**Using ARIMA for Predicting CPI in Dubai**

**Data Analysis**

* 1. Data Preparation

First of all, we need to add all bills for all quarters from 2018 to 2022 in one sheet (a total of 110 rows).

To accomplish this task, we are using Python and its Pandas library. Pandas is a powerful data analysis library that allows users to easily manipulate and analyze data in Python. Specifically, the team will use Pandas to read in the data from each quarter's bill report, combine the data into a single DataFrame, and write the combined data to a new sheet.

Once the data has been combined into a single sheet, the second task is to merge all of the required data and modify it to use as input data in an ARIMA prediction model. ARIMA (AutoRegressive Integrated Moving Average) is a popular time series forecasting model that can be used to make predictions based on historical data.

To prepare the data for use in an ARIMA model, we will need to perform some additional data manipulation and cleaning. This may include removing any duplicate or irrelevant data, filling in missing values, and transforming the data as needed to meet the requirements of the ARIMA model.

* 1. Data Preprocessing
     1. Data Cleaning

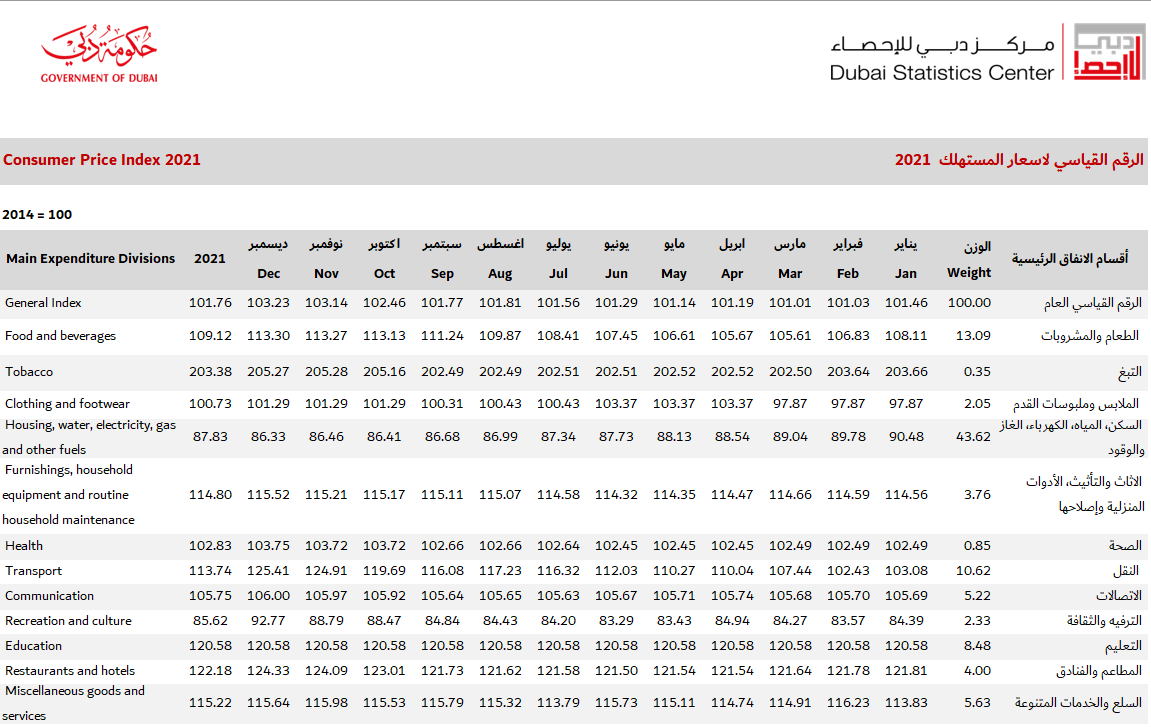
The cleaning process was conducted in .csv files using Python. The CPI files received from Dubai Statistics Center contain many unnecessary cells which some signals and Arabian letters. I followed these cleaning steps before we start joining the three datasets together.

Below are the cleaning steps for the CPI’s datasheet:

Python:

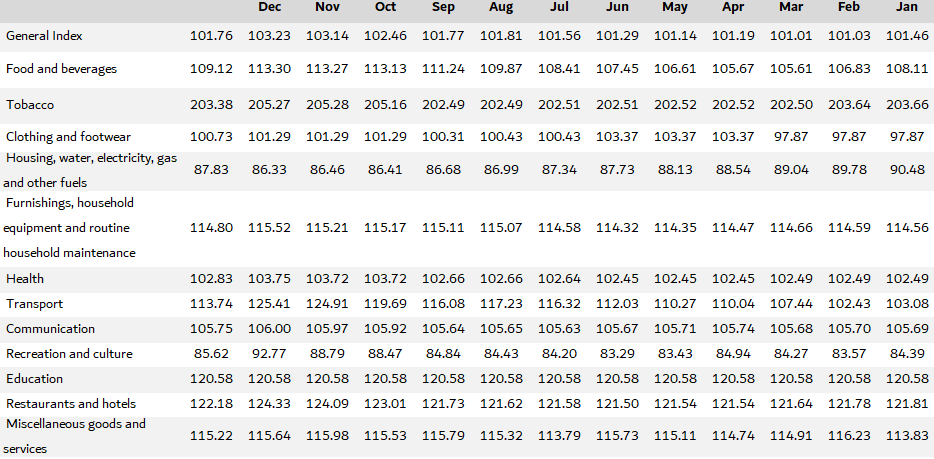
* Removing NAs, blanks, and zero values in CPI dataset.

9 rows and 1 column contain blanks and Arabian alphabets.



This datasheet is of the shape-23 rows, 17 columns.

To remove these cells, choose only necessary cells using Python.



if nYear < 2020:

        xls\_data = pd.read\_excel(xFile, index\_col=0)

    else:

        xls\_data = pd.read\_excel(xFile, index\_col=1)

    xls\_data.to\_csv('temp.csv')

    df = pd.read\_csv('temp.csv')

    print(df)

    if nYear >= 2020:

        df = df.drop(['Unnamed: 0.1', 'Unnamed: 0', 'Unnamed: 2', 'Unnamed: 15'], axis = 1)

    else:

        df = df.drop(['Unnamed: 0', 'Unnamed: 1', 'Unnamed: 14'], axis = 1)

    # if nYear == 2018 or nYear == 2020:

    print(df)

    if nYear >= 2020:

        df = df[9:23]

    else:

        df = df[5:18]

* + 1. Data Preparation

Merge the CPI datasheets from 2018 to 2022 into one sheet and modify it for analysis and forecasting.

* Transform each CPI data into one for analysis and forecasting.
* Identify the shape of all the CPI data by adding an ‘Insurance and financial services’ factor to 2018 and 2021 data and renaming some industry factors used for estimation of CPI.

  df = df.\_append(append\_df)

temp\_cols = df.columns.tolist()

    new\_cols = temp\_cols[-1:] + temp\_cols[:-1]

    df = df[new\_cols]

    dfT =df.T

    dfT = dfT.set\_axis(list(dfT.iloc[0]), axis = 1)

    dfT = dfT.drop(['Unnamed: 16'], axis = 0)

    dfT = dfT.set\_axis(new\_row, axis = 0)

    dfT\_shape = dfT.shape

    if dfT\_shape[1] == 14:

        finance = [100] \* 12

        dfT['Insurance and financial services'] = finance

        temp\_cols = dfT.columns.tolist()

        new\_cols = temp\_cols[0:2] + temp\_cols[-1:] + temp\_cols[2:-1]

        dfT = dfT[new\_cols]

dfT.rename(columns = {'Personal care, social protection and miscellaneous goods and services' : 'Miscellaneous goods and services', 'Restaurants and accommodation services' : 'Restaurants and hotels', 'Recreation, sport and culture' : 'Recreation and culture', 'Information and communication' : 'Communication'}, inplace = True)

* Transform 2021-based 2022-CPI data into 2014-based one and merge all data.

df18 = xls2csv("Consumer Price Index 2018.xls", 2018)

df19 = xls2csv("Consumer Price Index 2018.xls", 2019)

df20 = xls2csv("Consumer Price Index 2020.xls", 2020)

df21 = xls2csv("Consumer Price Index 2021.xls", 2021)

df22 = xls2csv("Consumer Price Index 2022.xls", 2022)

##########################Conver base-2021 to base-2014 for 2022#####################

print(df22.head())

for i in range(12):

    for j in range(1, 15):

        df22.iloc[i, j] = float(df22.iloc[i, j]) \* float(df21.iloc[i, j]) / 100

#####################################################################################

df = df18

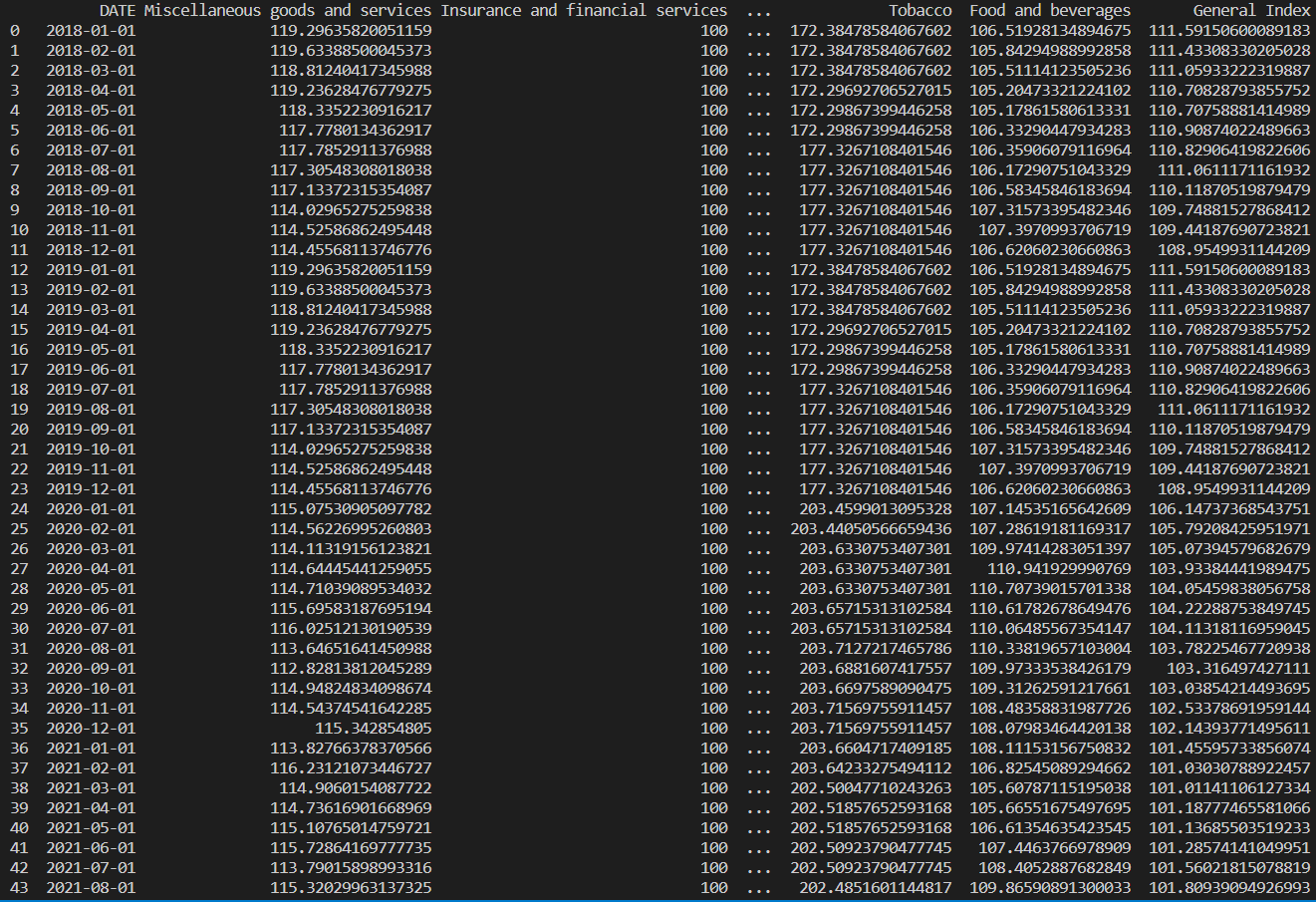
df = df.\_append(df19, ignore\_index = True)

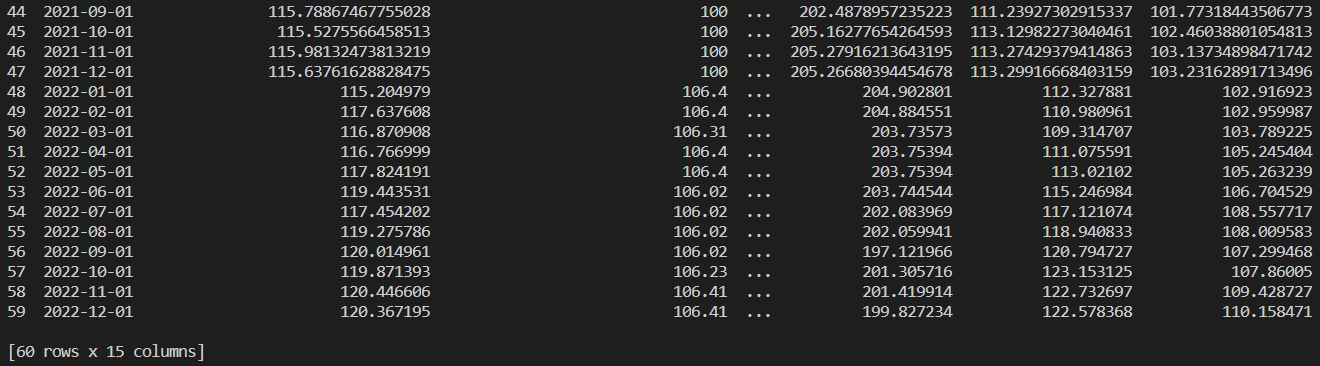
df = df.\_append(df20, ignore\_index = True)

df = df.\_append(df21, ignore\_index = True)

df = df.\_append(df22, ignore\_index = True)

The merged data is as follow.



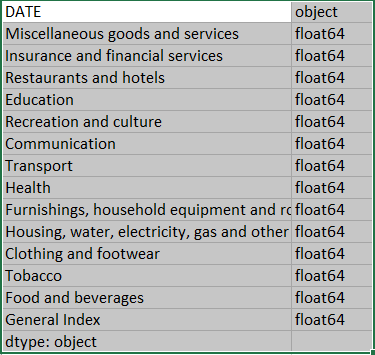


* 1. Data Understanding

 check on data types and convert date to datetime.

print(monthly\_raw.dtypes)

monthly\_raw.DATE = pd.to\_datetime(monthly\_raw.DATE)

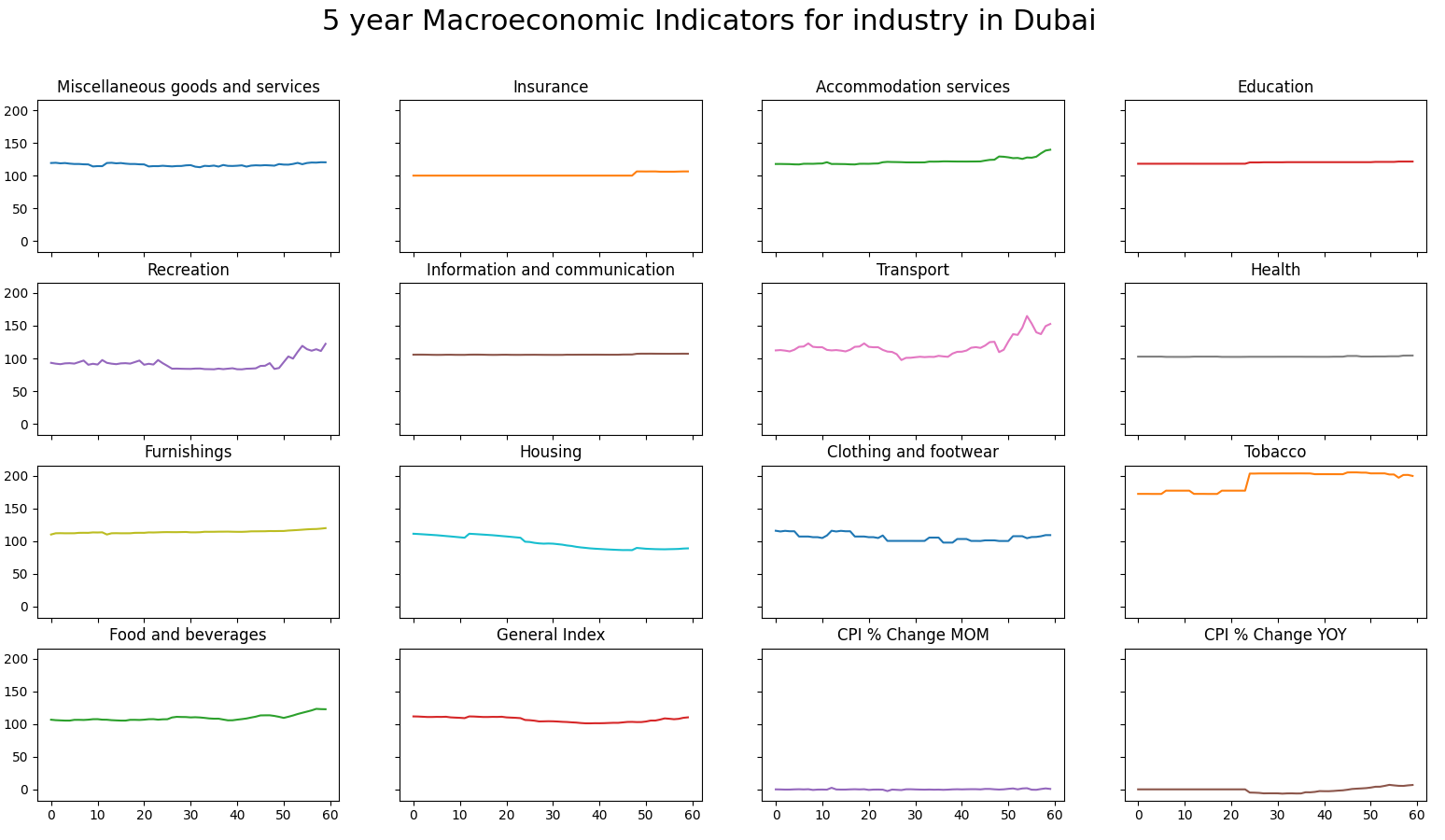


* + 1. Macroeconomic Indicator Trend.

We can perform feature generation first and create percentage change Month-over-month(cpi\_pct\_mom) and Year-over-year(cpi\_pct\_yoy).

monthly\_df['cpi\_pct\_mom'] = round((monthly\_df['General Index'].pct\_change().fillna(0)) \* 100, 2)

monthly\_df['cpi\_pct\_yoy'] = round((monthly\_df['General Index'].pct\_change(12).fillna(0)) \* 100, 2)

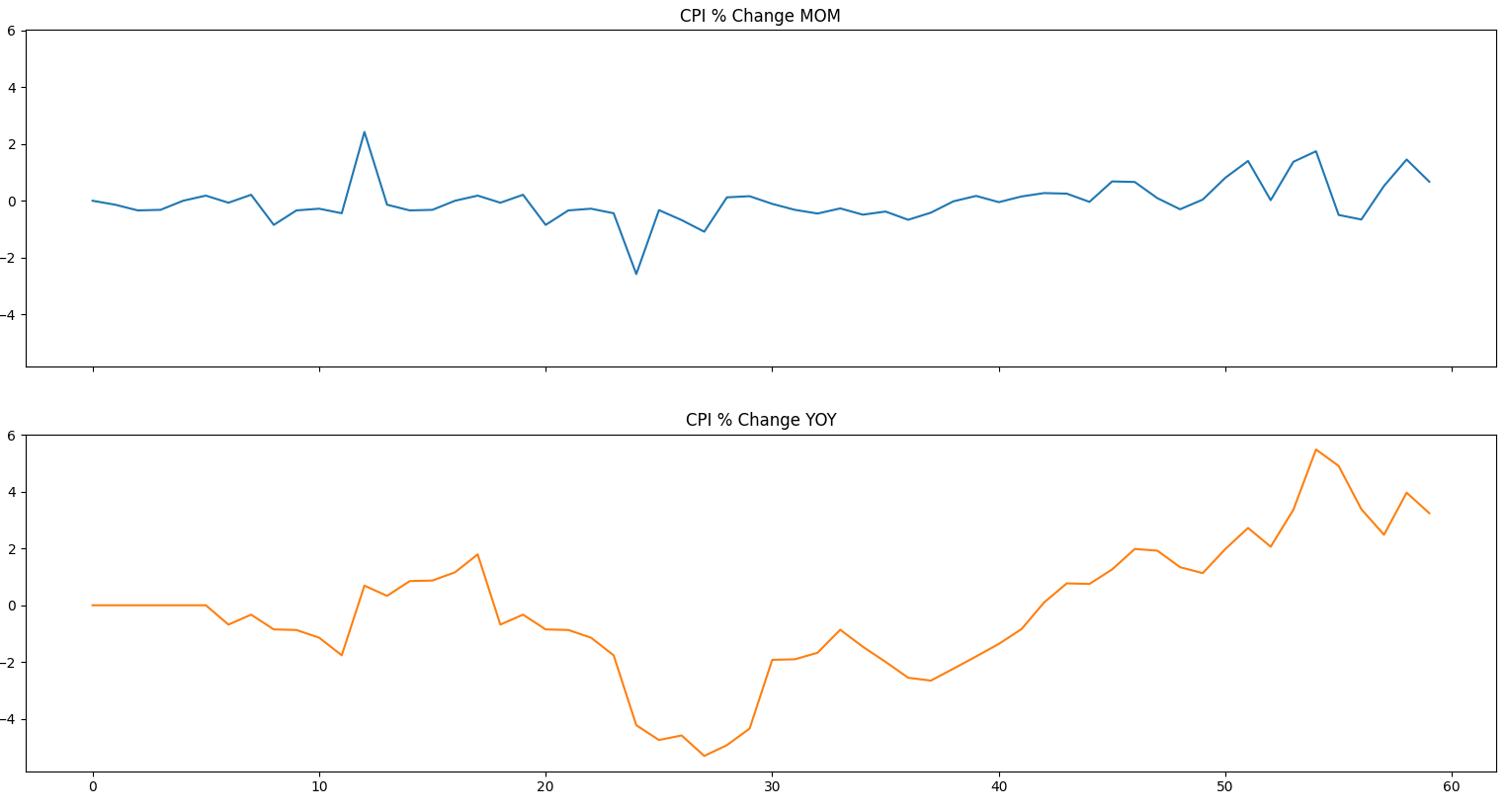


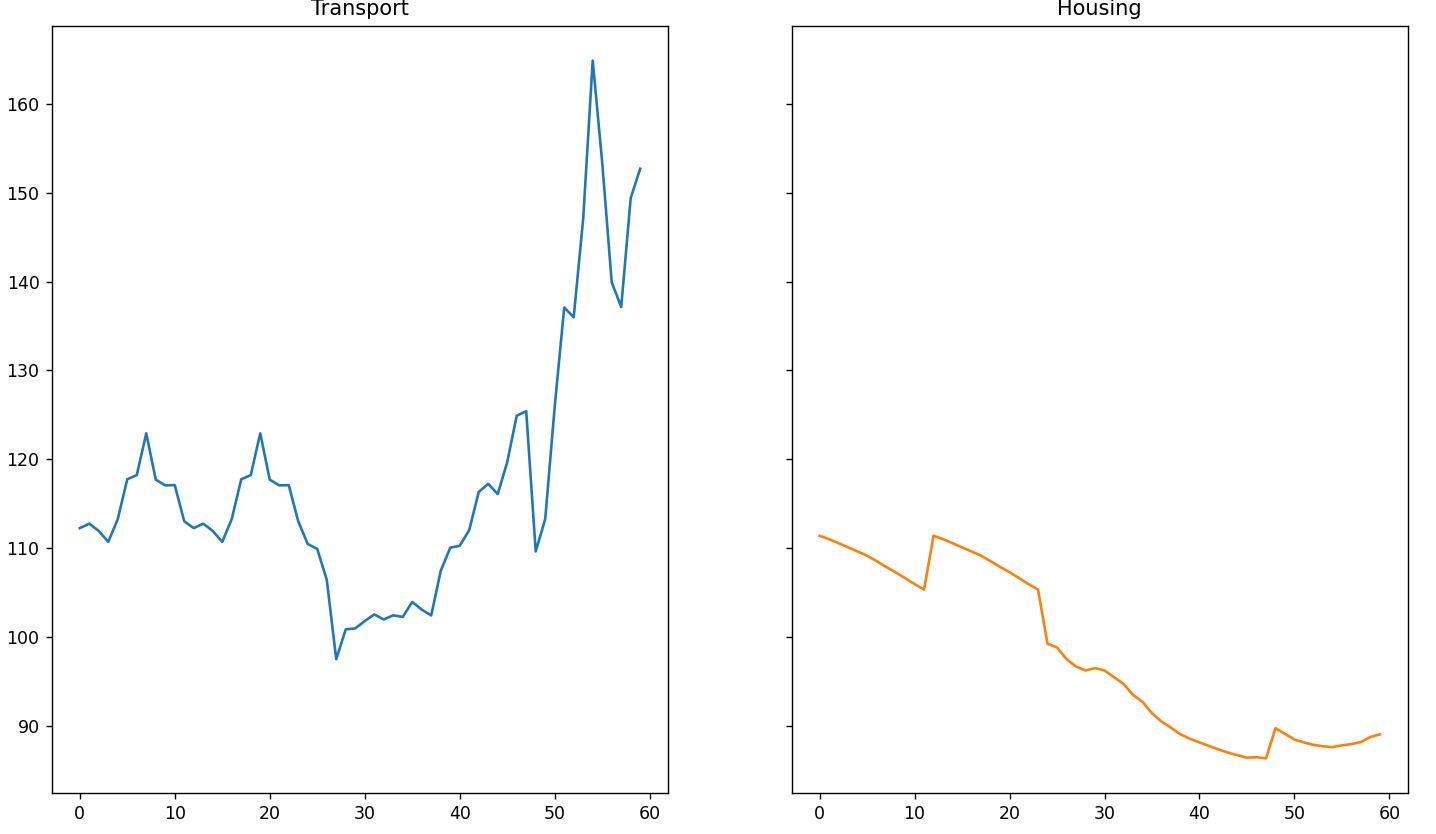
As the dataset shows, the most influence industries are housing and transport because of weight value.

|  |  |
| --- | --- |
| **Main Expenditure Divisions** |  |
| **Weight** |
| General Index | 100.00 |
| Food and beverages | 13.09 |
| Tobacco | 0.35 |
| Clothing and footwear | 2.05 |
| Housing, water, electricity, gas and other fuels | 43.62 |
| Furnishings, household equipment and routine household maintenance | 3.76 |
| Health | 0.85 |
| Transport | 10.62 |
| Communication | 5.22 |
| Recreation and culture | 2.33 |
| Education | 8.48 |
| Restaurants and hotels | 4.00 |
| Miscellaneous goods and services | 5.63 |

So, here, we are going to analyze the consumer price trend in housing and transport.

Here are graphs.





This figure gives a picture of how the consumer price rate in Housing, water, electricity, gas and other fuels, the one in transport and general CPI rate in Dubai has changed during a period of 5 years.

First, we can analyze the historical trend in Transport.

The transport-graph shows significant fluctuations in transport rates during the period in question, with a notable decline by 10% in early 2019 followed by a substantial increase by 50% in late of 2022. These changes suggest that external factors may have influenced transportation patterns during this time frame.

One such factor is the COVID-19 pandemic, which began in 2019 and ended in 2022. The pandemic may have contributed to the decline in early 2019 due to restrictions on travel and changes in consumer behavior. Conversely, the relaxation of restrictions and increased demand for travel may have contributed to the increase in mid-2022.

Another external factor that may have influenced transport rates is the FIFA World Cup held in Qatar in late 2022. The event likely led to an increase in travel demand and may have contributed to the observed 50% rise in transport rates during this period.

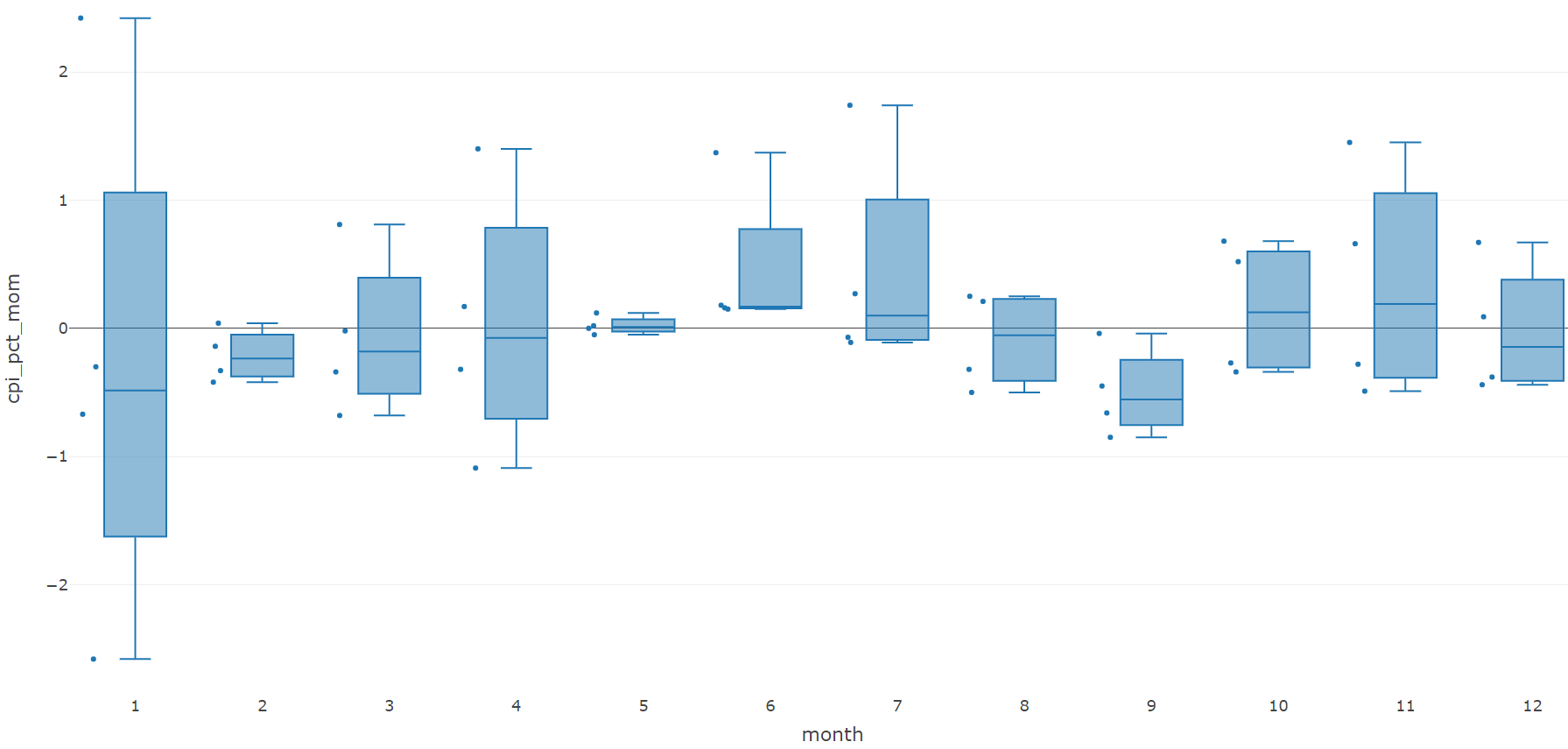
Further analysis is necessary to determine the extent to which these external factors impacted transport rates during this period. However, based on the available data, it appears that both the COVID-19 pandemic and the FIFA World Cup played a role in shaping transportation patterns during this time frame.

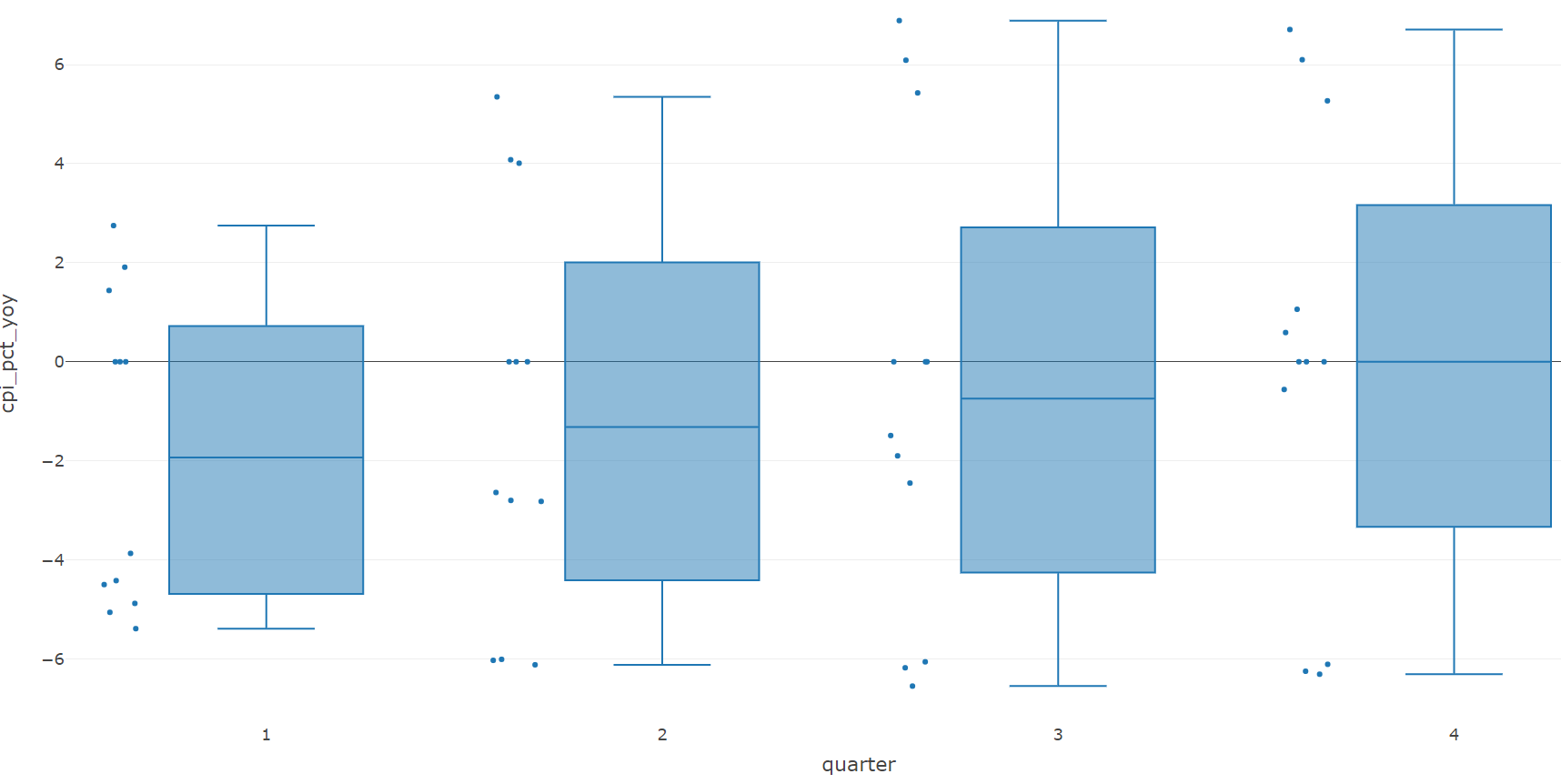
A clear positive period for the Housing in the late 2018 may be seen. This may be due to the finance crisis in Dubai. Later the housing rate reduced gradually, finally decreased by 25%. A positive trend for the housing rate may be seen in this figure which may indicate a non-stationary series.

As above figures show, we can see that the CPI trend is similar to Transport trend.

The CPI trend shows significant fluctuations in the CPI rates during the period in question, with a notable decline by 5% in early 2019 followed by a substantial increase by 5.5% in late of 2022.

* + 1. CPI trend by month and Quarter



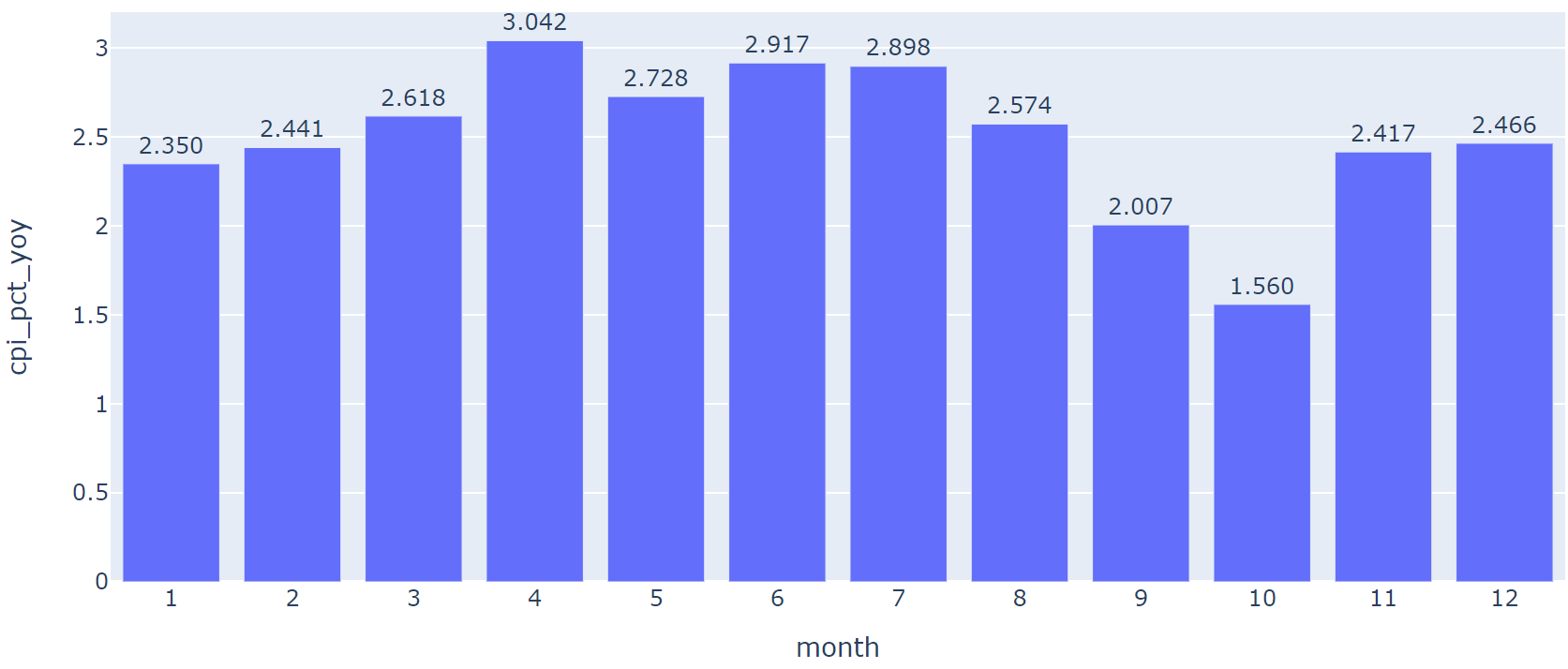


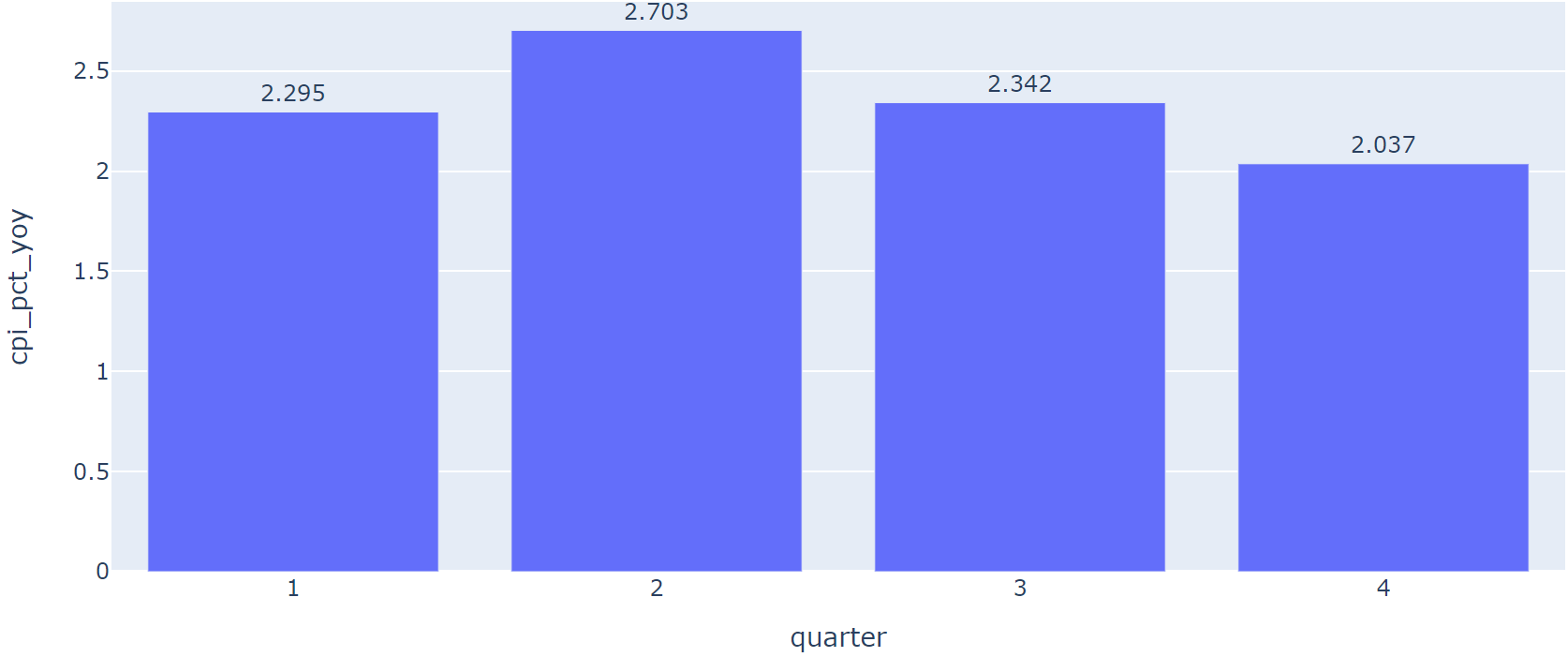
Based on an analysis of the CPI change on an annualized basis, it appears that the middle parts of the year showed more significant increases in CPI compared to the last few months. This can be seen by examining the data using a box plot, where the middle parts of the box (the second and third quartiles) are higher than the first and fourth quartiles.

Furthermore, the box plot shows that there are more outliers (data points that fall outside of the whiskers) in the last few months, which suggests that these months showed little change from the previous year. Specifically, these outliers are located below the lower whiskers, indicating that they are lower than the median value of the data set.

Overall, this analysis suggests that there may be seasonal trends in the CPI change, with middle parts of the year showing more significant increases compared to the last few months. However, more detailed analysis would be necessary to confirm these trends and identify any underlying factors that may be driving them.

We can further explore the volatility of the change in Core CPI YoY.

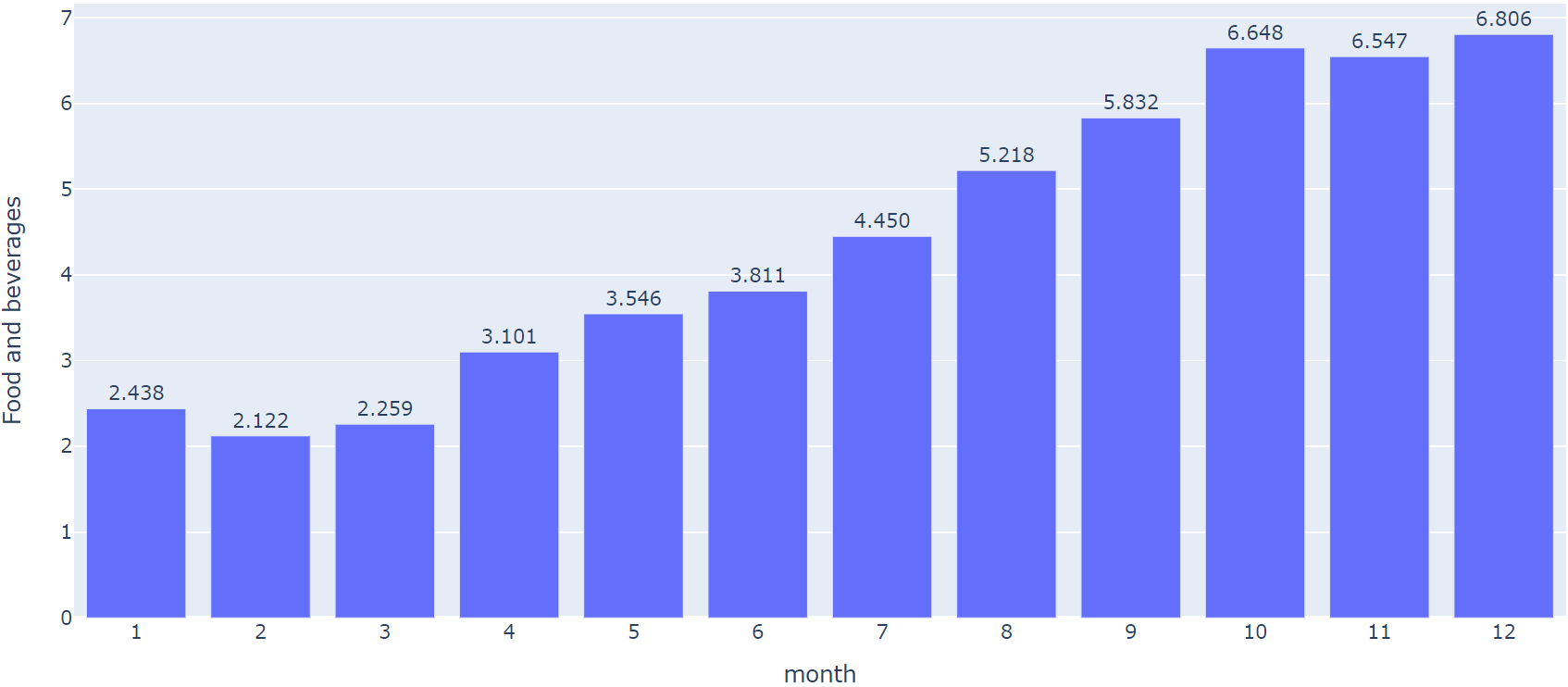
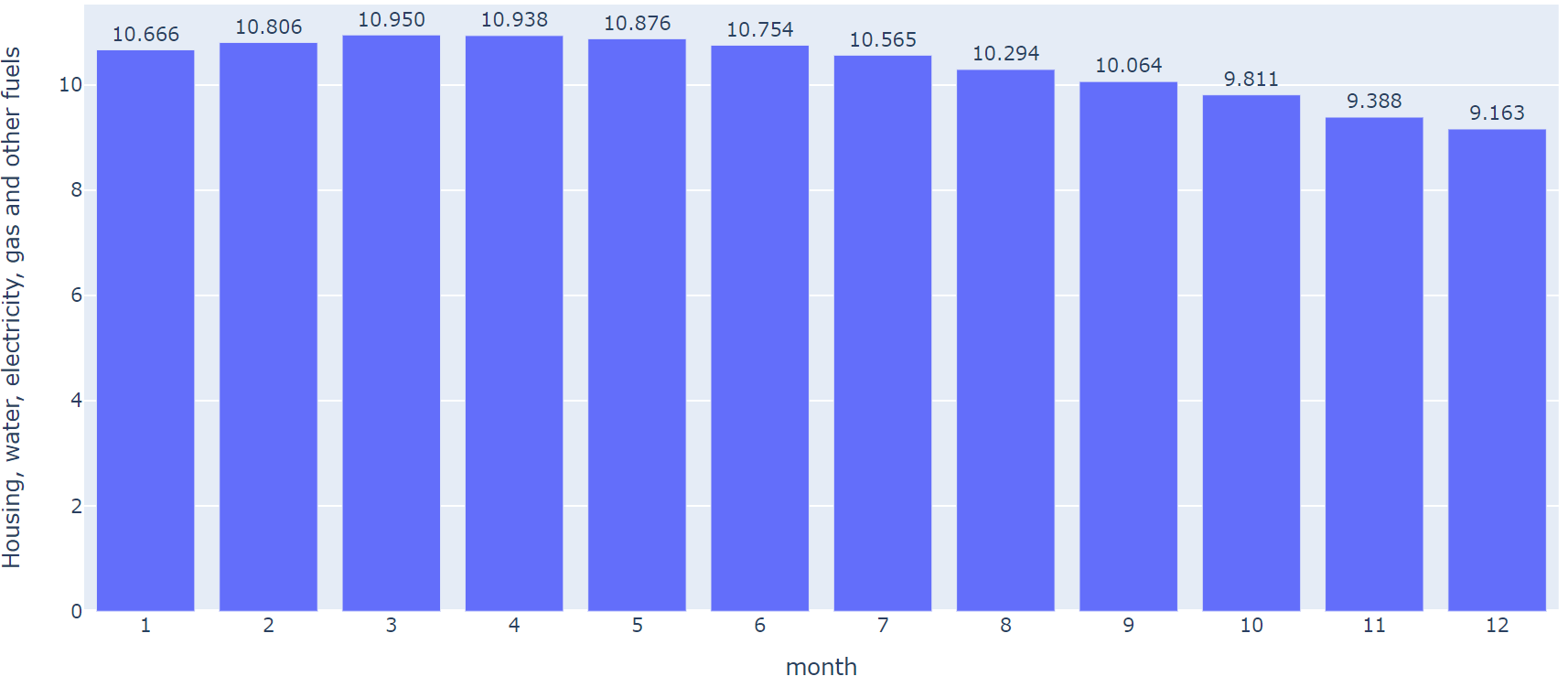
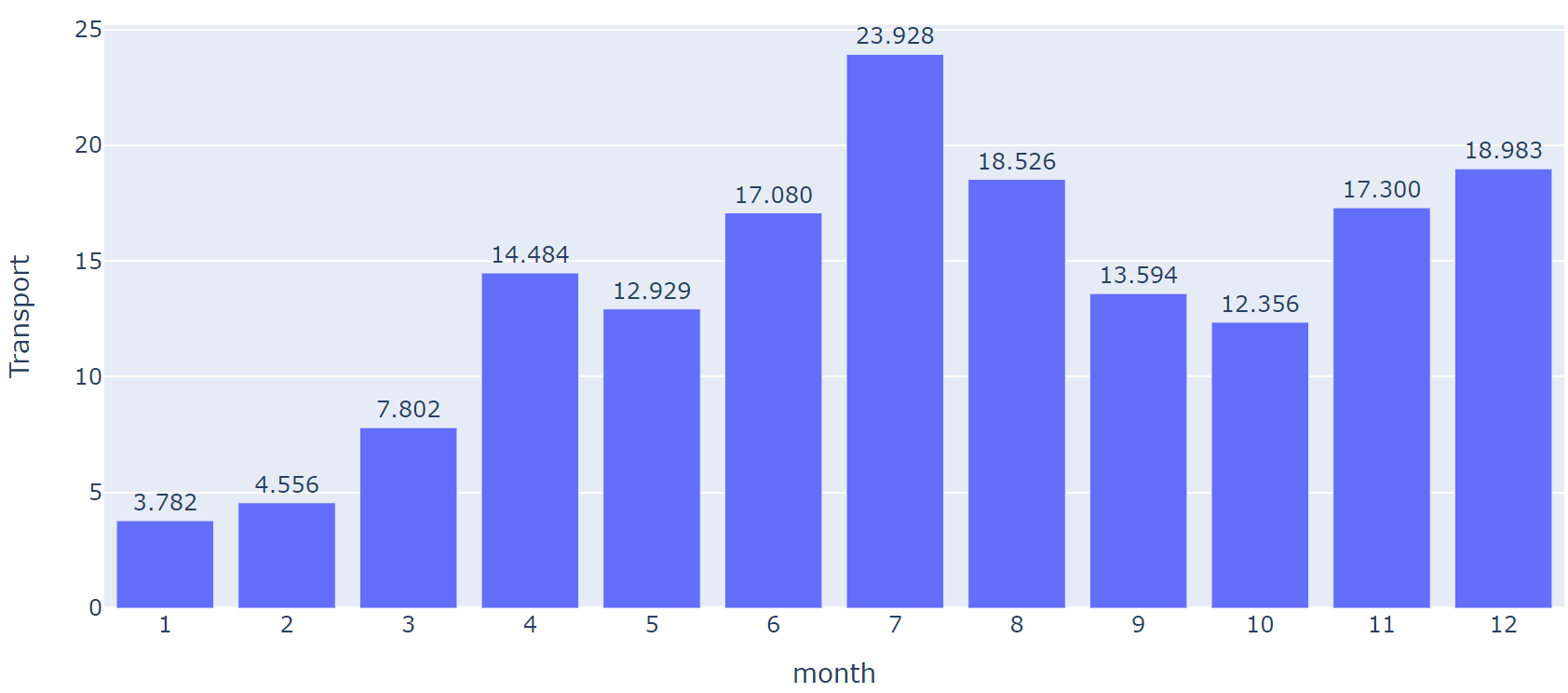


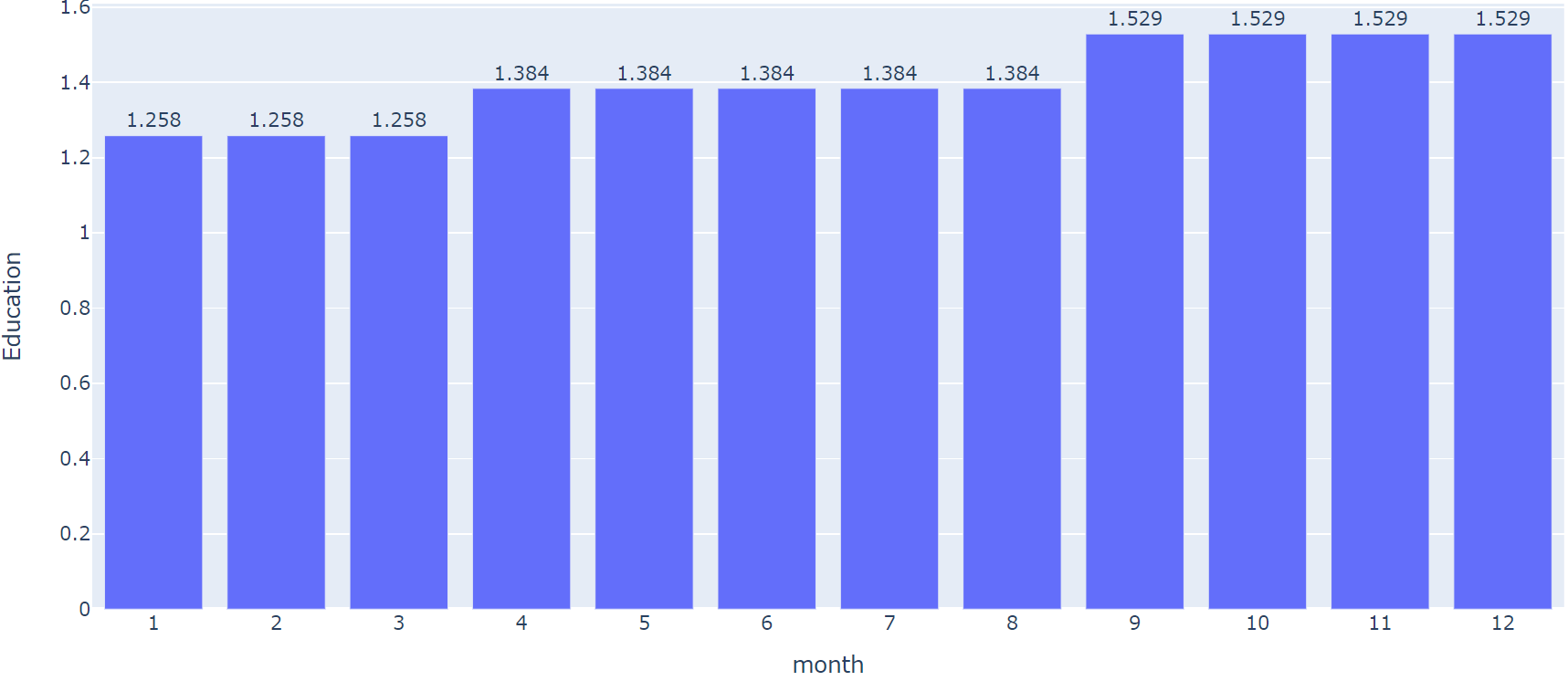


And then let’s interpret the month over month changes for 4 individual industries-Transport, Housing, Food, and Education, which are the main part of the over all CPI changes.

Analysis graphs are as follows.

From the trend for Transport price rate, we can see that the lowest rate is recorded in January and the highest is recorded in July. Same for the price rate of food and beverage, the highest rate is recorded in February and lowest in December.

In Housing, the rate gradually decreases and on the contrary, in education the rate gradually increases.



* 1. Data Modeling

This project will study the dataset in ARIMA model. This model will be discussed based on the following performance evaluation metrics:

* MAE:

The Mean Absolute Error (MAE) is an excellent metric for determining forecast accuracy. It is calculated by finding the mean of the absolute error.

The formula for MAE is:

MAE =

* MSE:

MSE measures the average squared difference between the predicted values and the actual values. It is calculated by taking the average of the squared differences between each predicted value and its corresponding actual value.

The formula for MSE is:

MSE =

1.5 ARIMA Model

ARIMA stands for autoregressive integrated moving average, and it's a statistical analysis model that employs time series data to better understand the data set or anticipate future trends.

An autoregressive integrated moving average model determines how powerful one dependent variable is in contrast to other variables associated. The model's goal is to predict future value of the independent variable by analyzing differences between values in a series rather than actual values.

Components of the ARIMA model:

• Autoregression AR: A model that displays a changing variable regressing on its own

lagged.

• Integrated (I): denotes the differencing of raw observations to allow the time series to

stabilize.

• Moving average (MA): A moving average model applied to lagged observations

incorporates the dependency between an observation and a residual error.

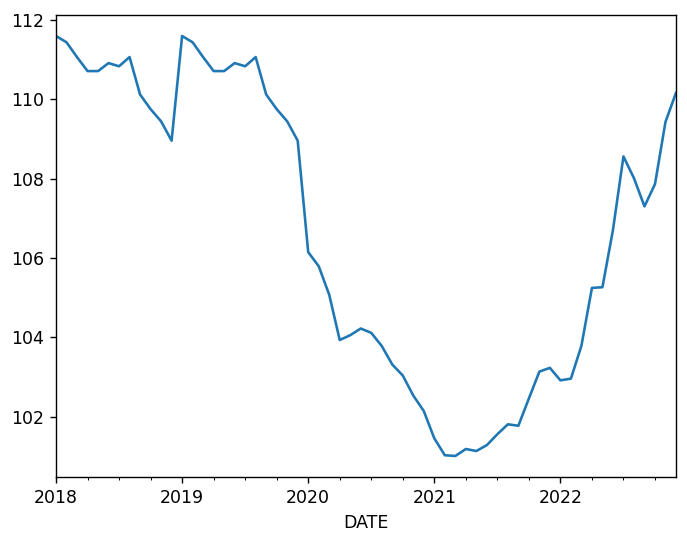
A univariate ARIMA time series model for water and electricity consumption was done. As we have chosen to forecast each department forecast, it is wiser to only consider a single variable-based ARIMA model.

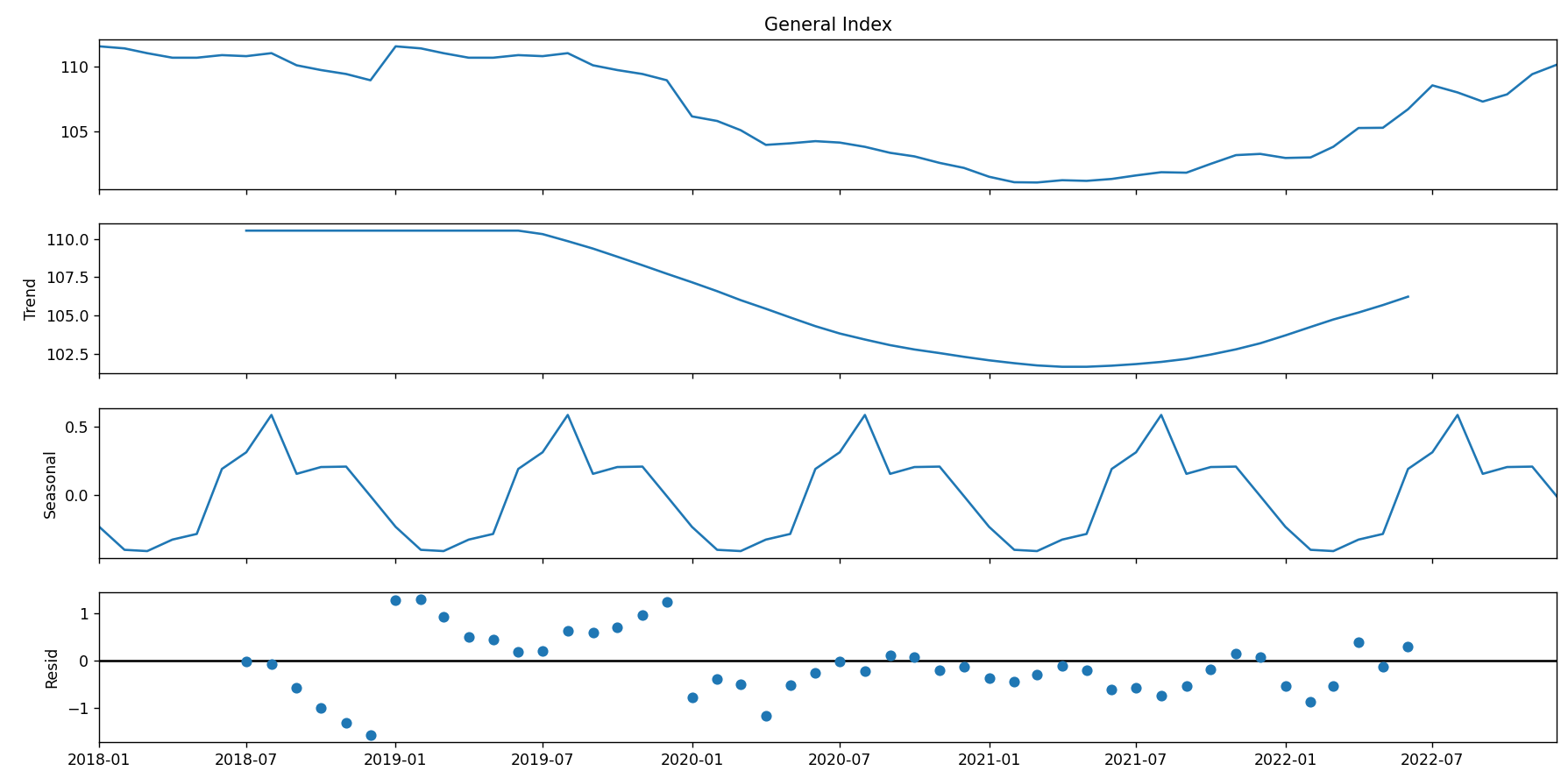
Some upsides of using ARIMA include its interpretability, ease of implementation and may even work better for relatively short series such as this case where the number of observation is not sufficient to apply more sophisticated models.

On the other hand, one limitation of ARIMA models is the assumption of constant variance and in financial time-series, most data exhibit volatility, asymmetries, irregular time intervals, sudden outbreaks, thus this model usually perform poorly on financial time series data.

1.5.1 Time Series Decomposition

Decompose the data into trend, seasonal and residual components.

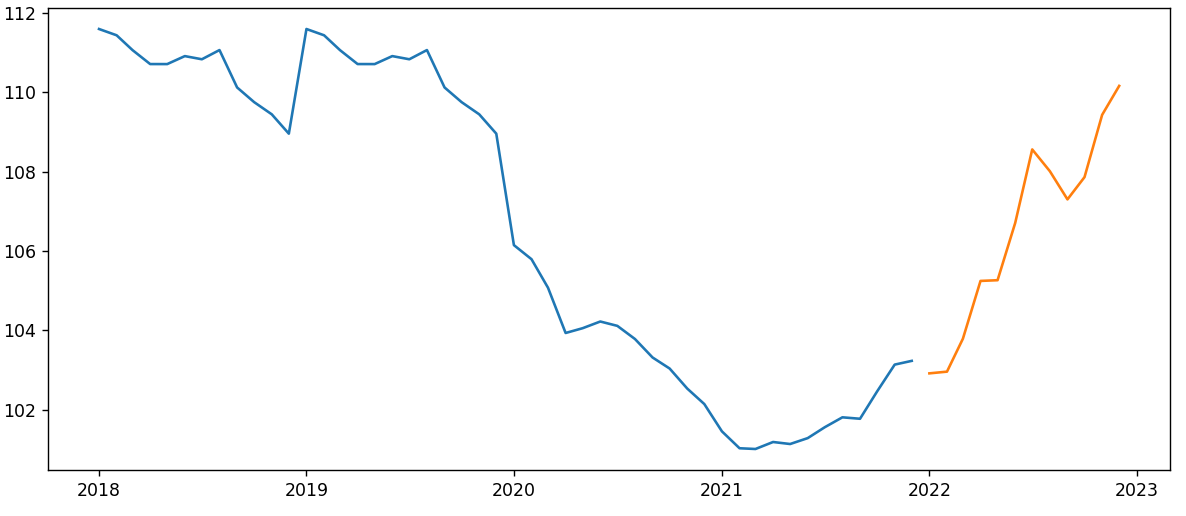




The data shows a clear upward and downward trend and is not stationary. As one of the key assumptions of the ARIMA model is that the time-series is stationary, we need to correct the non-stationarity later.

1.5.2 Splitting the Data

As the dataset is small, we will use the last 12 months as the out of sample test dataset.

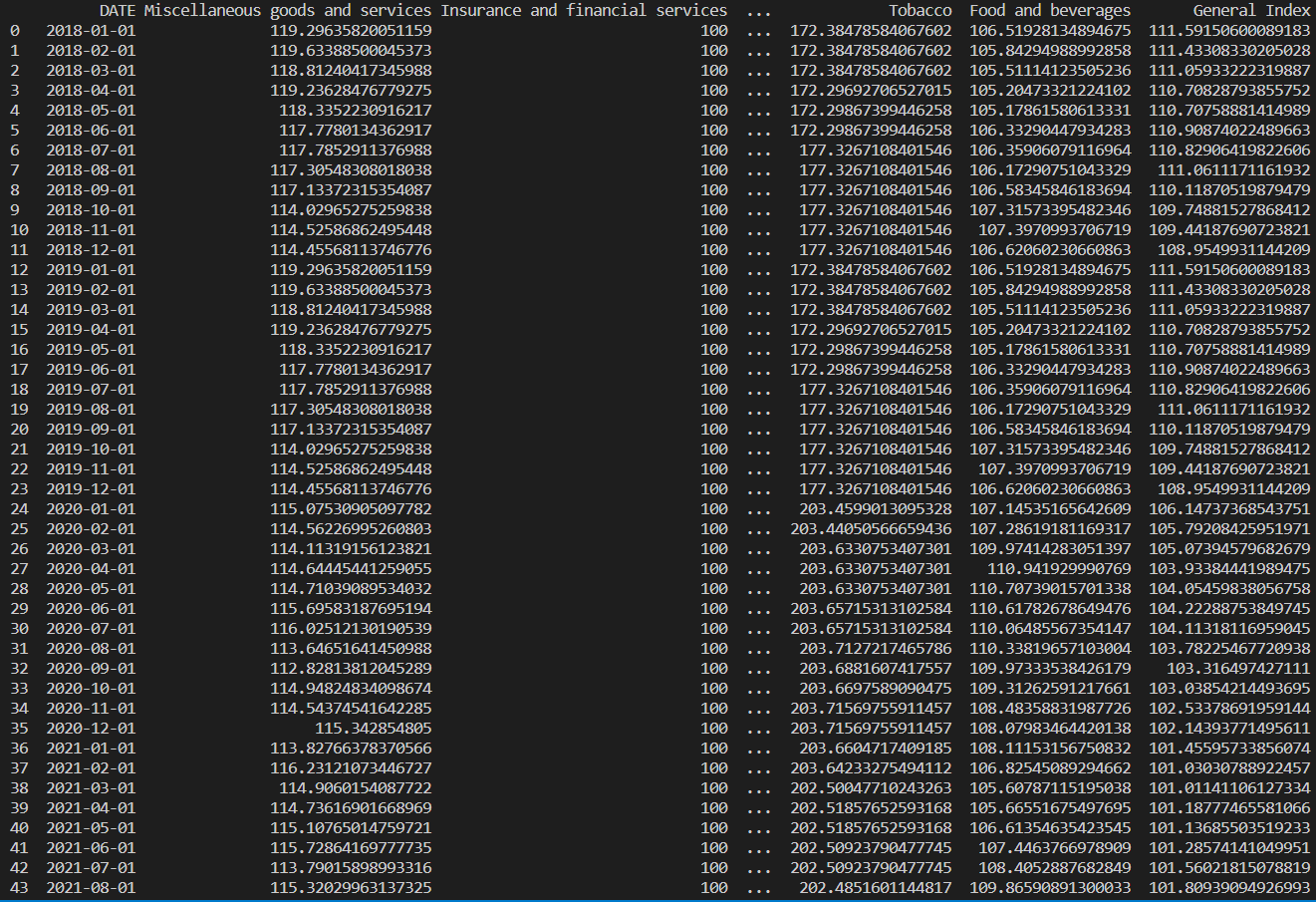


Blue: Train data

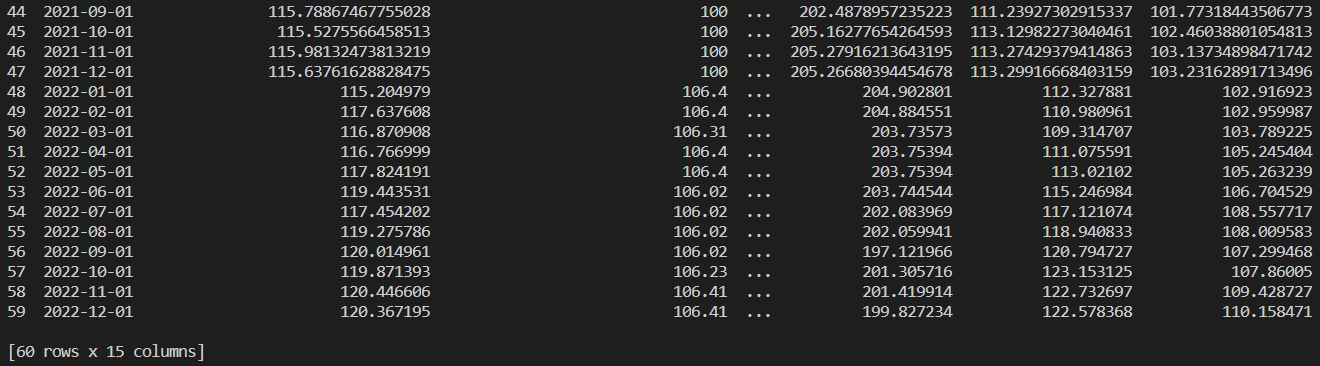
Orange: Test data

From the plot, we can see that 2019~2021 showed a dip due to the pandemic restrictions. The orange line indicates the Test set.

Train Dataset:

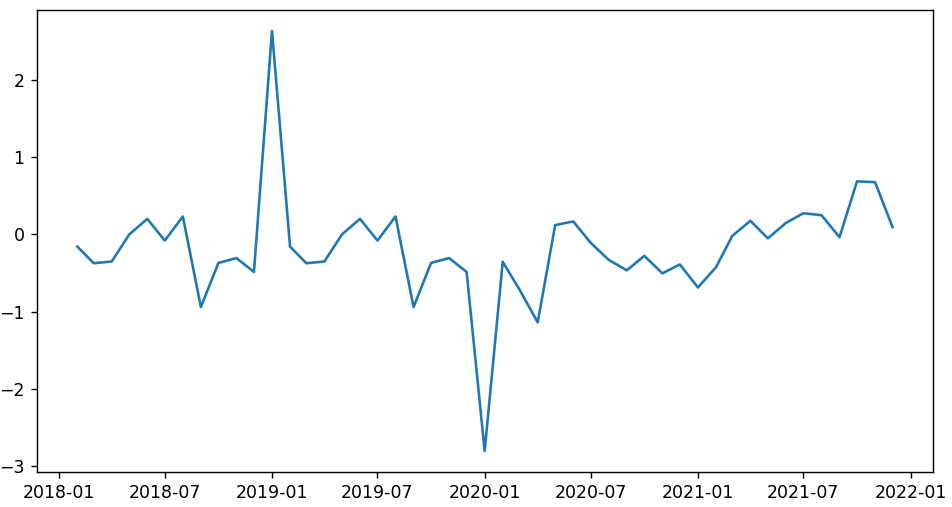


Test Dataset:



1.5.3 Take first differences

Here, we are finding the number of optimal differencing to remove unit root so that the time-series is stationary. This is done by using diff() function and testing with Augmented Dickey-Fuller test.



### 1.5.4 Augmented Dickey–Fuller test

With the small p-value, a 1 differencing is enough to remove unit root and make the series stationary.

diff = diff.dropna()

def adf\_test(df):

    result = adfuller(df.values, autolag = 'AIC')

    # print(result)

    if result[1] > 0.05:

        print("Series is not stationary")

    else:

        print("Series is stationary")

adf\_test(diff)

The result:



As you can see an above result, the series is stationary which tells us to apply ARIMA model.

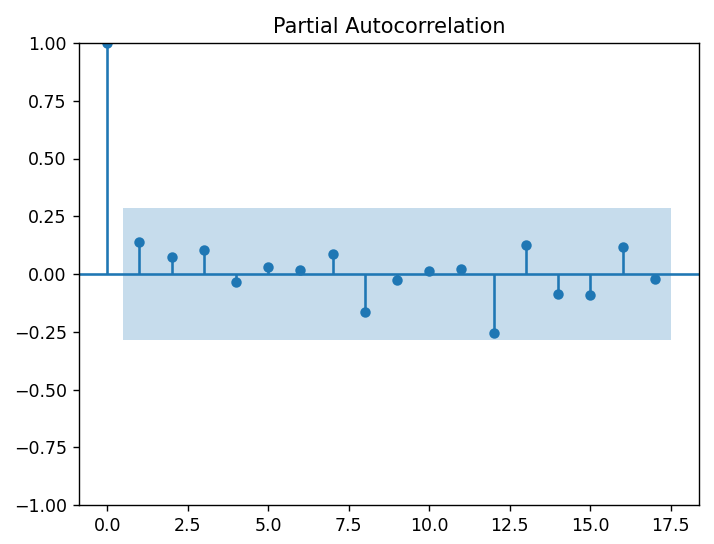
### 1.5.5 Plot ACF and PACF

The autocorrelation function (ACF) and the partial autocorrelation function (PACF) are to be examined to find an appropriate ARIMA model for the unemployment rate time series. ACF and PACF are also examined in order to see if autocorrelation is present in the unemployment rate time series.

Now, we need to find the optimal p and q using acf and pacf plot. Where p is the number of lags and q is the order of the MA term.

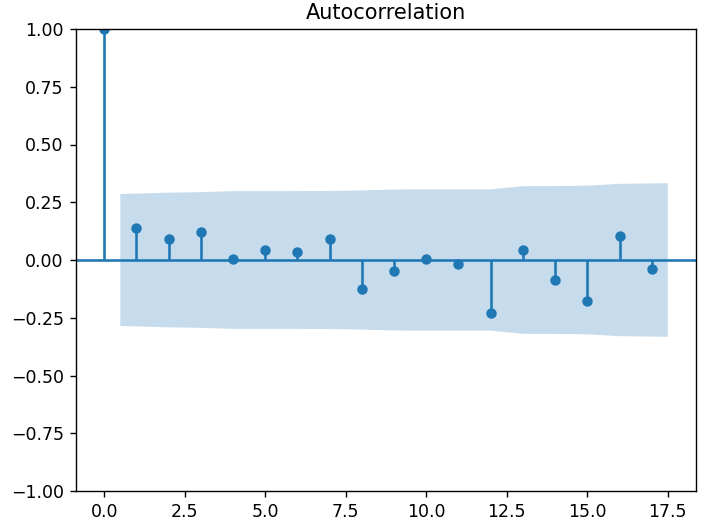
Finding the order of Auto Regressive Term (p)

* PACF lag 1 is significant



Finding the order of Moving Average Term (q)

* q = 1 and 2 is significant, try conservative take of q = 1.



Above diagram shows the autocorrelation function for the CPI rate in Dubai. Lags one, two and three are highly significant meanwhile lag number four are barely significant. However, the significance of the lags die out rather quickly, indicating that autocorrelation is not present. The figure also shows the partial autocorrelation function for the CPI rate. The first lag is highly significant, the second lags is just significant and the rest of the lags quickly drops to insignificance. However, lags number four and sixteen appears to be significant as well. After examining the above figure, a model containing an autoregressive component of order 1, 2 or 3 may be suggested.

1.5.6 Building the model

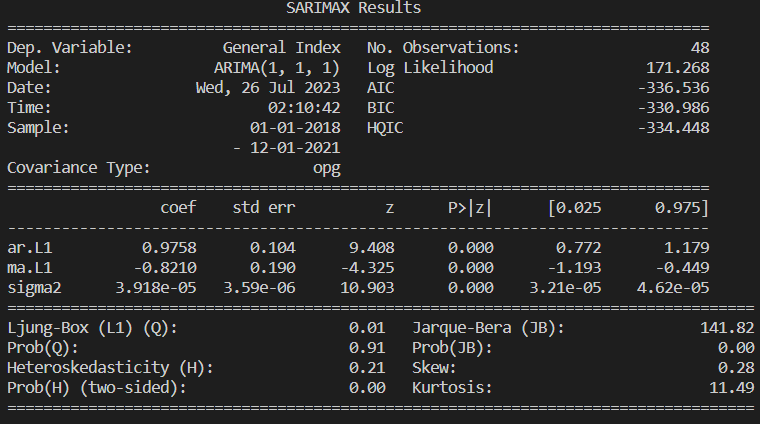
Since CPI exhibits exponential growth (variance increases), we build the model on the ln(CPI) e.g. converting the raw values to log values.

As earlier discovered, the ARIMA model parameters will be set as 1,1,1.

arima\_model = ARIMA(np.log(train['General Index']), order = (1,1,1))

arima\_fit = arima\_model.fit()

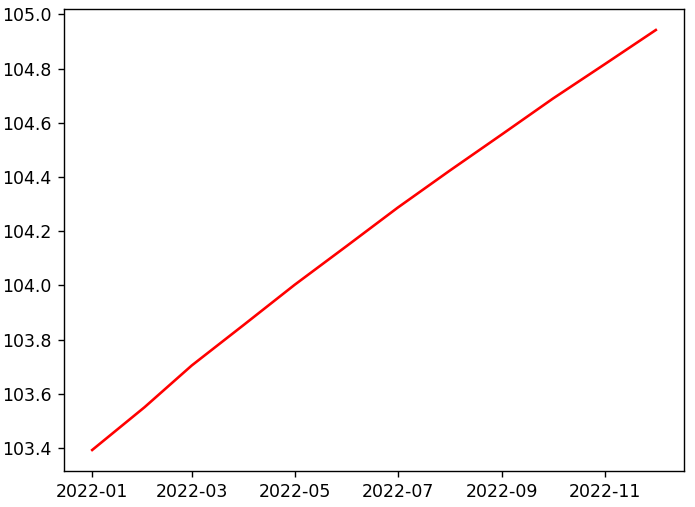
print(arima\_fit.summary())



1.5.7 Forecast for January 2022 – December 2022

Forecast for the next 12 months (12 out-of-sample)

When we forecast with time series models, we receive three values for each observation: Mean, Low, and high. The Mean value is the actual forecast. The low is the confidence interval of that forecast doing a downward trend, and the high is the confidence interval of that forecast going upward. We got a result for each department and police station. As shown for example in protective security emergency department electricity consumption model.





The forecast shows that for most of the groups, there is no clear seasonality or trend, which results in constant mean predictions for these groups (as ARIMA models are mean models). However, some groups average changes each quarter, so we can also see predictions accommodate that seasonality. The forecasts accuracy depends on past values as its essence of time series model).

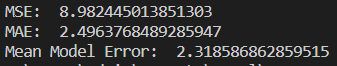
In plots, we can see that some locations have cyclical pattern and some locations have no clear trend. This can also be seen in forecasts. When ARIMA found clear trend plus seasonality. the forecasts are good. But when observations have no pattern, the predictions follow same mean for all forecasts. The visualizations with no seasonal pattern and no clear trend, results in poor predictions.

1.5.8 Evaluating the ARIMA model

mse = mean\_squared\_error(test['General Index'].values, forecast[:12])

mae = mean\_absolute\_error(test['General Index'].values, forecast[:12])

The result:



The mean squared error (MSE) of 8.982 indicates that the model's predictions were off by an average of 8.982 units squared. This metric is useful for evaluating the overall accuracy of the model, but it can be sensitive to outliers and may not provide a complete picture of the model's performance.

The mean absolute error (MAE) of 2.49 indicates that the model's predictions were off by an average of 2.49 units. This metric is useful for evaluating the magnitude of the errors and can be more robust to outliers than MSE.

The mean model error of 2.318 indicates that the model's predictions were consistently off by an average of 2.318 units. This metric is useful for evaluating the bias of the model and can help identify any systematic errors or trends in the model's predictions.

Overall, these evaluation metrics suggest that the ARIMA model may have some room for improvement in terms of accuracy and bias. As a data science engineer, I would investigate potential ways to improve the model, such as adjusting the model parameters, or using other approaches like LSTM.

1.5.9 Forecast for January 2023

arima\_model = ARIMA(np.log(test['General Index']), order = (1,1,1),freq=test.index.inferred\_freq)

arima\_fit = arima\_model.fit()

forecast = arima\_fit.forecast(steps=1)

forecast = np.exp(forecast)

The result:



Conclusion

2.1 Summary

The summary of ARIMA model. Showing some model accuracy measures.

|  |  |
| --- | --- |
|  | General Index |
| MSE | 8.98 |
| MAE | 2.49 |

2.2 Conclusion

This research has resulted in several conclusions as follows. Firstly, ARIMA can be used to predict the Dubai CPI value from January 2018 - December 2022. The ARIMA model

applied to this forecast is ARIMA (1,1,1) and produces CPI value of 103.39.

Secondly, this method can be used for forecasting, produced an MSE of 8.89.

The resulting MSE is quite small and so it can be said that the ARIMA forecast results are at a reasonable level and can still be calculated.

2.3 Future Works

It is known that ANN and LSTM are better than ARIMA in performance due to their non-stationary.

Although for this research work the dataset was somehow limited.

For improving of the accuracy of forecasting, we need more dataset, at least for more than 20 years.

Then we can use LSTM model to predict the trend of CPI with high accuracy.