for the Holts' linear trend model for fitting the data,



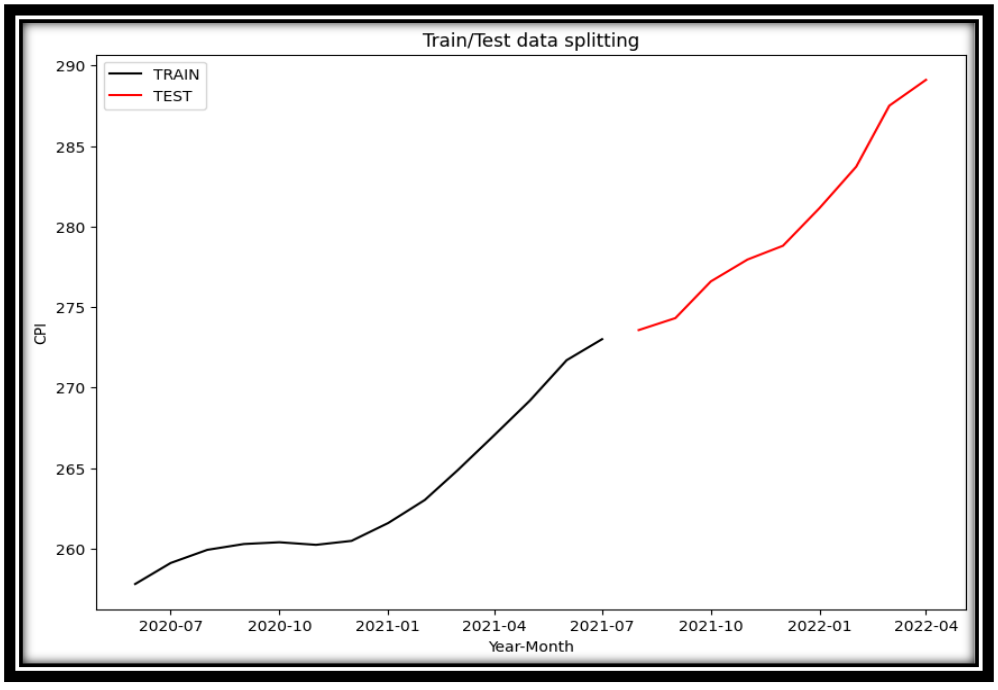
But, the error of this fit is quite high,

Mean square error for Holt's Model is 6.633590682765833

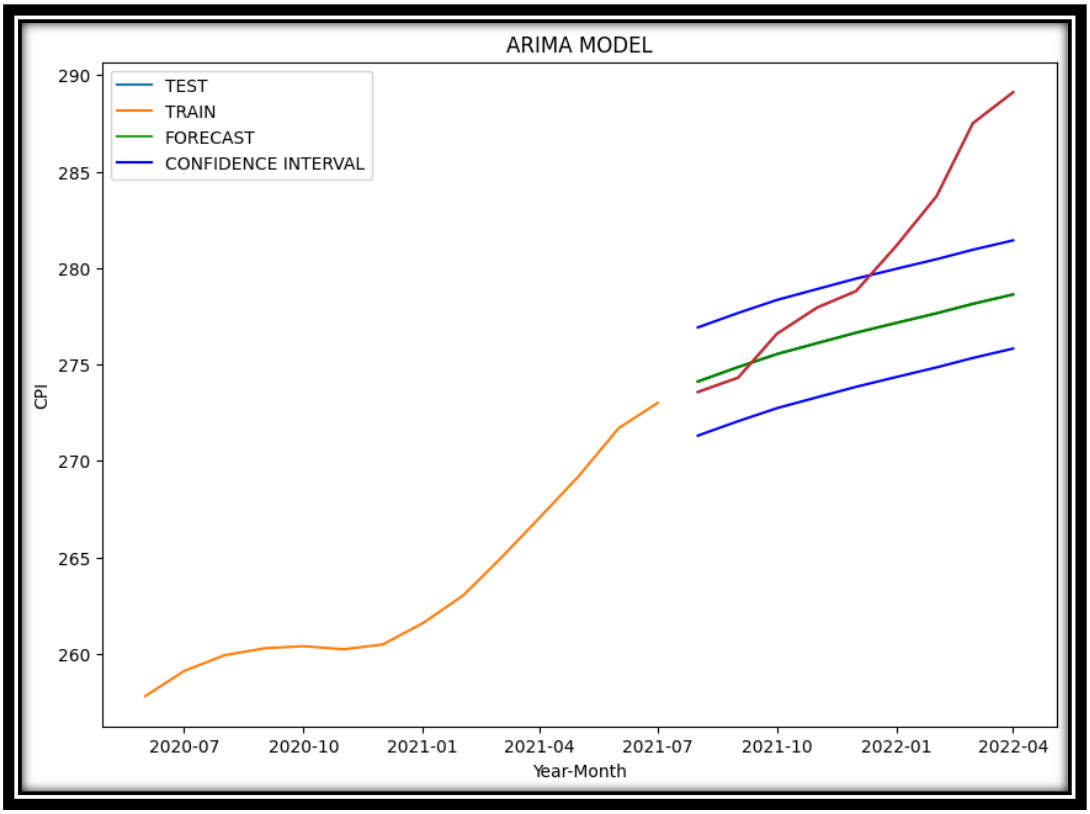
- So, ARIMA(1,2,2) is the best model fit sot this section of the data.

➢ For the SECOND Half:

- Training and testing data is split as:



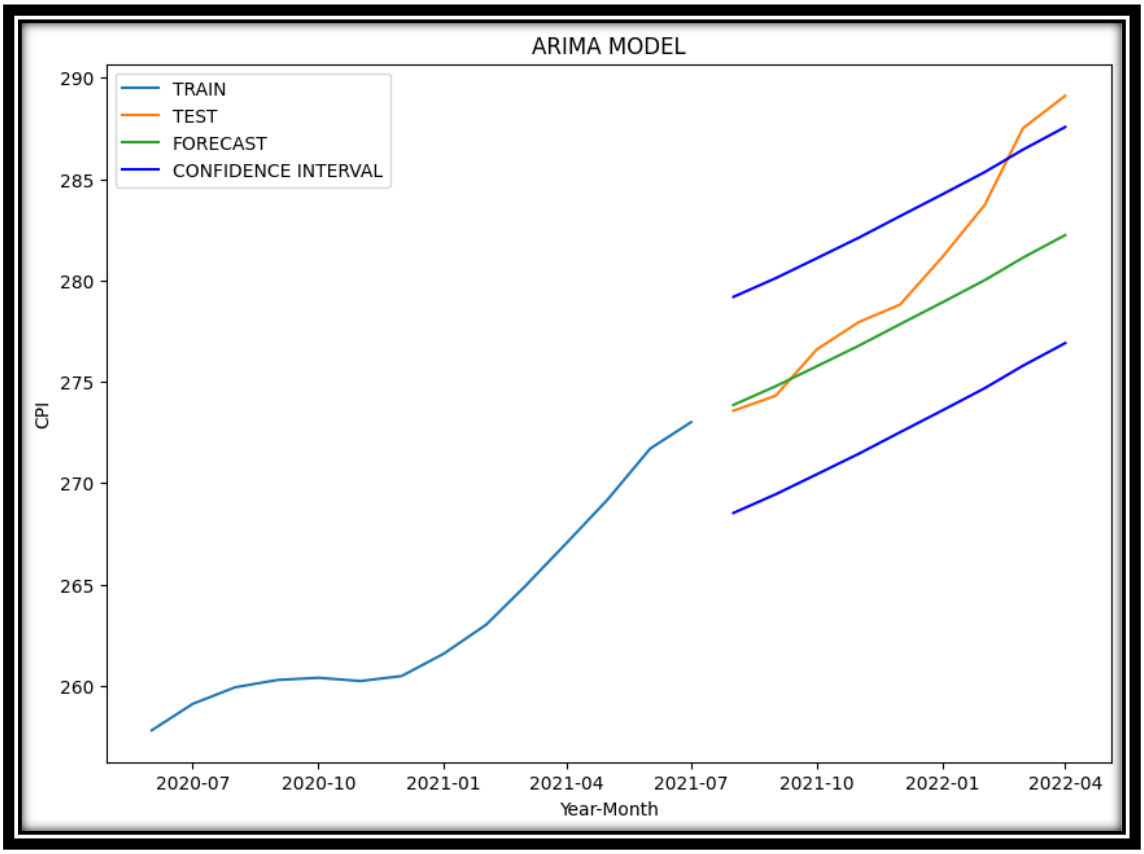
❖ Now as suggested by the AIC and BIC Values, fitting the APIMA(2,2,1) model to the data as:



But the error for this model is quite high,

Mean square Error- 5.376770663736331

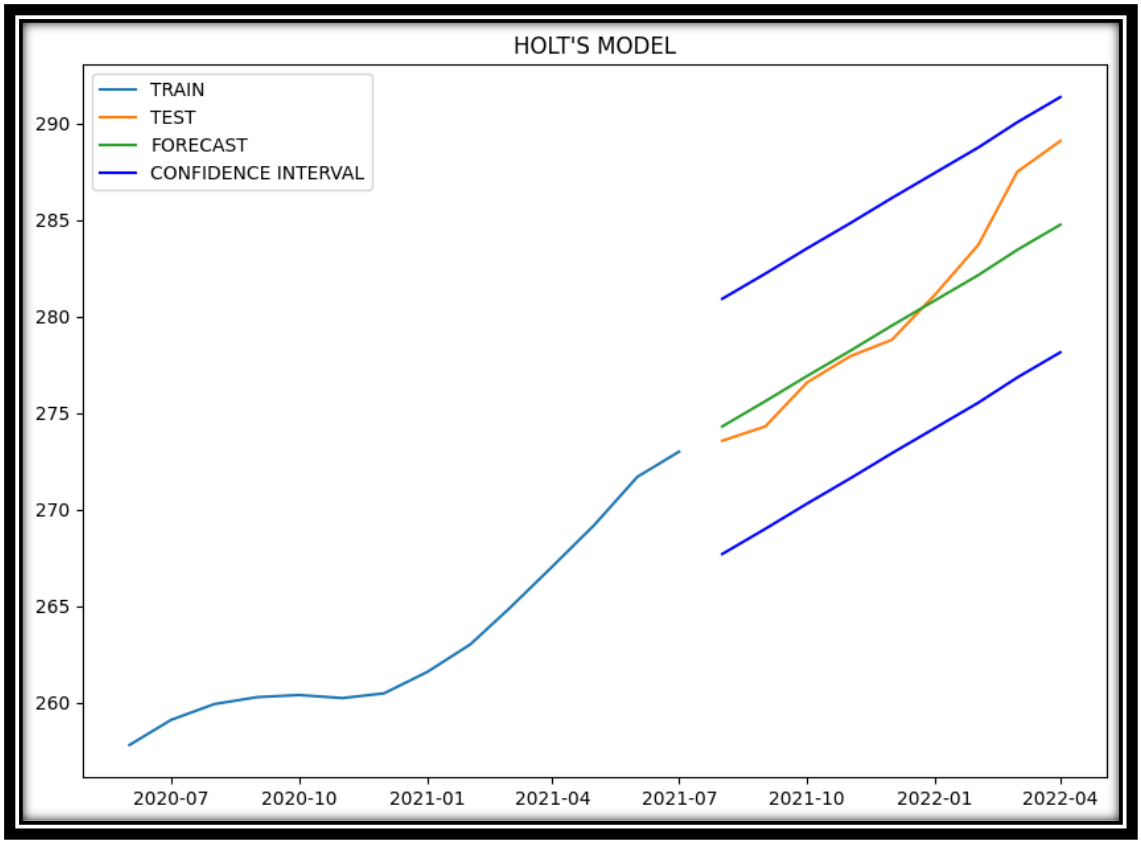
- Checking for next small BIC and AIC value, ARIMA(1,2,2) is found, which is also used for the first part of the data, and on fitting this model, it is observed that:



Mean square error for ARIMA(1,2,2) model is 3.496655809821376

Now, the error is quite low, but on visualizing it is found that the values are going out of the confidence interval. So, it cannot be a good fit for the data.

❖ Now, trying with the Holts' model for fitting the data,

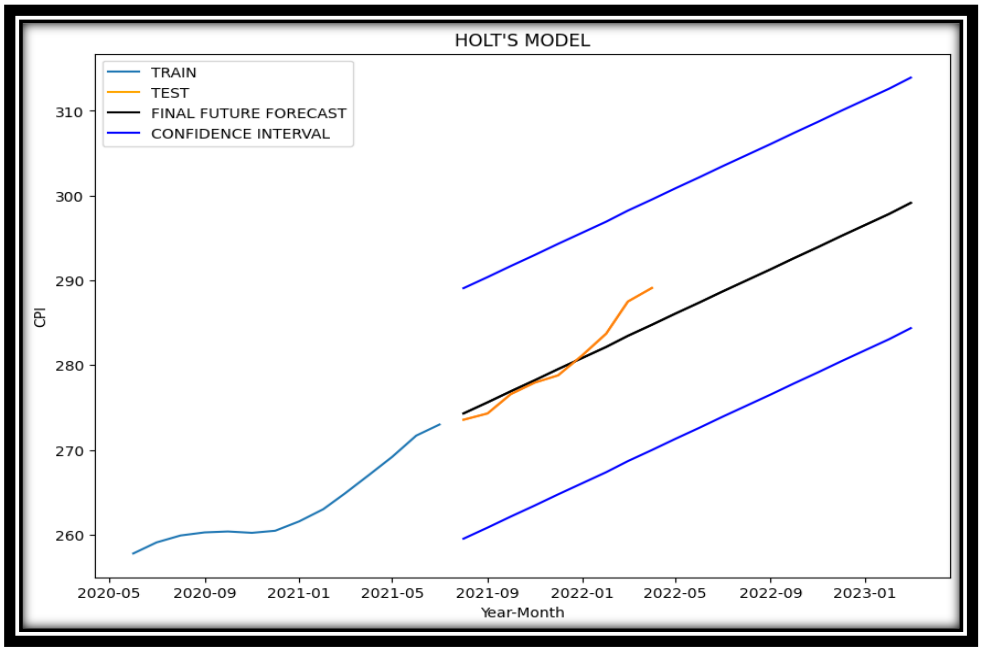


Mean square error for Holts' model 2.1280110852563934

Now, the error is quite low, and thus this model is a good fit for this portion of data.

So, Holts' linear trend model is used for final future predictions,

The forecasted values and the graph is shown as:



FORECAST DATE	FORECAST VALUES
2021-08-01	274.31
2021-09-01	275.617
2021-10-01	276.924
2021-11-01	278.231
2021-12-01	279.538
2022-01-01	280.845
2022-02-01	282.152
2022-03-01	283.459
2022-04-01	284.766
2022-05-01	286.073
2022-06-01	287.38
2022-07-01	288.687
2022-08-01	289.994
2022-09-01	291.301
2022-10-01	292.608
2022-11-01	293.915
2022-12-01	295.222
2023-01-01	296.529
2023-02-01	297.836
2023-03-01	299.143
2023-04-01	300.450001
2023-05-01	301.757001
2023-06-01	303.064001

CONCLUSION:

This project is based on the Consumer Price Index (CPI) values of the United States from January 2010- May 2022 and aims to predict the next 12 monthly values. From the data it is observed that, upto mid of 2020 the CPI values are increasing approximately in a same pace and the ARIMA(1,2,2) model efficiently fits the data. It is also observed that the data has a steep rise after the mid of 2020, for which the ARIMA(1,2,2) model loses its efficiency. This rise in CPI values can be interpreted as an effect of Covid 19 pandemic on the economic condition of the United States. It is found that Holts' Linear Trend Model gives a good fit for the data from mid of 2020 to May 2022. It can be concluded that, if the current rise in CPI due to the pandemic is not controlled, the CPI values would increase in a higher rate and the forecasts given by the Holts' Linear Trend Model will be an approximate match.

This forecast can also be used by the governing bodies as a warning to control the unexpected rise of the CPI values which is not desirable.

Thus, besides predicting the future Consumer Price Index Values the project also aims to analyses the cause and effect of the other factors on the Consumer Price Index Values.

REFERENCE:

Referred Links:

- Webpage: Analytics Vidhya
- https://www.analyticsvidhya.com/blog/2021/08/holt-winters-method-for-time-series-analysis/
- https://www.analyticsvidhya.com/blog/2018/08/auto-arima-time-series-modeling-python-r/

Books:

- Forecasting: Principles and Practice by Rob J Hyndman and George Athanasopoulos, Monash University, Australia
- Fundamentals of Statistics (Volume Two) by A.M. GUN, M.K. GUPTA, B. DASGUPTA
- Applied Statistics by Parimal Mukhopadhyay
- Time Series Analysis: Forecasting and Control by George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel
- The Analysis of Time Series: An Introduction by Chris Chatfield

APPENDIX:

```
from google.colab import drive
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose
from dateutil.parser import parse
import numpy as np
import pandas as pd
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_pacf,plot_acf
from statsmodels.tsa.stattools import acf, pacf
from statsmodels.tsa.arima.model import ARIMA
from sklearn.metrics import mean_absolute_error as mae
import itertools
import scipy.stats as st
import warnings
warnings.filterwarnings('ignore')
from datetime import datetime
from datetime import timedelta
from statsmodels.tsa.holtwinters import ExponentialSmoothing
from statsmodels.tsa.holtwinters import Holt
drive.mount('/content/gdrive')

Data= pd.read_excel(r'/content/gdrive/MyDrive/mydata.xlsx')
print(Data)

df = pd.DataFrame(Data, columns = ['sl no.', 'data'])
df.index = df['sl no.']
del df['sl no.']
plt.figure(figsize=(10,7))
plt.plot(df)
plt.title("CPI Data of US from Jan 2010- April 2022")
plt.xlabel("Date")
plt.ylabel("Data")

atest=adfuller(df['data'])
print("ADF statistic: ",atest[0])
print("P-value: ",atest[1])
for key, value in atest[4].items():
   print('Critial Values:')
   print(f'   {key}, {value}')


d= seasonal_decompose(df['data'],model="multiplicative",period=1)
d.plot()


d.trend

df['data_diff'] = df['data'].diff(1).diff(1)
```

```
df['data_diff'] =df['data_diff'].dropna()
df['data_diff'].plot()
df['data_diff']
plt.xlabel("Date")
plt.ylabel("Second Order Differencing Values")


atest2=adfuller(df['data_diff'].dropna(), autolag='AIC')
print("ADF statistic: ",atest2[0])
print("P-value: ",atest2[1])
for key, value in atest2[4].items():
   print('Critial Values:')
   print(f'   {key}, {value}')


plot_acf(df['data_diff'].dropna())
plot_pacf(df['data_diff'].dropna())


for i in range(1,4):
  for j in range(1,4):
    mod = ARIMA(df['data'], order=(i,2,j))
    rslt1 = mod.fit()
    print('ARIMA{} - AIC:{} - BIC:{}'.format((i,2,j), rslt1.aic, rslt1.bic))


df['sl no.'] = df.index
train = df[df['sl no.'] < pd.to_datetime("2017-05-01", format='%Y-%m-%d')]
train['train'] = train['data']
del train['sl no.']

t = df[df['sl no.'] >= pd.to_datetime("2017-05-01", format='%Y-%m-%d')]
t['sl no.'] = t.index
test=t[t['sl no.'] < pd.to_datetime("2020-05-01", format='%Y-%m-%d')]
test['test'] = test['data']
del test['sl no.']

plt.figure(figsize=(10,7))
plt.plot(train['train'], color = "black")
plt.plot(test['test'], color = "red")
plt.title("Train/Test data splitting")
plt.ylabel("CPI")
plt.xlabel('Year')
plt.legend(["TRAIN","TEST"])


model = ARIMA(train['train'], order=(1,2,2))
result=model.fit()
forecast = result.predict('2017-05-01','2020-04-01')
plt.figure(figsize=(10,7))
plt.plot(test['test'])
```

```
plt.plot(train['train'])

plt.plot(forecast)
plt.title("ARIMA MODEL")

y = (forecast)
ci = 1.96 * np.std(y)
# Plot the sinus function
#plt.plot(y)
# Plot the confidence interval
plt.plot( (y-ci), color='red')
plt.plot( (y+ci), color='red')
plt.plot(y,color='green')

plt.legend(["TEST","TRAIN","FORECAST","CONFIDENCE INTERVAL"])
plt.ylabel("CPI")
plt.xlabel('Year')


from math import sqrt
from sklearn.metrics import mean_squared_error

rms = sqrt(mean_squared_error(test['test'] ,forecast))
print("MEAN SQ ERROR FOR ARIMA(1,2,2) MODEL", rms)

fit1 = Holt(train['train']).fit()
forecast2= fit1.forecast(36)
plt.figure(figsize=(10,7))
plt.plot(test['test'])
plt.plot(train['train'])

plt.plot(forecast2)
plt.title("HOLT'S MODEL")

y = (forecast2)
ci = 1.96 * np.std(y)
plt.plot( (y-ci), color='red')
plt.plot( (y+ci), color='red')
plt.plot(y,color='green')

plt.legend(["TEST","TRAIN","FORECAST","CONFIDENCE INTERVAL"])
plt.ylabel("CPI")
plt.xlabel('Year')
rms2 = sqrt(mean_squared_error(test['test'] ,forecast2))
print("MEAN SQ ERROR FOR HOLT'S MODEL", rms2)


df['sl no.'] = df.index
t2= df[df['sl no.'] > pd.to_datetime("2020-05-01", format='%Y-%m-%d')]
train2= t2[t2['sl no.'] < pd.to_datetime("2021-08-01", format='%Y-%m-%d')]
```

```
train2['train'] = train2['data']
del train2['sl no.']

test2 = df[df['sl no.'] >= pd.to_datetime("2021-08-01", format='%Y-%m-%d')]
test2['test'] = test2['data']
del test2['sl no.']

plt.figure(figsize=(10,7))
plt.plot(train2['train'], color = "black")
plt.plot(test2['test'], color = "red")
plt.title("Train/Test data splitting")
plt.ylabel("CPI")
plt.xlabel('Year-Month')
plt.legend(["TRAIN","TEST"])

model = ARIMA(train2['train'], order=(1,2,2))
result=model.fit()
forecastn = result.predict('2021-08-01','2022-04-01')
plt.figure(figsize=(10,7))

plt.plot(train2['train'])
plt.plot(test2['test'])

plt.title("ARIMA MODEL")

y = (forecastn)
ci = 1.96 * np.std(y)
# Plot the sinus function
#plt.plot(y)
# Plot the confidence interval
plt.plot(y)
plt.plot( (y-ci), color='blue')
plt.plot( (y+ci), color='blue')

plt.ylabel("CPI")
plt.xlabel('Year-Month')

plt.legend(["TRAIN","TEST","FORECAST","CONFIDENCE INTERVAL"])
rms = sqrt(mean_squared_error(test2['test'] ,forecastn))
print(rms)
fit12 = Holt(train2['train']).fit()
forecastn2= fit12.forecast(9)
plt.figure(figsize=(10,7))

plt.plot(train2['train'])
plt.plot(test2['test'])

plt.plot(forecastn2)
plt.title("HOLT'S MODEL")

y = (forecastn2)
```

```
ci = 1.96 * np.std(y)
plt.plot( (y-ci), color='blue')
plt.plot( (y+ci), color='blue')

plt.legend(["TRAIN","TEST","FORECAST","CONFIDENCE INTERVAL"])

forecastfinal= fit12.forecast(23)
plt.figure(figsize=(10,7))
plt.plot(train2['train'])
plt.plot(test2['test'],color='orange')

plt.title("HOLT'S MODEL")

y = (forecastfinal)
ci = 1.96 * np.std(y)
# Plot the sinus function
#plt.plot(y)
# Plot the confidence interval
plt.plot(y,color='black')
plt.plot( (y-ci), color='blue')
plt.plot( (y+ci), color='blue')
plt.plot(y,color='black')

plt.ylabel("CPI")
plt.xlabel('Year-Month')
plt.plot(test2['test'])
plt.legend(["TRAIN","TEST","FINAL FUTURE FORECAST","CONFIDENCE INTERVAL"])
forecastfinal

rmsn2 = sqrt(mean_squared_error(test2['test'] ,forecastn2))
print("MEAN SQ ERROR FOR HOLT'S MODEL", rmsn2)

forecastfinal.to_excel('MY WORK.xlsx')
```