Forecasting Georgian Inflation Using Machine Learning and Econometric Models
(A Comparative Study)

Beka Abuladze

Bachelor's Project Report is submitted for the Economics Program of International School of Economics at Tbilisi State University

Supervisor: Shota Natenatdze, MA

Tbilisi, 2022

Contents

List of Abbreviations

ACF	Autocorrelation Function
ADF	Augmented Dickey-Fuller
AR	Auto-Regressive
ARIMA	Autoregressive Integrated Moving Average
CPI	Consumer Price Index
CPU	Central Processing Unit
GDP	Gross Domestic Product
GPU	Graphical Processing Unit
LSTM	Long Short-term Memory
MA	Moving Average
ML	Machine Learning
NBG	National Bank of Georgia
PACF	Partial Autocorrelation Function
RMSE	Root Mean Squared Error
SARIMA	Seasonal Autoregressive Integrated Moving Average

1 Introduction

Inflation is one of the most important statistical metrics in whole economics. Having an accurate forecast of inflation beforehand is very important for central bank officials and policymakers, as well as for economic agents to form their expectations. Since monetary policy transmission is associated with significant lags, central banks aiming to achieve price stability need to be forward-looking in their decisions as well – which underscores the importance of inflation forecasting (Garcia, 2018). Yet inflation dynamics have considerably evolved during the past decades and the search for a reliable model to forecast the metric is still an ongoing research question. As a part of my bachelor's thesis, I want to forecast inflation in Georgia in all, short, medium, and long-term time horizons using predominantly deep learning-based models. I also want to make a comparative analysis between classical econometric and machine learning/deep learning time series forecasting models.

The traditional literature on inflation forecasting mainly refers to Phillips curve-based models, where inflation is explained by some explanatory variables and autoregressive processes. This approach might be well established but the forecasts based on such models tend to show varying performance over time. They can quickly be outperformed by univariate models such as the unobserved components with stochastic volatility (UC-SV) model (Stock&Watson, 2007), or the random walk (Atkeson& Ohanian, 2001). There is a hand full of research papers conducted on forecasting inflation using multi and univariate econometric and machine learning approaches. The studies have been conducted worldwide, but rarely in Georgia, especially in multivariable setup. This paper aims to contribute in particular to Georgian data-based inflation forecasting literature by studying the ability of neural networks to consistently estimate inflation rates.

During the last decade, increasing growth in R&D and usage of ML models has not left the field of economics without its trace. Studies conducted on inflation forecasting using machine learning models have shown great promise in delivering consistent and accurate predictions for future inflation. These studies usually involve models that describe the nonlinear mappings between explanatory and dependent variables. Such models include support vector regressions (Siermpinis et al., 2014) and random forests (Medeiros et al., 2019). There also is a growing amount of research projects addressing the ability of neural networks to forecast inflation, but they mostly use the basic feed-forward neural network algorithms. Examples of such approaches are discussed in the works of Nakamura (2005) and Chakraborty&Joseph (2017). I want to enrich these studies by focusing on the LSTM model which belongs to a different class of neural networks and is relatively new in the field of inflation forecasting.

Accurately predicting inflation is a very interesting and at the same time, an extremely difficult challenge for both - academic researchers focusing on economics and practitioners such as data scientists. Following the points that were made by Stock & Watson (2007), inflation in the US is at the same time easier and harder to predict. On one side, after the “Black Monday” in 1987, the highly volatile inflation became more stable, thus giving the professionals the ability to predict it more easily. On the other hand, more or less stable and stationary inflation makes it hard for more advanced models to outperform naive random walk-type approaches. The same can be said for Georgia as well. The days of extremely high inflation are over, especially after the NBG decided to introduce an inflation targeting regime in 2009 (NBG, 2009), but the metric still behaves relatively unpredictably in times of moderately serious outside shock on the economy. Discussing the reasons for that is outside the scope of this paper, but the comparatively unstable inflation rate in Georgia gives us the ability to make use of the big and complex models.

It is clear that Georgia is not the best country in terms of data collection and the issues might arise because of the small sample data, but that would be yet another test to see how the models perform with limited data points and irregularities.

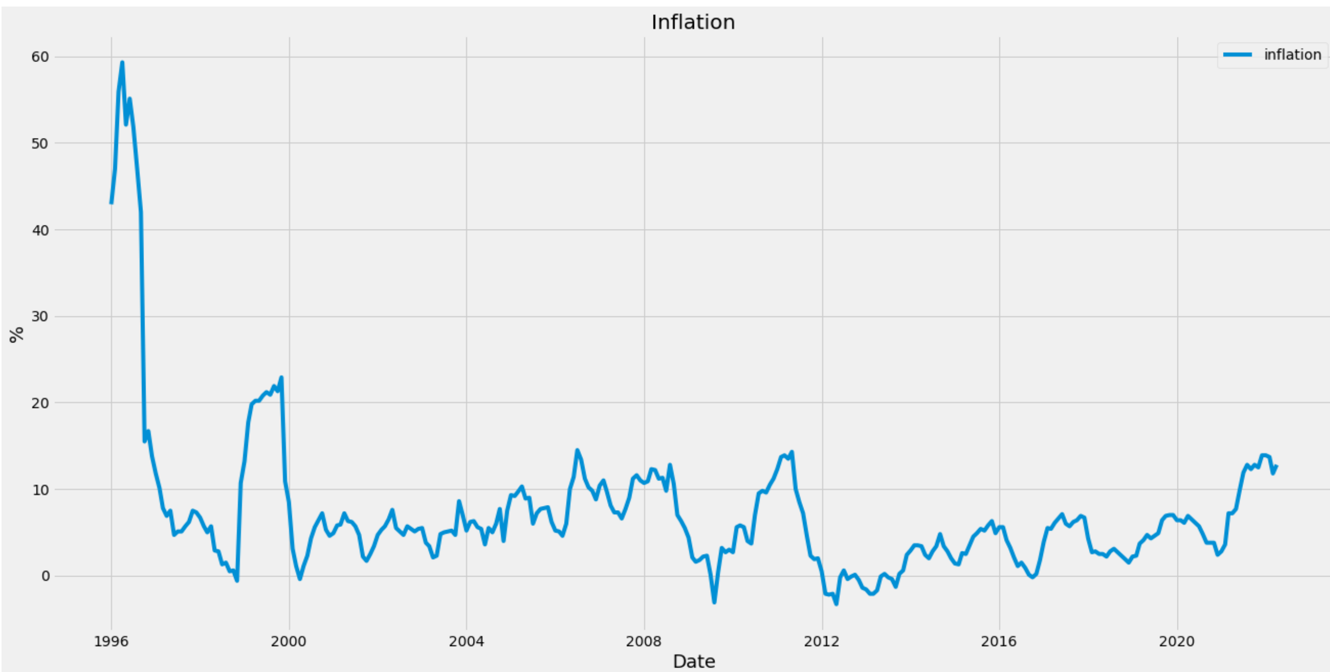
This paper aims to answer the following 2 questions: (i) Which of the models shows better results when forecasting inflation in Georgia in the short, medium, and long term perspectives. (ii) What is the forecast that the best model predicts for the Georgian inflation in the short, medium, and long term settings.

The organization of the paper is as follows. The following section introduces and thoroughly describes the data that the models were trained on. Section 3 presents the econometric framework used for the research, the models that were tested out against each other, alongside the results and further empirical analysis. Appendices A and B provide information about the estimation procedure and model specification respectively. Section 4 concludes the study.

2 Description of The Data

The models were trained on the Georgian monthly inflation rate series retrieved from the official database of the NBG. The dataset consists of 316 data points which represent the time period from January 1996 to April 2022. Figure 1 shows the graphical illustration of the inflation rate over time:

Figure 1: Georgian inflation rate over time



It can be clearly seen that the data has outliers and they might affect the predictions, but it is still important to include them in the forecast for two main reasons. The first one is that the number of data points is already pretty small for ML models and even further decreasing the number of inputs by removing the outliers would make it harder to make predictions. The second reason is that Georgia is heavily dependent on imports and the outside world in general. Any small international economic shock translates very heavily on the Georgian economy, and the unreal inflation rates could easily return in the future if a serious international shock occurs. A good example of that would be the COVID-19 pandemic and its effects on the inflation rate. If one takes a closer look at Figure 1, it is clearly visible that the inflation rate started increasing after the beginning of 2020, that is when the news about the pandemic started spreading and people in Georgia started emptying the shells in hypermarkets. That was followed up by the Russian annexation and occupation of innocent Ukraine, which further increased the fear index in the economy, fueling a further uptick in the inflation rate. Such events can still happen in the future and it is important that our models (especially LSTM) have the knowledge of such precedents happening in the past. That is the reason behind not removing the outliers of the late 1990s.

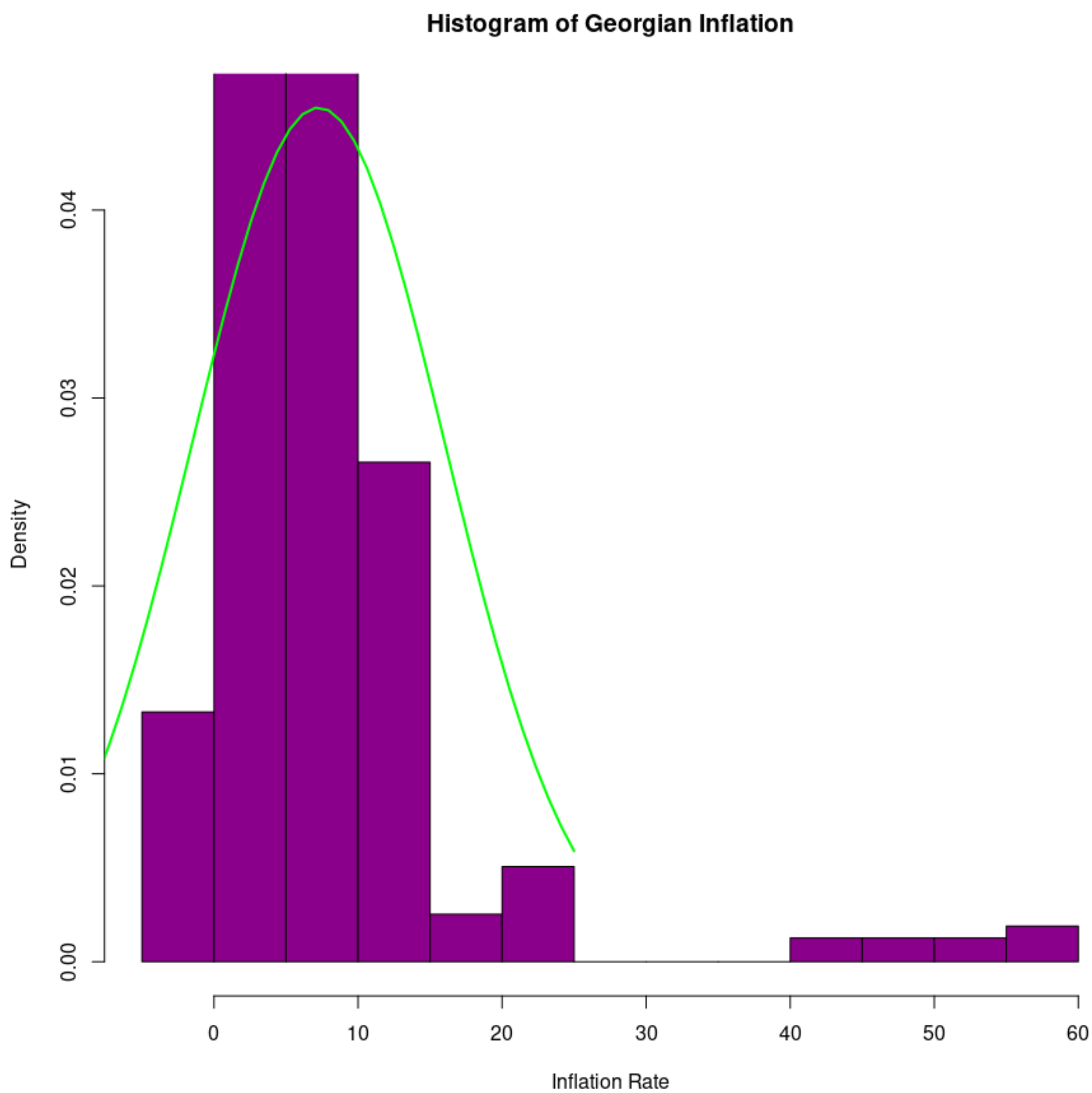
Table 1 below summarizes the general statistical characteristics of the inflation data, and Figure 2 shows the histogram representation of the data. It is once again visible that the inflation rate is mainly concentrated around the 5-10% mark, with only a couple of, mostly earlier data points being in the outliers. It is also worth mentioning that the distribution density in the "main island" of the data follows normal distribution centered at 8.2% having a standard deviation of 4.2%.

Table 1: Descriptive Statistics of Georgian Inflation

Number of values	316
Min	-3.3
Max	59.3
Range	62.6
Median	5.55
Mean	7.26
Standard deviation	8.77
Mean Standard Error	0.49
Variance	77.02

The existence of outliers can be easily explained. The high inflation rates during the period from 1996 to 2000 are associated with the financial crisis in Russia which officially began in 1998 after several years of having a large fiscal deficit and low growth in the economy. Russia had to devalue its national currency and default on its foreign debt (Chiodo&Owyang, 2002). This event had a pretty significant impact on Georgia. Not even a decade had passed after the dissolution of the Soviet Union and officially announcing the independence of Georgia. The economy was still heavily dependent on Russia causing slow growth of GDP together with an extremely volatile national currency and sky-high inflation rates.

Figure 2: Empirical Histogram of Georgian Inflation



Generally when working with the time series data using econometric models it is one of the prerequisites in most cases to ensure that the data is stationary. Stationarity is a mandatory requirement when modeling AR processes, so the data was examined on being stationary using the ADF test for a different number of lags. The results are given in Table 3.

Table 3: ADF test results

Lag Order	Dickey-Fuller Test Statistic	P-Value
0	-3.8032	0.01921
1	-4.6931	<0.01
2	-6.6694	<0.01
3	-7.9073	<0.01
4	-6.6386	<0.01
5	-8.3502	<0.01
6	-8.3457	<0.01

The P-Values being less than 0.05 in all of the cases gives us enough evidence to state that the data is stationary, so it is possible to conduct AR modeling without extra power transformations. It is worth noting that ML and especially DL models do not place the requirement of stationarity on the data. LSTM can handle the series in a similar fashion, whether the data is stationary or not (Preeti et al., 2019).

Figures 3 and 4 show ACF and PACF graphs respectfully. ACF describes the autocorrelation between observation and another observation at a prior point in time that includes direct and indirect dependence information. This means that if the ACF of the time series shows a strong dependence on a lag k and then the dependence weakens as the lags go past k, there is a good chance that the series was derived from an AR(k) process (Brownlee, 2019). It is also known that PACF only describes the direct relationship between an observation and its lag. This would suggest that there is no correlation for the lag values beyond k. The graphs in figures 3 and 4 strengthen these assumptions, suggesting that the time series was derived from an AR(k) process.

The argument can be extended for the MA(k) processes too. Since the MA process is an autoregression model of the time series of residual errors from prior predictions we would expect that the ACF of the MA(k) process shows a strong dependence on recent values up until the lag k, following a drastic decline to almost no correlation. As for PACF, we would expect the plot to show a strong dependence on the previous lag and an instant dropoff afterward. The plots are again consistent with these assumptions meaning that the time series was likely generated from AR and MA processes, indicating that the ARIMA model would be a good candidate to forecast the future of the series.

Figure 3: Autocorrelation

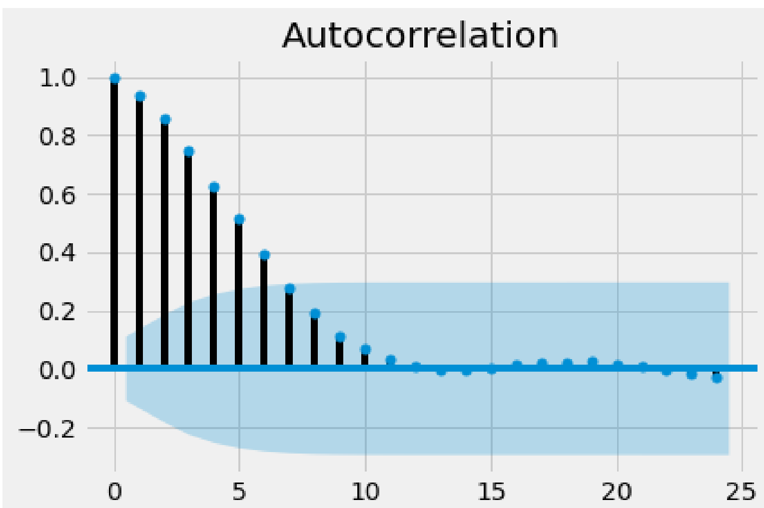
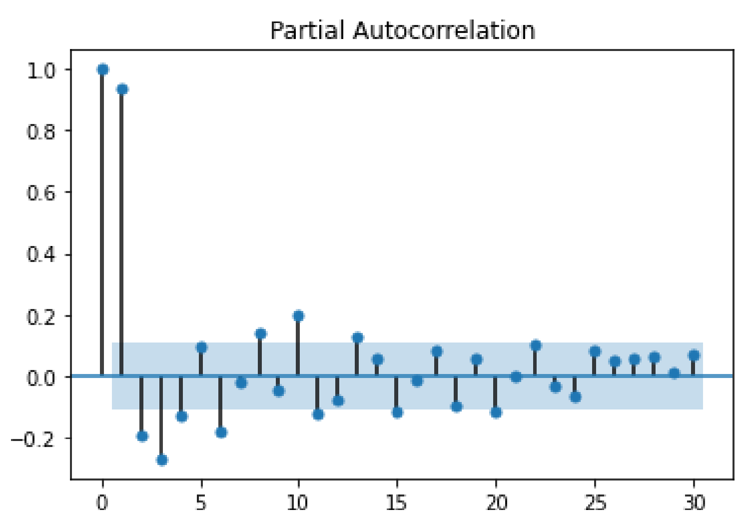


Figure 4: Partial Autocorrelation



3 Models and Methodology

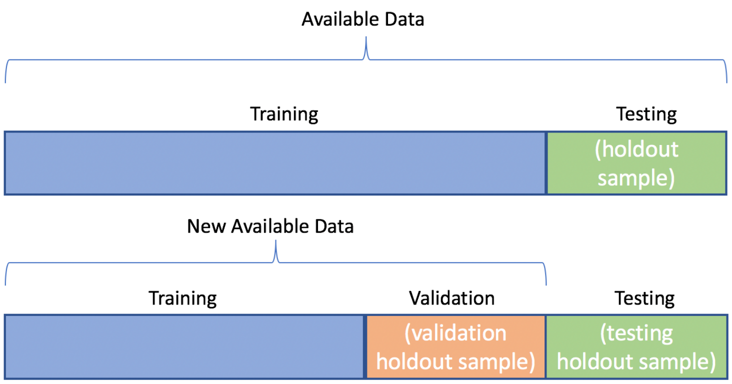
The selection of the models was made based on the previous literature namely the work of Paranhos (2020), Medeiros et al. (2019), Almosova&Andresen (2019), Stock&Watson(1998), Atkeson& Ohanian (2001), and Nakamura (2005).

3.1 Forecast Accuracy Metrics

One of the most important aspects of a forecasting process is to figure out how to measure the accuracy of the models used for predictions. Following the examples from the literature (Paranhos, 2021; Almosova&Andresen, 2019; Baybuza, 2018), a well-established RMSE was used for calculating the error terms. It is one of the go-to metrics when it comes to measuring the quality of the model. It punishes larger variations in the forecast more than other metrics, it is also sensitive to the outliers, because each individual error in RMSE is proportional to the square root of the original error term, therefore the results might be exaggerated if the data contains significant outliers. This fact might lead to different decisions regarding the inclusion of the outliers in the modeling data after the initial review of the first draft of the paper.

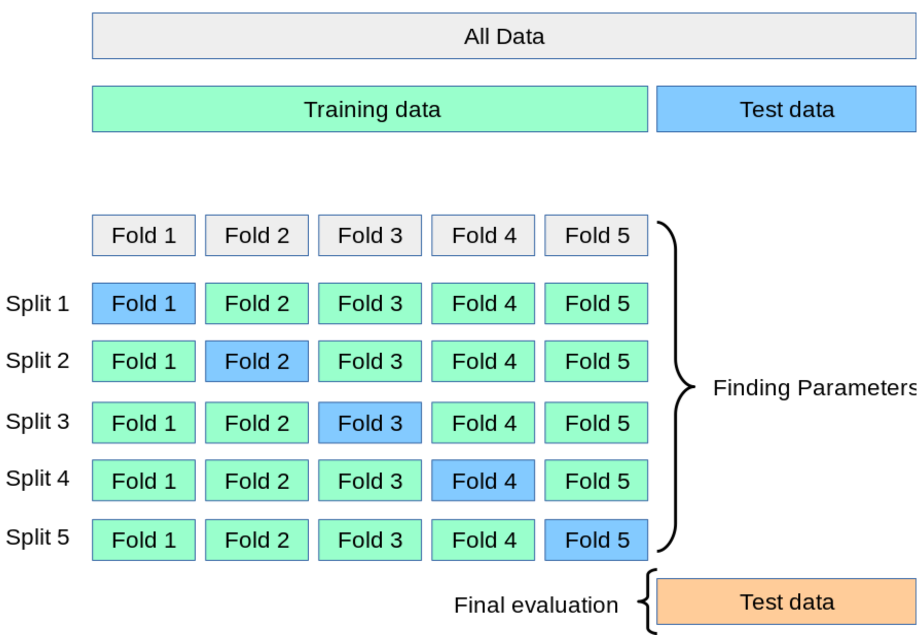
Another important point when measuring the accuracy of the models is to determine the testing framework. A well-known method for doing this is a so-called train-test splitting of the data, which means splitting the dataset into multiple parts (2 if conducting a train-test split and 3 parts when conducting a train-validation-test split). The general rule of thumb for relatively small datasets is to use an 80%, 20% split of the whole data for the train-test framework, and an 80%, 10%, and 10% split for the train-validation-test framework (Brownlee, 2020). When the size of the dataset increases dramatically, 98%, 1%, and 1% splitting can be used. The data from the training set is used to train the model, data from the validation set is used to validate the performance of the model and make sure that it works relatively well before trying to fit it for before unseen test data, and the data from the test set is used to test out how the model tackles the examples from the outside world. Figure 5 shows a visualization of the train-train and train-validate-test split frameworks.

Figure 5: Train-test split framework



Another method for checking the validity and accuracy of the model which is very highly regarded in the literature is K-fold Cross-Validation. The general procedure for testing using this methodology is as follows. The dataset is divided into two parts, the training dataset, and the test dataset. The test dataset is hidden away at the beginning. The training dataset is divided into K smaller parts, called folds (hence the name, K-Fold Cross-Validation), each unique fold is used as a test dataset, while the remaining (k-1) folds are used as the training dataset. The model is fitted on the training set and evaluated on the test set. The test score is recorded and averaged over all k iterations. In the end, the final model is tested out on the hidden test data produced during the first step of the whole process. Figure 6 shows the visualization of the process.

Figure 6: K-Fold Cross-Validation



Source: [*Scikit-Learn*]

The literature suggests that cross-validation is a more robust and reliable measure of the performance of the model since it considers every aspect of the dataset for training as well as testing. Since Georgian inflation data is less volatile after the 2010s regular 80%-20% splitting of the data for testing might provide a misleading RMSE metric, while the metric derived from cross-validation will take into account the errors generated by the highly volatile and unpredictable data from the late 1990s to early 2000s time period, thus providing a more vigorous testing and validation results.

This study is going to use the first framework for the first draft of the paper since it is easier to set up and experiment with. But it is planned to evaluate the models according to both methodologies eventually.

3.2 Forecasting Models

Random Walk

Using the random walk-type model as a basic benchmark is an approved strategy according to the literature (Stock&Watson, 2007; Atkenson & Ohanian, 2001; Almosova&Andresen 2019). It is a simple average over the previous n lags:

$$\hat{y}_{t|t-1} = \frac{1}{n} \sum_{i=1}^{n} y_{t-1}$$

Random walk will be model will is generated for n=1, 3, 6, 12 months to obtain short, medium and long-term predictions. This model is used as a minimal basic benchmark that all of the models should be able to beat

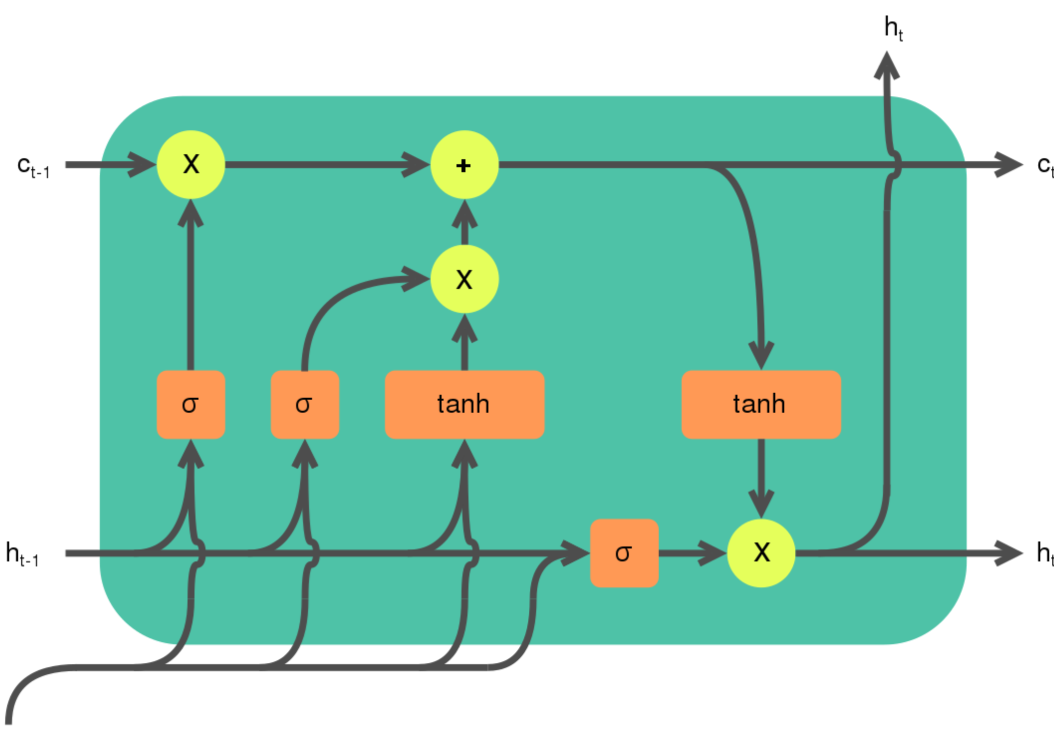
Seasonal Autoregressive Integrated Moving Average Model (SARIMA)

SARIMA(p, d, q)(P, D, Q)m model is one of the best-regarded models when it comes to time series forecasting according to the econometrics literature (Gujarati, D. 2009). SARIMA model contrary to the ARIMA model considers the seasonality of the data and uses additional P, D, Q, and m seasonal elements to control that trend. (p), (q), and (d) are the same as in ARIMA, namely trend autoregression order, trend difference order, and a trend moving average order respectively. While (P), (D), and (Q) bring the same values but control for the seasonality. (m) is the number of time steps for a single seasonal period, and in our example, we are setting it equal to 12 months. The estimation of the hyperparameters of this model is conducted using the Hyndman-Khandakar algorithm which is an iterative algorithm and tries to choose such (p) and (q) values that minimize the AIC error. The estimation and testing are done automatically using the auto-Arima package in the R programming language.

LSTM Architecture

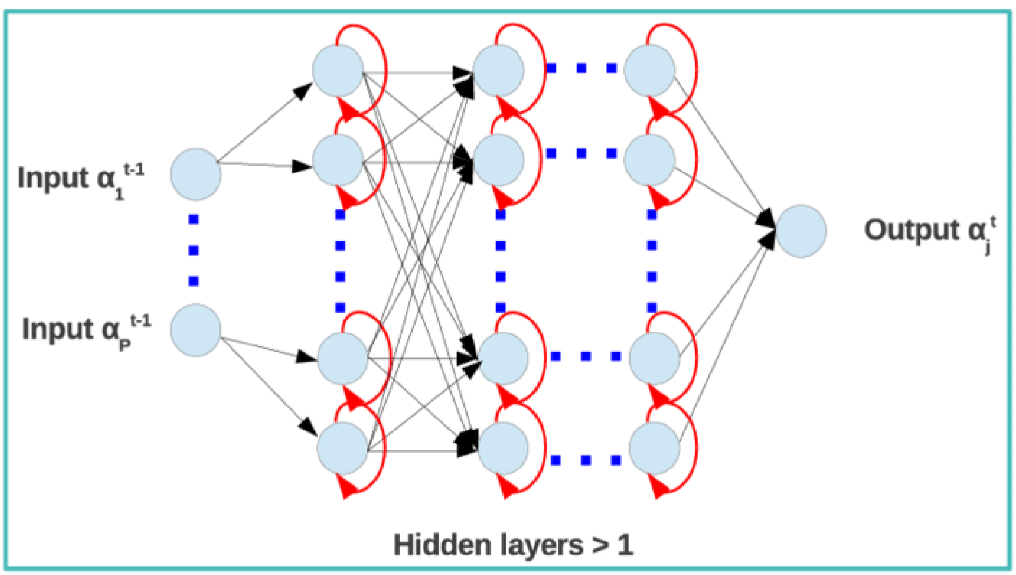
The LSTM model is one of the best machine learning models when it comes to analyzing the trends and tendencies in the time series data. LSTMs are a special kind of recurrent neural network that is capable of learning long-term dependencies. They were first introduced by Hochreiter&Schmidhuber (1997) and have been researched thoroughly ever since. LSTMs like every RNN have a chain-like structure but instead of a repeating module with a single tanh layer, there are four hidden layers interacting in a very special way. Figure 7 illustrates the architecture of an LSTM neuron.

Figure 7: LSTM neuron architecture



There are three sigmoid and one tanh layer in the LSTM models. The first sigmoid layer determines how much of the input information is going to be kept in the model and what portion of it would be forgotten after the data is processed. Sigmoid returns the value between 0 and 1 where 0 means completely forgetting the data and 1 means completely keeping it. The next step is determining what information is going to be stored in the cell state. This is done by the second sigmoid layer called the “input gate layer”. Once the sigmoid has finished working, the first tanh layer creates a vector of candidate data values that could be added to the state. The next step is the one during which the actual storing of the new data is happening. And the last step of the process is to determine the output of the neuron. This is based on the cell state, and after running the last sigmoid layer to decide what parts of the cell state are going to be included in the output, the final step is to run everything through a tanh gate to actually create the output. This is in short the way LSTM neurons work, and since the network takes in new arguments as well as the outputs from the previous neurons no data is being lost.

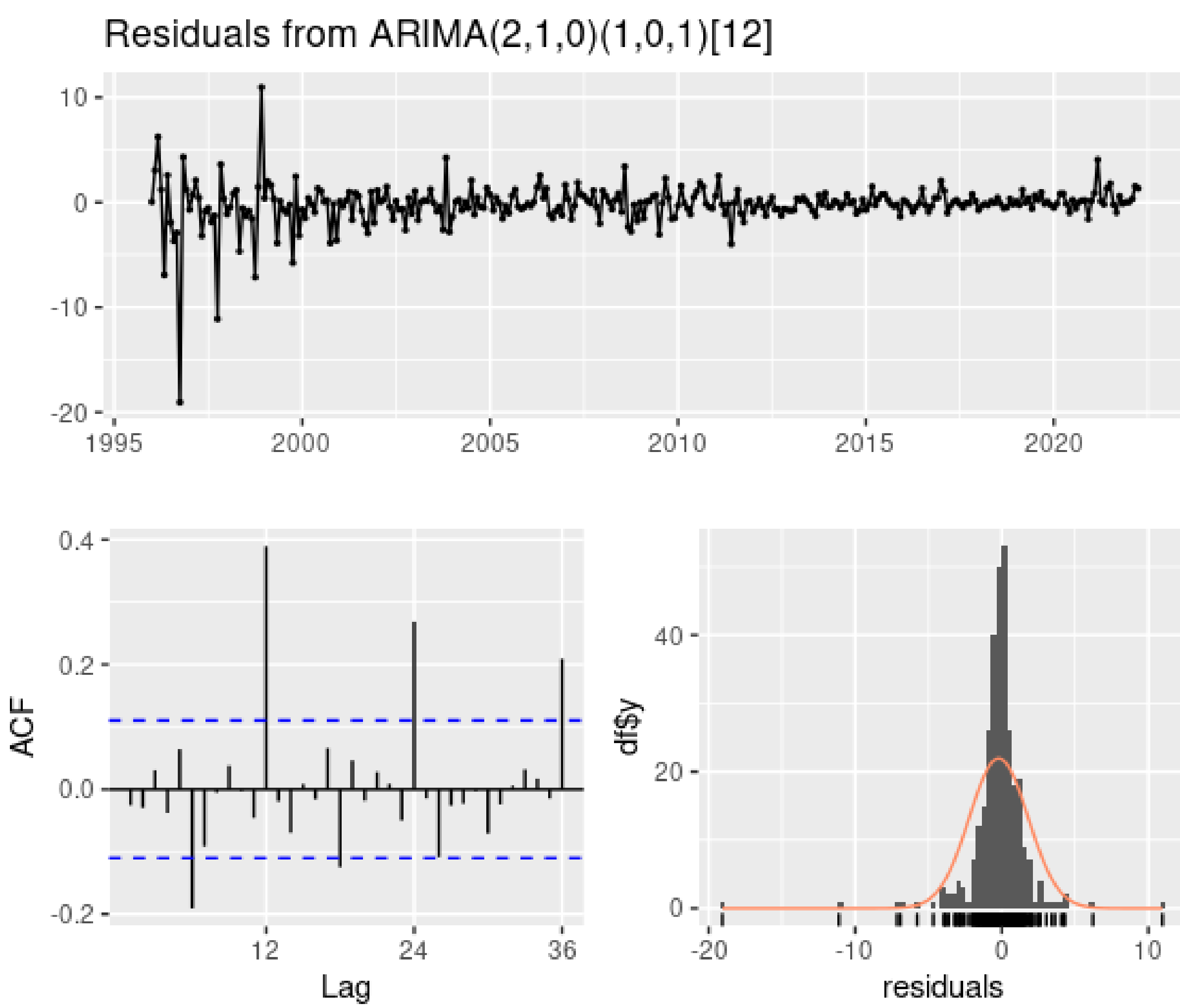
Figure 8: LSTM architecture



4 Results

Working on the results of the models is still a work in progress, so far the SARIMA model has been trained using the automatic hyperparameter selection provided by the auto Arima package in the R programming language and the model has predicted values that are 1, 3 and 6 months ahead. Appendix A shows the results of those predictions and the plots of the graphs illustrating the numbers. Figure 9 shows different metrics for the SARIMA model, we can see that the residuals are becoming less volatile as we move near the present time, meaning that the SARIMA model is able to accurately predict the data that is more stationary and less unpredictable. RMSE over the training dataset is equal to 2.014% which is really high accuracy. And the Ljung-Box test results in a p-value equal to 2.711e-13 which means that we have a slight serial correlation within the residuals.

Figure 9: Metrics for the SARIMA model



LSTM model on the other hand showed an RMSE equal to 6.84% on the validation dataset, The whole dataset was divided into 80% and 20% as discussed in the methodology part of the paper. The current model has an architecture that looks as follows, 2 LSTM layers with 500 neurons each, 2 densely connected layers each containing 50 neurons, a layer with 25 Densely connected neurons, and the last layers consisting of only one output neuron. The model uses 12 previous lags as inputs meaning that a year's worth of information is needed to produce a prediction for the following month. This is the architecture that has shown the best results so far, but the fine-tuning of the hyperparameters and the general architecture is still not finally decided, it will take more trial and error to get to the best possible solution. Appendix B shows the behavior of the model on the validation dataset.

5 Conclusions

The paper tries to compare machine learning and econometric models with each other to determine which of those is better for forecasting volatile time-series data such as inflation. So far SARIMA model has been performing better compared to the LSTM model, that fact can be explained by the methodology by which the models work. SARIMA has a built-in optimizer for the best hyperparameters, the algorithm used for that model produces hyperparameters that yield the best results pretty much all the time, while LSTM needs manual tuning to find the best model architecture, a number of lags, input parameters, etc. The final results and conclusions are not yet drawn and it will still be seen which of the models is better eventually and for the long-run forecasts, but so far the ARIMA family of models has had an upper hand, the main pros being the ease of use, automatic hyperparameter optimization, faster learning times and being less resource-intensive. LSTM is a more powerful framework that naturally requires more time, resources, and fine-tuning.

The results thus far are consistent with the existing literature and prove that both of the models are better than simple dummy random walk models.

The next steps of this study include testing out models using cross-validation, working on designing a better architecture and parameters for the LSTM model, and comparing the models to each other in short, medium, and long runs.

Bibliography

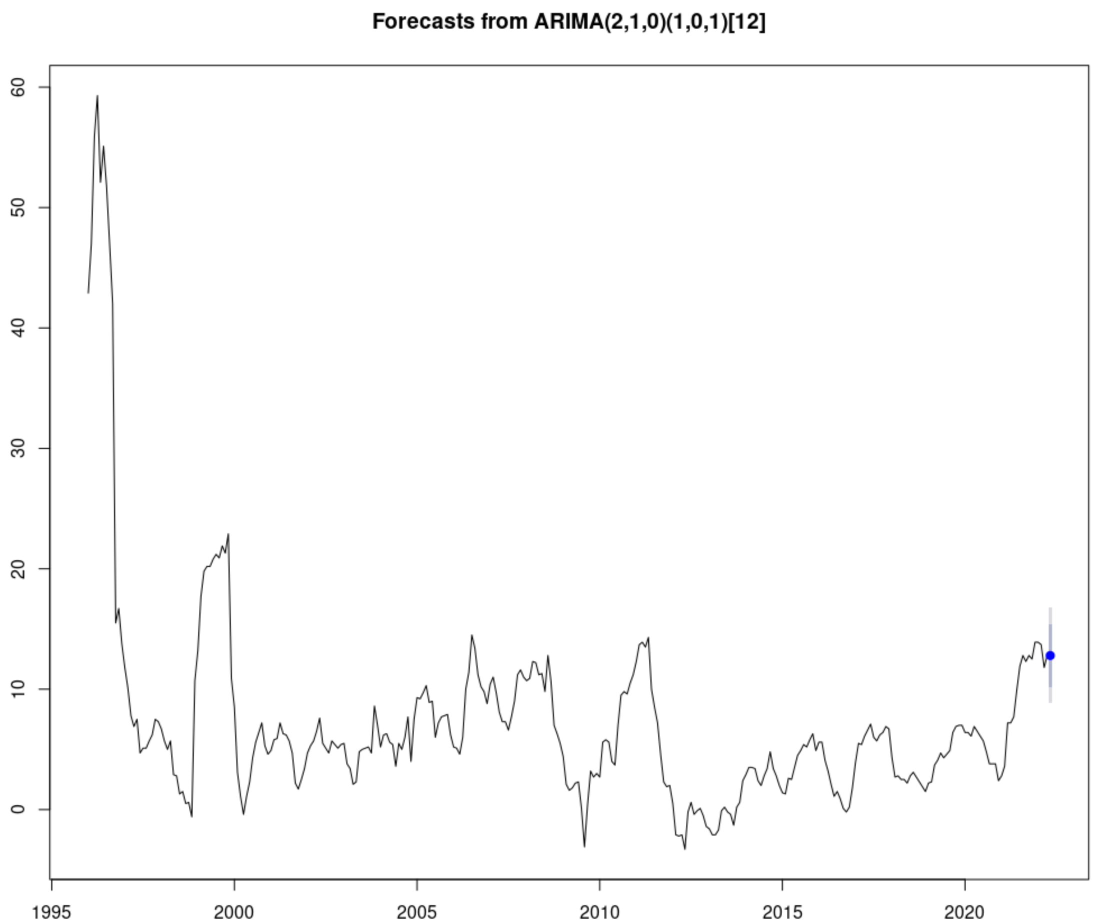
1. A. J. Chiodo and M. Owyang (2002). “A case study of a currency crisis: the Russian default of 1998”
2. Athey, S. (2019). “The Impact of Machine Learning on Economics.” In “Ajay Agrawal, Joshua Gans, and Avi Goldfarb (Eds.), The Economics of Artificial Intelligence: An Agenda,” Chicago: University of Chicago Press.
3. Atkeson, A., and Ohanian L. E. (2001). “Are Phillips Curves Useful for Forecasting Inflation?” Federal Reserve Bank of Minneapolis Quarterly Review.
4. Baybuza, I. (2018). Inflation Forecasting Using Machine Learning Methods. Russian Journal of Money and Finance, 77(4), pp. 42–59.
5. Brownlee J. (2017), “A Gentle Introduction to Autocorrelation and Partial Autocorrelation”
6. Brownlee, J. (2020). “Train-Test Split for Evaluating Machine Learning Algorithms”
7. Chakraborty, C., and A. Joseph. (2017). “Machine learning at central banks.” Bank of England Working Papers 674.
8. Duncan, R., and Garcia, E. (2018). “As good as a random walk: Inflation forecasting in emerging market economies”
9. Gujarati, D., and Porter, D. (2009). “Basic Econometrics, fifth edition”.
10. Kingma, D., and J. Ba. (2015). “Adam: A method for stochastic optimization.” In “ICLR,” .
11. Litterman, R., (1986). “Forecasting with Bayesian Vector Autoregressions - Five Years of Experience”, Journal of Business and Economic Statistics
12. McCracken, M. W., and S. Ng. (2016). “FRED-MD: A monthly database for Macroeconomic Research.” Journal of Business & Economic Statistics.
13. Medeiros, M. C., G. Vasconcelos, A. Veiga, and E. Zilberman. (2019). “Forecasting Inflation in a Data-Rich Environment: The Benefits of Machine Learning Methods.” Journal of Business & Economic Statistics.
14. Nakamura, E. (2005). “Inflation forecasting using a neural network.” Economics Letters 86.

15. National Bank of Georgia (2021). “Inflation Targeting”. Retrieved from: https://www.nbg.gov.ge/index.php?m=538&lng=eng.
16. Paranhos, L., (2021). “Predicting Inflation with Neural Networks”, University of Warwick
17. Preeti, R. Bala, and R. P. Singh, "Financial and Non-Stationary Time Series Forecasting using LSTM Recurrent Neural Network for Short and Long Horizon," 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019, pp. 1-7, DOI: 10.1109/ICCCNT45670.2019.8944624.
18. Sermpinis, G., C. Stasinakis, K. Theofilatos, and A. Karathanasopoulos. (2014). “Inflation and Unemployment Forecasting with Genetic Support Vector Regression.” Journal of Forecasting 33, 471–487.
19. Stock, J. H., and M. W. Watson. (2007). “Why Has U.S. Inflation Become Harder to Forecast?” Journal of Money, Credit and Banking.
20. Stockton, D., and J. Glassman, (1987). “An Evaluation of the Forecast Performance of Alternative Models of Inflation”, Review of Economics and Statistics

Appendices

A. Results and Forecast of SARIMA Model

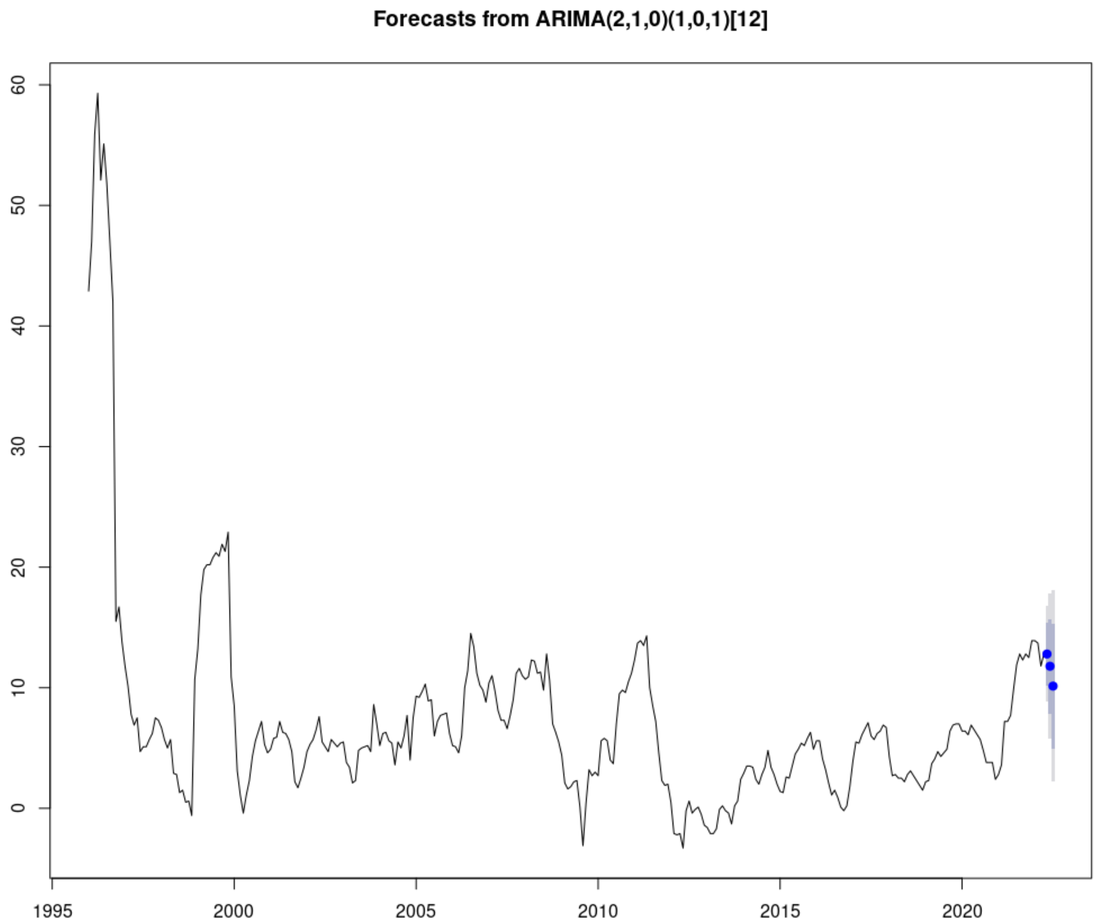
1 month ahead:



Forecasts:

Point	Forecast	Lo 80	Hi 80	Lo 95	Hi 95
May 2022	12.79314	10.191	15.39527	8.813516	16.77275

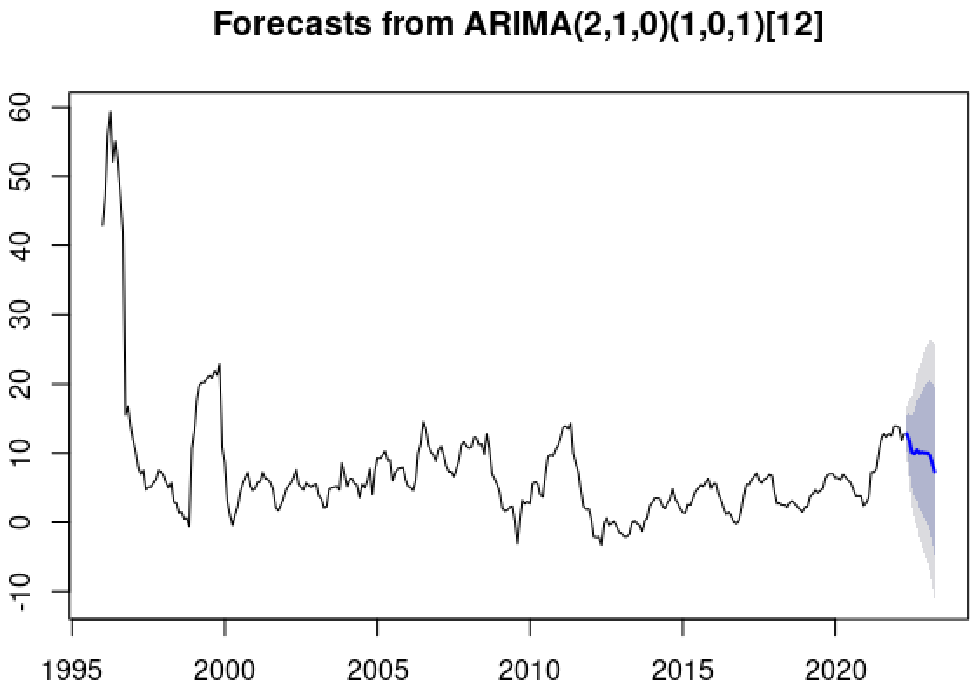
3 months ahead:



Forecasts:

Point	Forecast	Lo 80	Hi 80	Lo 95	Hi 95
May 2022	12.79314	10.191002	15.39527	8.813516	16.77275
Jun 2022	11.78047	7.854001	15.70694	5.775453	17.78549
Jul 2022	10.13377	4.938738	15.32879	2.188656	18.07888

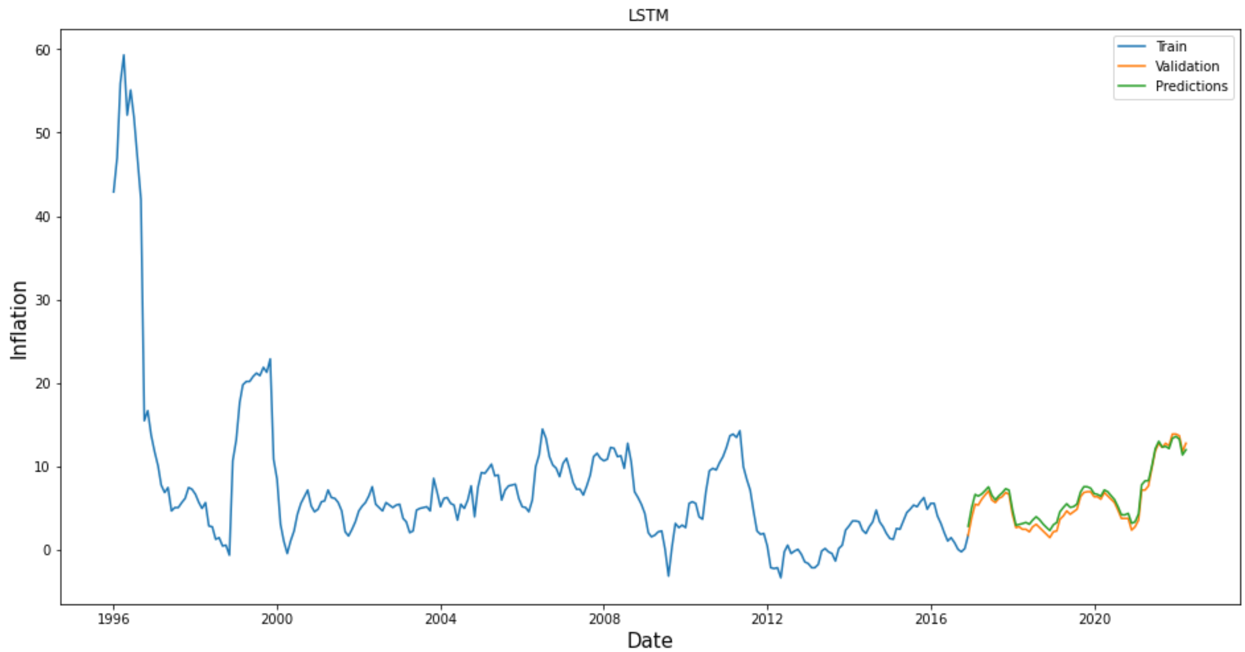
6 months ahead:



Forecasts:

Point	Forecast	Lo 80	Hi 80	Lo 95	Hi 95
May 2022	12.793135	10.191002	15.39527	8.8135160	16.77275
Jun 2022	11.780471	7.854001	15.70694	5.7754534	17.78549
Jul 2022	10.133766	4.938738	15.32879	2.1886560	18.07888
Aug 2022	9.908837	3.635922	16.18175	0.3152412	19.50243
Sep 2022	10.438623	3.203782	17.67346	-0.6261111	21.50336
Oct 2022	9.993351	1.896900	18.08980	-2.3891021	22.37580

B. LSTM Model on Validation Dataset



1 month ahead prediction:

Date	Forecast
May 2022	12.05