INFLATION FORECASTING USING LSTM NETWORKS: CASE OF TURKEY

Gundem, Korel

DAT6303 Instructor: Amir Jafari, PhD

Contents

1. Introduction

Inflation forecasting is a critical component of economic decision-making and policymaking (James H Stock, 1999). Inflation, which refers to the general rise in the prices of goods and services in an economy, can have a significant impact on various economic activities, including investment decisions, monetary policy, and interest rates. Thus, accurate inflation forecasting is essential to make informed decisions that can impact the overall economic health of a country (Jan J. J. Groen, 2013).

Fortunately, advances in machine learning and artificial intelligence have provided new and more sophisticated methods for forecasting economic time series data, including inflation (Nakamura, 2005). One such method that has shown promise is Long Short-Term Memory (LSTM) networks, which belong to the family of recurrent neural networks (RNNs). LSTMs are known for their ability to capture long-term dependencies in time series data and make accurate predictions based on past trends (Yong Yu, 2019).

The primary objective of this project is to leverage the power of LSTM networks to forecast inflation in Turkey accurately. To achieve this goal, the project will utilize historical data on inflation, macroeconomic variables, and other relevant factors. This data will be preprocessed and used to train the LSTM network model, enabling it to predict future inflation rates.

The trained LSTM model will be evaluated against actual inflation rates to assess its accuracy in predicting future inflation rates. This evaluation will help determine the effectiveness of the LSTM model in forecasting inflation in Turkey and provide insights into the suitability of LSTMs as a forecasting method for other economic indicators.

Overall, this project is significant because it seeks to improve the accuracy of inflation forecasting in Turkey, a critical economic indicator that influences investment decisions, monetary policy, and other economic activities. By utilizing cutting-edge machine learning techniques such as LSTM networks, this project has the potential to contribute to the development of more accurate and reliable methods for forecasting inflation, which can lead to more informed economic decision-making and better policymaking.

The rest of this report is structured as follows. In the upcoming section, we will provide a comprehensive overview of the dataset that was utilized in this analysis. Additionally, we will give a brief summary of the LSTM network and its history to establish the foundation for the study. We will then delve into the experimental setup and elaborate on how the data will be processed, along with the various techniques that can be utilized to tune hyperparameters. Subsequently, we will present the obtained results and their implications in a coherent and comprehensive manner. Finally, we will summarize the entire project and discuss potential extensions that can be pursued to further improve the performance of the study.

2. Description of dataset

The data utilized for both the training and testing phases of the neural network is a comprehensive time series, sourced from esteemed organizations including the Turkish Statistical Institute, Federal Reserve Economic Data, and the World Bank. This invaluable series is comprised of monthly observations spanning a remarkable period, commencing in January 1994 and culminating in January 2023. The variables utilized in the analysis are diverse, including the inflation rate, policy rate established by the esteemed Central Bank of Turkey, the prevailing value of the Turkish Lira relative to the US dollar, the M2 money supply, the industrial production index, the business tendency index, and the registered unemployment figures within the Turkish economy.

The selection of these variables is based on their recognized significance in forecasting inflation within an economy. The inflation rate serves as a fundamental variable in this regard, as it is a crucial indicator of the general level of prices and the purchasing power of consumers. The policy rate established by the Central Bank of Turkey is also vital, as it provides insights into the monetary policy stance of the institution and its willingness to adjust interest rates in response to prevailing economic conditions. The value of the Turkish Lira against the US dollar is another significant variable, as it has a substantial influence on the exchange rate regime and, in turn, the overall price level within the economy. Additionally, the M2 money supply, industrial production index, business tendency index, and registered unemployed people in Turkey are all fundamental variables that provide insights into the broader economic conditions, including the level of economic activity, capacity utilization, and labor market dynamics. By including all these variables in the analysis, the neural network can effectively capture the underlying trends and dynamics that drive inflation and deliver more accurate and reliable predictions.

3. Description of the Network and Background Information

The Long Short-Term Memory (LSTM) network is a type of recurrent neural network (RNN) that is specifically designed to handle long-term dependencies in time series data. Unlike traditional RNNs, which can suffer from the vanishing gradient problem, LSTM networks use specialized memory cells and gating mechanisms to store and selectively retrieve information from previous time steps (Alex Graves, 2007).

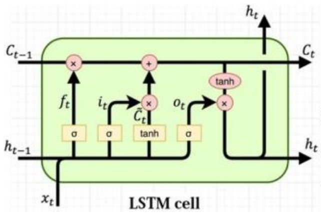
The development of the LSTM algorithm can be traced back to the mid-1990s when Sepp Hochreiter and Jürgen Schmidhuber first proposed the concept of Long Short-Term Memory. Their idea was to create a neural network architecture that could learn long-term dependencies in time series data while avoiding the vanishing gradient problem that often plagues traditional RNNs (Sepp Hochreiter, 2001).

The LSTM network consists of multiple memory cells, each of which is designed to store information for an extended period of time. The memory cells are connected by a series of gating mechanisms, which control the flow of information into and out of the cells. These gates include an input gate, an output gate, and a forget gate, which allow the network to selectively remember or forget information from previous time steps (Felix A. Gers, 2002).

The training algorithm for the LSTM network involves optimizing a set of parameters to minimize the difference between the predicted output and the actual output. This is typically done using backpropagation through time (BPTT), a variant of the backpropagation algorithm that is used to train recurrent neural networks (Jeses & Hagan, 2001). During training, the LSTM network is fed a sequence of input data, and the corresponding output data is compared to the predicted output. The error is then propagated backward through time, and the weights and biases of the network are adjusted to minimize the error (Jesus & Hagan, 2007).

One key feature of the LSTM network is its ability to handle variable-length input sequences (Ilya Sutskever, 2010). This is accomplished through the use of padding and masking, which allow the network to process sequences of different lengths without introducing errors or inconsistencies.

In terms of equations, the LSTM network can be described using the following equations:



$$i_t = \sigma(x_t U^i + h_{t-1} W^i)$$
$$f_t = \sigma(x_t U^f + h_{t-1} W^f)$$
$$o_t = \sigma(x_t U^o + h_{t-1} W^o)$$
$$\tilde{C}_t = \tanh(x_t U^g + h_{t-1} W^g)$$
$$C_t = \sigma(f_t * C_{t-1} + i_t * \tilde{C}_t)$$
$$h_t = \tanh(C_t) * o_t$$

where i_t, f_t, and o_t are the input, forget, and output gates, respectively, $\backslash tilde\{C\}_t$ is the candidate memory cell, C_t is the current memory cell, and h_t is the hidden state at time step t. W^i, W^f, W^o, and W^g are the weight matrices for the input, forget, output, and candidate memory cell, respectively. σ represents the sigmoid function, and tanh represents the hyperbolic tangent function.

Overall, the LSTM network and its training algorithm represent a significant advancement in the field of machine learning, particularly in the area of time series forecasting (Graves, ve diğerleri, 2008). Its ability to capture long-term dependencies and handle variable-length input sequences make it a powerful tool for a wide range of applications, from speech recognition to natural language processing to financial forecasting (C. LEE GILES, 2001).

4. Experimental Setup

The first step is to split data into training and test sets. I will be using data from the beginning until 2021 for training and from 2021 to 2022 for validation. Since we plan to predict a 12-step out forecast, using 12 input time steps, I will leave the last 12 observations (2022-2023) for out-of-sample testing later. The testing inputs are data from 2020-2021.

To implement the LSTM network for the inflation forecasting project, we will use the deep learning framework PyTorch. The first step will be to preprocess the data, which may include normalizing the data, splitting it into training and testing sets, and potentially applying other techniques such as data augmentation to improve the robustness of the model.

Next, we will define the architecture of the LSTM network, including the number of memory cells, the number of hidden layers, and the various gating mechanisms. We will also define the loss function and the optimization algorithm used to train the network.

Once the network has been defined, we will train it using the training data and evaluate its performance on the testing data. To judge the performance of the network, we use metrics such as mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE) to compare the predicted inflation rates to the actual inflation rates. We may also use other metrics such as the coefficient of determination (R-squared) or the mean absolute percentage error (MAPE) to further evaluate the performance of the model.

Overall, the implementation of the LSTM network in the chosen framework involve several steps, including data preprocessing, network architecture design, training and evaluation, and cross-validation. By carefully evaluating the performance of the model using appropriate metrics, we can ensure that the model is accurate and effective in predicting inflation rates in Turkey.

Determining appropriate training parameters, such as the learning rate, is an important aspect of training an LSTM network. The learning rate determines how much the weights of the network are updated during each iteration of the optimization algorithm. If the learning rate is too high, the network may fail to converge, while if it is too low, the training process may be slow and inefficient.

There are several techniques that can be used to determine the optimal learning rate for a given network architecture and dataset. One popular technique is known as "learning rate scheduling," where the learning rate is gradually decreased over time as the training progresses. This can help prevent the network from diverging or getting stuck in local minima.

Another technique is to use a learning rate range test, where the learning rate is increased gradually until the loss function starts to increase. This can help identify a suitable range of learning rates that allows the network to converge quickly and effectively.

Alternatively, we can use optimization algorithms such as Adam, which adaptively adjusts the learning rate during training based on the gradient of the loss function. This can help ensure that the learning rate is appropriate for the given network and dataset.

In addition to the learning rate, other training parameters such as the batch size and number of epochs may also need to be tuned to achieve optimal performance. These can be determined using techniques such as grid search or random search, where different combinations of parameters are tried and evaluated based on their performance on a validation set.

Overall, determining the appropriate training parameters for the LSTM network in our project will involve a combination of trial and error, as well as techniques such as learning rate scheduling, learning rate range testing, and optimization algorithms such as Adam. The goal will be to find the optimal set of parameters that allows the network to converge quickly and accurately while avoiding overfitting.

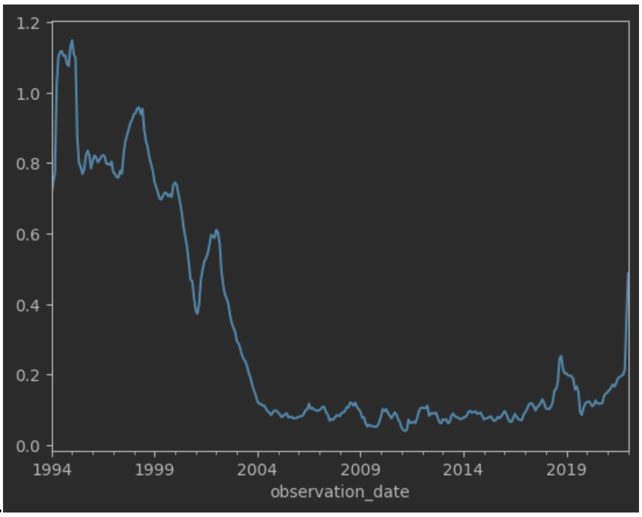
5. Results

As the primary focus of this study is to analyze the behavior of inflation in Turkey, it is important to carefully consider the methodology employed for data analysis. The data set under consideration spans a time period of thirty years, from 1994 to 2023, and is collected at a monthly frequency. The variable of interest is inflation, which is a crucial economic indicator that reflects the rate at which prices of goods and services increase over time.

In order to gain insights into the behavior of inflation over time, it is necessary to split the data set into training and test sets. This is a common practice in statistical modeling and allows for the evaluation of the model's performance on unseen data. In this study, the training data will consist of all observations from the beginning of the time period until the year 2021, while the testing data will consist of observations from 2021 to 2022. Additionally, the last 12 observations will be reserved for out-of-sample testing to further validate the model's accuracy.

Once the data has been appropriately divided, the next step is to analyze the behavior of inflation in Turkey over the entire time period. This involves examining trends, patterns, and fluctuations in the data set to gain a deeper understanding of the factors that contribute to changes in inflation rates. By doing so, we can identify potential drivers of inflation and develop effective policies to mitigate its negative effects on the economy.

In conclusion, the process of analyzing inflation in Turkey requires careful consideration of the data set and appropriate methodology for data analysis. By examining the behavior of inflation over time, we can gain valuable insights into the economic factors that influence this important indicator and develop effective policies to ensure long-term economic stability.



.

The graphical representation of the inflation data over the time period of 1994-2023 reveals some interesting patterns in the behavior of inflation in Turkey. Notably, from 1994 to 2004, there was a significant decline in inflation, as can be observed from the aforementioned graph. This trend could be attributed to the implementation of economic policies aimed at reducing inflationary pressures and stabilizing the economy.

Furthermore, from 2004 to 2018, there was a period of relative stability in inflation, with the inflation rate hovering around 0.15. This period was characterized by a stable political environment and the implementation of sound macroeconomic policies, which contributed to the overall economic stability of Turkey.

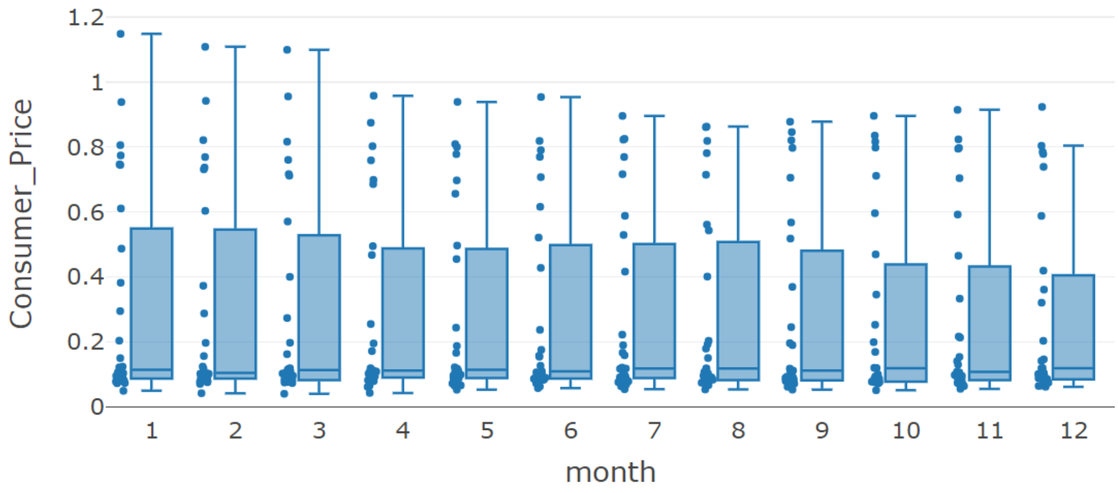
However, as noted in the statement, there was a structural break in 2019, which led to a sharp increase in inflation. This could be attributed to a variety of factors, such as political instability, economic shocks, and changes in global economic conditions. The recent surge in inflation underscores the importance of continued monitoring and analysis of economic indicators, as well as the need for sound economic policies that can mitigate the negative effects of economic shocks and ensure long-term economic stability.

To gain a deeper understanding of the behavior of inflation in Turkey, it is important to analyze the data at a more granular level. One way to do this is to examine the inflation on each month over time and looking at its distribution categorized by each month.

As shown in the graph, the yearly changes in inflation exhibit some interesting patterns over the time period of observation. There appears to be a seasonal pattern in the data, with the earlier parts of the year showing higher inflation compared to the latter parts of the year. This could be attributed to factors such as seasonal fluctuations in demand for certain goods and services, as well as changes in production and distribution costs.

Furthermore, in the last few months of the time period under consideration, there appears to be a slight decrease in inflation compared to the previous year. This could be due to a variety of factors, such as changes in consumer behavior, shifts in global economic conditions, or changes in government policies aimed at controlling inflation.

As shown in the data, there appears to be a higher degree of volatility in inflation in the earlier part of the year compared to the latter part. This suggests that there may be seasonal factors or other drivers that contribute to increased uncertainty in inflation during this time. It is important to note, however, that volatility can also be influenced by other factors such as changes in global economic conditions or shifts in government policies.



5.1.ARIMA

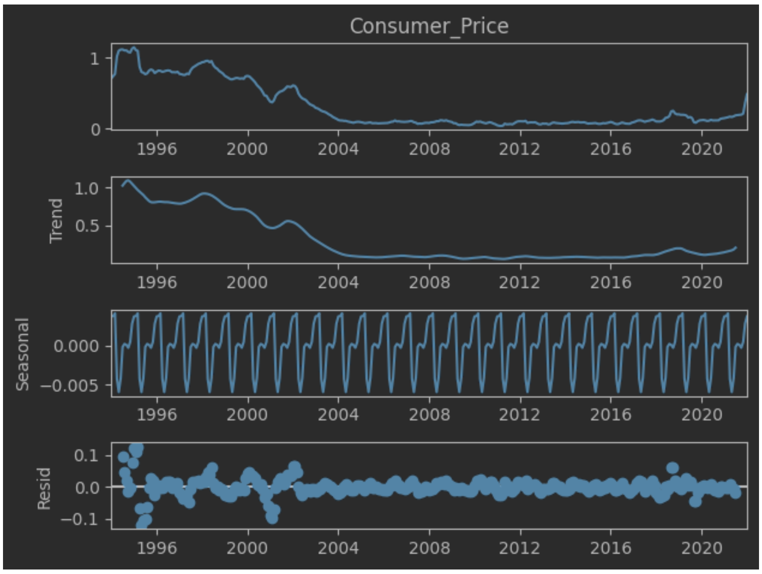
As the goal is to forecast future trends/values in a time-series dataset, one approach that can be used is the Autoregressive Integrated Moving Average (ARIMA) model, which combines AR and MA models. ARIMA models are commonly used for time-series analysis due to their interpretability and ease of implementation. Additionally, they can be effective for relatively short series such as this case where the number of observations is not sufficient to apply more sophisticated models. We set ARIMA model as a benchmark to compare how LSTM performs on the same task.

One potential limitation of ARIMA models, however, is the assumption of stationarity, which may not hold in financial time-series data where volatility, asymmetries, irregular time intervals, and sudden outbreaks are common. As a result, ARIMA models may not perform well on financial time-series data, and more advanced models may be needed to capture these complexities.

Before running the ARIMA model, it is important to check the stationarity of the inflation time series data. One commonly used method for testing stationarity is the Augmented Dickey-Fuller (ADF) test, which tests the null hypothesis that a unit root is present in a time series sample.

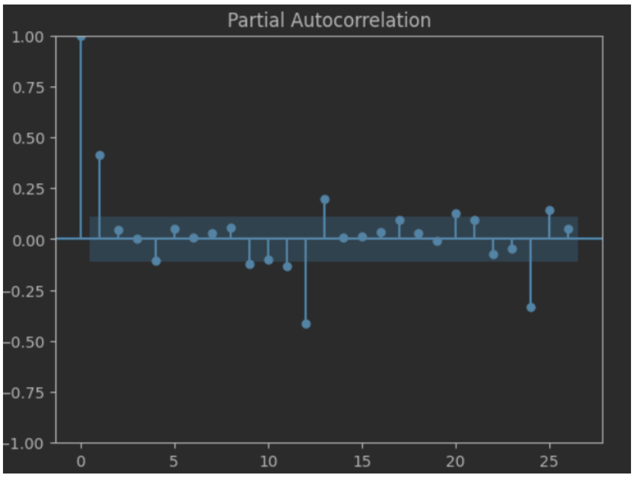
To visually examine the inflation time series data, we can use a decomposition plot. From the plot, it appears that there is a downward trend in inflation initially, which then appears to break.

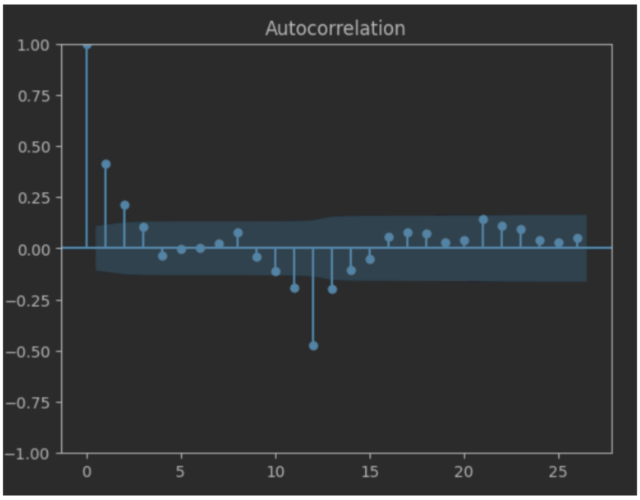
However, it is important to note that visual inspection alone is not sufficient to determine the stationarity of a time series. The ADF test provides a more formal and statistically rigorous method for testing stationarity, and should be performed before running any time-series models.



After running the ADF test on the inflation time series data, we fail to reject the null hypothesis that a unit root is present in the series. This suggests that the series is non-stationary. One commonly used technique to remove the unit root and make the series stationary is to take the first difference of the data. After taking the first difference and running the ADF test again, we can see that we reject the null hypothesis of a unit root with a significance level of 0.05. This indicates that the first differenced inflation time series data is stationary, and is a suitable candidate for fitting an ARIMA model.

After differencing the inflation time series data, we can use partial autocorrelation function (PACF) and autocorrelation function (ACF) plots to determine the optimal number of lags and the order of the MA term for the ARIMA model.





In the PACF plot, we look for significant spikes that extend beyond the confidence interval. From the plot, we can see that the PACF at lag 1 is significant. Similarly, in the ACF plot, we look for significant spikes that extend beyond the confidence interval. From the plot, we can see that the ACF at lag 1 is also significant. These results suggest that an ARIMA(1,1,1) model might be appropriate for fitting the inflation time series data. Since CPI exhibits exponential growth (variance increases), we build the model on the ln(inf) e.g. converting the raw values to log values. Below the reader can see the summary table of ARIMA(1,1,1) model.

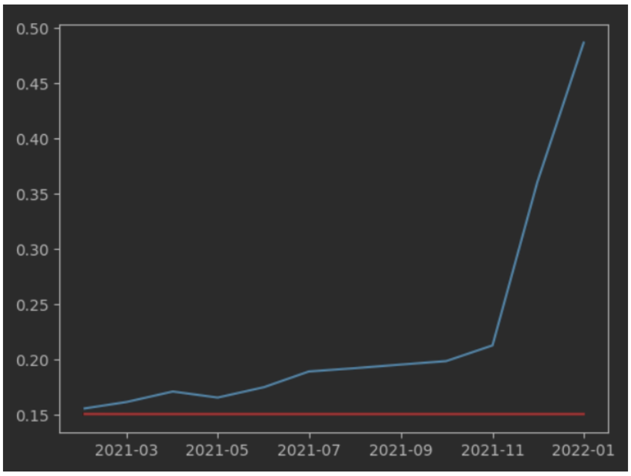
Dep. Variable:	Consumer_Price	**No. Observations:**	325
Model:	ARIMA(1, 1, 1)	**Log Likelihood**	318.225
Date:	Thu, 20 Apr 2023	**AIC**	-630.449
Time:	18:40:11	**BIC**	-619.107
Sample:	01-01-1994	**HQIC**	-625.922
	- 01-01-2021		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.0392	0.169	-0.232	0.817	-0.371	0.292
ma.L1	0.3359	0.158	2.120	0.034	0.025	0.646
sigma2	0.0082	0.000	23.145	0.000	0.008	0.009

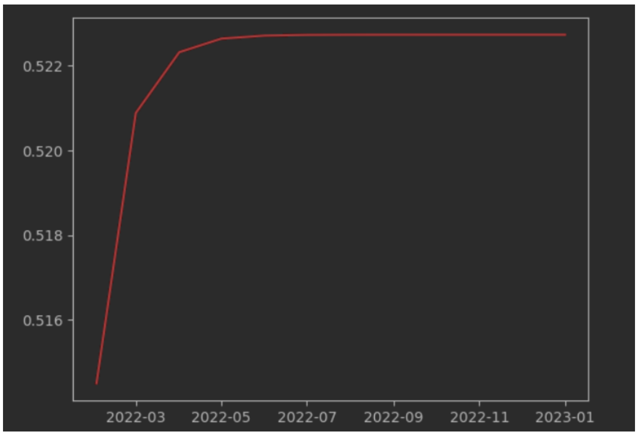
Ljung-Box (L1) (Q):	0.00	**Jarque-Bera (JB):**	340.30
Prob(Q):	1.00	**Prob(JB):**	0.00
Heteroskedasticity (H):	4.25	**Skew:**	0.13
Prob(H) (two-sided):	0.00	**Kurtosis:**	8.01

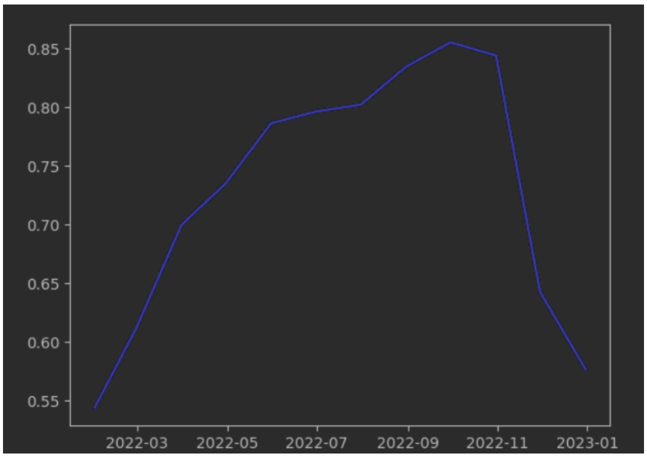
To evaluate the performance of the ARIMA (1,1,1) model on the test set, we can calculate the mean squared error (MSE). The MSE is a measure of the average squared difference between the predicted and actual values. The calculated MSE is 0.014.

To better evaluate the model, we can also forecast the next 12 months on the test set using the ARIMA(1,1,1) model. The red line in the plot represents the forecasted values, and the blue line represents the actual values. As it can be seen from the plot, the it fares badly on the validation set by constantly underpredicting the actual inflation. This result suggests that the ARIMA(1,1,1) model may not be the best fit for the data, and further analysis and model selection should be performed to improve the forecasting performance. The reason behind the almost straight line of ARIMA forecast is that ARIMA models assume that the time series has a stationary mean and variance, and any trend or seasonality is accounted for through differencing or seasonal differencing. When the time series has no or weak seasonality, the ARIMA model can only rely on the trend and past values to make predictions, which results in a straight-line forecast. This is one of the limitations of ARIMA models in forecasting time series with weak or no seasonality.



Evaluating a model's performance out of sample is crucial to assess its reliability and usefulness for real-world applications. In this case, we compared the forecasted values from the ARIMA model over the period of 2022-2023 to the actual inflation data. The results show that the model consistently underpredicted the actual inflation, indicating that it may not be the best choice for forecasting inflation in the current economic climate. It is important to note that forecasting financial time series can be challenging due to their complex and volatile nature. Additionally, the current inflation rate has been heavily influenced by various economic and social factors, such as supply chain disruptions, labor shortages, and government stimulus programs, which can make it difficult for a model to accurately capture all the nuances and changes in the data. Therefore, it may be necessary to explore more advanced modeling techniques, such as neural networks or other deep learning methods, to better capture the non-linear relationships and patterns in the data. Additionally, incorporating external factors such as economic indicators or policy changes may also improve the accuracy of the model's predictions.

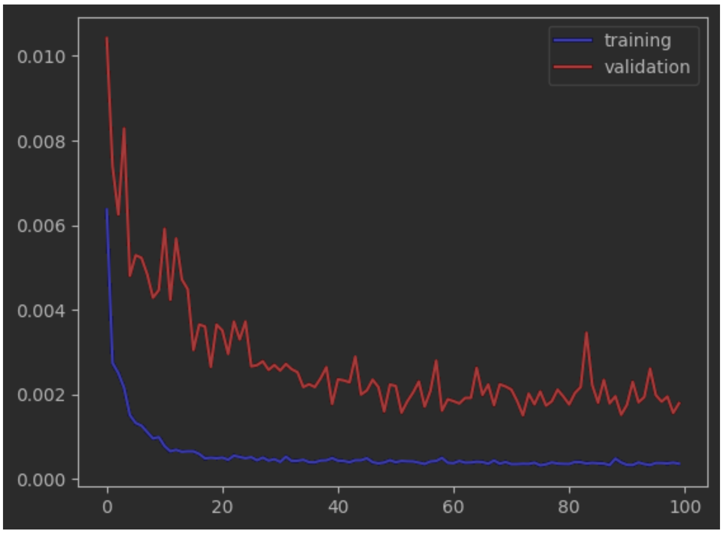




5.2 Univariate Time Series Forecasting with LSTM

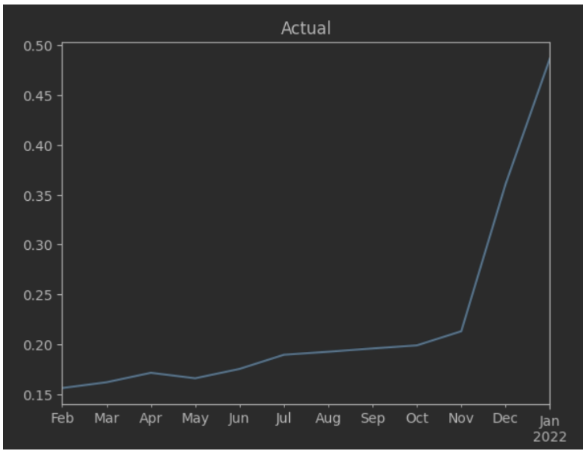
After running ARIMA(1,1,1) model as a benchmark model we see that it does not produce strong results in terms of forecasting. LSTM models have been shown to be effective in various time series forecasting problems, including financial forecasting. By leveraging the memory of past inputs and learning complex temporal dependencies, LSTM models can capture and learn the underlying patterns and trends in the data, leading to improved forecasting accuracy. However, due to their complex architecture, LSTM models require significant computational resources for training and can be prone to overfitting if not properly regularized. In this specific case, we can use LSTM to forecast inflation values with a one-step out forecast, using 12 input time steps. The LSTM architecture will consist of a simple, vanilla LSTM with one hidden layer and the default activation function of hyperbolic tangent (tanh). The model will be trained using historical inflation data and validated using a holdout set of data. One potential limitation of the LSTM model is the risk of overfitting to the training data, which can result in poor generalization to new data. To mitigate this risk, we can use techniques such as dropout regularization and early stopping during training. Another potential challenge is the presence of noise or outliers in the data, which can impact the model's ability to learn the underlying patterns and make accurate predictions. In such cases, preprocessing steps such as outlier removal or smoothing techniques can be applied to improve model performance.

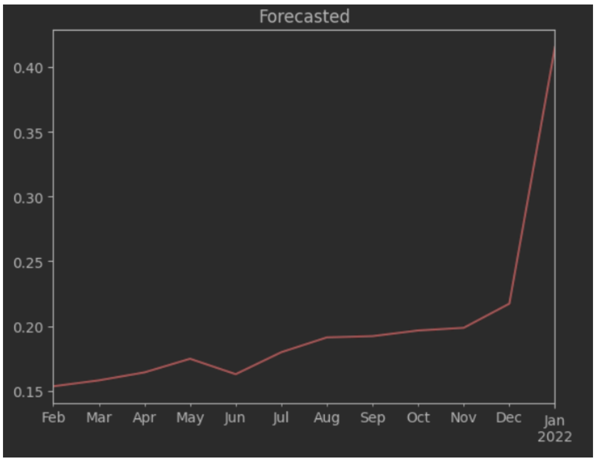
The LSTM model architecture was designed with a single layer and a high number of neurons, 100, in order to capture the complex trends present in the inflation data. Dropout regularization was not employed due to the simplicity of the problem, with only one layer present in the neural network, and the limited size of the dataset. It was found that adding dropout regularization for LSTM significantly decreased the model's performance on the test set. This can be attributed to the erasure of important contextual information, particularly in the presence of limited timesteps and a single layer. To ensure that the model did not overfit, the train and validation loss were closely monitored throughout the training process. A small learning rate of 0.001 was selected to account for the size of the neural network and the limited size of the dataset. The batch size was set to 100 epochs, and was fine-tuned based on the observed performance of the model. The reader can see the plot for training and validation error below.



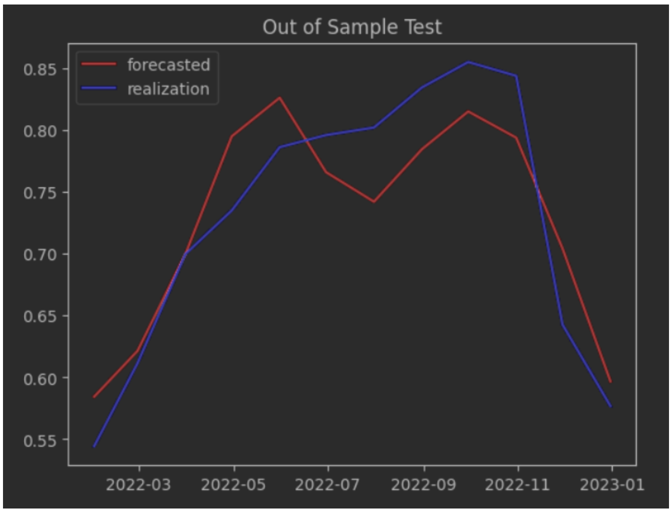
To compare the models, we will use MSE to evaluate the model's forecasting power. In this case, the MSE for the LSTM network is 0.00219, indicating that the model has a relatively low prediction error. Compared to the benchmark ARIMA(1,1,1) model, the LSTM network shows a significant improvement in terms of its forecasting power, indicating that the use of deep learning techniques can result in more accurate predictions of time series data.

To evaluate the performance of the model on the validation dataset, we can plot the forecasted values against the actual inflation values. The plot shows that the model is able to capture the overall shape of the actual inflation, but as time progresses, it seems to underpredict the actual inflation. This suggests that the model may not be accurately capturing the trends and patterns in the data. Further analysis and fine-tuning may be required to improve the model's forecasting power.





To test the predictive ability of the LSTM model on unseen data, we use the out of sample dataset. The model is trained to forecast one step ahead, given the previous 12 time steps as input. Hence, the input data for the model starts from 2021-02-01 and ends at 2022-01-01, and then shifts one time period for each new forecast. The performance of the model on the out of sample dataset is evaluated by comparing the predicted inflation values to the actual values. As can be observed from the plot, the LSTM model shows a significant improvement in performance compared to the benchmark ARIMA model. However, there are still areas where the model underpredicts or overpredicts the actual inflation values. At the beginning of the forecasting period, the model still underpredicts the inflation values. However, as the forecasting period progresses, the model starts to overpredict the inflation values. Despite these fluctuations, the model overall captures the underlying movement of the inflation between the periods of 2022-2023. This observation suggests that the model has the ability to capture the long-term trend of the inflation, which is a promising result for future forecasting efforts.



5.3 Multivariate Time Series Forecasting with LSTMs

The capacity of univariate LSTM network is limited as it can be seen from the previous section. In order to improve the accuracy of the inflation rate forecast, it is common practice to include other relevant variables that may affect its behavior. In the case of the analysis in question, several variables have been identified as potentially influential, including the policy rate established by the Central Bank of Turkey, the value of the Turkish Lira relative to the US dollar, the M2 money supply, the industrial production index, the business tendency index, and the registered unemployment figures within the Turkish economy.

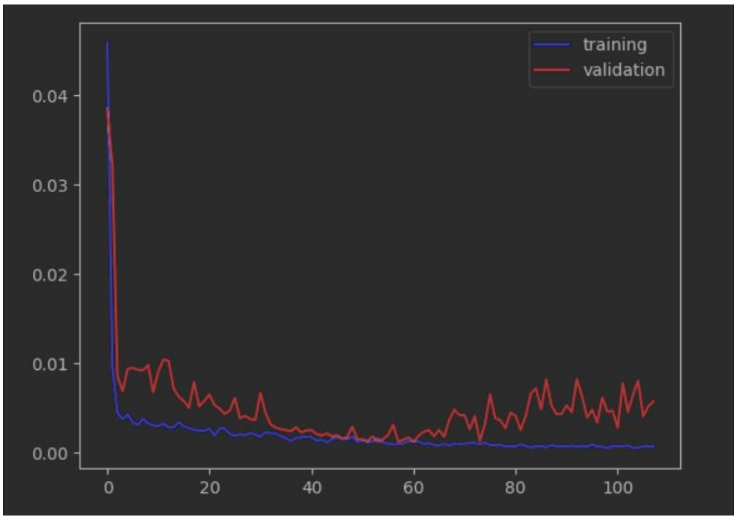
To determine which of these variables may play a more significant role in forecasting inflation, a Granger causality test is conducted. This test examines the causality relationship between two variables by analyzing whether the past values of one variable can be used to predict the future values of the other variable. However, it is important to note that one of the main assumptions of the Granger causality test is that the data is stationary. Therefore, prior to conducting the Granger causality test, the Augmented Dickey-Fuller (ADF) test is employed to determine whether the variables of interest are stationary. By checking the stationarity of the variables using the ADF test, we can ensure that the Granger causality test results are reliable and accurate. Once the variables have been tested and confirmed as stationary, the Granger causality test can be conducted to determine the causal relationship between the variables and the inflation rate. This information can then be used to select the most relevant variables to include in the LSTM network to improve the accuracy of the inflation rate forecast.

Once the stationarity of the variables has been verified using the ADF test, the next step is to prepare the data for the Granger causality test by taking the difference of several non-stationary variables. Taking the difference of the variables can help to remove any underlying trends and make the data stationary. After preparing the data, the Granger causality test is conducted to examine the causal relationship between the variables and inflation rate. The output of the test is then examined using a significance level of 0.05 to determine whether the null hypothesis can be rejected. If the null hypothesis cannot be rejected, it indicates that the variable in question does not Granger cause the inflation rate and can be excluded from the analysis. In the case of the current analysis, the Granger causality test results show that for the industrial production index, the number of registered unemployed, and the business tendency index, we fail to reject the null hypothesis at the 0.05 significance level. Therefore, it can be concluded that these variables do not Granger cause inflation rate and can be excluded from the analysis. This suggests that these variables are not significantly related to the inflation rate and are not useful in forecasting its behavior.

	Consumer_Price_x	interest_rate_x	M2_x	Real_ef_Ex_x	Ind_Prod_x	registered_unemp_x	Buisness_ten_x
Consumer_Price_y	1.0000	0.0000	0.0000	0.0000	0.4261	0.5353	0.6765
interest_rate_y	0.0004	1.0000	0.7439	0.8312	0.0190	0.7908	0.1315
M2_y	0.0140	0.0664	1.0000	0.0000	0.0000	0.1059	0.7490
Real_ef_Ex_y	0.0061	0.0723	0.0000	1.0000	0.0000	0.0765	0.3308
Ind_Prod_y	0.7907	0.5307	0.0000	0.0001	1.0000	0.0095	0.1002
registered_unemp_y	0.8525	0.7101	0.0256	0.1856	0.0417	1.0000	0.7163
Buisness_ten_y	0.2476	0.2184	0.3100	0.2890	0.1682	0.1114	1.0000

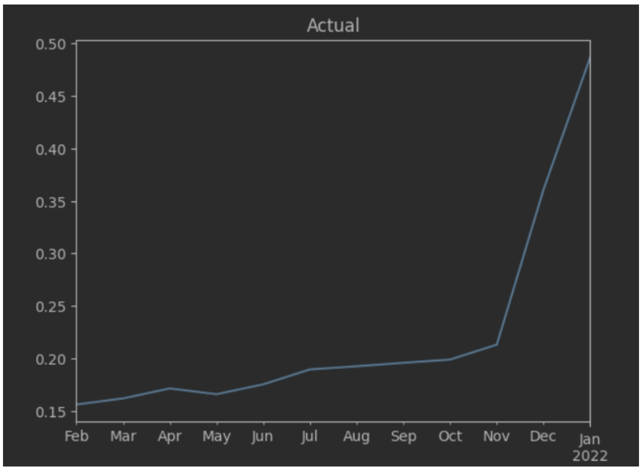
After doing the necessary analysis, we are ready to build the model. Similar to the univariate model, we will leave the last 12 months as the test set. We configure the inputs to be from 1994 to 2021for the train set and 2021 to 2022 for the validation set, and from 2022 to 2023 for out-of-sample test set. Specifically, the number of nodes in each LSTM layer was varied to find the optimal configuration that provided the best performance in terms of forecasting accuracy. After several experimental runs, it was found that the final model architecture consisting of two LSTM layers with 64 and 32 nodes respectively, followed by a Dropout layer with a rate of 20%, and a Dense output layer provided the best results. The model was then trained for 500 epochs, and the EarlyStopping and Dropout techniques were implemented to address the overfitting issue. The EarlyStopping patience was set at 50, allowing the model to stop training if there was no improvement in the validation loss for 50 consecutive epochs. The Dropout layer was added between the LSTM layers and the Dense output layer to randomly drop out 20% of the nodes, preventing the model from overfitting.

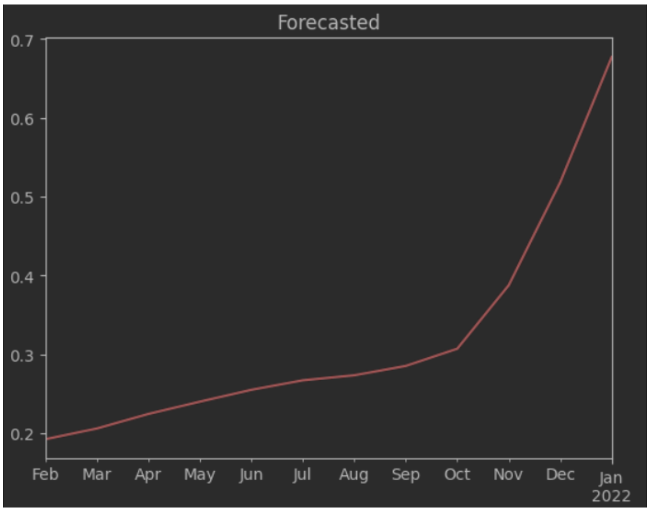
To further fine-tune the hyperparameters, a trial-error approach was used to find the optimal values for the number of epochs and batch size. This approach involved testing the model with different combinations of hyperparameters to find the optimal configuration that provided the best performance. In summary, the final multivariate LSTM model was designed with a careful selection of hyperparameters, including the number of nodes in the LSTM layers, the Dropout rate, and the use of regularization techniques such as EarlyStopping and Dropout to prevent overfitting. The hyperparameters were fine-tuned using manual experimental runs and a GridSearchCV approach to ensure optimal performance. Here is the plot for training and validation error.



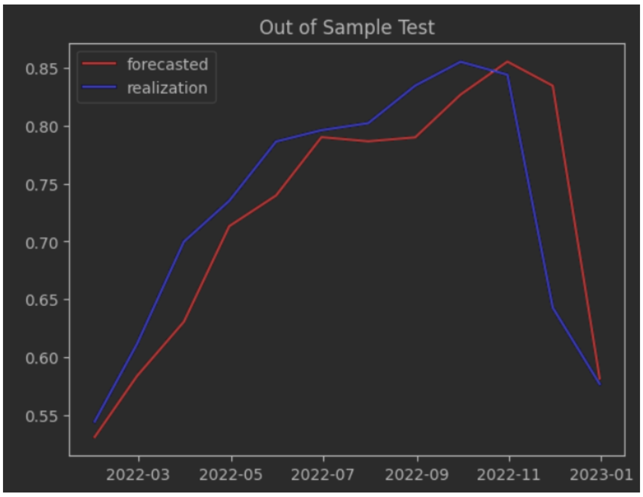
By measuring the difference in predicted y between the original model and the perturbed model, we can determine which features are more important in predicting the output. The greater the difference in predicted y, the more important the corresponding feature is in the model's output. In practice, we perturb each feature individually by introducing a small amount of random noise, usually drawn from a normal distribution. We then observe the change in the model's output for each perturbed feature and calculate the difference in the predicted output compared to the original prediction. This process can be repeated for each feature in the input data, allowing us to rank them by their importance in the model's output. Perturbation analysis provides a simple and effective way to quantify the contribution of each feature to the model's output and is commonly used in machine learning to identify the most important features in a given dataset. However, it is important to note that the results of perturbation analysis may not always be reliable, especially when dealing with highly correlated or interdependent features. In such cases, more sophisticated methods such as SHAP or permutation importance may be more appropriate. With the perturbation effect results, we can observe that the most important features for forecasting inflation are past inflation, exchange rate, and the level of M2 money supply.

The performance of the multivariate model compared to the univariate model appears to be quite impressive, as it captures the shape of inflation better and has a lower test mean squared error (MSE). Specifically, the test MSE for the model is 0.00035 whereas for univariate model it was 0.00219. It is also worth noting that the multivariate model seems to overpredict the inflation rate, while the univariate model underpredicted it. This suggests that the multivariate model may be capturing more of the complex relationships between the various input features and the target variable. Overall, the results suggest that the multivariate LSTM model is a significant improvement over the univariate model for predicting inflation, and could be a useful tool for decision-making in areas such as monetary policy and financial forecasting. However, it is important to continue evaluating and refining the model to ensure that it is robust and accurate in a variety of contexts.





To assess the performance of the multivariate LSTM model on previously unseen data, we utilized an out-of-sample dataset. The model was trained to forecast one step ahead, based on the previous 12 time steps as input. The input data range was from 2021-02-01 to 2022-01-01, and the forecast shifted one time period for each new prediction. To evaluate the model's performance, we compared the predicted inflation values to the actual values. The results showed that the multivariate LSTM model outperformed the benchmark univariate model, indicating a significant improvement in performance. However, there were still instances where the model either overpredicted or underpredicted the actual inflation values. At the beginning of the forecasting period, the model tended to underpredict inflation, while later in the forecasting period, it overpredicted inflation. Nevertheless, the model was able to capture the underlying movement of inflation between the periods of 2022-2023, suggesting its potential in predicting long-term trends of inflation.



6 Summary

The primary aim of this project was to employ LSTM networks for accurate inflation forecasting in Turkey, compare them with traditional methods like ARIMA, and evaluate both univariate and multivariate LSTM models. This goal was achieved by preprocessing historical data on inflation, macroeconomic variables, and other relevant factors and training the models on this data to predict future inflation rates. The trained models were then evaluated against actual inflation rates to assess their accuracy and effectiveness in forecasting inflation in Turkey, providing insights into the suitability of LSTMs as a forecasting method for other economic indicators.

Three models were considered and compared in the analysis, namely ARIMA(1,1,1), univariate LSTM, and multivariate LSTM. Firstly, the ARIMA(1,1,1) model had a calculated MSE of approximately 0.014. The out-of-sample forecast for this model consistently underpredicted actual inflation, resulting in a nearly straight-line forecast over the entire period. This behavior can be attributed to the ARIMA model's assumption of a stationary mean and variance in the time series, with any trend or seasonality accounted for through differencing or seasonal differencing. In the absence of strong seasonality, the ARIMA model relies solely on past values and trend to make predictions, resulting in a straight-line forecast.

Next, we trained a univariate LSTM network, which showed a significant improvement in forecasting power compared to the benchmark ARIMA(1,1,1) model. The MSE of the univariate LSTM was

0.00040, indicating a reasonable level of prediction accuracy. These results suggest that deep learning techniques such as LSTM networks can result in more accurate predictions of time series data.

The final model considered in this study was the multivariate LSTM network. Prior to training the model, a granger causality test was conducted to identify which features should be included. Based on the test results, several features were found to not granger cause inflation and were therefore excluded from the analysis. The model was then trained using the remaining variables. The resulting mean squared error (MSE) for the model was 0.00035, which indicates a relatively low prediction error. Compared to the benchmark ARIMA(1,1,1) model and the univariate LSTM network, the multivariate LSTM network exhibited a significant improvement in terms of its forecasting ability. This highlights the potential benefits of using multivariate LSTM models in forecasting time series data, as it allows for the incorporation of multiple variables that may have a causal relationship with the target variable. Overall, the results suggest that the multivariate LSTM model may be a promising tool for accurately predicting inflation rates in Turkey and potentially other economic indicators in the future.

In terms of future research directions, this report can be further augmented by also including Bidirectional LSTMs into analysis and comparison to see whether BiLSTM models provide better predictions compared to ARIMA and LSTM models and whether BiLSTM models reach the equilibrium much slower than LSTM-based models. Furthermore, the Gated Recurrent Unit can also be put into the analysis to see how it performs.

References

Alex Graves, S. F. (2007). Multi-dimensional Recurrent Neural Networks. *Artificial Neural Networks* .

C. LEE GILES, S. L. (2001). Noisy Time Series Prediction using Recurrent Neural Networks and Grammatical Inference. *Machine Learning*.

Felix A. Gers, N. N. (2002). Learning Precise Timing with LSTM Recurrent Networks. *Journal of Machine Learning Research*.

Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., & Schmidhuber, J. (2008). A Novel Connectionist System for Unconstrained Handwriting Recognition. *IEEE*.

Ilya Sutskever, G. H. (2010). Temporal-Kernel Recurrent Neural Networks. *Neural Networks*.

James H Stock, M. W. (1999). Forecasting inflation. *Journal of Monetary Economics*.

Jan J. J. Groen, R. P. (2013). Real-Time Inflation Forecasting in a Changing World. *Journal of Business & Economic Statistics* .

Jeses, O. D., & Hagan, M. (2001). Backpropagation through time for a general class of recurrent network. *IEEE*.

Jesus, O. D., & Hagan, M. T. (2007). Backpropagation Algorithms for a Broad Class of Dynamic Networks. *IEEE*.

Nakamura, E. (2005). Inflation forecasting using a neural network. *Economic Letters*.

Sepp Hochreiter, Y. B. (2001). Gradient Flow in Recurrent Nets: the Difficulty of Learning Long-Term Dependencies. *IEEE press*.

Yong Yu, X. S. (2019). A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Computation* .

NOTE: While carrying the analysis, approximately %60 of the code was found either on the internet or provided by ChatGPT. Furthermore, this work is mainly inspired by the following GITHUB repository: https://github.com/asaenzg/mscthesis