Executive Summary

The Consumer Price Index (CPI) is one of the most widely used statistics to determine annual cost-of-living allowances for social security retirees and other recipients of federal payments, to index the federal income tax system for inflation and deflation. The report is aimed to develop a predictive model to forecast the quarterly CPI measures from Jan 2020 to Dec 2021 based on a provided dataset containing historical quarterly values. Further, the analysis of mean squared error (MSE) is allowed to use for better checking the accuracy of the predicted model.

Background

The provided CPI_train.csv includes the 120 data points of the quarterly CPI dataset with some added noise for the de-identification purposes during the time of Jan 1990 until Dec 2019, which will be used to forecast 8 data points from Jan 2020 to Dec 2021.

Firstly, the technique of Exploratory Data Analysis (EDA) will be analyzed to help us get a general understanding of the number of train dataset and their data type, as well as the problem of train datasets, which has been solved. As for the part of the methodology, there are seven models to be used for prediction, such as decomposition, drift method, Naïve, Holt-Winter, ARIMA and Feedforward Neural Network and Recurrent Neural Network (LSTM). Eventually, we are allowed to analyze the mean squared error (MSE) to better check the accuracy of the predicted model.

Exploratory data analysis

To get a general understanding of the data for further analysis, it is important to use the technique of exploratory data analysis (EDA). In this way, a suitable analysis can be selected for different data. The first step is to load data that checks what kind of columns are in the dataset. It is obvious to observe that the training dataset belongs to time series data. Then the type of date column has been changed from object to DateTime so that it can be processed quickly and ordered by the exact date when plotting. The technique of EDA is supported in the process of data understanding, which is analyzed for its information in the following:

- It records the exact time when the feature of CPI.
- There are no missing values or duplicates.

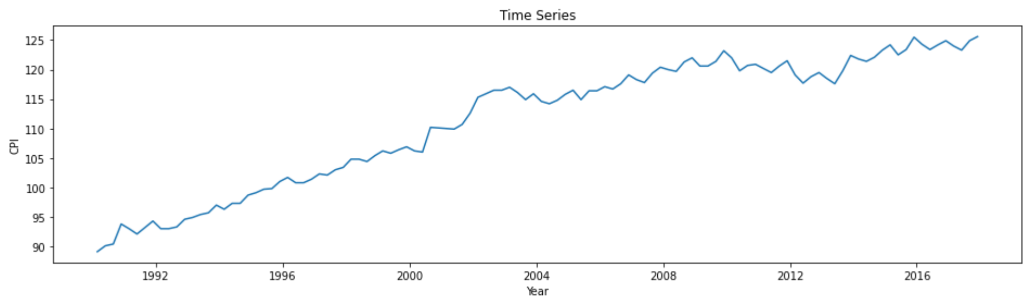
Likewise, the test dataset has been applied to analysis in the same technique, which has changed the type of date columns object to DateTime and identified its information to get a deeper understanding. That shows there are 8 non-null values, and it recorded the exact time for every CPI value. Importantly, it is necessary to check both the detailed description of the train and the test dataset.

As the purpose of this project is to forecast the quarterly CPI measures from Jan 2020 to Dec 2021 which has a total of 8 data points, it is significant to split the training and validation set from the train dataset, respectively for 112 and 8 data points. Therefore, the 112-training set will be used for analysis and the 8 data points in the validation set will be forecasted to check the accuracy.

1. The Analysis of Decomposition

To plot the time series of training set data, it is very hard to observe if it belongs to a multiplicative or additive trend. Therefore, it is important to use the structural decomposition model that allows to identify the main trend of observed changes over time of CPI and impacts, as well as provide a better understanding of problems during the process of time series analysis and prediction.

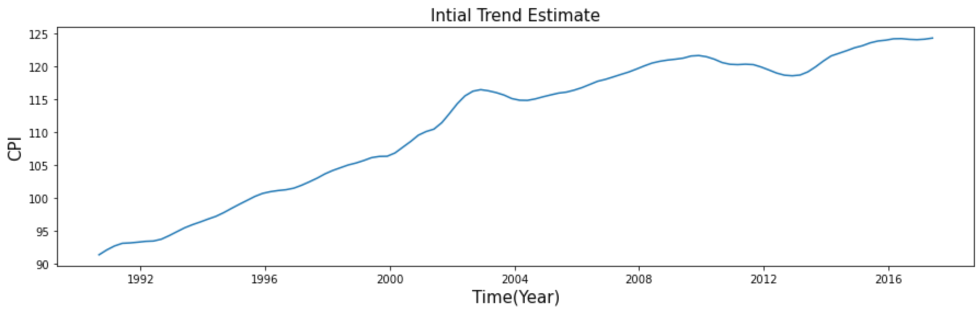
- It is obviously an uptrend with quarterly seasonality.



Time series decomposition includes a series thinking, such as a combination of level, trend, seasonality, and remainder components. The challenge of using a decomposition model to forecast a time series is to calculate the future values for each separate component and then add them back together to obtain a prediction. There are five steps in the process of decomposition as shown the following steps:

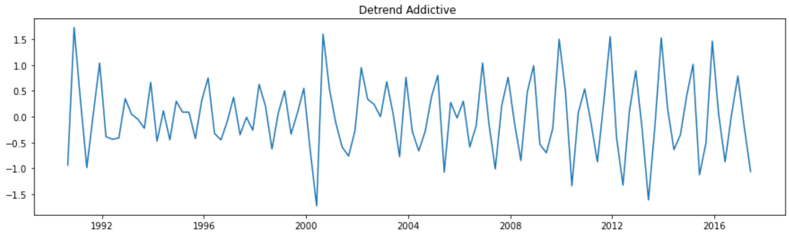
1.1 Smooth the data to remove seasonality

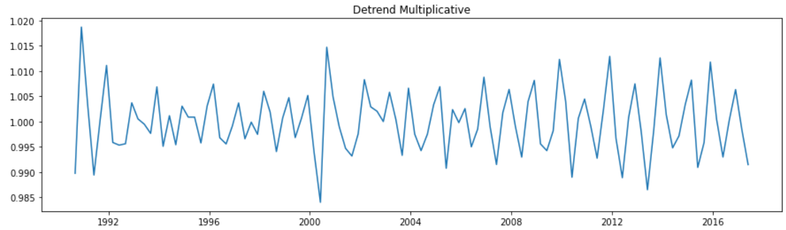
It is clear to see time-series datasets containing a seasonal component, which is a quarterly cycle that repeats over time that may obscure the signal to predictive models. Therefore, we have decided to apply the moving average (MA = 4) period for removing the seasonal component, and this process is called a seasonal adjustment, in which the trend in the data can be more linear so that the plot of the initial trend estimate can be shown in the below.



1.2 De-trend the original series.

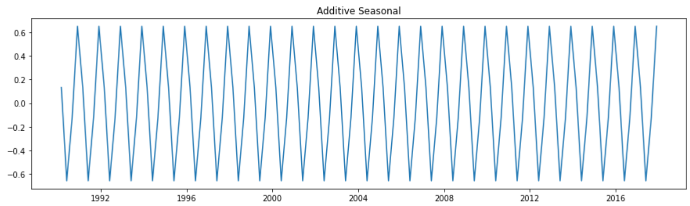
It is important to distinguish if it is for the additive or multiplicative decomposition from seasonal variation. However, from the following plots, it is very hard to observe, so we decided to apply both additive and multiplicative decomposition to analyze. There is only one difference between them, which is that an additive decomposition is done by subtracting the trend estimates from the series, and a multiplicative decomposition is done by dividing the series by the trend values.

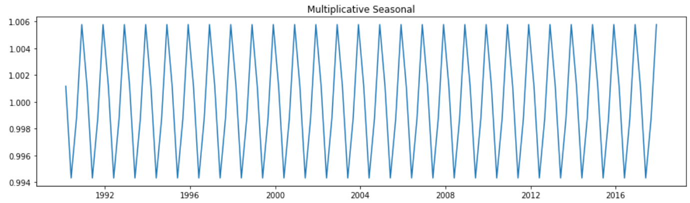




1.3 Estimate the seasonal indices based on the detrend component and normalization

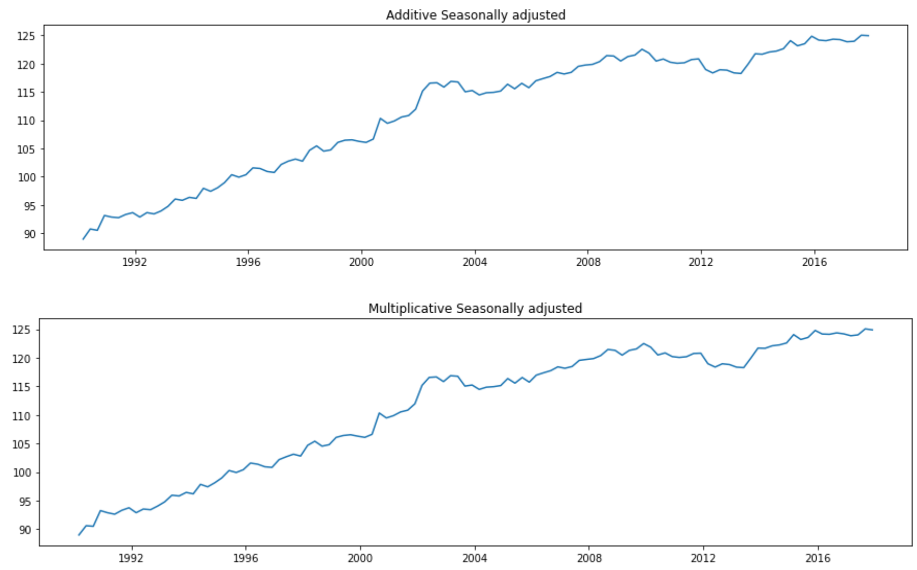
In this step, it is essential to calculate the mean (or median) values of the detrended series for quarterly periods, estimating the seasonal effect and averaging the de-trended values. Then normalization plays an important role in the seasonal effects, for instance, an additive model is adjusted so that their sum is equal to 0. As for the multiplicative model, the d seasonal effects are adjusted so that they average.





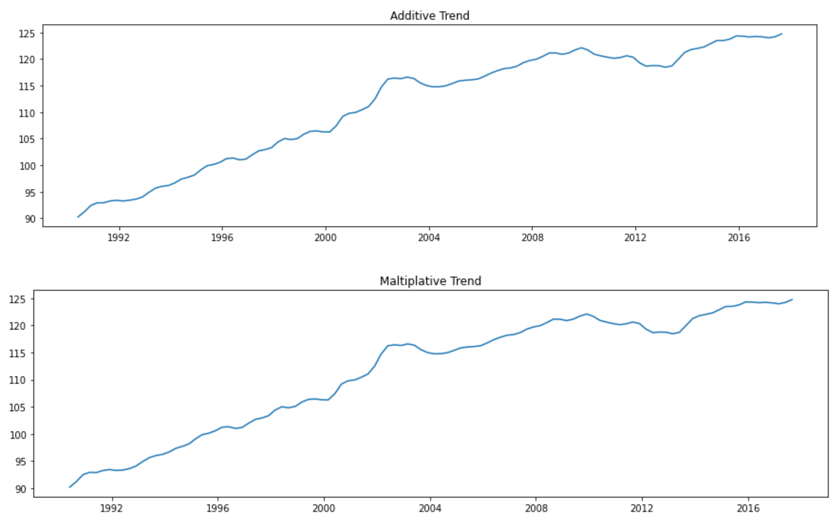
1.4 Seasonally adjusted

When forecasting, it is advantageous to remove seasonality that allows a forecaster to concentrate on predicting the general trend of the data.



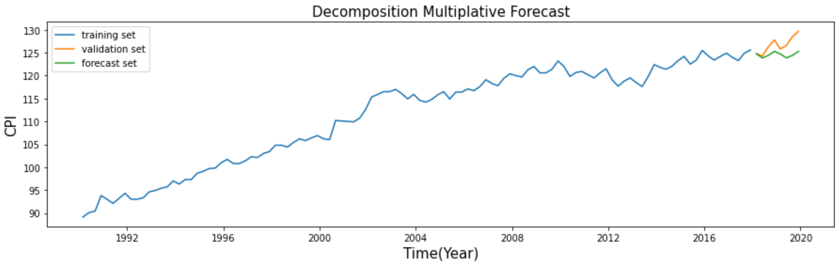
1.5 Obtain trend component

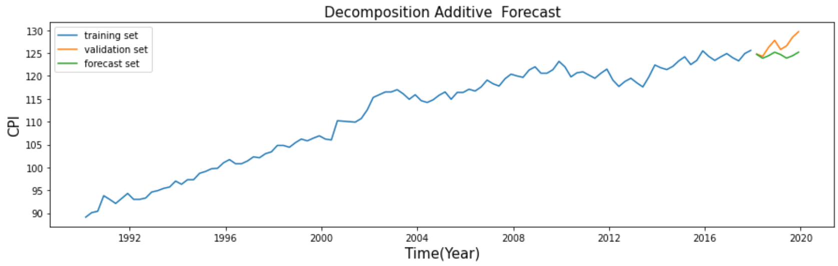
The final step is to determine the trend component by using to calculate a moving average cantered on MA = 2.



1.6 Forecasting

After using the decomposition methods, we gain the final trend to further forecast the validation set. As mentioned above, the train dataset would be split into two parts which is the training set and the validation set. To observe the following plots, there is a slight difference between the forecast set and validation set. According to GeeksforGeeks (2019), regarding the value of Mean Squared Error (MSE) that the average squared difference between the estimated values and true value, it shows that the additive model is calculated to 6.94 and the value of 6.83 is for the multiplicative model.





1.7 Advantages and Disadvantages

The method of decomposition largely increases the productivity that the complexity model can be braked into small sections in order to run many lines of code at the same time, which are more manageable and easier to understand. However, compared with exponential smoothing, it is superior in forecasting because it places relatively more weight on the most recent observation. The technique of decomposition simply assumes the equal weight, ignoring the uncertainty in the forecasts.

2. Seasonal Naïve Method

Model descriptions

Naïve Method Formula: $\hat{Y}_{t+h}$=Y

Seasonal Naïve Method Formula: $\hat{Y}_{t+h}$=Y_{t+h-M}

The Naïve method is the simplest way to predict but is very efficient sometimes and it can be used as basis for other prediction methods. This method considers what happened recently and predicts that the same thing will happen in the next period ("Forecasting Methods: What is Naive Forecasting Method?", 2022). In other words, the Naïve method is a technique that uses data from the previous time as a forecast value for the next period. The Seasonal Naïve forecasting method is a technique based on Naïve forecasting method. It uses the value of the corresponding time of the previous period as the forecast value of the same time of the next periods ("How You Can Use Naïve Forecasting for Your Business", 2022). For example, the value of May of the last year as the forecast value of May of all subsequent periods. According to above analysis, for following fitting and forecasting process of this report, the period is 4 months.

The reason of choose this model

The Seasonal Naïve method is the most basic of the models that consider seasonality, so it is necessary to use it in the report because it can be seen as a benchmark to compare with other models.

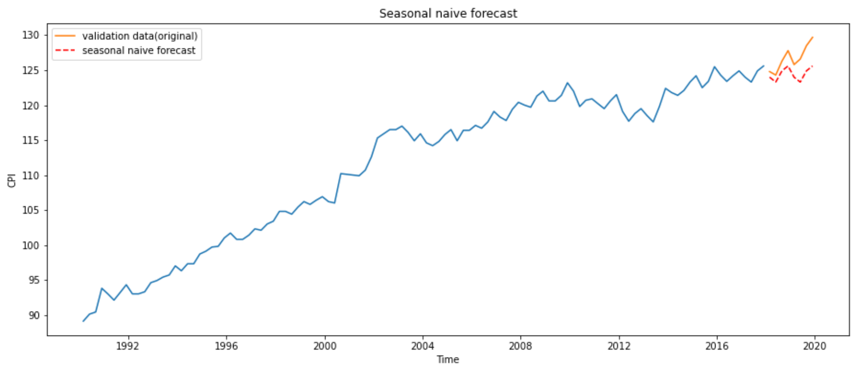
Processing steps and forecasting

The original data has been divided into training data and validation data above, where training data is used for training the model and validation data is used for evaluating the accuracy of prediction choosing the model. Since this report needs to predict eight values, and the period of seasonal is four, we pulled out the last four values of training data, and then copy these four values once to get eight values. Then, according to the characteristics of the Seasonal Naïve model, the eight values obtained are the predicted values.

Result visualization and Forecasting error

It can be seen from the graph below that the predicted results of Seasonal Naïve deviate from the actual values. It ignored the increasing trend of CPI data, so it is lower than the real value.

In addition to the graph, this report also measures the forecasting error based on the MSE, and the value of forecasting error calculated by MSE is 6.5425, which shows that the Seasonal Naïve method is not good enough for the CPI data.



Advantages and Disadvantages

First, because the Seasonal Naïve forecasting method uses previous data to make predictions, it is very simple and easy to implement. In addition, unlike the black-box algorithm, seasonal naïve is also very easy to interpret (McHenry, 2022).

But this method also has disadvantages. Since this method only considers the seasonal factor and not the trend (up or down), it only can be used in dataset where the values are stable. Otherwise, the forecasting result will not accurate when the data have significant trend changes like the CPI data set.

3. **Drift Method**

Model descriptions

Formula: $\widehat{Y_{t+h|t}}=Y_t+\frac{h(Y_t-Y_1)}{t-1}$

Unlike the naive method, the drift is not a constant prediction, it increases or decreases over time, which indicates that it considers the trend changes ("Some simple forecasting methods", 2022). This method takes two data, the first number of the training set and the last data of the training set. Then, drawing a line between these two points and extends it, the prediction value will locate at the point of the extension line corresponding to the predicted time.

The reason of choose this model

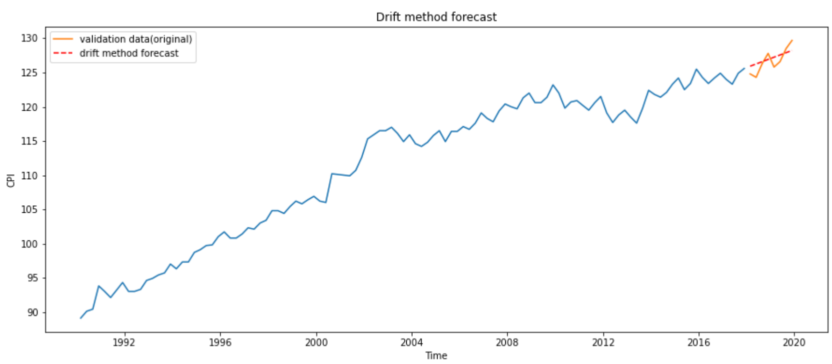
According to plots in EDA and decomposition, we found that the CPI dataset does not only has seasonality, but also a clear upward trend. Therefore, we must take the trend into consideration when predicting, and the Drift model is the basic model to allow the forecasts to increase or decrease over time.

Processing steps and forecasting

First, I found the first and last numbers in the training set by code, which are 89.1 and 125.6, respectively. That is, Y_1=89.1 and Y_t=125.6. In addition, the number of training set is 112, so t=112. Then I used a for loop to put h equal to 1 to 8 and the data known above into the equation to calculate eight values in turn, which are the prediction values for the period from March 1, 2018, to December 1, 2019.

Result visualization and Forecasting error

According to the figure below, it can be seen that the predicted values of the Drift method are close to the true values. This may be due to the more stable upward trend of the training set since 2013. The data demonstrates this result more visually. The MSE of the drift method is 1.4401, which is a significant decrease compared to the seasonal naïve, indicating that the drift method performs significantly better than the Seasonal Naïve method on the CPI data set.



Advantages and Disadvantages

First of all, the Drift method, like the Seasonal Naïve method, is a very simple forecasting method that does not require code to calculate it, people can calculate it manually by using formulas, which leads it to be easy to implement and understand ("Forecasting - Drift Method", 2022).

However, the Drift method only considers the trend but lacks consideration of seasonality. In addition, this method only cares about two values, the first and the last, which indicates that it ignores all other data. Although this method performs well in this report, it is because there is a relatively stable upward trend in the training data, which favours this method to fit. However, this is a coincidence and ignoring data cause errors most time when making forecasting using unstable data sets ("Forecasting - Drift Method", 2022).

4. **Holt-Winters Seasonal Method**

Model descriptions

Additive

$l_t = \alpha(y_t - S_{t-M}) + (1-\alpha)(l_{t-1} + b_{t-1})$

$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1}$

$S_t = \gamma(y_t - l_t) + (1-\gamma)S_{t-M}$

$\widehat{y_{t+1}} = l_t + b_t + S_{t+1-M}$

$\text{SSE} = \sum_{t=1}^{T}(y_t - l_{t-1} - b_{t-1} - S_{t-M})^2$

Multiplicative

$l_t = \alpha(y_t / S_{t-M}) + (1-\alpha)(l_{t-1} + b_{t-1})$

$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1}$

$S_t = \gamma(y_t / l_t) + (1-\gamma)S_{t-M}$

$\widehat{y_{t+1}} = (l_t + b_t)S_{t+1-M}$

$\text{SSE} = \sum_{t=1}^{T}(y_t - (l_{t-1} + b_{t-1})S_{t-M})^2$

Overall, the Exponential Smoothing method of forecasting is a weighted average of previous observations, and the weights decay exponentially over time. (Qbus6840, Lecture4, s1, 2022). There are three forms of this method, which are classified as single exponential smoothing, quadratic exponential smoothing, and triple exponential smoothing. The single exponential smoothing method, also called Simple Exponential Smoothing (SES), does not consider trend and seasonality, therefore not suitable for forecasting CPI data. Quadratic exponential smoothing, also called Trend corrected exponential smoothing (TCES), considers trend but not seasonality, therefore not suitable for forecasting CPI data too. The triple exponential smoothing method also called the Holt-Winters smoothing method, which considers level, trend, and seasonality and this is why it is called triple exponential smoothing. In addition, the Holt-Winters smoothing method can be divided into additive smoothing and multiplicative smoothing depending on the calculation method. According to the analysis above, the seasonality of the training set has some variation but is not significant, so both methods will be used for forecasting. When forecasting, we use the forecasting equation and three smoothing equations, $\boldsymbol{l_t}$for level, $\boldsymbol{b_t}$ for trend, and $\boldsymbol{S_t}$ for seasonality, and the corresponding smoothing weights α, β, and γ. In addition, t represents the time and M is the period. The formulas are as above (Qbus6840, Lecture4, s1, 2022).

The reason of choose this model

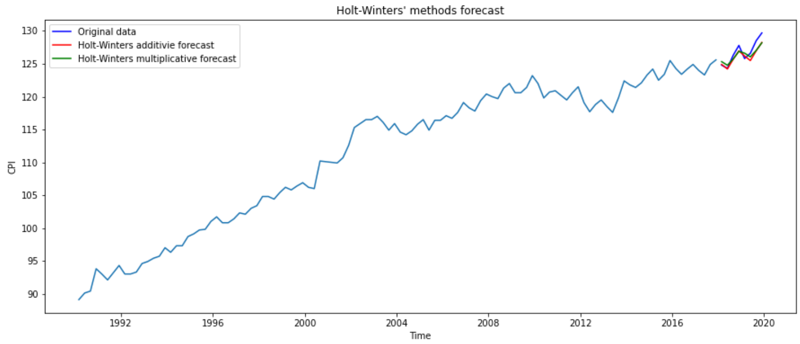
First, in terms of model interpretability, Holt-Winters is logically clear and easily interpretable. And in terms of the size of the dataset, the CPI dataset provided is small, and Holt-Winters is capable of predicting a dataset of this magnitude without overfitting, whereas a deep learning model would be more suitable for predictions with large data volumes (Chawla, 2020). In addition, it is repeatedly mentioned that Holt-Winters takes into account both seasonal and trend factors and is well suited for forecasting CPI datasets.

Processing steps and forecasting

First, we need to import the libraries, here we use the package Exponential Smoothing. Then, we call the Exponential Smoothing function to smooth the data. Since we use both additive smoothing and multiplicative smoothing, the keyword seasonal needs to be equal to add and mul respectively. Next, I call .params[] and call different attributes like smoothing level to get the values of α, β, γ, $l0$,$b0$. After getting the best parameters respectively additive method and multiplicative method, we call .forecast() to make the prediction and plot the forecasting values.

Result visualization and Forecasting error

In the figure below, the red line is the holt-winters additive forecast, and the green line is the holt-winters multiplicative forecast. The difference between the predicted data using the additive method and the multiplicative method is not significant, and the predicted data using both methods are close to the real data.



The MSE values also show that Holt-Winters method performs well. The deviation of Holt-Winters additive method is 0.8627 and the deviation of Holt-Winters multiplicative method is 0.8761 for the. Although the difference between the two is not significant, but Holt-winters additive fits the data better.

Advantages and Disadvantages

Compared with deep learning forecasting methods, Holter-Winters smoothing method is relatively easier to understand and apply. In addition, compared with the Naive method and Drift method, Holter-Winters smoothing method can effectively respond to changes in time series because it considers not only the level but also the seasonality and the trend. And most of the real-world data are usually seasonal and trendy, so Holt-Winters is also more widely applicable.

Although Holt-Winters has many benefits, it requires rigorous data pre-processing to remove anomalous data, which would otherwise produce serious prediction errors (Chief, 2016).

5. SARIMA Model Analysis

This section is about using the Seasonal ARIMA model (SARIMA), which is a formal statistical time series model that can be used to help capture complex underlying patterns in the time series rather than trends and seasonality, as the training dataset contains seasonality.

The first step is to examine the plots of the ACF and PACF for the raw data. By using the plots below and combining them with the relevant ACF definitions, the ACF dies down extremely slowly, then it should be considered nonstationary, and also presumably, the mean of the data is also nonstationary. So it is necessary to do the data transformation for model fitting.

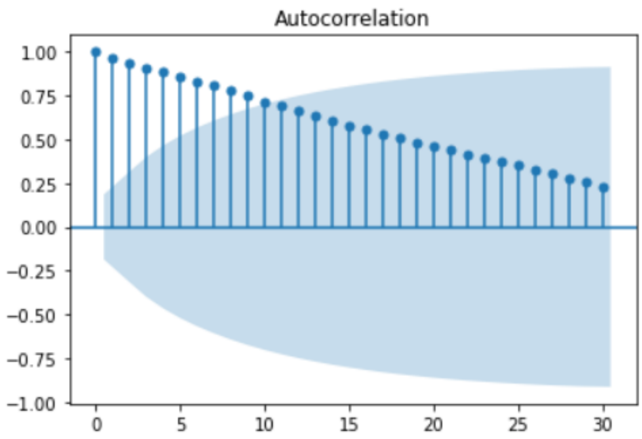


Figure 1 Original data ACF plot

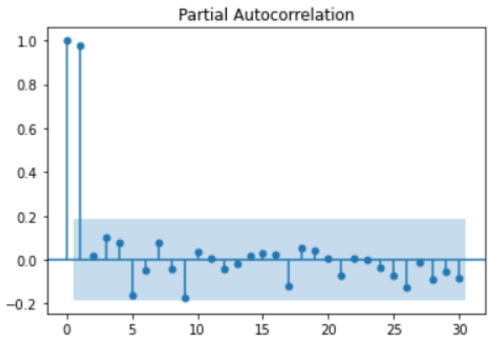


Figure 2 Original data PACF plot

The next step is to log the data (log_train), which reduces the volatility of the data in terms of variance, but is not very helpful for the previous problem - the mean, and does not actually change significantly when compared to the previous image of the original data.

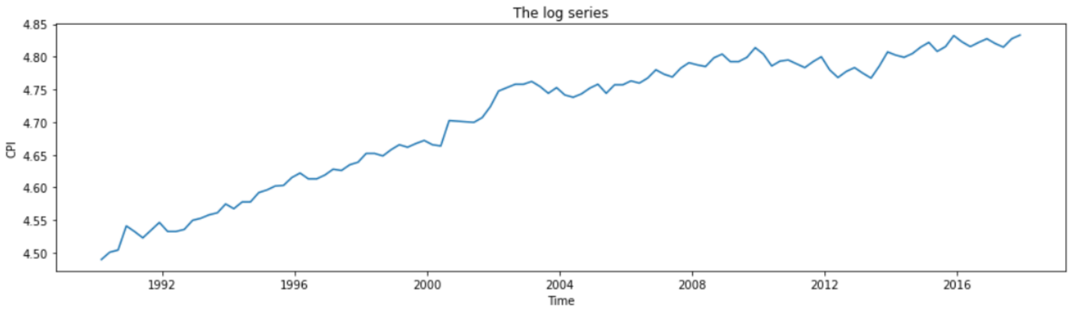


Figure 3 The log data plot

Also, the ACF and PACF for this part did not change very much compared to Figures 1 and 2, suggesting that log is limited in its optimization of the data, as the ACF did not disappear rapidly. Thus, the next step is to perform a 1st order difference on log_train and re-check the ACF and PACF of the difference data.

Step 3 is to do the 1st order differencing. It is clear from picture 3 that time is not stationary as it tends to increase. And there is a fact that SARIMA is needed to be stationary data as input, and the stationarity of the time series is helpful to select the appropriate integration to go for model fitting.

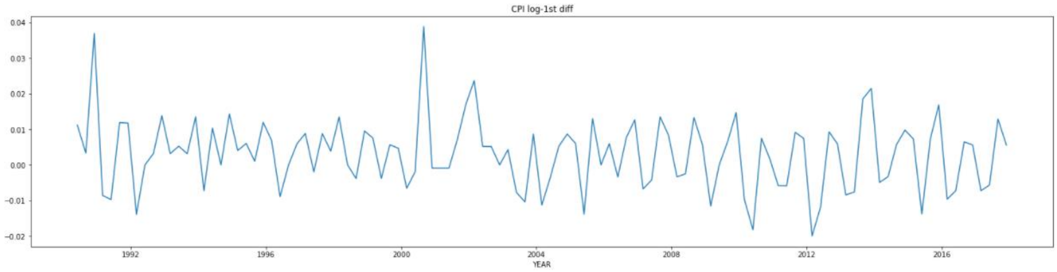


Figure 4 The 1st-differencing plot

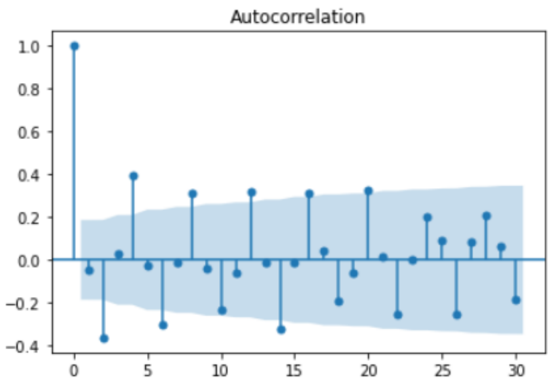


Figure 5 The 1st-differencing ACF plot

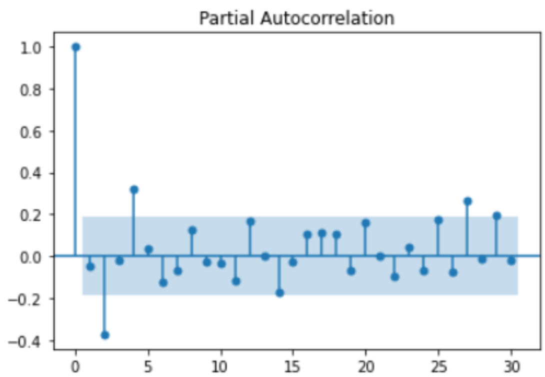


Figure 6 The 1st-differencing PACF plot

A comparison of Figures 3 and 4 shows that the data has been processed to meet the requirements, that the data is smooth over the entire time series and that there are very obvious cut off points in both the ACF and PACF where the parameters of the model can be selected for fitting.

The ACF and PACF above can be used to check the stationarity of the data. However, for some complex cases, it may not be easy to check whether the data is stationary or not. Therefore, to avoid controversy, step 4 will use the Dickey-fuller test to ensure that the data being processed is stationary. The null hypothesis for this test is $\boldsymbol{H_0}$: the time series is non-stationary. The p-value of the test can be checked to determine whether the time series is smooth by deciding whether to reject the null hypothesis $\boldsymbol{H_0}$.

```
Results of Dickey-Fuller Test:
Test Statistic                  -1.751297
pvalue                           0.404895
#Lags Used                       6.000000
Number of Observations Used    105.000000
Critical Value (1%)             -3.494220
Critical Value (5%)             -2.889485
Critical Value (10%)            -2.581676
dtype: float64
```

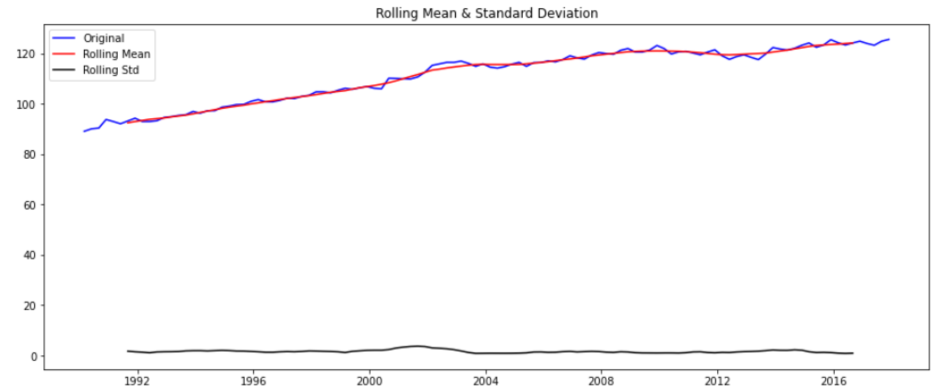


Figure7 Dickey-Fuller test for the original series

```
Results of Dickey-Fuller Test:
Test Statistic                  -4.492389
pvalue                           0.000203
#Lags Used                       3.000000
Number of Observations Used    107.000000
Critical Value (1%)             -3.492996
Critical Value (5%)             -2.888955
Critical Value (10%)            -2.581393
dtype: float64
```

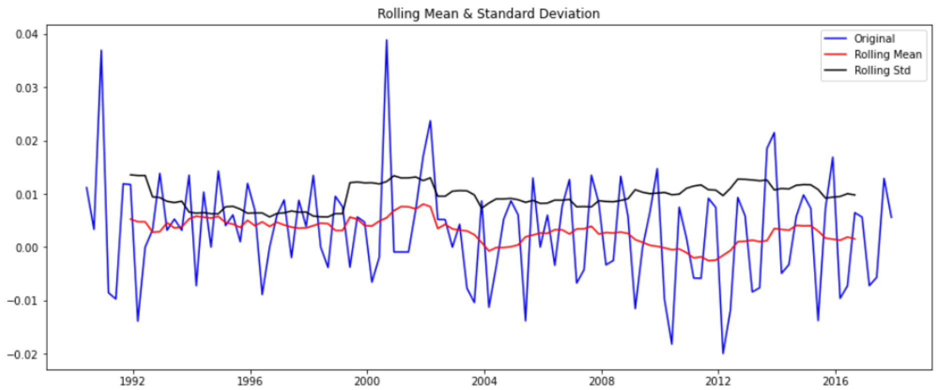


Figure 8 Dickey-Fuller test for 1st differencing series

As can be observed in Figure 7, the p-value is 0.40, which is above the significant levels at 1%, 5% and 10% then we do not reject the $\boldsymbol{H_0}$, which indicates that the original time series is non-stationary. However, from Figure 8 it can be observed that the p-value is 0.000203, which is smaller than significant levels at 1% then we reject the $\boldsymbol{H_0}$ at significant level 1%. So this proves that the judgement made in the previous section is correct and that the data can be fitted using the first order difference.

The step 5 is to select the best model by means of the Akaike Information Criteria (AIC). After screening by AIC, the optimal (*p, q*) of the model is (2, 2)

Through the ACF, PACF plot and AIC, which suggest to use *p*=2 and *q*=2.There is also a big value at lag 4 in the ACF plot (Picture 5), so the season will be 4. Since, PACF at lag *m*=4 is positive it suggests *P*=1. Similarly, ACF at lag 4 is positive and thus *Q*=0. And there is a differenced series for SARIMA and the seasonal pattern is stable over time, so *d*=1, *D*=1. Finally, the values for SARIMA model are $(2,1,2)(1,1,0)_4$.

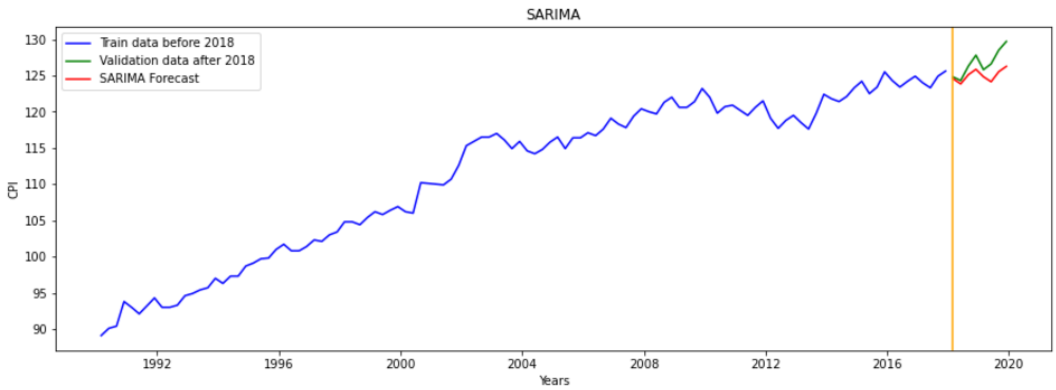


Figure 9 The SARIMA model plot

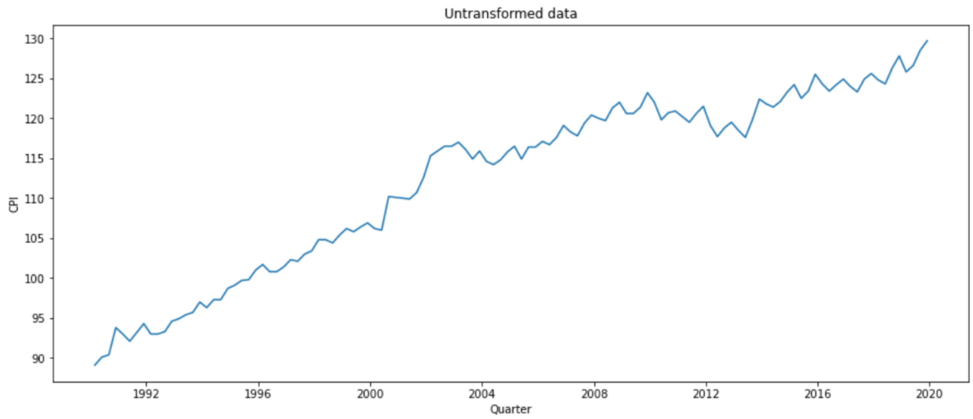
Step 6 is to do the model fitting, as can be seen through image 9, there is a slight difference between the SARIMA prediction and the original validation data, in this part, the MSE of SARIMA is 4.4480. It is worth mentioning that the author found that the larger the selection of *p, q*, the smaller the MSE of the model will be and the better the fit will be, but due to the limitation of the equipment, it is not possible to do the data calculation after lag 5 through AIC, so under the principle of maintaining rigour, the AIC suggested p, q was still chosen.

6. Feedforward Neural Network Model

Model description

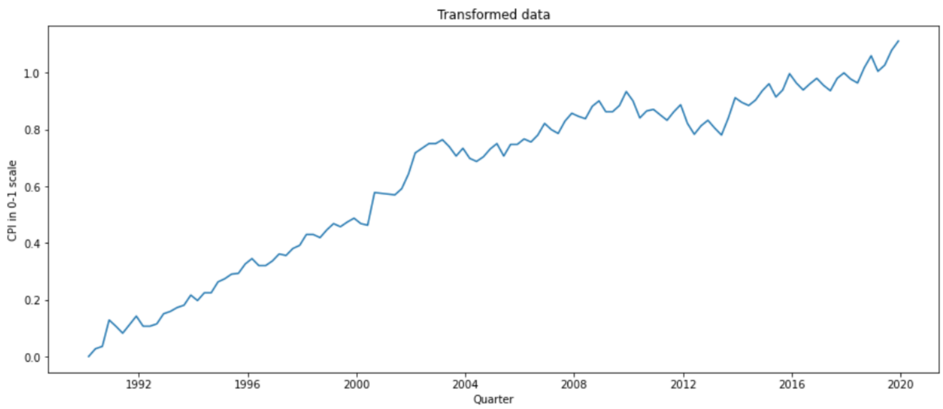
The Feedforward neural network is a simple artificial neural network. A simple neural network consists of an input layer, a hidden layer and an output layer, and the nodes connect the layers to form a neural network (IBM, 2021). The advantage of deep learning is that it can infer new features from the limited features set contained in the training set. To further investigate the trend and direction of CPI and measure the inflation of the country, we will use a neural network model for analysis.

Firstly, we read all 120 quarters of CPI data from 1990 to 2019, cleaned the data by removing Nan values, and converted the pandas data frame to an array format. We found that all the 120 data were clean with no nan values. Then, we plot the time series of the consumer price index. The horizontal axis represents time, and the vertical axis represents the consumer price index. As can be seen, the line chart shows that CPI has a gradually increasing trend. However, it had a drastic fluctuation between 2009 and 2013, which may be due to the global financial crisis and economic depression resulting in deflation.



Preprocessing Data & Train/Test Split

Secondly, we preprocessed the data to standardize the data set and scaled the smallest data to 0, the largest data to 1, and the rest data are between 0 and 1. The purpose of scaling data is to facilitate the construction of a neural network model and data processing. The time series chart below clearly demonstrates that the range of the CPI is between 0 and 1.



In addition, we will split the data and take the first 112 data as training data and the last 8 data as validation data to access the prediction performance of the trained neural network model.

Model building & Hyper-parameters

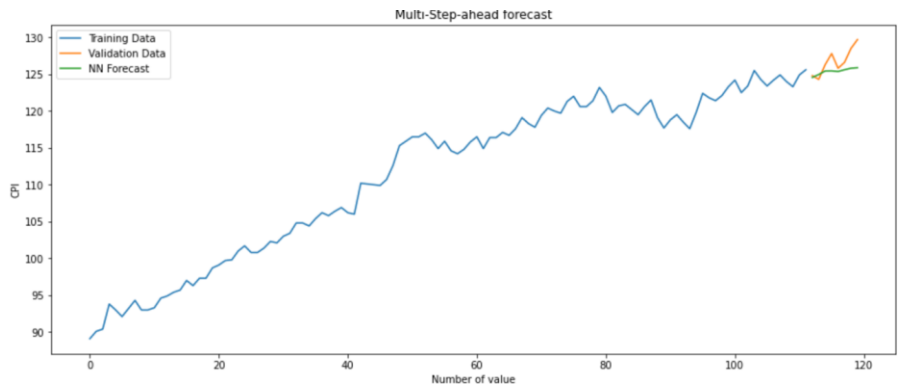
Thirdly, we're going to create the data feature that represents time series in a tabular format. The length of the window is how many past observations we use to predict a new observation. The analysis of the decomposition model in the first part proved that this group of data is seasonal, and the seasonal cycle is 4. According to Nick (2022), the input feature window must be greater than or equal to the seasonal period if the data have seasonality. Therefore, the window size was set to 9 after adjusting the hyperparameters several times.

Then, we started to build the neural network model. We set only one hidden layer because there's only 120 data. Furthermore, we also adjusted the hyperparameter of the number of neurons in the hidden layer to 22, and the number of input neurons was equal to the window size. We use Rectified Linear Unit (ReLU) as the activation function because its gradient is zero for all negative values of the inputs, which makes the parameters update faster (Jason, 20219). Moreover, we take a mean squared loss function and optimize it with Adam. It is an algorithm to perform a step optimization for the random objective function, and it is easy to implement (GeeksforGeeks, 2020).

Finally, we train our network by adjusting the hyperparameters of the epochs and the batch size. We set epochs to 150, batch size to 40. After training 150 epochs, the models tried to understand the patterns and behavior of the data. Because we split the data into training and validation sets, we can now predict the validation data values and compare them to the actual validation values.

Forecasting

The following chart illustrates the results of the multi-step forecasting method using the past 112 quarters of CPI data from 1990 to 2017 to forecast 8 quarters of CPI data from 2018 to 2019. We can see that this model does not perform well because the forecast CPI fluctuation is smoother while the real CPI growth is significant.



MSE

Substituting the true and predicted values into the equation of mean squared error to calculate the MSE is 3.7379. By comparing it with other models, we found that the neural network model does not have the best MSE. This is because the neural network model also has many disadvantages. For example, it usually requires a large amount of data, and the relatively small number of 120 data in this project leads to the poor performance of this model. Moreover, Tu (1996) pointed out that neural network modelling requires more computing power, which means that the calculation of the algorithm needs to spend a lot of time and cost to minimize the error. Also, the neural network model is easy to overfit, especially the external test data to the poor performance of the training data.

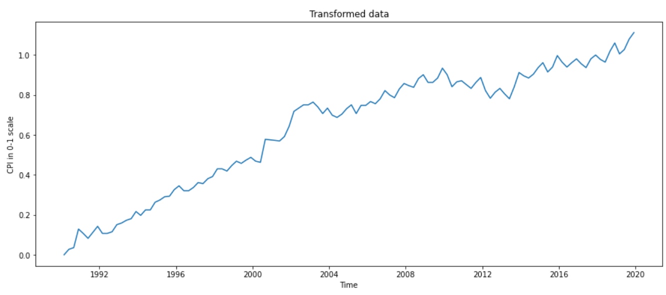
7. LSTM Model

Model description

(Recurrent neural networks) RNN is one of artificial neural networks, dealing with sequential data where data points have defined ordering (Schuster & Paliwal, 1997). RNN was inspired by David Rumelhart's work in 1986 (Rumelhart, Hinton, & Williams, 1986). There is an obvious limitation in the basic RNN. It is suffering from long-distance dependence problems (Bender et al. 2003) which means that it has difficulty dealing with long sequences, often referred to as exploding or gradient vanishing problem (Hochreiter, Bengio, Frasconi, & Schmidhuber, 2001). Long Short Term Memory (LSTM), a special type of RNN, was introduced in 1997 to overcome this limitation (Hochreiter & Schmidhuber, 1997). LSTM allows passed information to skip the processing of the current cell and move on to the next which enables it to retain the memory for a long sequence (Ghoshal, n.d.). This characteristic is extremely useful when dealing with time series data.

Model preparation

Since neural networks converge faster with scaled data, the time series data is scaled to the range between 0 and 1 using MinMaxScaler() function of scikit-learn. The following figure shows the time series after scaling.



Then the time series data is reconstructed in tabular format and splitted into training and validation set. The validation set is used to fit and evaluate the model performance.The size of validation set is defined as 8 in order to be consistent with the size of test set.

Modelling & Tuning hyper-parameters

The second step is to reshape the input data of train and validation sets to 3D array with the format of (# sample, # time window, # 1). The third step is to build and train the LSTM model. We define a LSTM model with one hidden layer. Adam opitimizer, MSE loss function, relu and linear activation functions are applied to train LSTM model.

In terms of hyper-parameters, ‘time window’, ‘units’, ‘batch size’ and ‘epoch’ are tuned to improve the model performance.

Time window is the number of past observations used to predict a new observation. It should be large enough to cover the seasonal period. While larger time window will lead to higher accuracy considering that more input data are used in prediction, it will lead to longer training time.

Units is the number of LSTM neurons in the hidden layer which affects the learning capacity of the network. In general, more neurons lead to higher accuracy but also causes the problem of overfitting.

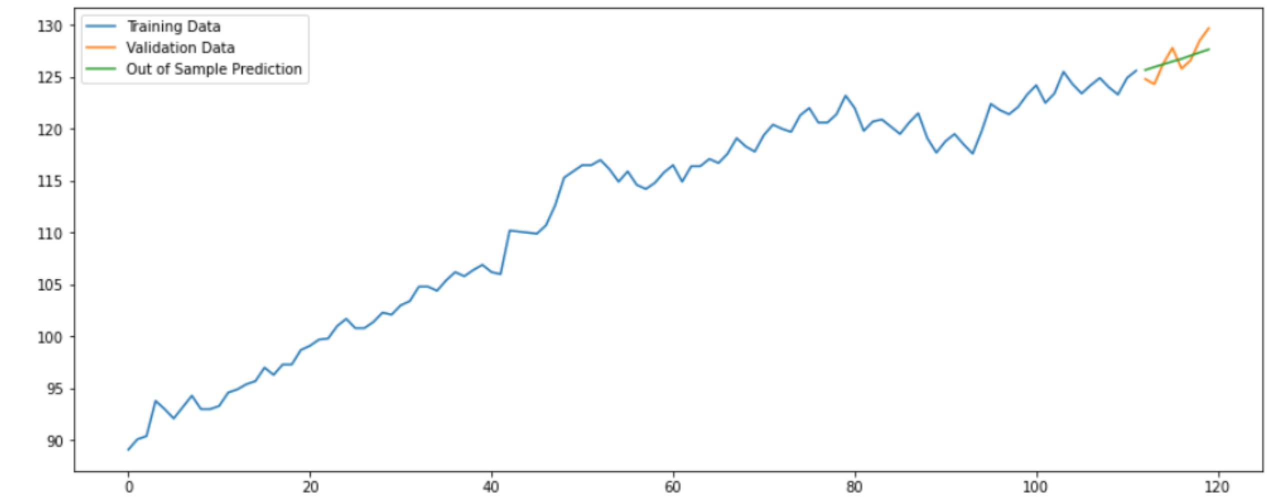
Batch size is the number of training samples used in each iteration. Larger batch size can speed up model training but tends to decrease model accuracy.

Epoch is a forward and backward pass through the neural network. As the number of epochs increases, accuracy will be improved since a greater number of times the weight are undated but it may lead to overfitting. Therefore, early stopping technique is used to prevent over-training. With early stopping, training will be terminated when the performance gets worse.

After carefully adjusting the hyperparameters, the best model was trained where the time_window equals 15, units and batch size equal 16 with 42 epochs. The validation MSE is 1.4763.

Forecasting

Dynamic forecast method is applied with LSTM model for 8-step-ahead forecast. We predict one time step at a time and feed that prediction back into the input window to estimate the next observation. The following figure shows the result of predictions and true values.



LSTM has several advantages forecasting time series data. Unlike ARIMA, LSTM is not based on specific assumptions about time series data such as stationarity (Kutzkov, 2022). Besides, the model is capable of capturing seasonality ("Understanding LSTM in Time Series Forecasting", n.d.). LSTM can provide better result than traditional methods since it is capable of learning long term correlations in a sequence (Brownlee, 2017). However, it also has some disadvantages. LSTM is difficult to interpret and requires careful hyper-parameter tuning to achieve high accuracy (Brownlee, 2017).

Result analysis

Method	MSE	Advantages	Disadvantage
Additive Decomposition	6.9434	Breaking down complex problems into more manageable parts.	Ignoring the uncertainty in the forecasts.
Multiplicative Decomposition	6.8313		
Seasonal Naïve	6.5425	Simple, easy to implement and interpret.	Only considers the seasonal factor and not the trend (up or down).
Drift	1.4401	Simple, easy to implement and interpret.	1. Lacks consideration of seasonality. 2. Only consider two values, so ignoring data may cause errors most time when making forecasting using unstable datasets.

Additive Holt-Winters	0.8627	Simple model which considers level, seasonality, trend.	Requires rigorous data pre-processing to remove anomalous data, which would otherwise produce serious prediction errors.
Multiplicative Holt-Winters	0.8761		
SARIMA	4.4480	Capable of capturing complex underlying patterns in the time series rather than trends and seasonality.	Due to technical problems with the equipment and python, the optimal model parameters could not be filtered by AIC.
Neural Network	3.7379	It can infer new features from the limited features set contained in the training set.	1. It usually requires a large amount of data. 2. Requires more computing power, which means that the calculation of the algorithm needs to spend a lot of time and cost to minimize the error. 3. The neural network model is easy to overfitting.
LSTM	1.4763	1. Not based on specific assumptions such as stationarity. 2. Capable of capturing seasonality. 3. Capable of learning long term correlations in a sequence.	Difficult to interpret and requires careful hyper-parameter tuning.

Conclusion

To sum up, exploratory data analysis of the quarterly CPI data has provided us with a good understanding of the data's type, meaning, and quantity. The decomposition model also helps us understand that the data has obvious seasonal characteristics and increasing trends. In addition, we also studied simpler prediction models such as the seasonal Naïve model, drift model and holt-winters seasonal models. The seasonal Naïve model takes the previous period data as the predicted value for the next period, and it can be the comparison benchmark of other models. However, its performance is not good because it ignores the growth trend of CPI data, resulting in a large MSE. The drift model considers data trends, and its MSE is small, but it ignores seasonal and other data. The Holt-Winters Seasonal model considers the level, seasonality and data trend, and it can effectively respond to time series changes with a mean square error of only 0.8627. Furthermore, the SARIMA model as a formal time series model helps capture complex underlying patterns in time series. It uses the means of the Akaike Information Criteria to filter optimal parameters P and Q. Then, do model fitting and found that the predicted result was different from the original value, and the MSE was 4.4480. We also develop deep learning to analyze the data set, such as the feedforward neural network and Long Short-Term Memory model. Feedforward neural network model is a simple artificial neural network in which the signal propagates unidirectional from the input layer to the output layer. Through the analysis of the model creation process, parameter adjustment and model prediction, it is found that there was a slight difference between the predicted value and the real value. Thus, the model is not the optimal model. Finally, we analyze quarterly consumer price index data using the LSTM model, and it is a special type of recurrent neural network. It can more easily remember past data in memory. After training the model, we found that the Additive Holt-Winters model is the best model to predict the quarterly CPI because it provides higher quality results than the other six models, with a mean square error of only 0.8627. Although LSTM has produced small MSE result, the model's performance is not stable. Thus, we chose Holt-Winters as final model and made prediction on test data.

Reference lists:

Bender, E., Sag, A., & Wasow, T. (2003).*Syntactic Theory: a formal introduction(Second Edition).*CSLI Publications.

Brownlee, J. (2017, May 22). *The Promise of Recurrent Neural Networks for Time Series Forecasting.* Retrieved from https://machinelearningmastery.com/promise-recurrent-neural-networks-time-series-forecasting/

Chawla, V. (2020, September 19). *Is More Data Always Better For Building Analytics Models?*. Analytics India Magazine. Retrieved from https://analyticsindiamag.com/is-more-data-always-better-for-building-analytics-models/.

Chief, E. (2016, October 16). *5 Advantages and Disadvantages of Exponential Smoothing*. ConnectUS. Retrieved from https://connectusfund.org/5-advantages-and-disadvantages-of-exponential-smoothing.

Exponential Smoothing (Seasonal). (2022). (pp. 1-52). Retrieved May 24, 2022, from QBUS6840, Lecture6, s1, 2022.

Forecasting - Drift Method. TradingView. (2022). Retrieved May 24, 2022, from https://www.tradingview.com/script/5MKvTvrq-Forecasting-Drift-Method/.

Forecasting Methods: What is Naive Forecasting Method?. Avercast.eu. (2022). Retrieved May 22, 2022, from https://www.avercast.eu/post/naive-forecasting.

GeeksforGeeks. (2020, October 24). *Intuition of Adam optimizer.* https://www.geeksforgeeks.org/intuition-of-adam-optimizer/

GeeksforGeeks. (2019, June 30). *Python | Mean Squared Error.* https://www.geeksforgeeks.org/python-mean-squared-error/

Ghoshal, A. (n.d.). Part 1: Journey of BERT. Retrieved from https://www.arpanghoshal.com/post/part-1-journey-of-bert

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.

How You Can Use Naïve Forecasting for Your Business. Avercast.in. (2022). Retrieved May 23, 2022, from https://www.avercast.in/blog/what-is-naive-forecasting-and-how-can-be-used-to-calculate-future-demand#:~:text=Seasonal%20na%C3%AFve%20forecasting%20is%20similar,uses%20decreasing%20weights%20for%20forecasts.

IBM. (2021). *The Neural Network Model* https://www.ibm.com/docs/en/spss modeler/18.0.0?topic=networks-neural-model

Jason B.(2019, January 9). *A Gentle Introduction to the Rectified Linear Unit (ReLU)* https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

Kutzkov, K. (2022). ARIMA vs Prophet vs LSTM for Time Series Prediction. Retrieved May13, 2022, from https://neptune.ai/blog/arima-vs-prophet-vs-lstm

McHenry, J. (2022). Rstudio-pubs-static.s3.amazonaws.com. Retrieved May 22, 2022, from https://rstudio-pubs-static.s3.amazonaws.com/255895_602e0142078543f49cd0ec65e82b1f07.html.

Nick, N. (2022, May 18). QBUS6840 Predictive Analytics, Module Group Assignment[Tutorial 11]. University of Sydney Business School. https://canvas.sydney.edu.au/courses/40417/pages/tutorial-11?module_item_id=1505643

Rumelhart, D., Hinton, G., & Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533-536. doi: 10.1038/323533a

Schuster, M., & Paliwal, K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions On Signal Processing*, *45*(11), 2673-2681. doi: 10.1109/78.650093

Some simple forecasting methods. Otexts.com. (2022). Retrieved May 23, 2022, from https://otexts.com/fpp2/simple-methods.html.

Tu. (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology*, *49*(11), 1225–1231. https://doi.org/10.1016/S0895-4356(96)00002-9

Understanding LSTM in Time Series Forecasting. (n.d.). Retrieved from https://www.predicthq.com/events/lstm-time-series-forecasting