BANGKO SENTRAL NG PILIPINAS

BSP Working Paper Series

Series No. 2020-10

October 2020

Forecasting regional inflation in the Philippines using machine learning techniques: A new approach

Vidal Marvin C. Gabriel,
Dennis M. Bautista, and
Cherrie R. Mapa

Forecasting regional inflation in the Philippines using machine learning techniques: A new approach

Vidal Marvin C. Gabriel, Dennis M. Bautista, and Cherrie R. Mapa

Abstract

This study develops the first regional inflation forecasting models for the Philippines employing non-linear machine learning approaches for a few representative regions of the country. These regional forecasting models are expected to supplement the BSP's suite of macroeconomic models used for forecasting and policy analysis. In particular, three machine learning methods are employed: support vector regression (SVR), artificial neural networks (ANN), and long-short term memory (LSTM) to forecast inflation for the selected regions using univariate and multivariate processes. These models are evaluated based on root mean square error (RMSE) and mean absolute error (MAE) in one-month ahead static forecasting and 12-month ahead dynamic forecasting. The results indicate relatively good performance of the models for month-ahead forecasting while SVR models dominated in the 12-month ahead dynamic forecasting exercises. Furthermore, the models are evaluated vis-à-vis traditional ARIMA models and this paper finds evidence that machine learning methods do outperform ARIMA models in forecasting.

JEL classification: E31, E37

Keywords: forecasting, ARIMA, support vector regression, artificial neural networks, long-short term memory, regional inflation forecasting

Corresponding authors: Vidal Marvin C. Gabriel (vmgmarv@gmail.com),
Dennis M. Bautista (dmbautista@bsp.gov.ph),
Cherrie R. Mapa (CRamos@bsp.gov.ph)

Table of Contents

Forecasting regional inflation in the Philippines using machine learning techniques: A new approach

Vidal Marvin C. Gabriel, Dennis M. Bautista, and Cherrie R. Mapa[*]

1. Introduction

The Bangko Sentral ng Pilipinas (BSP) officially adopted inflation targeting as framework for monetary policymaking in 2002. Since then, inflation forecasting models have been an integral component of the policymaking process at the central bank. The BSP employs a suite of models to nowcast and forecast inflation and other key macroeconomic variables such as output and money supply. These models range from simple autoregressive integrated moving average (ARIMA) models to a more complex macroeconomic model with an endogenous monetary policy rule. The main objective of this paper is to develop the first regional inflation forecasting models for the Philippines, employing three non-linear machine learning approaches for selected representative regions of the country. These regional forecasting models are expected to supplement the BSP's suite of macroeconomic models used for forecasting and policy analysis.

The government's inflation target is expressed in terms of nationwide, year-on-year inflation rate and as such, forecasts used for policymaking should likewise refer to the nationwide inflation. Aggregated regional inflation forecasts could provide an alternative method of nowcasting the nationwide inflation. For this paper, models are built for seven regions representing Luzon (NCR, Central Luzon and CALABARZON), Visayas (Western Visayas and Central Visayas), and Mindanao (Northern Mindanao and Davao). Together, these regions already account for 73.1 percent of the country's consumer price index (CPI) basket. Moreover, these are pioneering models for generating regional inflation forecasts and could be helpful for surveillance of regional trends. While these algorithm-generated models have limited capability in providing insights on causal factors, these models can help policymakers spot a region that might have a diverging trend from the nationwide trend. The intent is not to help craft a monetary policy response to regional inflation. Rather, the generated forecasts can signal to policymakers whether a region merit closer scrutiny and if there are any other possible targeted government policy that could be deployed to preserve price stability in that region.

There is an existing large body of empirical literature devoted to modeling inflation. Similar to other time series data, the simplest and perhaps most commonly used statistical tools to forecast inflation are the auto regressive (AR) and ARIMA models (Moser et al., 2007; Baciu, 2015). More recently, machine learning algorithms have started to attract interest from researchers (Binner et al., 2005; Siami-Namini et al., 2020; Okasha & Yassen, 2013) in time series forecasting because of its non-linear capabilities and promising results. In fact, there are a number of studies that tried to compare these models and found out that some of the

[*] The authors are presently assigned at the Economic and Financial Forecasting Group of the Department of Economic Research. The expert assistance of Ms. Rosemarie A. Amodia is gratefully acknowledged. The usual disclaimer applies.

machine learning models can outperform the commonly-used statistical tools (Esquivel-Monge, 2009; Chakraborty, 2017; Hurtado & Cortes-Fregoso, 2013; Cortez et al., 2001). This paper follows this approach and explores the use of machine learning techniques for regional inflation forecasting. Specifically, three machine learning methodologies will be utilized: support vector regression (SVR), feed-forward artificial neural networks (ANNs), and recurrent neural networks – long-short term memory (LSTM). These models will be evaluated based on their root mean square error (RMSE) and mean absolute error (MAE) in one-month ahead static forecasting and 12-month ahead dynamic forecasting. Moreover, forecast performance of the machine learning models will be assessed vis-à-vis traditional ARIMA models.

The remainder of this paper is structured as follows: Section II presents an overview of the machine learning algorithms used in this paper. Section III provides a review of the empirical literature. Section IV covers data preparation, architecture of the models, and the performance metrics. Section V discusses the results and Section VI concludes.

2. Overview of Machine Learning Models

Machine learning is an approach to forecasting where there are no priori assumptions between the relationship of input and output variables but uses an algorithm to detect or learn any relationship between these variables. Based on this learning, the model then creates a function that helps predict the output variable given the input variable. There is no singular way of doing machine learning. Rather, there are a myriad of machine learning techniques. This paper utilizes SVR, ANN and LSTM.

2.1 Support vector regression (SVR)

2.1.1 Brief mathematical background of SVR

SVR is an application of support vector machines (SVM). In an ordinary least squares regression with one predictor, the goal is to minimize the objective function:

$$min\sum_{t=1}^{n}(y_t - w_t x_t)^2 \tag{1}$$

where y_t is the target, w_t is the coefficient, and x_t is the predictor (feature). SVR takes this a step further by allowing the definition of the acceptable level of error (Smola & Schölkopf, 2004). The objective of SVR is to minimize the coefficient w while the error term is handled in constraint, i.e., error should be less than the specified threshold, ε. This modifies **Equation 1** to:

$$min\frac{1}{2}||w||^2 \tag{2}$$

and the constraint is given by:

$$|y_t - w_t x_t| \leq \varepsilon \tag{3}$$

However, there will be cases wherein data will violate **Equation 3** (Smola & Schölkopf, 2004). Data can exceed ε with some deviation. To accommodate this instance, ξ is added in **Equation 2** as a regularization parameter which then yields:

$$min\frac{1}{2}||w||^2 + C\sum_{t=1}^{n}|\xi_t| \tag{4}$$

with a new constraint:

$$|y_t - w_t x_t| \leq \varepsilon + |\xi| \tag{5}$$

In this configuration, there is an additional hyperparameter, C that should be tuned. Furthermore, it can be noted that C and ε are directly proportional. This means that as C approaches 0, **Equation 4** reverts back to **Equation 2**.

2.1.2 Limitations of SVR

There may be instances when the algorithm may perform poorly. SVR may not perform very well when the dataset has more noise. SVR also underperforms when the number of features for each data point exceeds the number of training data sample. In other words, SVR is not suitable for large data sets. Moreover, since the model is bounded by ε and ξ, no probabilistic explanation can be formulated.

2.2 Feed-forward Artificial Neural Networks (ANNs)

Artificial neural networks (ANNs) are computational models that mimic the behavior of the human brain that has the capability to learn. Essentially, inputs are processed through a learning node or perceptron to help generate an output. In order to more adequately mirror complex real-life decision-making processes, researchers would typically use more than one perceptron, thereby producing a so-called neural network. Goodfellow & Bengio, 2016 is the main reference in illustrating the mathematical description of ANN.

ANNs are parameterized, oftentimes nonlinear functions, that can be fitted to data in order to get a desired forecast. They are composed of interconnected pool of neurons that are linked together. Consider an n-dimensional input vector of the j^{th} neuron $x = (x_{1,} \ldots, x_n)$ with vector weights $w = (w_{j1,} \ldots, w_{jn})$ and a bias b_j, the output $y_{j,}$ of the output neuron is then given by:

$$y_j = f\left(\sum_{t=1}^{n} w_{jt}x_t + b_j\right) \tag{6}$$

where function f is the activation function. Note the weights determines the connection between two neurons. It decides how much influence the input will have on the next neuron. Meanwhile, biases b are additional inputs into the next layer usually into the hidden layer. This is done to prevent zeroes in the input which will give an error in the activation function. This paper uses the hyperbolic tangent activation function presented below:

$$f(x) = \tanh(z) = \frac{\exp(z) - \exp(-z)}{\exp(z) + \exp(-z)} \tag{7}$$

2.2.1 Multi-layer perceptrons (MLPs)

Multi-layer perceptrons (MLPs) are ANNs with one or more layers of neurons that are interconnected together. In a feed-forward fashion, data is fed in the input layer then passed through to the hidden layer/s, and finally to the output layer. The output of each layer is then the input of the successive layer. On a side note, however, the neurons of the same layer are not connected.

For a simple mathematical representation of MLP, consider an MLP with n inputs, m neurons in the hidden layer and c neurons in the output layer. Assume $x \in \mathbb{R}^n$ be the input vector of the neural network, and w the matrix of the weights in the first layer, the resulting output of the j^{th} neuron in the hidden layer is:

$$a_j = f\left(\sum_{t=1}^{n} w_{ji}x_i + b_{j1}\right) \tag{8}$$

where w_{ji} is the weight parameter of the connection from the i^{th} input to the j^{th} neuron in the hidden layer, b_{j1} is the bias of the j^{th} neuron of the first layer.

After passing to the hidden layer, the connection between the output y_k of the k^{th} neuron and the hidden neuron a_j is presented in the following equation:

$$y_k = g\left(\sum_{j=1}^{m} v_{kj}a_j + b_{k2}\right) \tag{9}$$

where v_{kj} is the weight of the connection from the j^{th} hidden neuron to the k^{th} neuron in the output layer and b_{k2} is the bias of the kth neuron of the output layer. Combining **Equation 8** and **Equation 9** gives the explicit form of the output of the MLP network as shown below:

$$y_k = g\left(\sum_{j=1}^{m} v_{kj}f\left(\sum_{t=1}^{n} w_{ji}x_i + b_{j1}\right) + b_{k2}\right) \tag{10}$$

Equation 10 can be extended to the case of more than one hidden layer while maintaining the same iterative process. Due to these properties of the ANNs, it has gained popularity in forecasting and modelling activities of different types of data. Its popularity is related to its simplicity and ability to deal with complex multi-dimensional mappings.

2.2.2 Learning algorithm through back-propagation

Just like the human brain, ANNs need to learn in order to predict future data values. If the error of the MLP network has some error functions that is differentiable, it is then possible to estimate the derivative of the error with respect to the weights and modify them in order to minimize the error function.

Consider again the MLP in the above illustrative example with n inputs, m neurons in the hidden layer and c neurons in the output layer and assume y^p the forecasted output and d^p the corresponding actual data (or target) of the network. The network (MLP) should learn N input patterns. The goal then of back-propagation algorithm is to modify the weights of the network to minimize the error function:

$$E(w) = \frac{1}{Nn}\sum_{p=1}^{N}\sum_{k=1}^{c}\left[d_k^p - y_k^p\right]^2 \tag{11}$$

The error function (see **Equation 11**) is a continuous differentiable of the weights of the network. Further, gradient descent technique can be used for the minimization process. From the above **Equation 11**, the weights can be updated using the following equation:

$$w(l+1) = w(l) - \eta\frac{\partial E(w)}{\partial w} \tag{12}$$

where l is the l^{th} iteration of the algorithm and η is the learning rate. Updating the weights should follow the gradient of $E(w)$ where it decreases. It can be noted that if η is too low, training progress will be very slow. On the other hand, if it is too high, it could cause undesirable divergence behavior in $E(w)$.

2.2.3 Limitations of ANN

The best-known disadvantage of all feed-forward ANNs is their "black box" approach. In other words, one doesn't know exactly how or why the ANN came up with the output. This is because, as the layers and nodes are increased, it becomes extremely difficult to come up with an equation to explain the process.

Another disadvantage of feed-forward ANNs is that they require much more data than other machine learning algorithms. Although there are some instances where neural networks do well with little data, this is not often the case.

Feed-forward ANNs require higher computer capacity since they are usually computationally expensive. In a state-of-the-art deep learning algorithm, successful training of really deep neural networks can take several days or even weeks to train completely from scratch. Hence, ANNs may not be suitable for real-time forecasting.

Feed-forward ANNs also suffer in vanishing gradient phenomenon. This is a consequence of the chain rule differentiation process in the back-propagation algorithm. When the derivative of the error function becomes constant, the weights in the first few layers will no longer be updated. When this happens, the model will stop learning and will provide erratic forecasts.

2.3 Recurrent Neural Networks (RNN)

Data in a time series models are sequential in nature. Although some studies have shown that feed-forward ANNs are capable of time series forecasting, it can be improved when the decisions of the model are influenced by what it has learnt from the past. Basic ANNs remember things too, but they remember things they learnt during training. A special type of ANNs where it can store past memory are RNNs. While RNNs learn in training, they also remember things learnt from prior inputs while generating outputs. Hochreiter & Schmidhuber, 1997 provide a good discussion of RNN.

The hidden layers of RNNs act as internal storage for storing the information captured in earlier stages of reading sequential data. It performs the same task for every element of the sequence, with the ability of utilizing information captured earlier to predict future unseen sequential data, hence the name "recurrent".

In a sequential manner, RNN takes x_{t-1} from the sequence of input data (time series) and estimates h_{t-1}. Then, it takes x_t and h_{t-1} as the new inputs for generating h_t. This process of RNN makes it remember the previous data while training. The current state of RNN can be represented by **Equation 13** below:

$$h_t = f(h_{t-1}, x_t) \tag{13}$$

Using the above equation, we can now apply an activation function to produce **Equation 14**:

$$h_t = \tanh\,(W_{h-1}h_{t-1} + W_{ts}x_t) \tag{14}$$

where W_{h-1} is the weight at previous hidden state and W_{ts} the weight at current input state. Since the activation function in **Equation 14** was set to tanh, it introduces non-linearity to h_t.

2.3.1 Long short-term memory (LSTM)

Long short-term memory (LSTM) is a special type of RNN that has the capability to remember the values from earlier stages for the purpose of future prediction. In other words, it has additional features to memorize the sequence of data. This model was introduced by Hochreiter & Schmidhuber in 1997. Each LSTM is a set of cells, or system modules, where the data streams are captured and stored. Three types of gates are involved in each LSTM with the goal of controlling the state of each cell:

Forget gate: This gate assess what information will be thrown away. It outputs a number between 0 (forget information) and 1 (keep information) due to the sigmoid activation function - $\sigma(x) = \frac{1}{1+\,e^{-x}}$. **Equation 15** below is the mathematical representation of the forget gate.

$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] \; + \; b_f) \tag{15}$$

where h_{t-1} and x_t are previous hidden state and current input state, respectively.

Memory gate: This gate chooses new data that will be stored in the cell state. The sigmoid layer presented in **Equation 16** below chooses what information will be updated.

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t]) \tag{16}$$

It is called the "input gate layer". Meanwhile, **Equation 17** below makes a vector of new candidate values, $\bar{C}_t$ that could be stored to the cell state.

$$\bar{C}_t = tanh(w_c \cdot [h_{t-1}, x_t] + b_c) \tag{17}$$

To update the previous cell state, C_{t-1} to new cell state, C_t the forget gate (see **Equation 15**) is manipulated to the previous cell state, C_{t-1} and added to the product of **Equation 16** and **Equation 17**.

$$C_t = f_t * C_{t-1} + i_t * \bar{C}_t \tag{18}$$

Output gate: The output gate yields an output based on the cell state along with the filtered and newly added data. The sigmoid layer is first implemented to decide what parts of the cell state will be considered.

$$\sigma_t = \sigma(w_o \cdot [h_{t-1}, x_t]) \tag{19}$$

Then, the output of the sigmoid layer will be multiplied to **Equation 17**. The output gate is represented mathematically by the **Equation 20** below:

$$h_t = \sigma_t * tanh(C_t) \tag{20}$$

2.3.2 Limitations of LSTM

Just like the feed-forward ANNs, LSTMs also suffer from a "black box" approach and requires higher data resolution. Furthermore, since the structure of LSTM is more complex than feed-forward ANNs, it requires higher computational power and thus longer running time.

3. Review of Literature

There is a growing body of literature on the empirical application of machine learning algorithms for forecasting in the fields of economics and finance. Results, thus far, have generally been favorable, attesting to the usefulness of said techniques for these fields.

For example, SVR has been used to forecast stock prices. Shen et al., 2012 explored the capabilities of SVR in forecasting US stock prices. Their study used the lag values of the time series together with other market features to forecast prices. Based on their numerical tests, SVR forecasts tend to have high accuracy.

In terms of forecasting inflation, Zhang et al., 2012 used SVR to forecast inflation in China. They argued that SVR could outperform ANNs because an SVR does not need bigger dataset to perform optimal. They compared three models: linear regression, feed-forward ANNs, and SVR. Based on the MAE of the three models, SVR registered the lowest (0.2) followed by ANNs (2.3). Linear regression performed the worse with a score of 2.8. Similar conclusion was observed when RMSE is calculated instead of MAE.

ANN modelling has also been gaining attention as an attractive technique for estimation and forecasting time series in recent years due to its nonlinear properties. Adebiyi et al., 2014 and Okasya et al., 2013 compared the performance of ARIMA and ANN in forecasting stock prices. The former (Adebiyi et al., 2014) studied the performance of the two models in New York Stock Exchange while the latter (Okasya et al., 2013) in Al-Quds index of the Palestine stock exchange. Both studies showed that ANNs are superior relative to ARIMA.

Meanwhile, Binner, et al., 2005 compared nonlinear ANNs with linear univariate time series ARIMA model and multivariate vector auto regression (VAR) model in forecasting Euro inflation. Their research showed that ANNs outperform the traditionally-used linear ARIMA and VAR models in macroeconomic forecasting and are statistically superior to them. Further, they have noted that the gain in forecasting accuracy in ANNs is very much likely to have emerged from the capability of the ANNs to capture nonlinear relationships between macroeconomic variables.

Chavez-Hurtado et al., 2013 similarly utilized ANN to forecast inflation for Mexico. They categorized Mexico's inflation into three stages: volatile phase - from 1994 to 1998, transition phase - from 1998 to 2001, and stability phase - from 2001 to 2010. The authors found that during the volatile period, their ANN model registered better performance than the Bank of Mexico model. However, during the transition phase, the Bank of Mexico model had better accuracy than their ANN model. Nonetheless, as the ANN model learned from the updated database (during stability phase in 2001 to 2010), its performance has again improved compared to the Bank of Mexico model.

As discussed in the previous Section, ANNs have the capability to find patterns based on multiple inputs. Sari et al., 2002 for example, used ANN to forecast inflation rate in Indonesia and accommodate multiple features, (i.e. $m-1$ (inflation rate a month ago), $m-2$ (inflation rate two months ago), $m-3$ (inflation rate three months ago), and CPI). ANNs manifested a relatively low RMSE (0.204) which is superior to fuzzy logic, their reference model. Thakur et al., 2016 similarly designed ANNs for forecasting inflation in India with multiple features as inputs: output, exchange rate, export, money supply, imports, gold prices, oil prices, and the balance of trade. They found that their generated forecasts are closer to the actual inflation than forecasts of Organization for Economic Cooperation and Development (OECD) and International Monetary Fund (IMF).

There are also a number of studies that compared LSTM with traditionally-used statistical tools in modelling time series. Most of these studies lent support to the view that LSTM could be used for time series forecasting. Siami-Namini et al., 2018 compared LSTM with ARIMA in forecasting different stock prices. On average, LSTM improved their forecast accuracy by 85 percent compared with ARIMA.

A similar study was performed by Lee & Yoo, 2017 wherein they used LSTM to forecast the top 10 stocks in terms of market value from the Standard and Poor's 500 index (S&P500) and concluded that deep learning and machine learning algorithms produce more accurate financial time series forecasts.

Meanwhile, Almosova & Andresen, 2004 have compared different time series forecasting models to forecast US inflation. They used AR, ARIMA, Seasonal ARIMA (SARIMA), Random walk (RW), Markov switching model (MS-AR), standard ANN, and LSTM. Their findings showed that LSTM outperforms all other forecasting techniques especially at longer horizons. Further, they have investigated the real-time forecasting performance of the LSTM by computing one-month-ahead predictions in a rolling-window setting. They have found that the prediction of LSTM is significantly superior to the ANN and VAR forecasts with the exception of some years.

During the course of this literature review, this paper found two central banks (i.e. Bank of England and Banco Central de Costa Rica) (Chakraborty, 2017 and Monge, 2009) that explored machine learning algorithms to forecast inflation in their country. Bank of England (Chakraborty, 2017) provided an overview on how to integrate machine learning algorithms in economic analyses. They presented different algorithms such as ANNs, tree-based models, SVMs, and different clustering techniques. They performed three case studies, one of which is the projection of UK CPI inflation on a medium-term horizon of two years. They used AR, VAR, and ridge regression as reference models. Their results show that the performance of all the reference models is generally worse than most machine learning algorithms. Likewise, Banco Central de Costa Rica (Monge, 2009) studied non-linearity in inflation forecasting. The main objective is to examine whether to allow non-linearity in some of the economic models for forecasting inflation in Costa Rica and if it will yield any improvement in their performance. The paper investigated if the nonlinear methodology of ANNs can improve inflation forecasts obtained from linear, structural models. They have found that using ANNs reduces the RMSE of out-of-sample forecasts as compared when OLS estimation method was applied. Their findings support the existence of relations among variables that are not fully captured by standard linear econometric methods due to the nonlinear nature of some of them.

4. Data and Methodology

This paper implements three machine learning algorithms: (1) SVR, (2) feed-forward ANN, and (3) LSTM. Since all of them can perform univariate and multivariate processes, they are tested on both configurations. All of the models are executed in a computer with the following specifications: (1) 12 gigabytes of RAM, (2) 4 gigabytes of video card, (3) 4 chores, and (4) intel i5 – 5^{th} generation processor.

4.1 Data preparation

This study includes 7 regions representing the major island groupings in the country: Luzon, Visayas and Mindanao. The data are taken from the Philippine Statistics Authority (PSA) and covers the period January 1994 to December 2019. **Table 1** lists the covered regions and their corresponding weights in the nationwide CPI basket.

Table 1. Regions Covered and Corresponding Weights in the CPI basket

Region	Weight (in percent)
National Capital Region (NCR)	22.51
CALABARZON (Region 4-A)	17.99
Central Luzon (Region 3)	12.48
Western Visayas (Region 6)	6.37
Central Visayas (Region 7)	6.34
Northern Mindanao (Region 10)	3.34
Davao Region (Region 11)	4.11
Total	**73.14**

Source: Philippine Statistics Authority (PSA), 2012-based

Inflation is defined as the percent difference between the current CPI and the CPI of the previous year as seen in **Equation 21** below:

$$\pi_{year-on-year} = \left(\frac{CPI_t - CPI_{t-1}}{CPI_{t-1}}\right) * 100 \tag{21}$$

where CPI_t is the CPI of the current month while CPI_{t-1} is the same month's CPI from the previous year.

After obtaining the year-on-year inflation, the data is normalized using **Equation 22** wherein n is the element of the array x and is multiplied with the difference between the maximum and minimum values of array x. Normalizing the time series can be useful, and even required in some machine learning algorithms to make the scale of the input values uniform. It may be required for algorithms, like the models used in this paper which weight input values.

$$y = \frac{n - x_{min}}{x_{max} - x_{min}} \tag{22}$$

Finally, the data is structured to follow the requirements of the model. This involves preparing the input to the network as the feature vector $\left[y_t, y_{t-\tau}, \dots, y_{t-(m-1)_\tau}\right]$ and the corresponding output vector as $y_{t-\tau}$.

The models also explore the introduction of a shock variable, expressed in binary form. This essentially transforms the models into multivariate processes. A value of 1 for the shock variable implies that there is an ongoing shock which can cause inflation the following month. Meanwhile, a value of 0 indicates deflationary tendency or absence of shocks that could cause inflation the following month.

To classify the shock as 1 or 0, **Equation 23** below will be used. A value of 1 is assigned if the difference between π_n and π_{n-1} is positive which indicates inflationary movement. In this manner, every time the models encounter a value of 1, it will assume positive trend and negative otherwise. This is a way of helping the model arrive on a more precise forecast.

$$b = \begin{cases} 1, & if\ \pi_n - \pi_{n-1} > 0 \\ 0, & otherwise \end{cases} \tag{23}$$

4.2 SVR architecture

Two models will be developed using SVRs. The first model is a univariate process with lag. The second model is a multivariate process wherein the inputs are inflation and shocks (i.e. 0 or 1) as determined by **Equation 23**. Both models employ a lag value of $m = 1$.

Both models will use an ε of 0.1 and a radial base function (RBF) kernel. RBF is a type of algorithm wherein it reconstructs unknown functions from a known data. Further, the functions are multivariate and may be solutions to partial differential equations satisfying certain additional conditions. Since the kernel searches for a function to describe the data, a grid search is needed. In this paper, γ which is the kernel coefficient of RBF will be iterated from $1e^{-4}$ to 1 (see **Equation 24**). This is inserted in the model in the form of an array. Further, the regularization parameter will also be iterated from $1e^{-2}$ to $1e^{-3}$ which is also in the form of an array.

$$K(x, y) = \exp\left(-\gamma \|x - y\|^2\right) \tag{24}$$

4.3 Feed-forward ANNs architecture

Similar to SVR, two models will be developed using feed-forward ANNs. The first one is a univariate model wherein the inputs are purely the lags, m of the inflation. The lag is set to three steps in the past ($m = 3$). For the second model, shocks (i.e. 0 or 1) will be included. It will also follow $m = 3$ configuration.

Both ANN models will have a single hidden layer perceptron network following the work of Figueiredo, 1980, who argued that one hidden layer is sufficient enough to characterize any continuous function. The number of input perceptrons will be equal to the lag dimension m plus one bias perceptron. In order to create a consistent architecture for the feed-forward ANN regardless of the number of inputs, the number of hidden layer perceptrons is chosen using the geometric pyramid rule (see **Equation 25**):

$$N_h = \propto \sqrt{N_t N_o} \tag{25}$$

where N_h, N_t, and N_o are the number of hidden, input, and output layer nodes respectively. In this particular study, $\propto$ is set to 2 in order to cover the complexity of the problem. The hyperbolic tanh will be used for the hidden layer activation function while a linear function is intuitively chosen as the output activation function.

Stochastic gradient descent and back propagation error algorithm will be used to obtain the optimal weights in the network. During training, the initial weights are selected randomly. As a consequence, a minimum number of random starts before choosing the optimal classifier is required (Goodfellow et al., 2016). At 95 percent confidence level, the weights should at least comprise the 30 percent value, hence the number of random starts is given by **Equation 26** below.

$$N = \frac{\ln(1-0.95)}{\ln(1-0.30)} \approx 8 \tag{26}$$

Finally, to prevent over-fitting, an early stopping in the training phase is executed when the RMSE of the network in the validation data set no longer improves with the succeeding training iterations (Prechlet, 1998).

4.4 LSTM architecture

Just like SVR and feed-forward ANNs, two models will be developed in LSTM, i.e. univariate and multivariate processes. Further, Keras library will be used as a tool to build the model. Keras is an open-source library that is written in Python.

Both models will have a single hidden layer with different number of LSTM units (100-1000). The optimal number of LSTM units for each time series will be identified through trial and error. Moreover, 50 epochs will be used to keep the running time reasonable. A higher number of epochs will give a heavier load in the processors of the computer. To avoid over-fitting, a dropout function is added. Dropout is a type of regularization where randomly selected neurons are ignored during training phase. This means that their effect to the activation of downstream neurons is temporally removed on the forward pass and any weight updates not applied to the neuron on the backward pass (Srivastava et al., 2014).

At the tail of the model is a dense layer that specifies the output of one unit. The model is then compiled using the Adam optimizer in contrast to feed-forward's back propagation. Adam combines the advantages of adaptive gradient algorithm (AdaGrad) and root mean square propagation (RMSProp) (Kingma et al., 2014). AdaGrad maintains a per-parameter learning rate that improves performance on problems with sparse gradients while RMSProp maintains per-parameter learning rates that are adapted based on the average of recent magnitudes of the gradients for the weight. In this manner, Adam can thrive well on non-stationary dataset.

4.5 Model assessment

To assess each model's performance, the RMSE and MAE (see **Equation 27** and **Equation 28**) will be computed and compared for both the one-month ahead static inflation forecasts and the 12-month ahead dynamic forecasts. Since there is limited number of data, 20 percent of the whole time series will be set aside for the testing phase to evaluate the one-month ahead forecasts of the models. Meanwhile, the whole 2019 data will be removed and will be considered as the test set for the 12-month ahead dynamic forecasts. The remaining data will be used as train set wherein 5 percent of which will serve as the validation set.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2} \tag{27}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|e_i| \tag{28}$$

As of writing, no literature can be found with regards to the most optimal value of RMSE and MAE. The rule of thumb is that the lower the values of the two metrics, the better the model's performance.

The forecast performance of these machine learning techniques will also be compared vis-à-vis traditional ARIMA models. These statistical models will also be programmed in Python. Its parameters p – order of autoregressive terms and q – order of the moving average terms will be autotuned using different value combinations from 0 to 3 for model parsimony while d – degree of differencing will be set to one. The RMSE and MAE for both the month-ahead and 12 month-ahead forecasts of these ARIMA models will be computed and compared to those of the machine learning models.

5. Results and Discussions

The plots of the forecasts generated using the different models for the sample regions along with the actual inflation for the regions are shown in the Appendix. **Figures 1.1 – 1.28** depict the month-ahead forecasts while **Figures 2.1 – 2.28** show the dynamic forecasts. Notably, the plots indicate relatively good forecasting capability of the models for month-ahead forecasts. For the dynamic 12-month ahead forecasting, visual inspection of the plots suggests that SVR models outperformed both ANN and LSTM models.

Meanwhile, **Table 2** shows the RMSE and MAE scores of the forecasting models for the month-ahead forecasts. It can be noted that all the models have relatively good/low RMSE and MAE scores. Across regions, the ANN univariate models with an average RMSE of ~ 0.20 registered the best performance while LSTM with shocks had the worst mean RMSE of ~ 0.53. Meanwhile, in terms of MAE scores, SVR models had the lowest average MAE of ~ 0.13 across the regions while LSTM with shocks posted the highest average MAE of ~ 0.43 This means that despite the sophisticated method of LSTM with shocks (Goodfellow & Bengio, 2016), it showed the worst performance in forecasting activity for the one-month ahead forecast as it scored the poorest on both performance metrics.

Table 2. RMSE and MAE scores of the various models for one-month ahead forecasting

Forecasting model	NCR		Region 3		Region 4-A		Region 6		Region 7		Region 10		Region 11	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
SVR univariate	0.38	0.23	0.04	0.02	0.34	0.20	0.07	0.06	0.32	0.26	0.22	0.14	0.17	0.11
SVR with shocks	0.38	0.11	0.03	0.02	0.35	0.21	0.13	0.12	0.32	0.23	0.22	0.14	0.17	0.11
ANN univariate	0.26	0.17	0.09	0.08	0.18	0.13	0.13	0.10	0.26	0.19	0.34	0.25	0.13	0.13
ANN with shocks	0.76	0.21	0.73	0.25	0.65	0.17	0.60	0.25	0.68	0.23	0.63	0.23	0.75	0.19
LSTM univariate	0.21	0.17	0.18	0.15	0.28	0.21	0.28	0.24	0.51	0.43	0.20	0.14	0.28	0.21
LSTM with shocks	0.58	0.43	0.30	0.24	0.39	0.32	0.31	0.21	0.71	0.54	0.83	0.75	0.61	0.49
ARIMA	0.42	0.32	0.36	0.27	0.42	0.33	0.39	0.28	0.44	0.35	0.38	0.31	0.52	0.41

Source: Authors' estimates

Table 3 shows the optimal one-month ahead forecasting model for each region. It can be seen that no single model won it all. Nonetheless, it appears that all univariate models have the better RMSE scores. Hence, based on RMSE, it appears that a univariate process is already good enough to generate fairly reliable forecasts. Meanwhile, based on the MAE metric, four regions require multivariate processes (i.e. NCR, Region 3, Region 10, and Region 11).

Another key consideration in assessing the performance of the models is their running time. ANN with shocks took the longest time to finish forecasting at an average of 2 hours, 50 minutes. Meanwhile, SVR univariate has the fastest running time at an average of 38 minutes.

Table 3. Optimal one-month ahead forecasting model for each region

Region	RMSE	MAE
NCR	LSTM univariate	SVR with shocks
Region 3	SVR univariate	SVR univariate & with shocks
Region 4-A	ANN univariate	ANN univariate
Region 6	SVR univariate	SVR univariate
Region 7	ANN univariate	ANN univariate
Region 10	LSTM univariate	SVR univariate, SVR with shocks & LSTM univariate
Region 11	ANN univariate	SVR univariate & SVR with shocks

Source: Based on authors' estimates

This study also assessed results generated from dynamic forecasting. In this exercise, the forecasted inflation is inserted in the model to forecast the succeeding values. **Table 4** shows the RMSE and MAE scores of the different models for the sample regions based on their dynamic forecasts for 2019. In this exercise, SVR univariate, SVR with shocks, and ANN univariate posted the best RMSE and MAE results across all regions as the other models registered higher RMSE and MAE, even exceeding one.

In particular, based on RMSE, SVR with shocks with an average of ~0.65 registered the best performance among the six models while LSTM with shocks had the worst performance with a mean RMSE of ~1.48. The same conclusion could be arrived based on MAE scores. SVR with shocks with an average MAE of ~0.65 appeared to be the best model while LSTM with shocks with an average MAE of ~1.48 was the worst.

Table 4. RMSE and MAE scores of the various models for 12-month ahead forecasting

Forecasting Model	NCR		Region 3		Region 4-A		Region 6		Region 7		Region 10		Region 11	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
SVR univariate	0.63	0.52	0.36	0.28	0.60	0.48	0.67	0.55	0.87	0.80	0.76	0.61	0.81	0.67
SVR with shocks	0.55	0.45	0.37	0.29	0.61	0.48	0.67	0.54	0.80	0.72	0.74	0.59	0.82	0.68
ANN univariate	0.64	0.54	0.38	0.31	0.69	0.55	0.65	0.52	0.85	0.78	0.84	0.67	0.87	0.80
ANN with shocks	1.67	1.18	0.70	0.66	1.17	1.07	1.46	1.24	1.74	1.54	1.62	1.33	1.68	1.26
LSTM univariate	1.78	1.37	1.73	1.46	2.74	2.26	3.21	2.78	3.39	3.09	3.37	3.04	2.98	2.02
LSTM with shocks	2.01	1.69	2.27	1.94	2.11	1.72	2.43	1.95	3.42	2.93	2.57	2.27	2.00	1.62
ARIMA	1.08	0.98	0.84	0.82	1.18	1.15	1.04	1.02	1.91	1.85	2.35	2.27	1.73	1.65

Source: Authors' estimates

Table 5 shows the optimal 12-month ahead forecasting model for each region. It can be seen that SVR models have the best performance for the sample regions with the exception of Region 6. However, it should be pointed out that, ANN univariate models outperformed SVR models by a margin of 0.02 only on both metrics.

Table 5. Optimal 12-month ahead forecasting model for each region

Region	RMSE	MAE
NCR	SVR with shocks	SVR with shocks
Region 3	SVR univariate	SVR univariate
Region 4-A	SVR univariate	ANN univariate and SVR with shocks
Region 6	ANN univariate	ANN univariate
Region 7	SVR with shocks	SVR with shocks
Region 10	SVR with shocks	SVR with shocks
Region 11	SVR univariate	SVR univariate

Source: Based on authors' estimates

It could be noted that the relatively bad performance of LSTM models in dynamic forecasting may be a consequence of the memory storage of LSTM, wherein it remembers long and short-term memories (Hochreiter & Schmidhuber, 1997; Gibson & Patterson, 2017). In other words, if LSTM commits a significant error in the first few forecasts, the succeeding forecasts will start to suffer bigger errors since the value that it remembers already has a significant margin of error. This phenomenon was also noted in the previous study by Almosova & Andresen, 2004. Also, this behavior of LSTM maybe due to the resolution of data (Hochreiter & Schmidhuber, 1997; Gibson & Patterson, 2017).

Meanwhile, in terms of running time for dynamic forecasting, LSTM with shocks requires the longest running time of three hours, 51 minutes (average) while SVR univariate is the fastest to complete a run at only 32 minutes (average). As SVR models manifest superior performance for dynamic forecasting (based on RMSE and MAE metrics) along with shortest run time to complete, SVR models represent the best candidate to forecast inflation of the seven regions with longer forecasting horizons.

Nonetheless, this study notes that the running time and performance of the neural networks (LSTM and ANN) can be improved if a better computer is used. In this study, the hyperparameters were set at a good running condition based on the capability of the authors' computer. It may be worthwhile to investigate the performance of the two neural networks at higher speed computer with a more sophisticated hyperparameters.

Similar to other studies, this paper also evaluates the usefulness of machine learning methods compared to traditional ARIMA models. In the one-step ahead forecast, all the SVR and ANN univariate models outperformed ARIMA models in terms of RMSE and MAE metrics (see **Table 2**). However, the ANN with shocks models are inferior to ARIMA models in terms of the RMSE scores. From the definition of RMSE (see **Equation 27**), it can be seen that it gives relatively high weight to large errors. This means that ANN with shocks have predictions with higher deviation from the actual inflation values as compared to ARIMA. Thus, if all the individual errors are weighted equally, ANN with shocks outperformed ARIMA in all the regions, which is evident on the MAE scores. For LSTM, the univariate models outperformed ARIMA models on six regions while LSTM with shocks outperformed ARIMA on just three regions (see **Table 2**).

In the 12-month ahead forecasts, all SVR and ANN univariate models consistently outperformed ARIMA models (see **Table 4**). The ANN with shocks models outperformed ARIMA on five regions but are inferior for NCR and Region 6. Meanwhile, all the LSTM models are inferior to ARIMA for all the regions despite its sophisticated algorithm.

All the optimal machine learning models for each region-identified based on RMSE and MAE scores in the one-month ahead forecasting exercise-outperformed ARIMA with a difference of at least 0.18 for RMSE and 0.17 for MAE (see **Table 6**). The same goes with the optimal machine learning models for the 12-month ahead forecast with a higher difference of at least 0.39 for RMSE and 0.33 for MAE (see **Table 7**). The difference was computed by subtracting from RMSE/MAE of ARIMA models the RMSE/MAE of the optimal machine learning models. Hence, a positive value means the optimal machine learning model outperformed ARIMA.

Table 6. RMSE and MAE of the optimal machine learning models vis-à-vis ARIMA for one-month ahead forecasting

Region	Optimal forecasting model		Difference with ARIMA ($ARIMA_{RMSE/MAE} - Optimal\ model_{RMSE/MAE}$)	
	RMSE	MAE	RMSE	MAE
NCR	LSTM univariate	SVR with shocks	0.21	0.21
Region 3	SVR univariate	SVR univariate & with shocks	0.33	0.25
Region 4-A	ANN univariate	ANN univariate	0.24	0.20
Region 6	SVR univariate	SVR univariate	0.32	0.22
Region 7	ANN univariate	ANN univariate	0.18	0.16
Region 10	LSTM univariate	SVR univariate, SVR with shocks & LSTM univariate	0.18	0.17
Region 11	ANN univariate	SVR univariate & SVR with shocks	0.35	0.30

Source: Based on authors' estimates

Table 7. RMSE and MAE of the optimal machine learning models vis-à-vis ARIMA for 12-month ahead forecasting

Region	Optimal forecasting model		Difference with ARIMA ($ARIMA_{RMSE/MAE} - Optimal\ model_{RMSE/MAE}$)	
	RMSE	MAE	RMSE	MAE
NCR	SVR with shocks	SVR with shocks	0.53	0.53
Region 3	SVR univariate	SVR univariate	0.48	0.54
Region 4-A	SVR univariate	ANN univariate and SVR with shocks	0.58	0.33
Region 6	ANN univariate	ANN univariate	0.39	0.50
Region 7	SVR with shocks	SVR with shocks	1.11	1.13
Region 10	SVR with shocks	SVR with shocks	1.61	1.68
Region 11	SVR univariate	SVR univariate	0.92	0.98

Source: Based on authors' estimates

6. Conclusions and Recommendations

Six models were developed to forecast inflation for each of the representative regions identified in this study. Two forecasting activities were done: (1) one-month ahead static forecasting wherein 10 percent of the data points were used as a test set; and (2) 12-month ahead dynamic forecasting wherein the whole 2019 inflation data were used as a test set.

In generating one-month ahead forecasts, all the models displayed relatively good forecasting capability. The ANN univariate models had the best average RMSE while the SVR with shock models had the best MAE. For the dynamic forecasting activity, SVR with shock models registered the best performance for both metrics. Furthermore, all the optimal machine learning models identified for each region outperformed ARIMA on both performance metrics. Hence, it is recommended that the optimal machine learning models for each region be used for regional forecasting.

Although it was observed that LSTM has the poorest performance in dynamic forecasting, further studies can be conducted in the future when data resolution is increased. This is because from previous studies (Siami-Namini et al., 2018; Thakur et al., 2016; Almosova & Andresen, 2004) LSTM performs better when significant data resolution is achieved. With regards to the running time, a better computer capacity is recommended to further increase the hyperparameters of the neural networks since in this study, the hyperparameters for ANN and LSTM were set to a running condition of the computer being used.

Further work in this subject includes creating inflation forecasting models for the remaining regions in the country using the ANN and SVR, determining optimal models based on RMSE and MAE and aggregating the regional forecasts in order to generate a nowcast for the nationwide inflation rate.

References

Adebiyi, A.A., Adewumi, A.O. and Ayo, C.K., 2014. Comparison of ARIMA and artificial neural networks models for stock price prediction. *Journal of Applied Mathematics*, *2014*.

Almosova A., Andresen N. Nonlinear inflation forecasting with recurrent neural networks, European Central Bank. Retrieved from: https://www.ecb.europa.eu/pub/conferences/shared/pdf/20190923_inflation_conference/L2_Almosova.pdf

Baciu, I., 2015. Stochastic Models for Forecasting Inflation Rate. Empirical Evidence from Romania. *Procedia Economics and Finance*, 20, pp.44-52.

Binner, J., Bissoondeeal, R., Elger, T., Gazely, A. and Mullineux, A., 2005. A comparison of linear forecasting models and neural networks: an application to Euro inflation and Euro Divisia. *Applied Economics*, 37(6), pp.665-680.

Chakraborty, C. and Joseph, A., 2017. Machine learning at central banks.

Cortez, P., Rocha, M. and Neves, J., 2001. A meta-genetic algorithm for time series forecasting.

de Figueiredo, R., 1980. Implications and applications of Kolmogorov's superposition theorem. *IEEE Transactions on Automatic Control*, *25*(6), pp.1227-1231.

Esquivel-Monge, M., 2009. Performance of Artificial Neural Networks in Forecasting Costa Rican Inflation.

Goodfellow, I., Bengio, Y. and Courville, A., 2016. *Deep learning*. MIT press.

Hurtado, J.L.C. and Cortes-Fregoso, J.H., 2013, March. Forecasting Mexican inflation using neural networks. In *CONIELECOMP 2013, 23rd International Conference on Electronics, Communications and Computing* (pp. 32-35). IEEE.

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

Kingma, D.P. and Ba, J., 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lee, S.I. and Yoo, S.J., 2017. A deep efficient frontier method for optimal investments. *Department of Computer Engineering, Sejong University, Seoul*, *5006*.

Ma, X., Li, X., Zhang, Q., Tang, K., Liang, Z., Xie, W. and Zhu, Z., 2018. A survey on cooperative co-evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, *23*(3), pp.421-441.

Moser, G., Rumler, F. and Scharler, J., 2007. Forecasting Austrian inflation. *Economic Modelling*, 24(3), pp.470-480.

Okasha, M.K. and Yassen, A.A., 2013. The application of artificial neural networks in forecasting economic time series. *The International Journal of Statistics and Analysis (IJSA)*, *3*(2), pp.123-143. [6] M. E. Monge, Performance of artificial neural networks in forecasting costarican inflation, Division Economica, Banco Central de Costa Rica (2009).

Patterson, J. and Gibson, A., 2017. *Deep learning: A practitioner's approach*. " O'Reilly Media,Inc.".

Prechelt, L., 1998. Automatic early stopping using cross validation: quantifying the criteria. *Neural Networks*, *11*(4), pp.761-767.

Sari, N.R., Mahmudy, W.F. and Wibawa, A.P., 2016. Backpropagation on neural network method for inflation rate forecasting in Indonesia. *Int. J. Soft Comput. Its Appl*.

Shen, S., Jiang, H. and Zhang, T., 2012. Stock market forecasting using machine learning algorithms. *Department of Electrical Engineering, Stanford University, Stanford, CA*, pp.1-5.

Siami-Namini, S., Tavakoli, N., Adl Khanghah, M., Mirza Soltani, F. and Siami Namin, A., 2020. An autoencoder-based deep learning approach for clustering time series data. *SN Applied Sciences*, 2(5).

Smola, A.J. and Schölkopf, B., 2004. A tutorial on support vector regression. *Statistics and computing*, *14*(3), pp.199-222.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), pp.1929-1958.

Thakur, G.S.M., Bhattacharyya, R. and Mondal, S.S., 2016. Artificial neural network-based model for forecasting of inflation in India. *Fuzzy Information and Engineering*, *8*(1), pp.87-100.

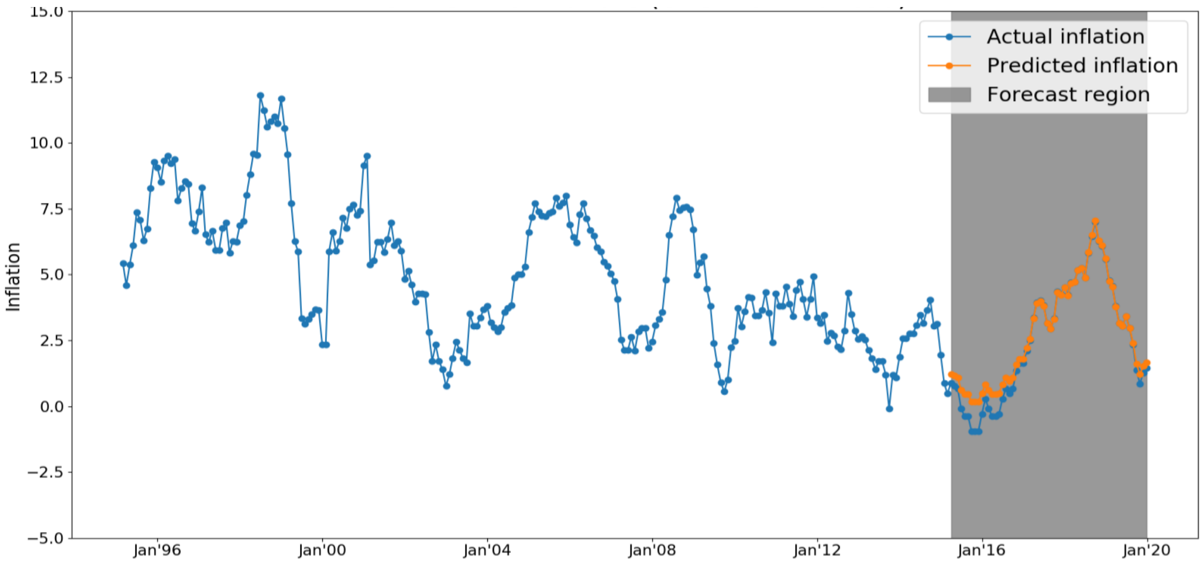
Appendix

I. One-month ahead forecasts

A. NCR

Figure 1.1. SVR models for NCR

(i) Univariate



(ii) Multivariate

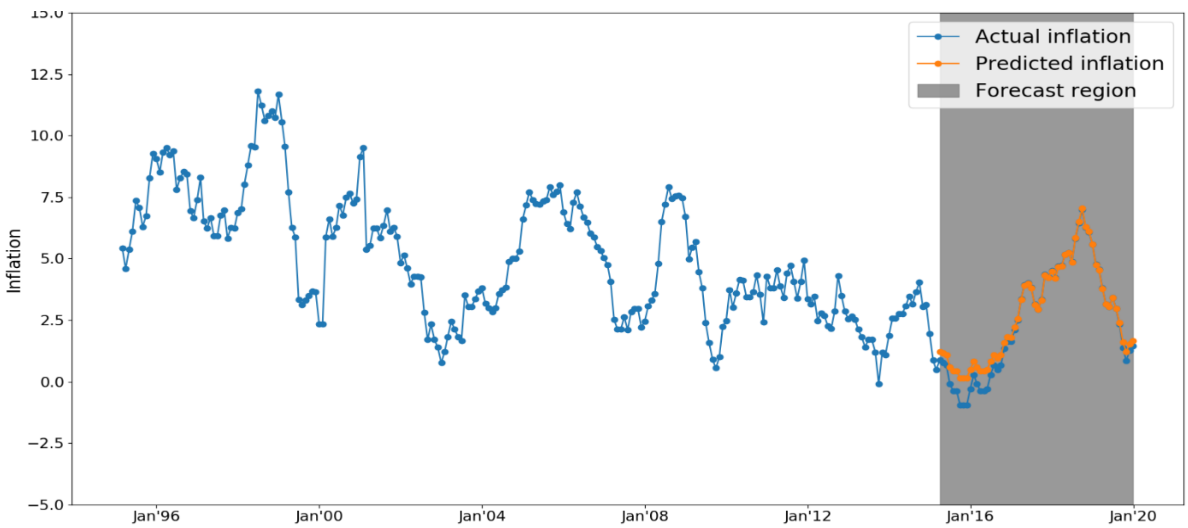
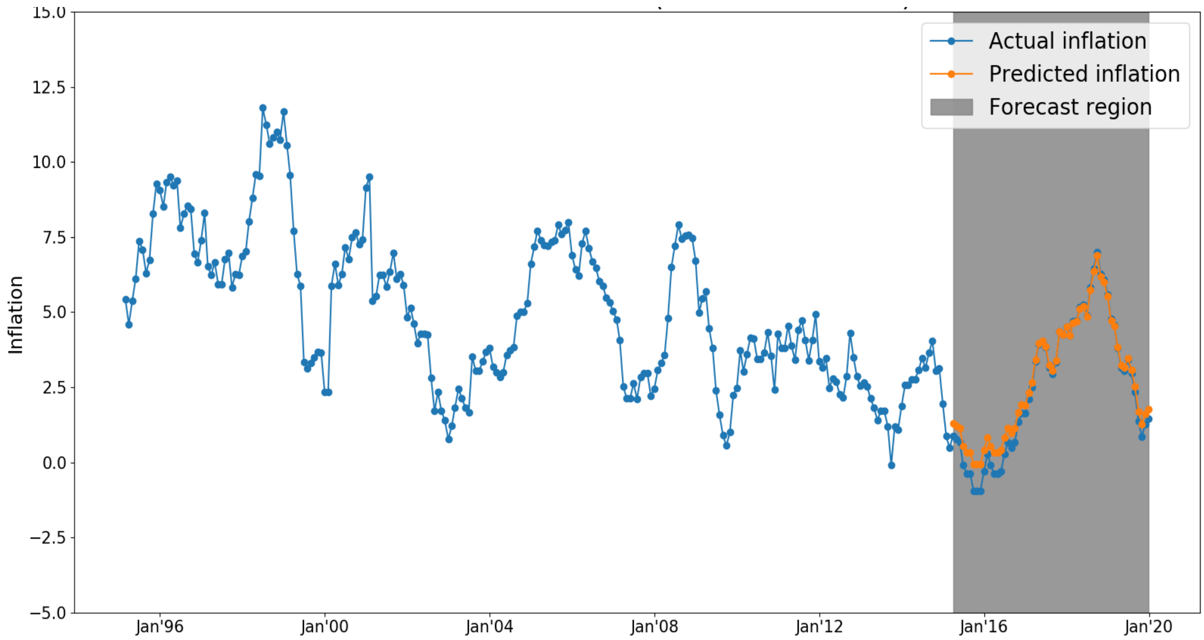


Figure 1.2. ANN models for NCR

(i) Univariate



(ii) Multivariate

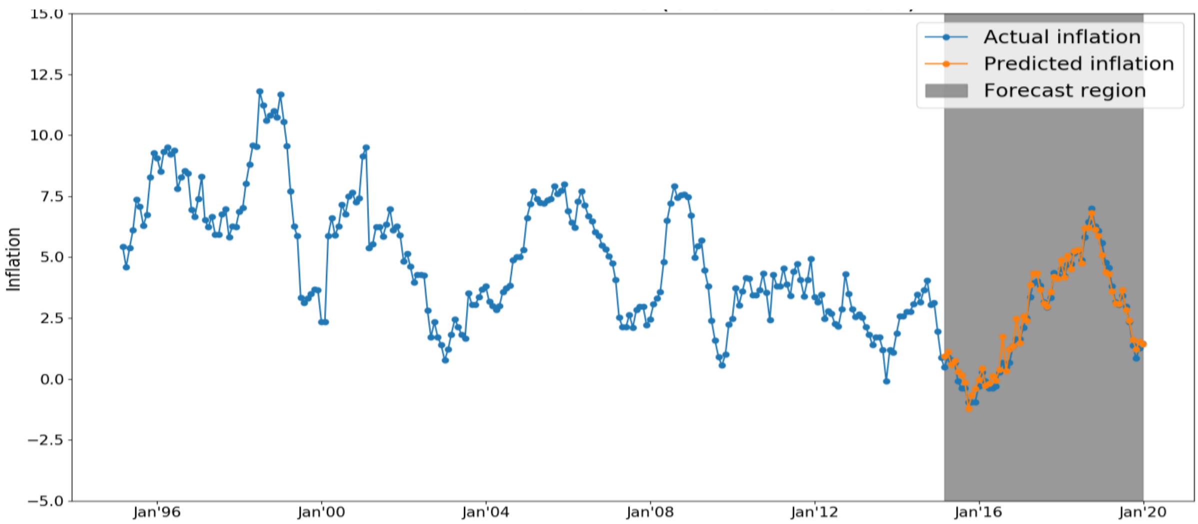
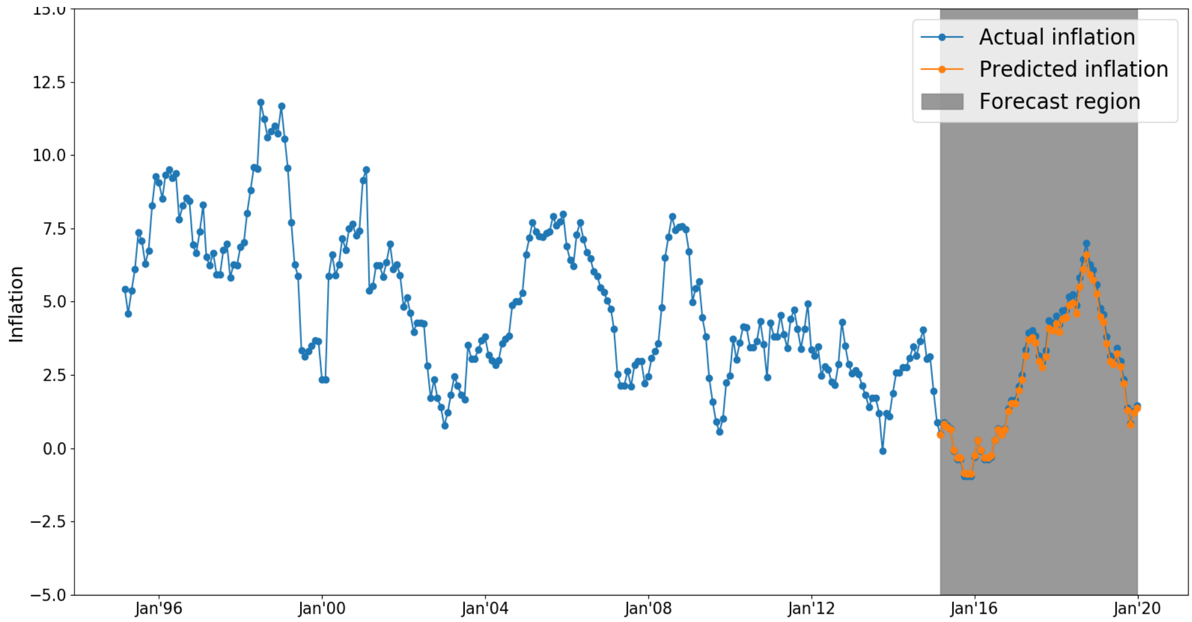


Figure 1.3. LSTM models for NCR

(i) Univariate



(ii) Multivariate

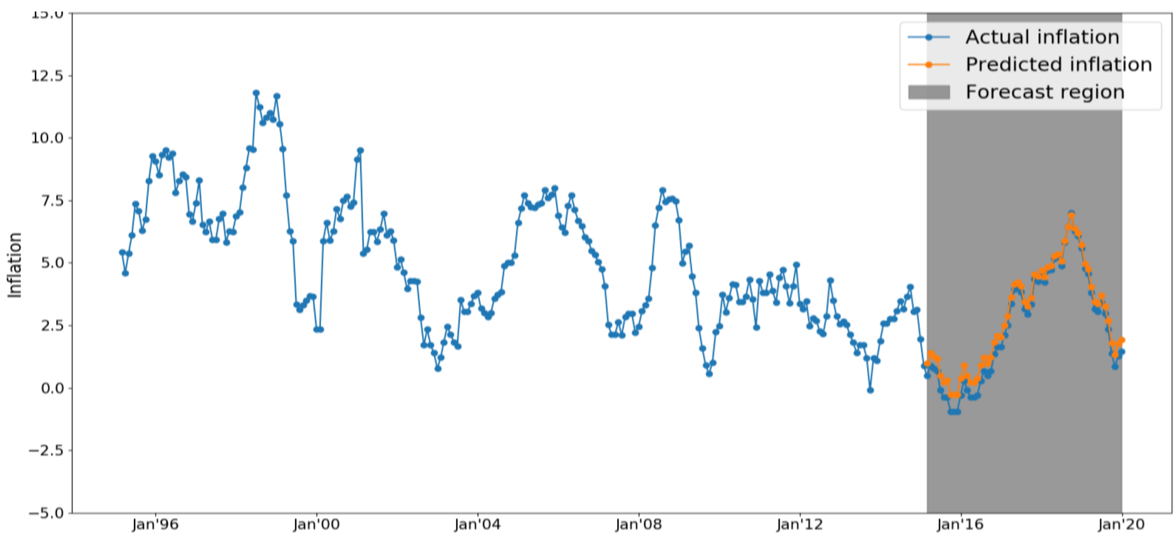
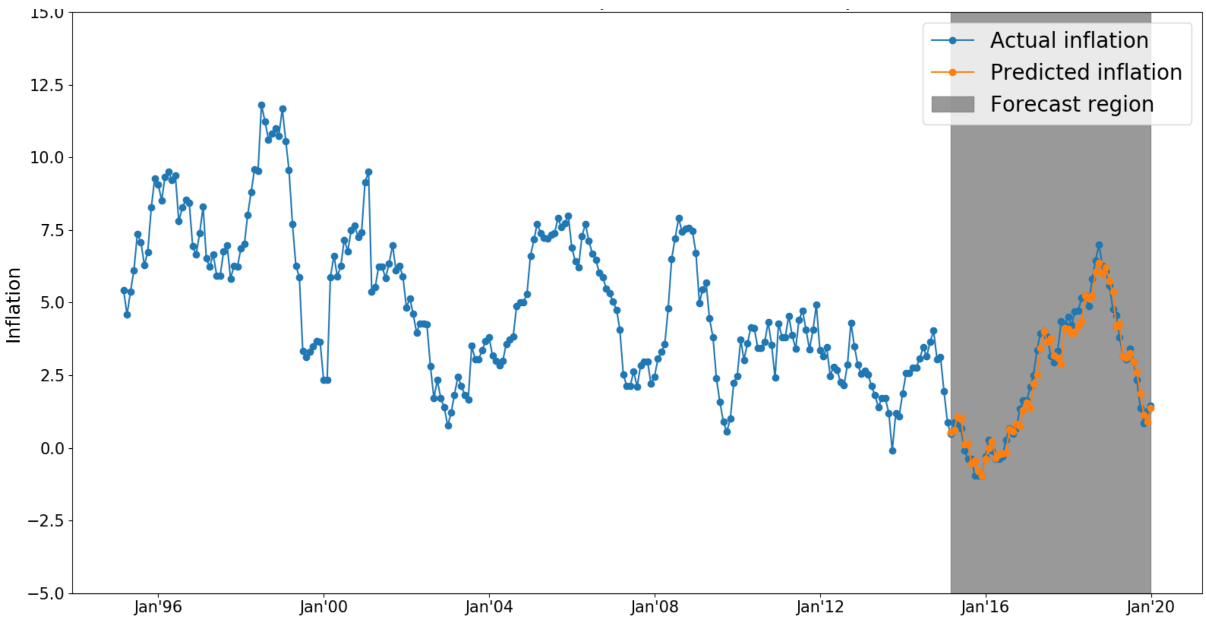


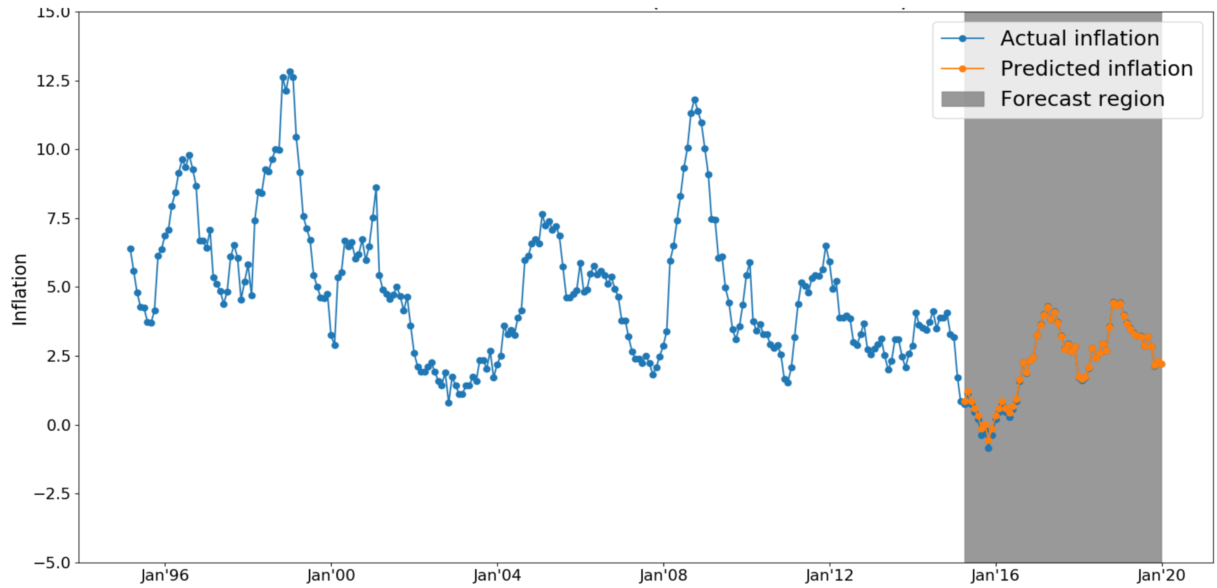
Figure 1.4. ARIMA model for NCR



B. Region 3

Figure 1.5. SVR models for Region 3

(i) Univariate



(ii) Multivariate

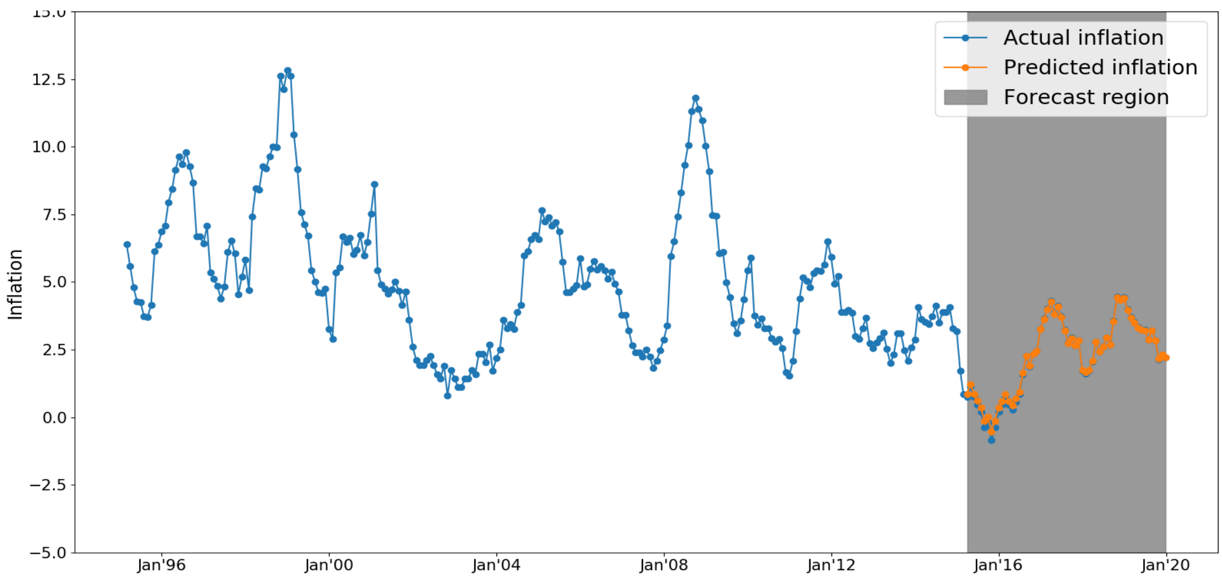
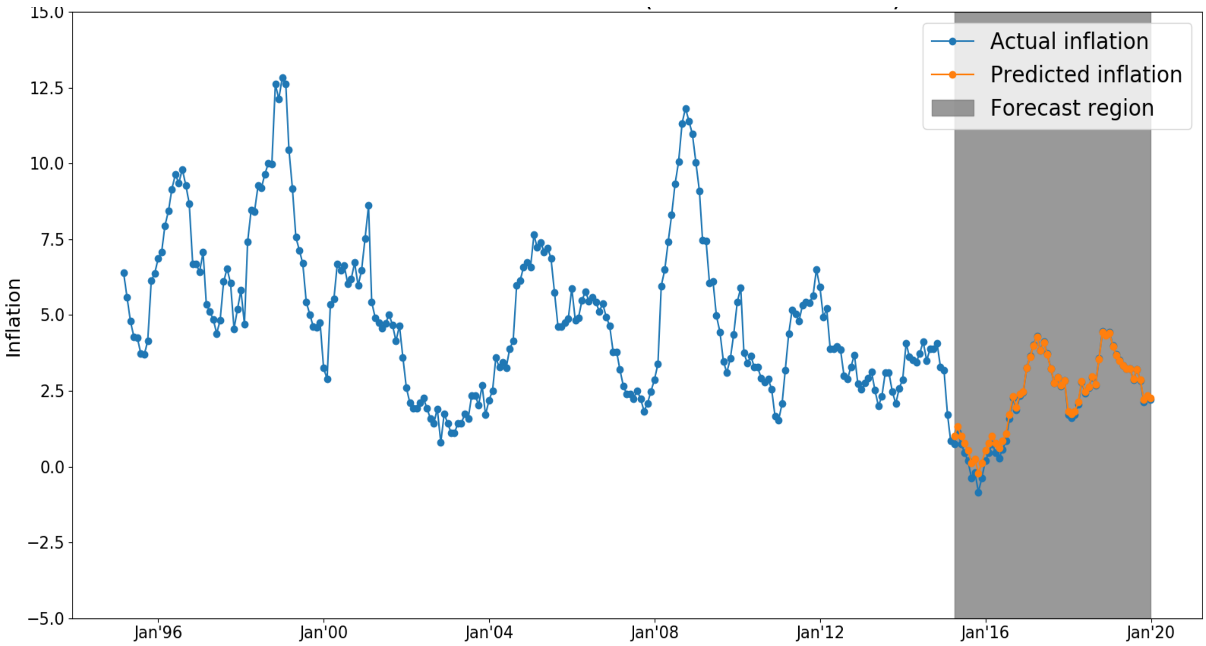


Figure 1.6. ANN models for Region 3

(i) Univariate



(ii) Multivariate

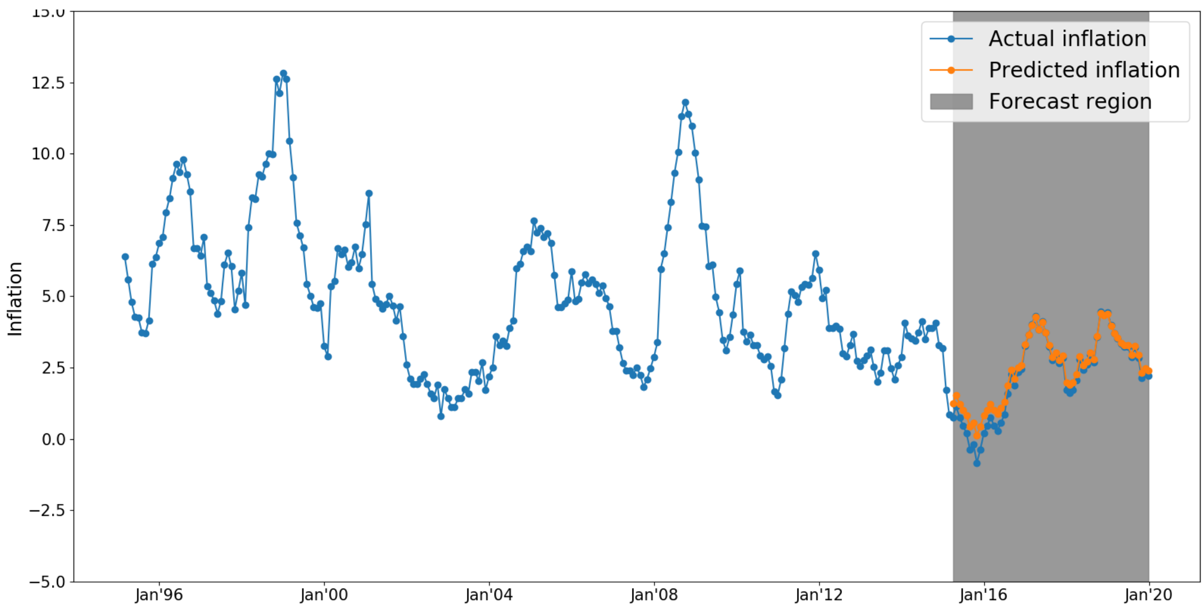
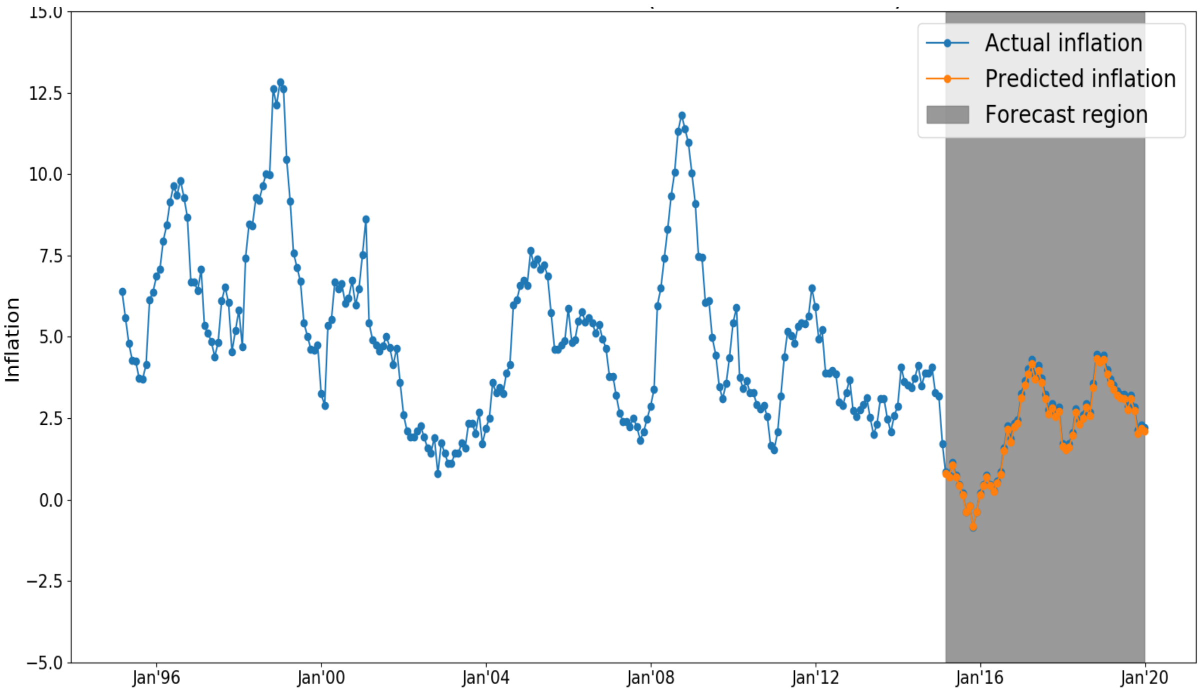


Figure 1.7. LSTM models for Region 3

(i) Univariate



(ii) Multivariate

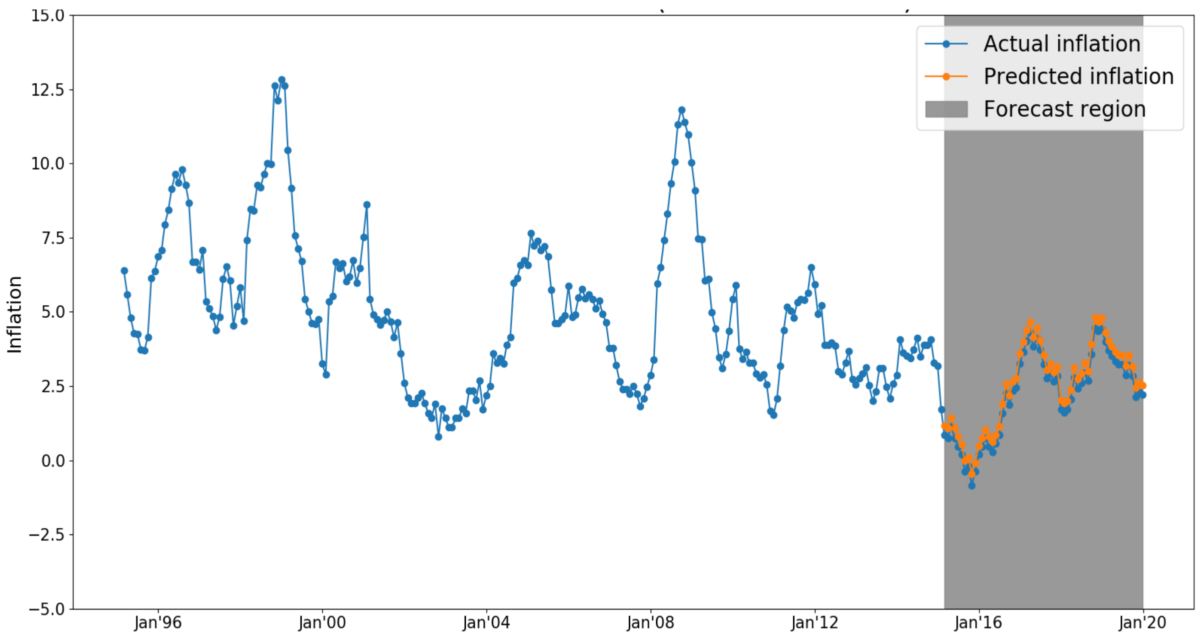
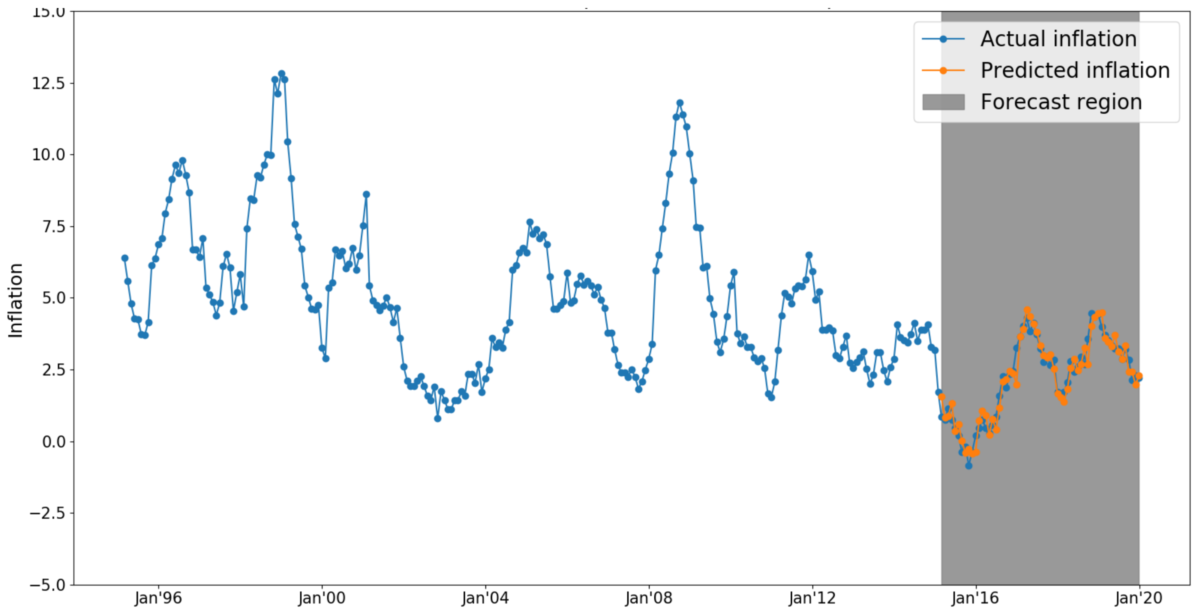


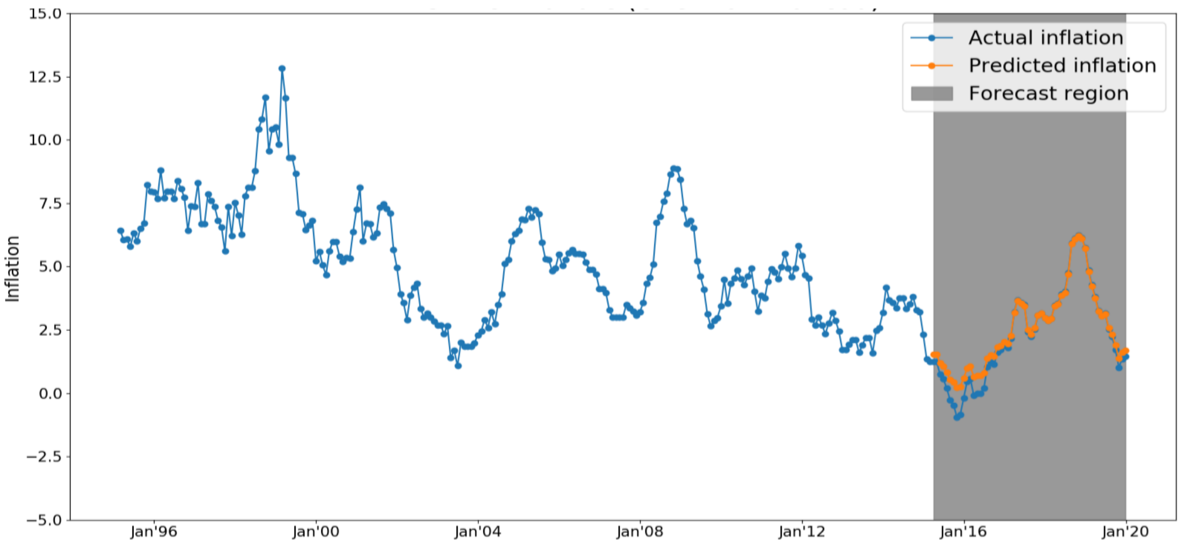
Figure 1.8. ARIMA model for Region 3



C. Region 4-A

Figure 1.9. SVR models for Region 4-A

(i) Univariate



(ii) Multivariate

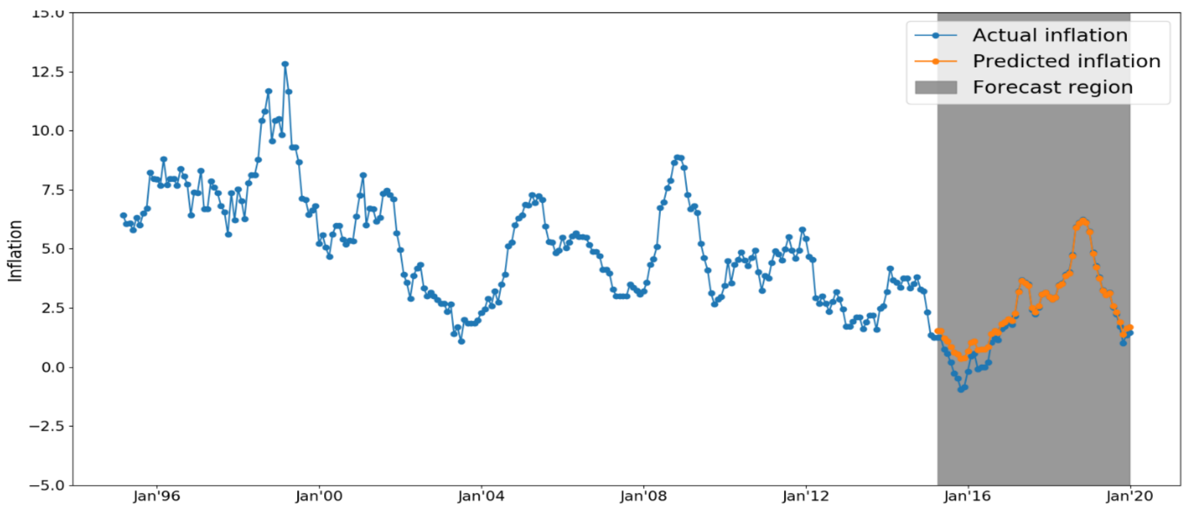
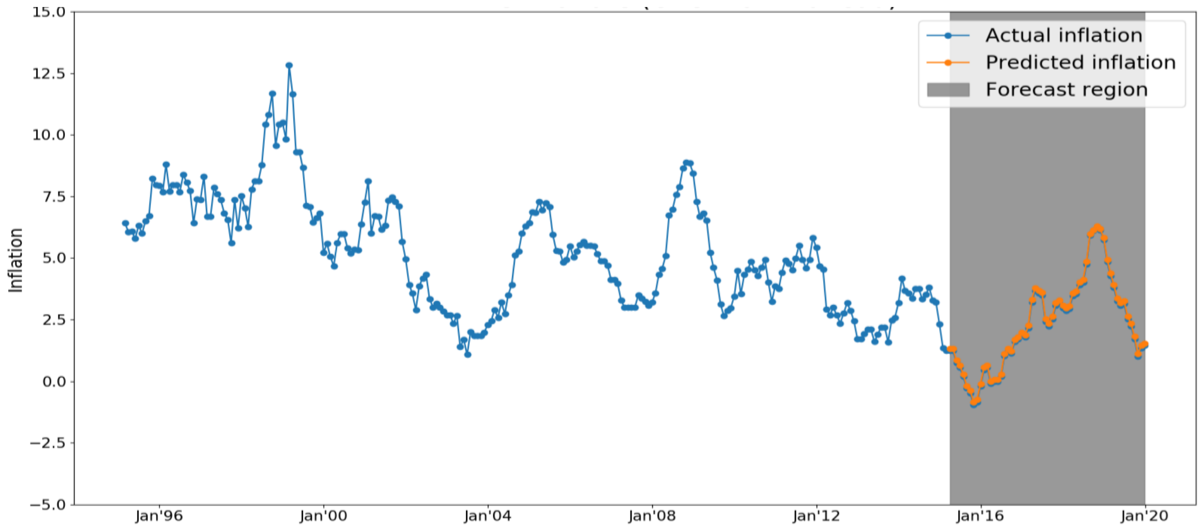


Figure 1.10. ANN models for Region 4-A

(i) Univariate



(ii) Multivariate

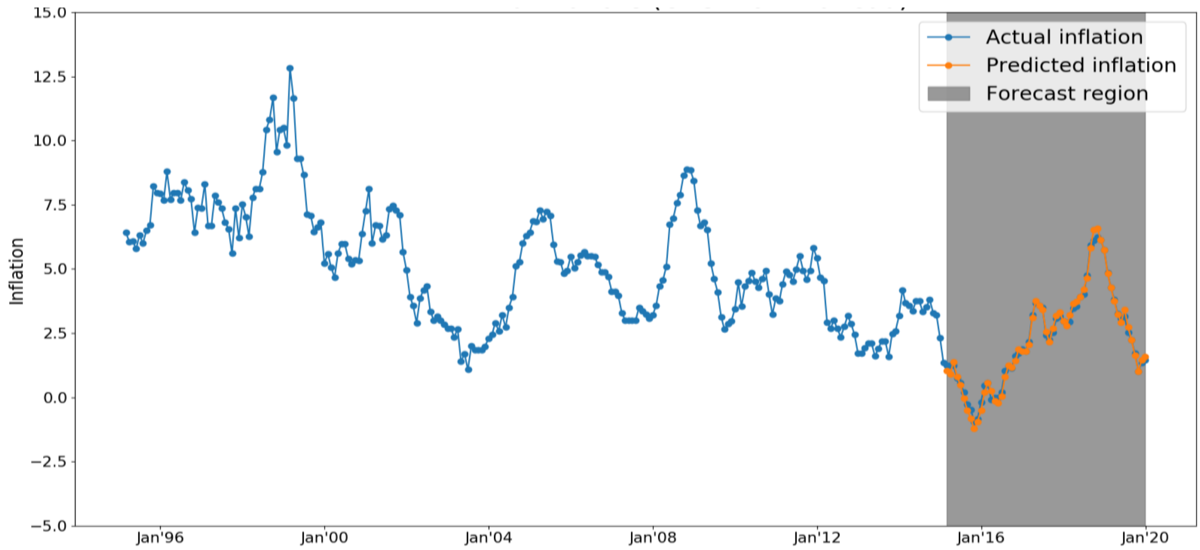
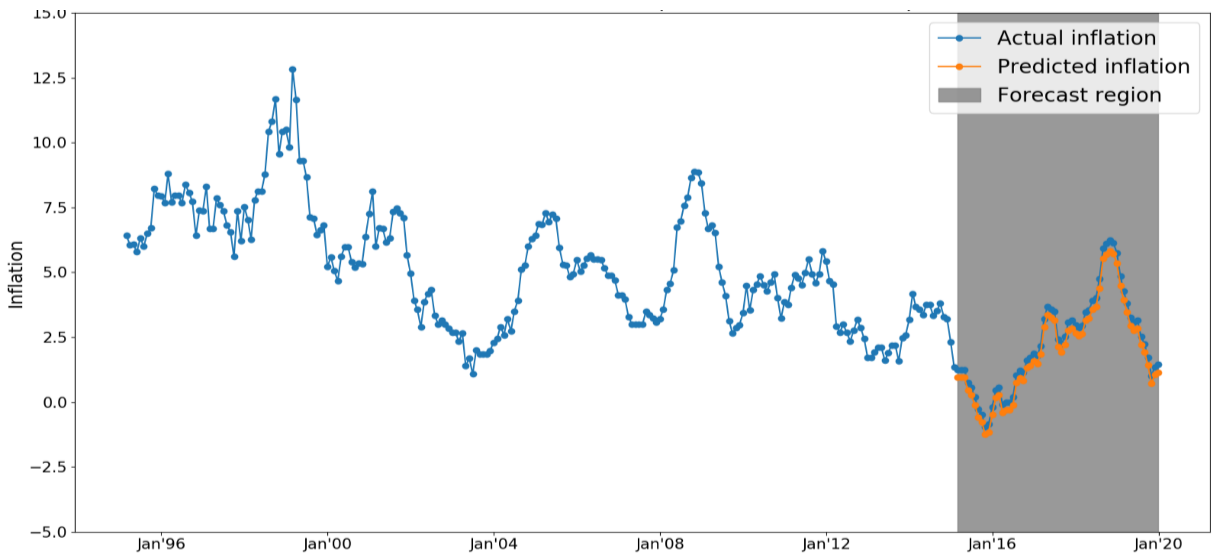


Figure 1.11. LSTM models for Region 4-A

(i) Univariate



(ii) Multivariate

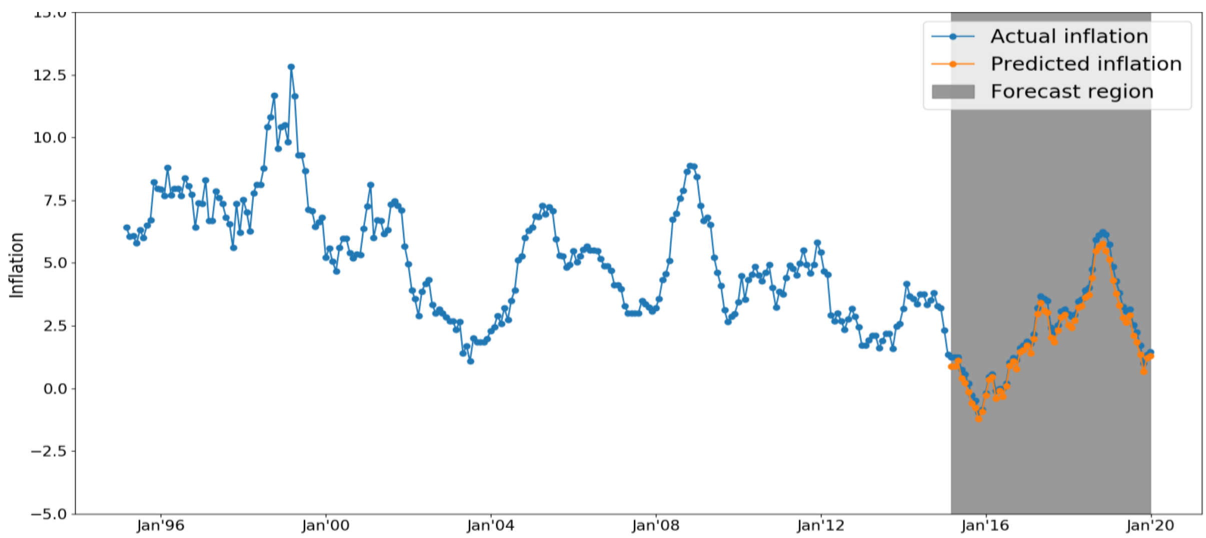
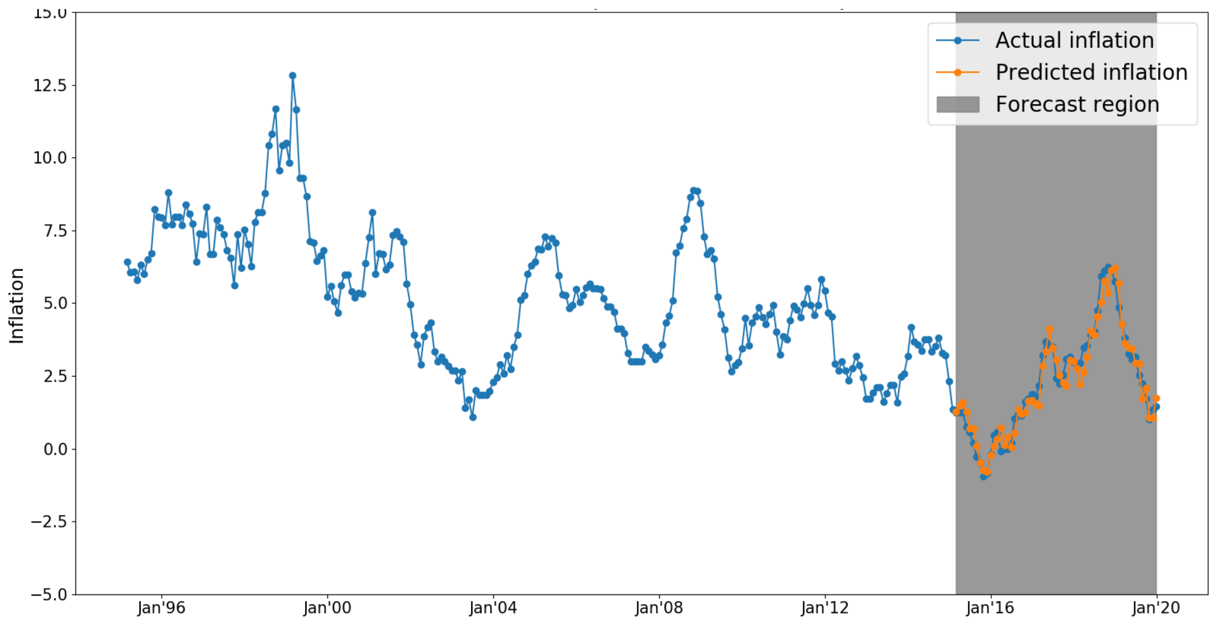


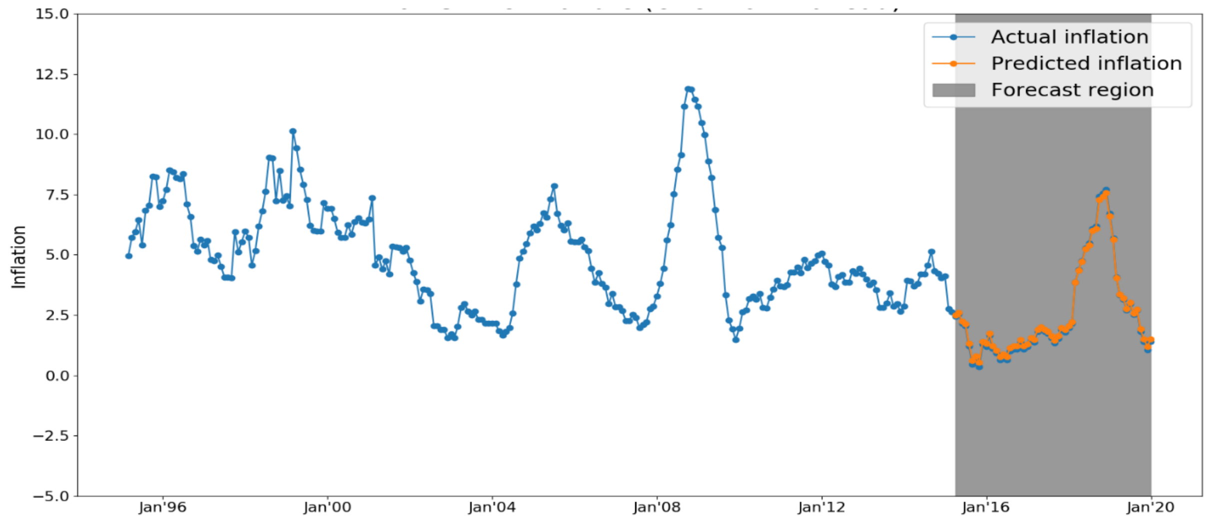
Figure 1.12. ARIMA model for Region 4-A



D. Region 6

Figure 1.13. SVR models for Region 6

(i) Univariate



(ii) Multivariate

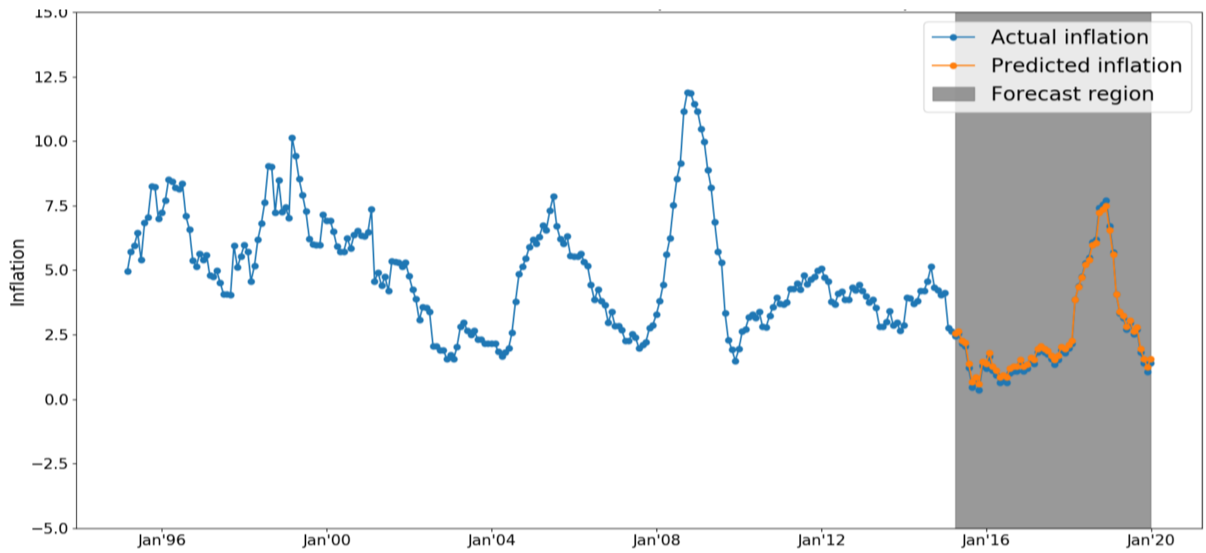
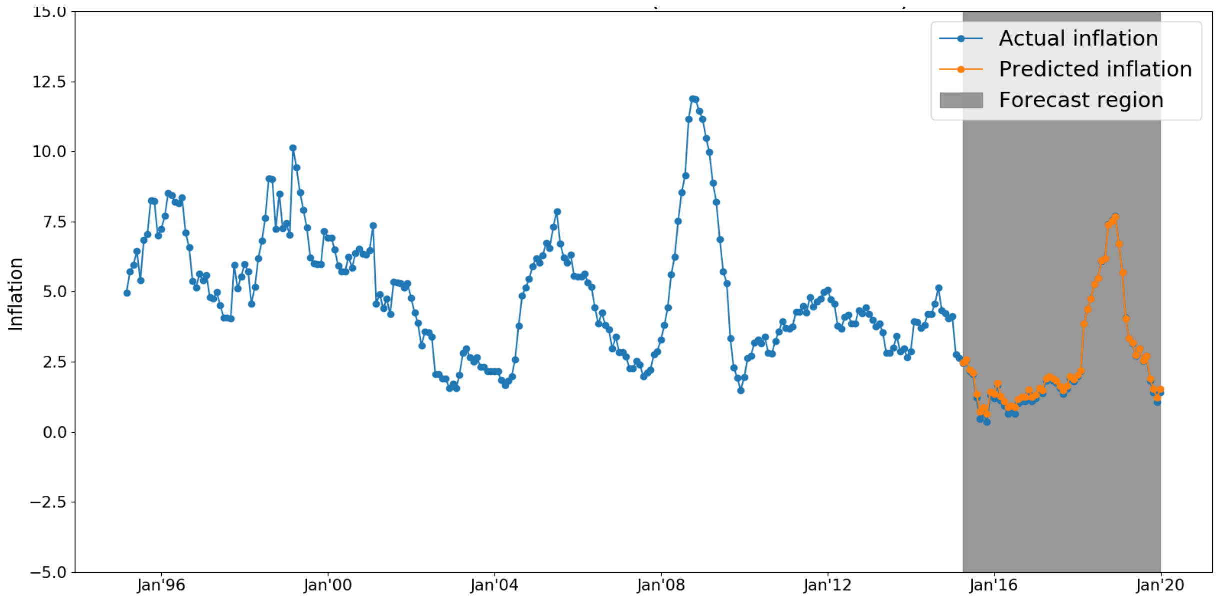


Figure 1.14. ANN models for Region 6

(i) Univariate



(ii) Multivariate

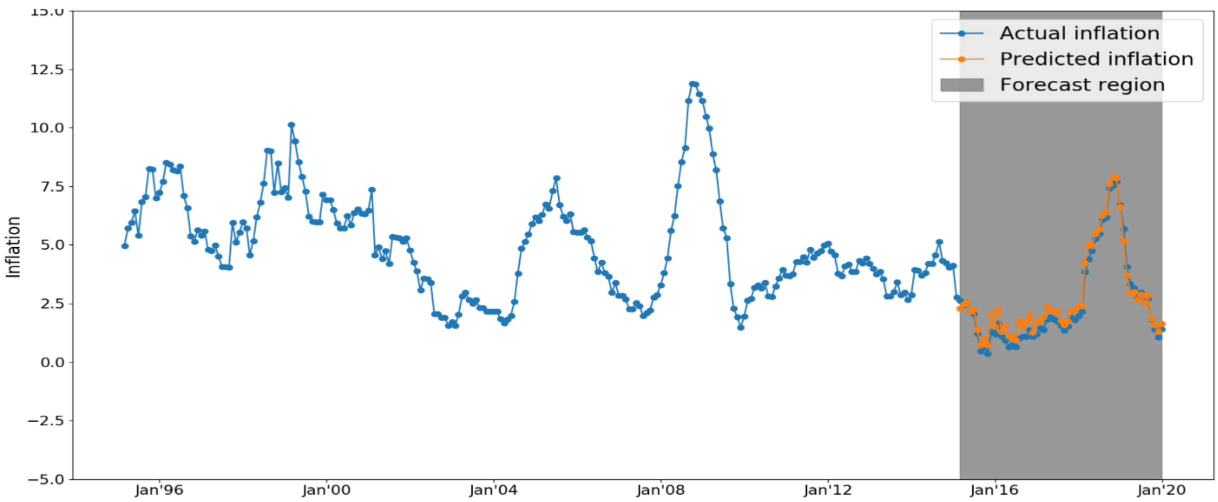
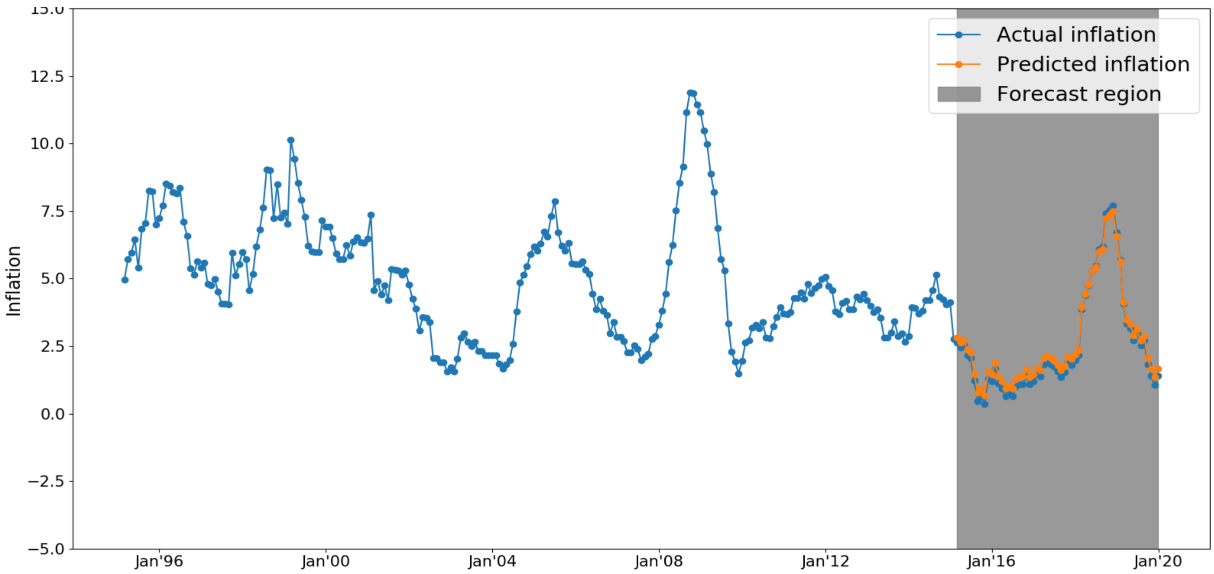


Figure 1.15. LSTM models for Region 6

(i) Univariate



(ii) Multivariate

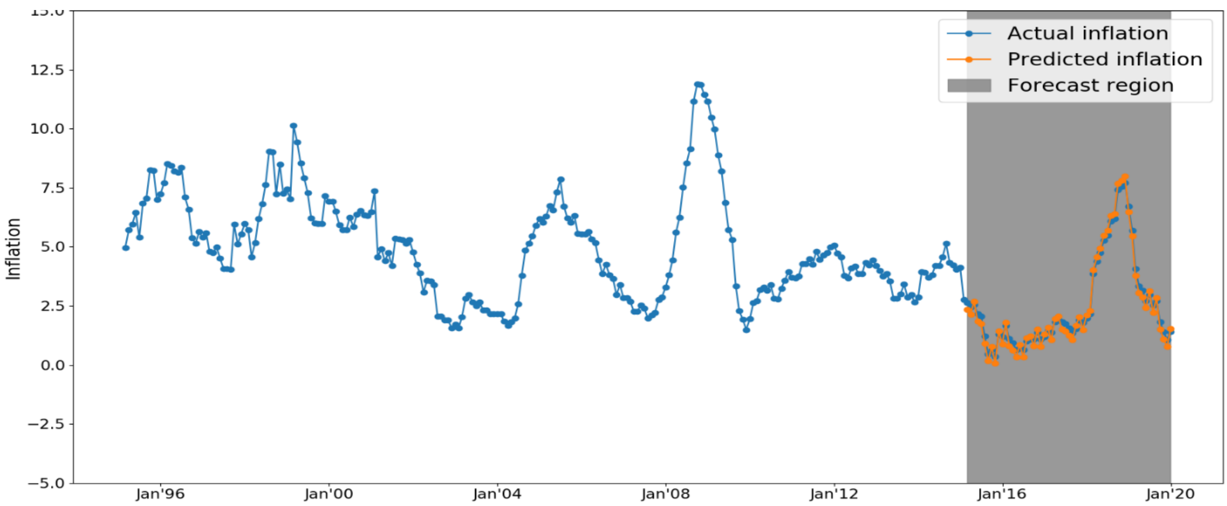
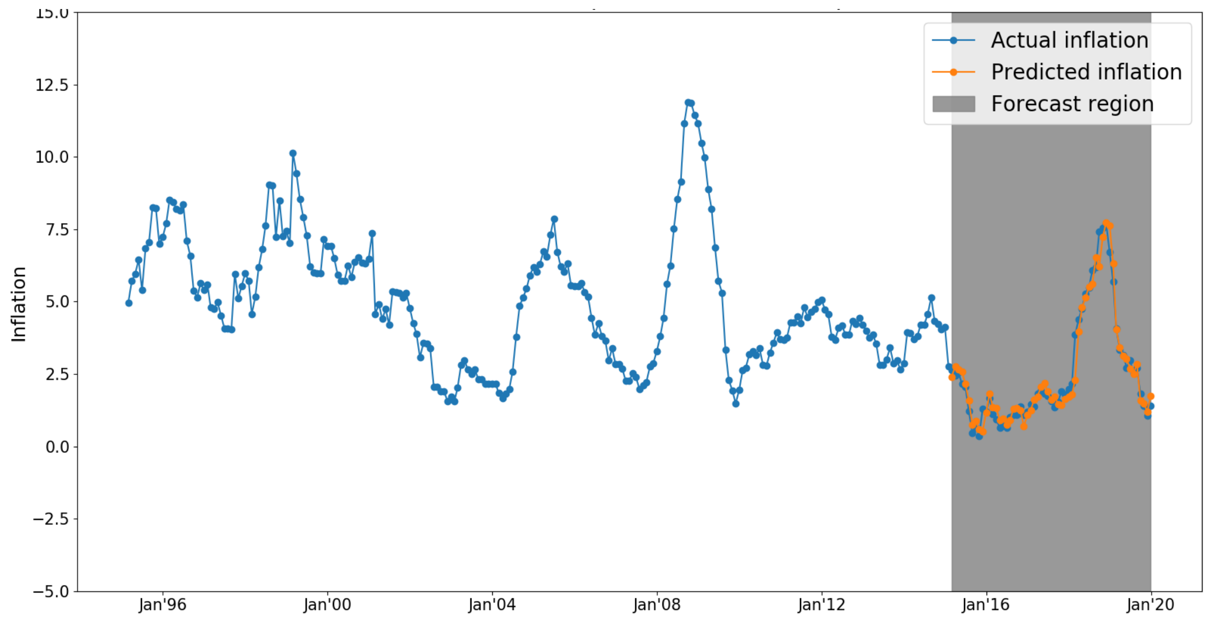


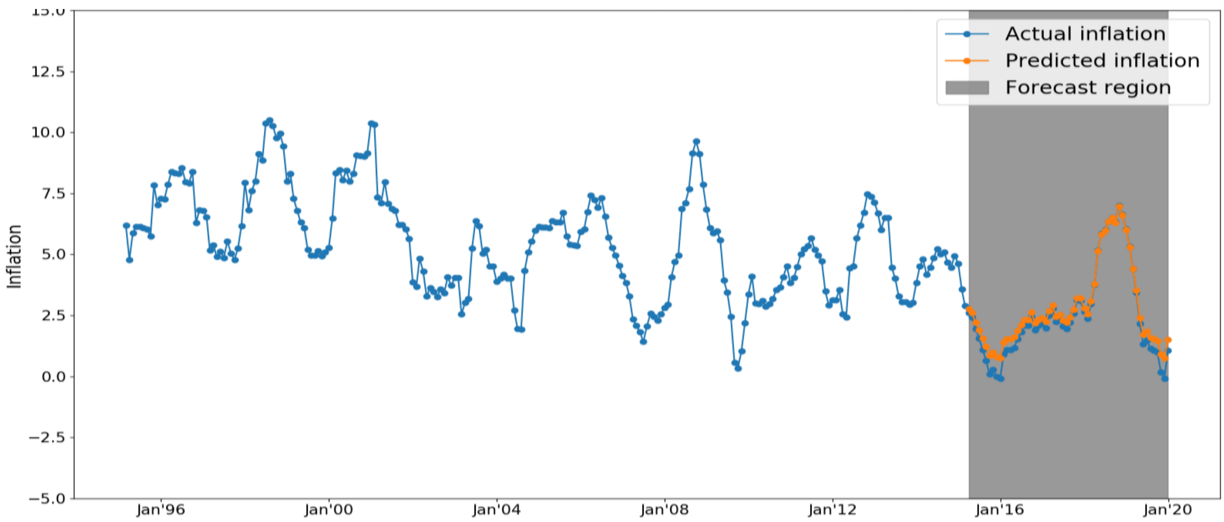
Figure 1.16. ARIMA model for Region 6



E. Region 7

Figure 1.17. SVR models for Region 7

(i) Univariate



(ii) Multivariate

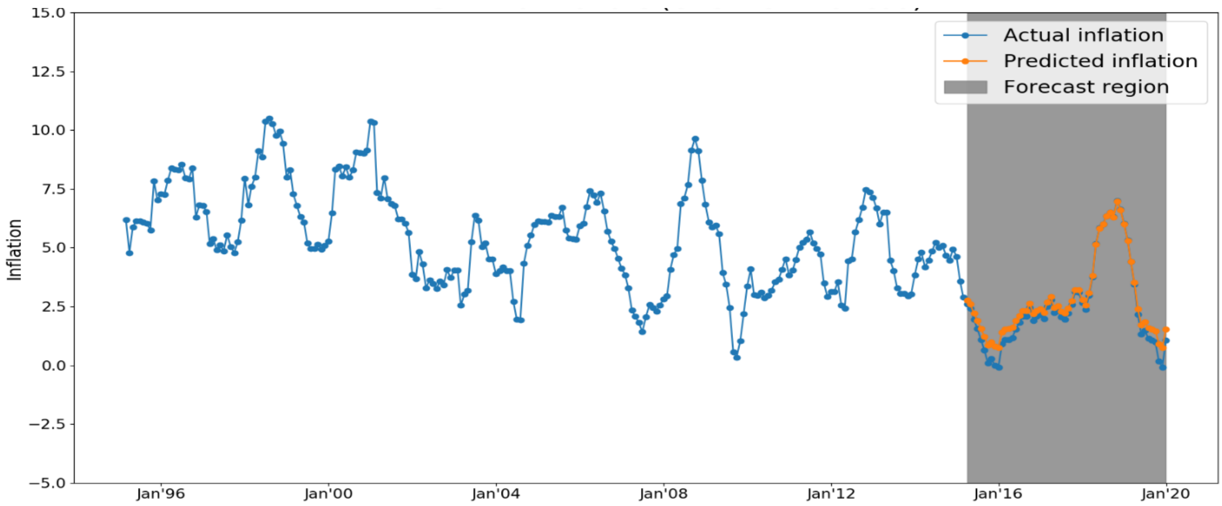
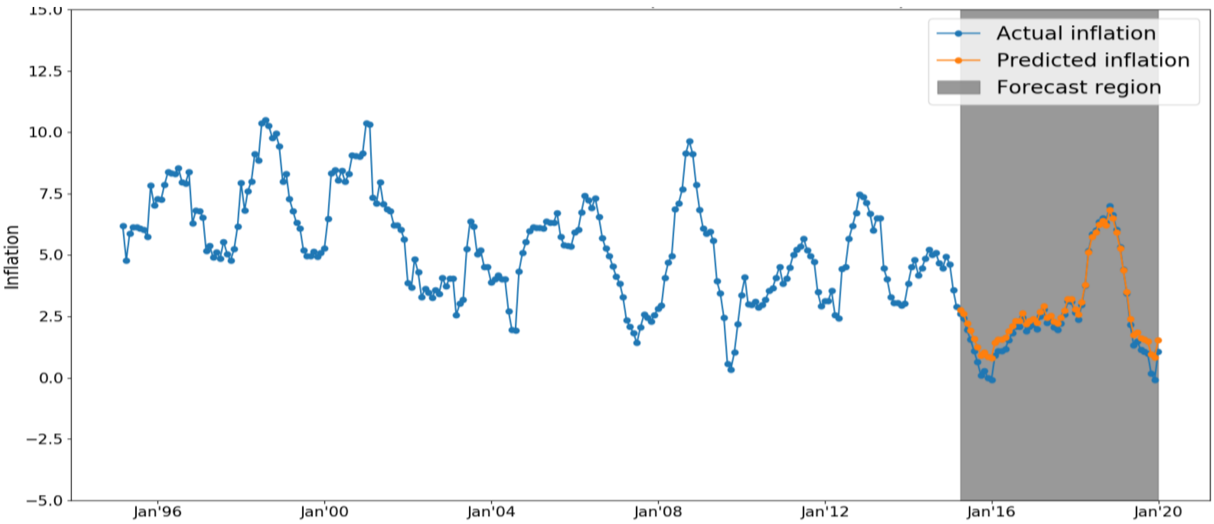


Figure 1.18. ANN models for Region 7

(i) Univariate



(ii) Multivariate

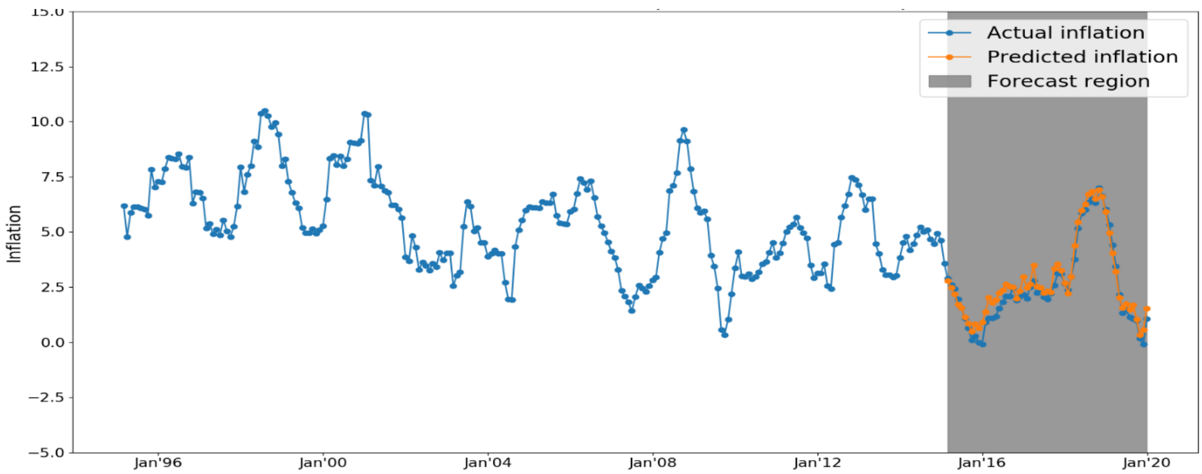
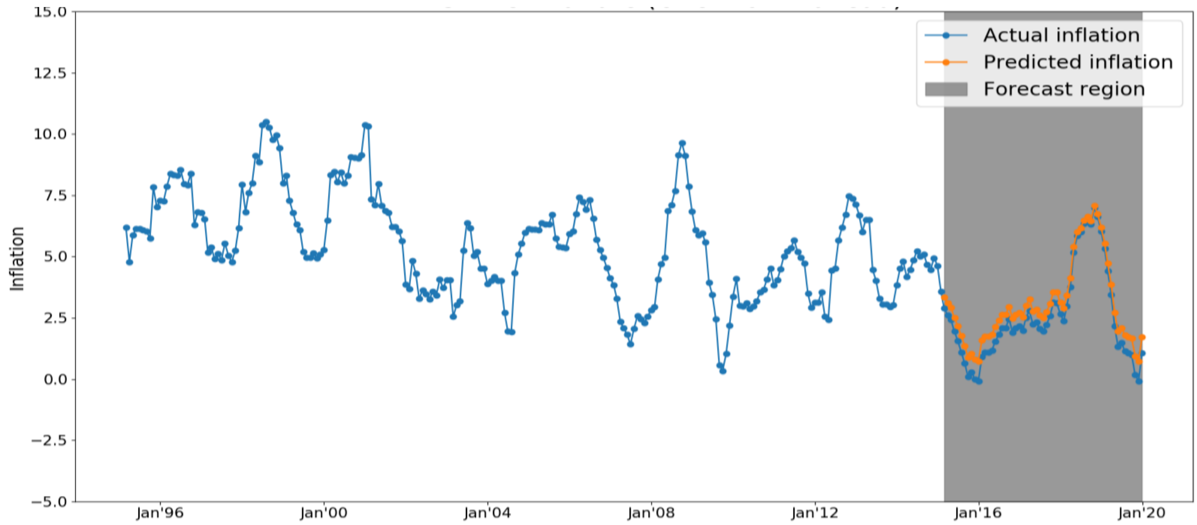


Figure 1.19. LSTM models for Region 7

(i) Univariate



(ii) Multivariate

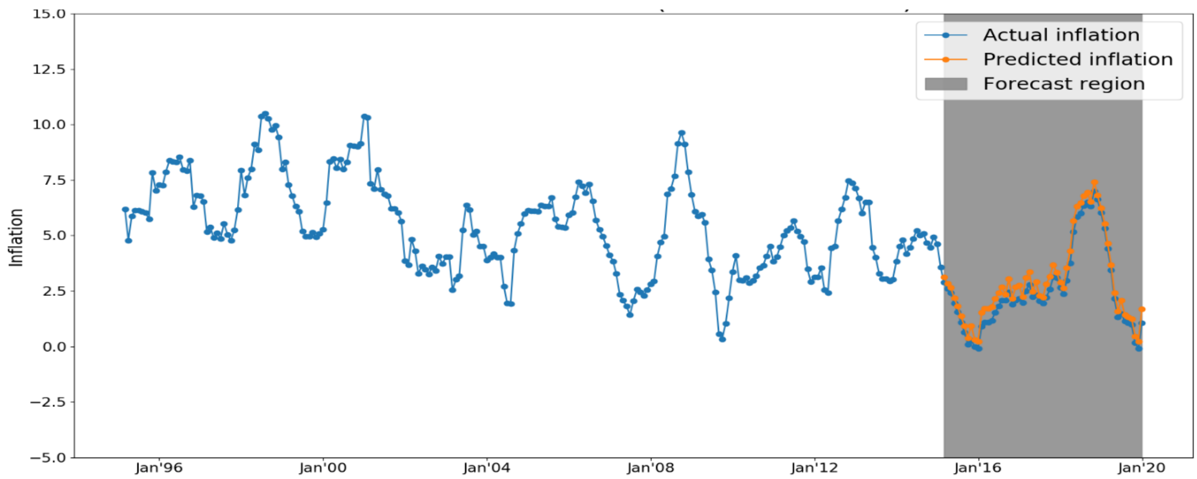
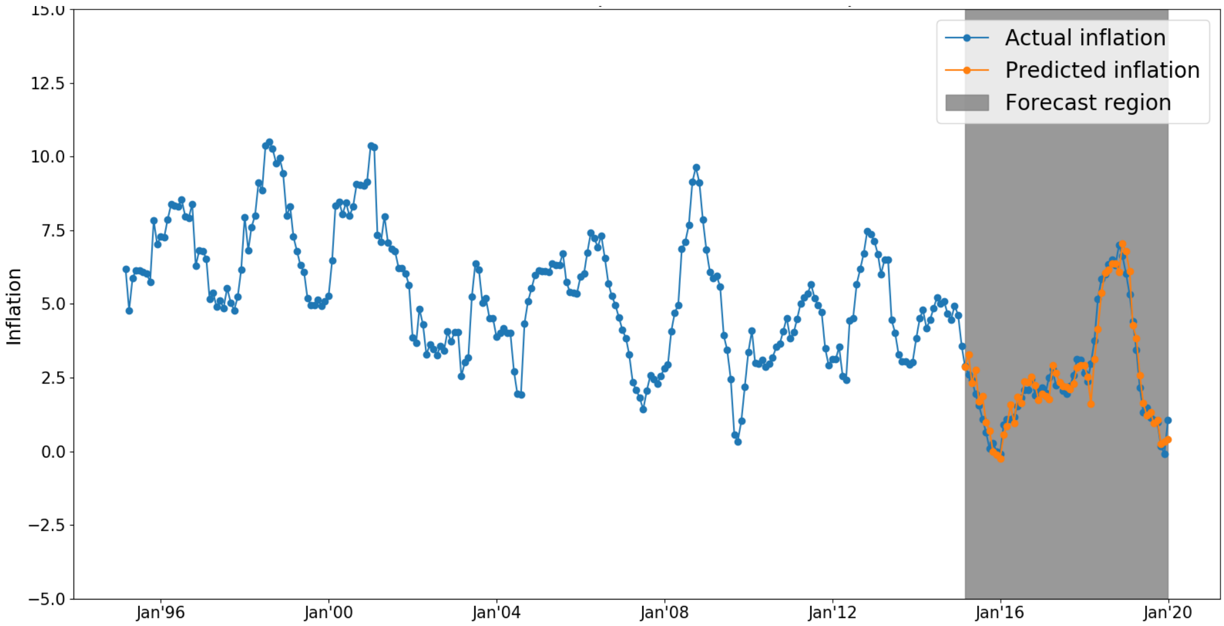


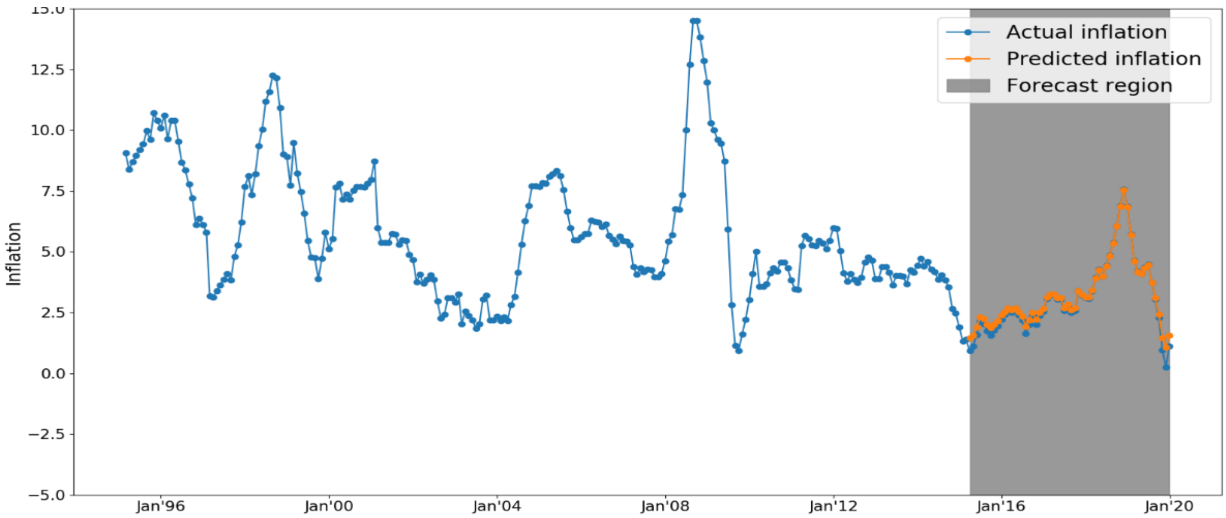
Figure 1.20. ARIMA model for Region 7



F. Region 10

Figure 1.21. SVR models for Region 10

(i) Univariate



(ii) Multivariate

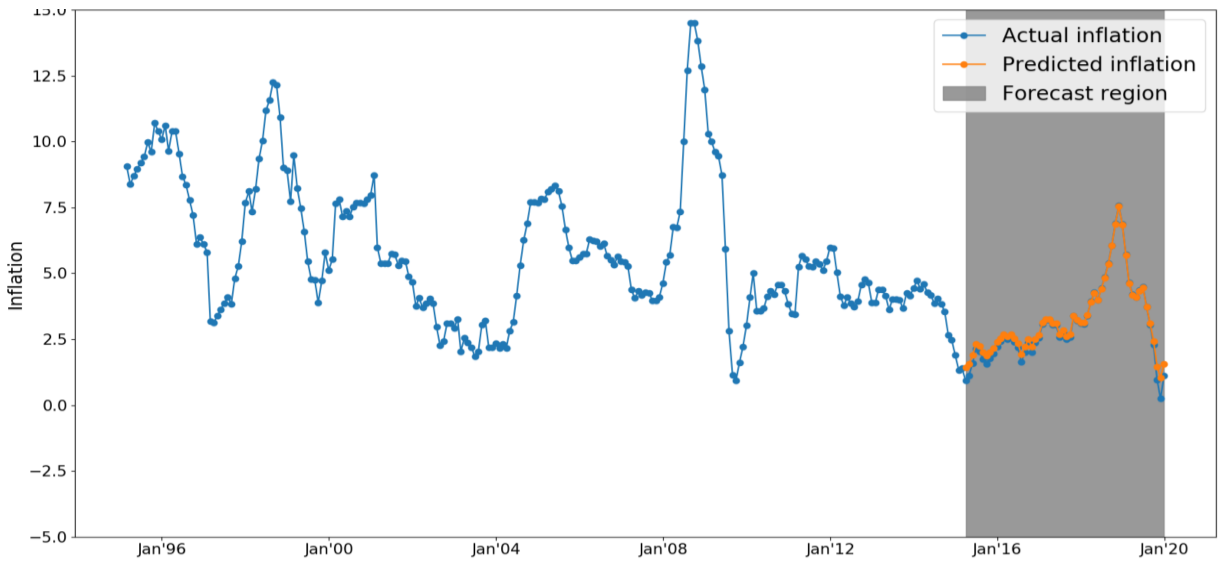
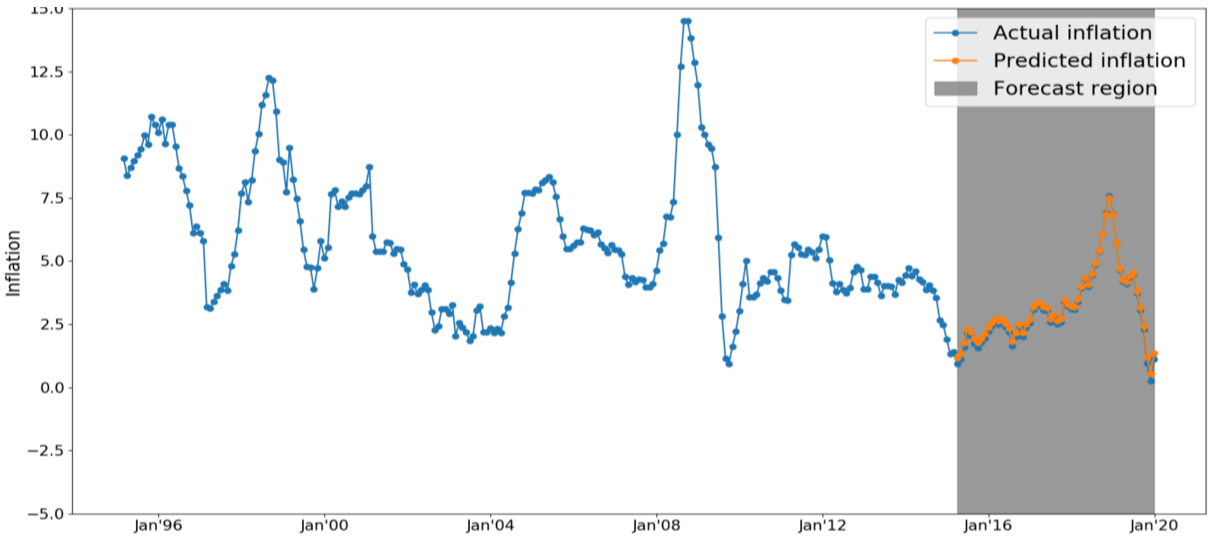


Figure 1.22. ANN models for Region 10

(i) Univariate



(ii) Multivariate

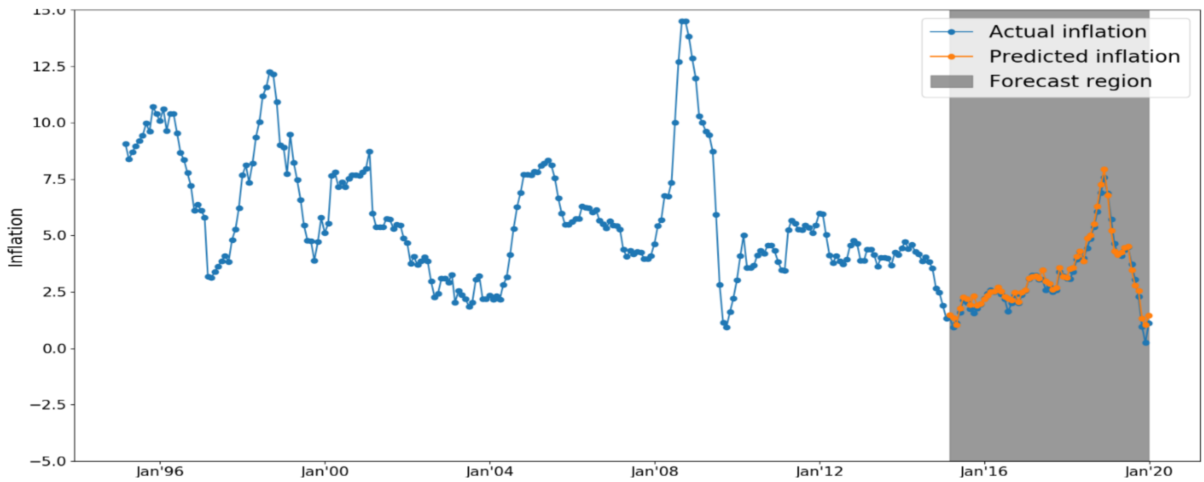
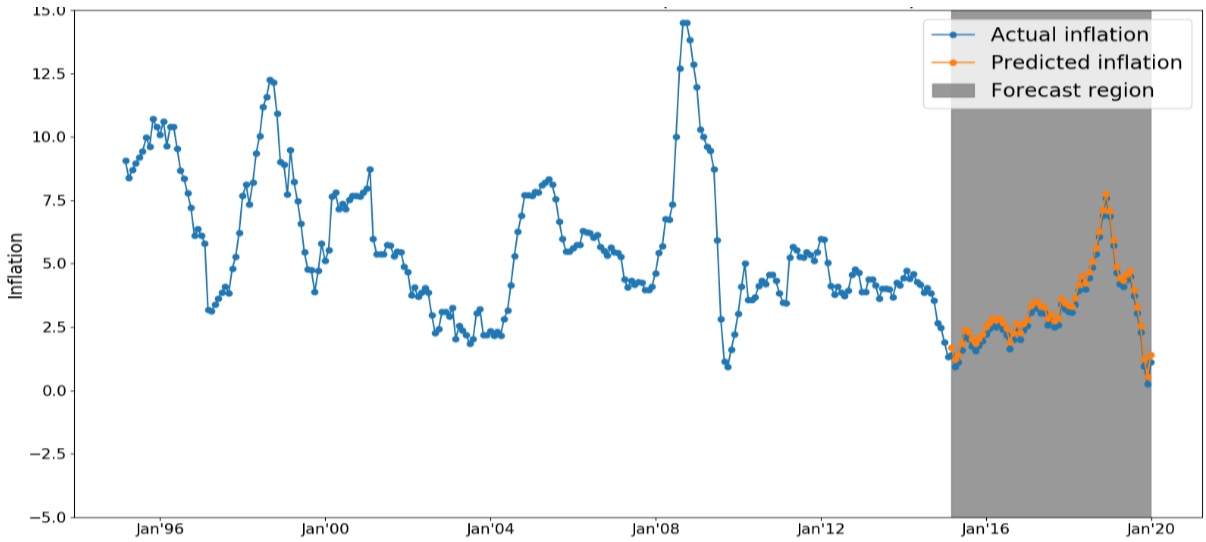


Figure 1.23. LSTM models for Region 10

(i) Univariate



(ii) Multivariate

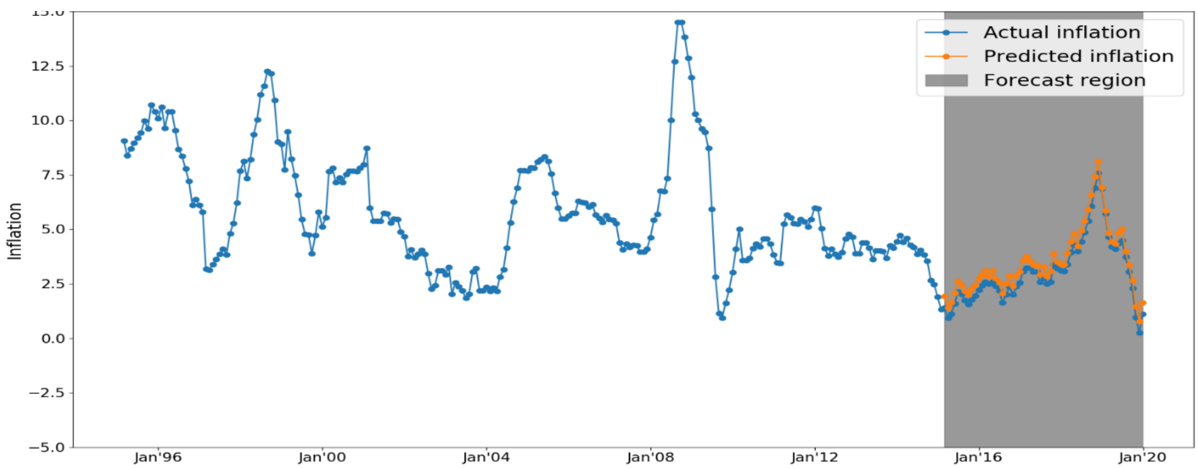
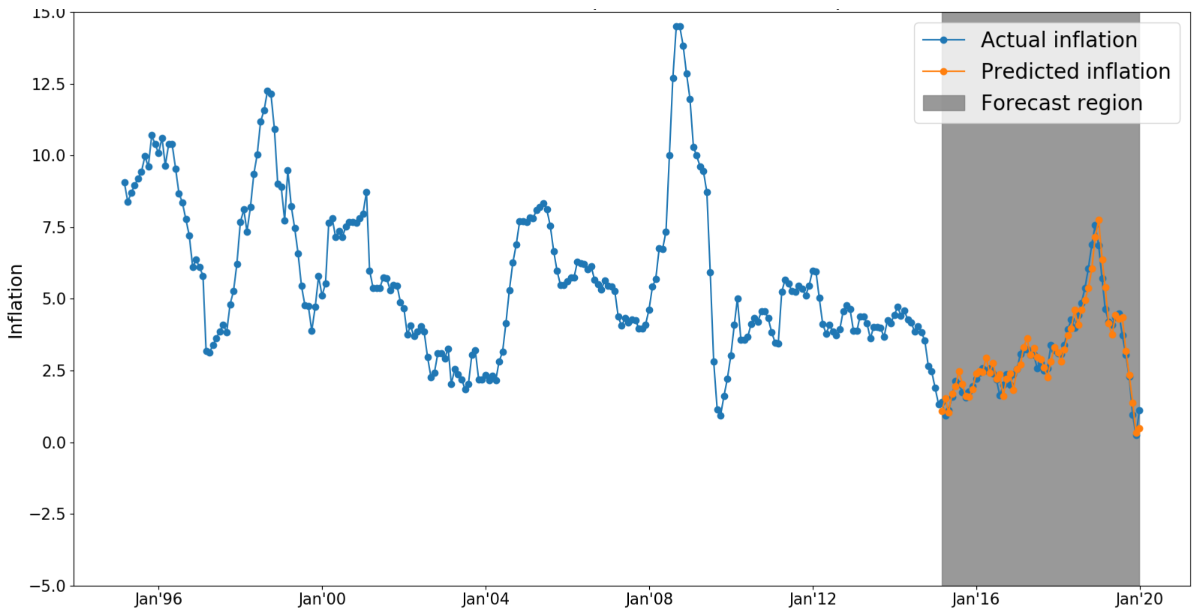


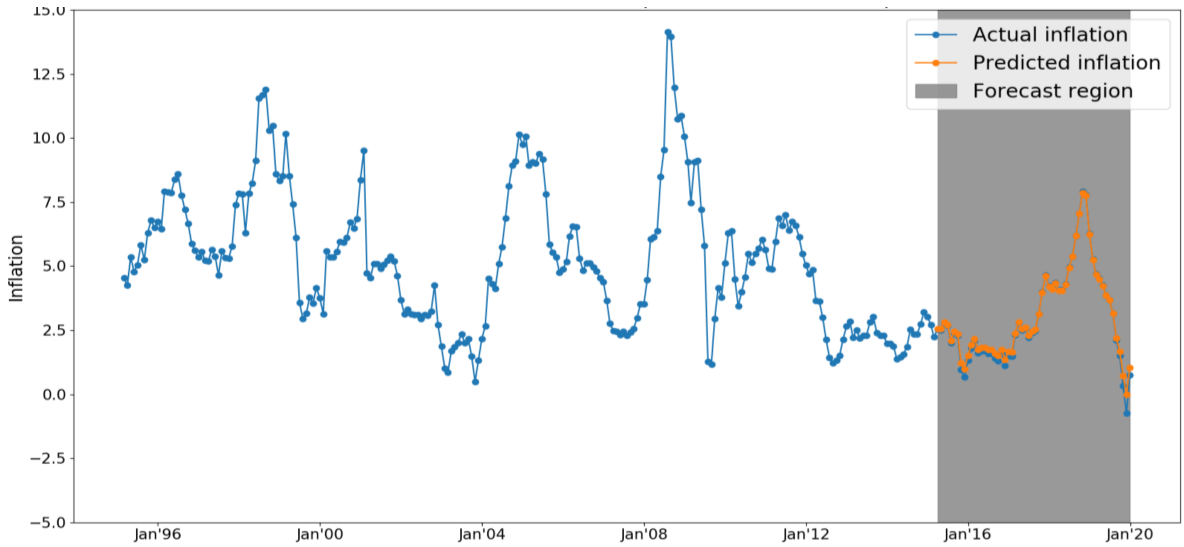
Figure 1.24. ARIMA model for Region 10



G. Region 11

Figure 1.25. SVR models for Region 11

(i) Univariate



(ii) Multivariate

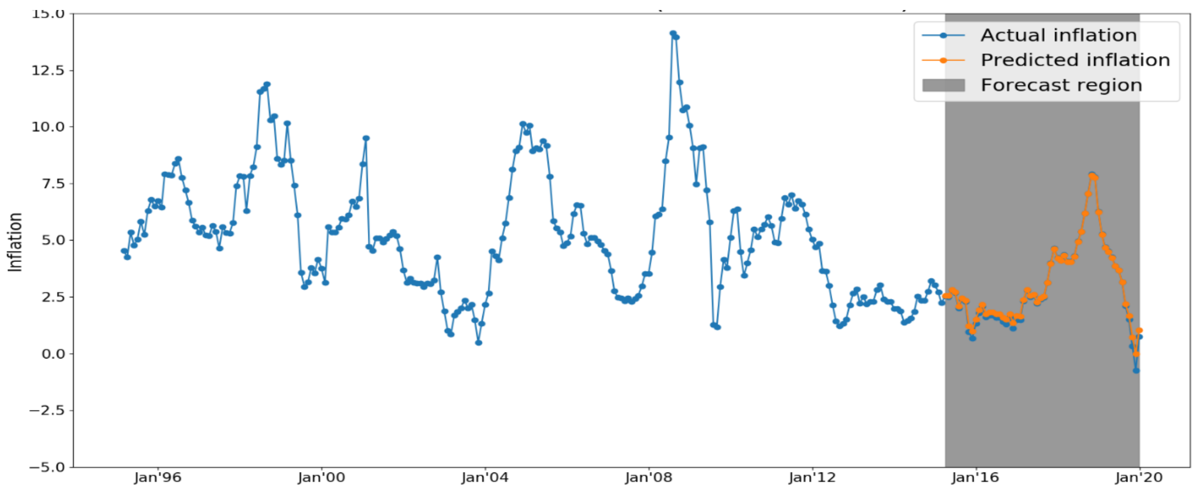
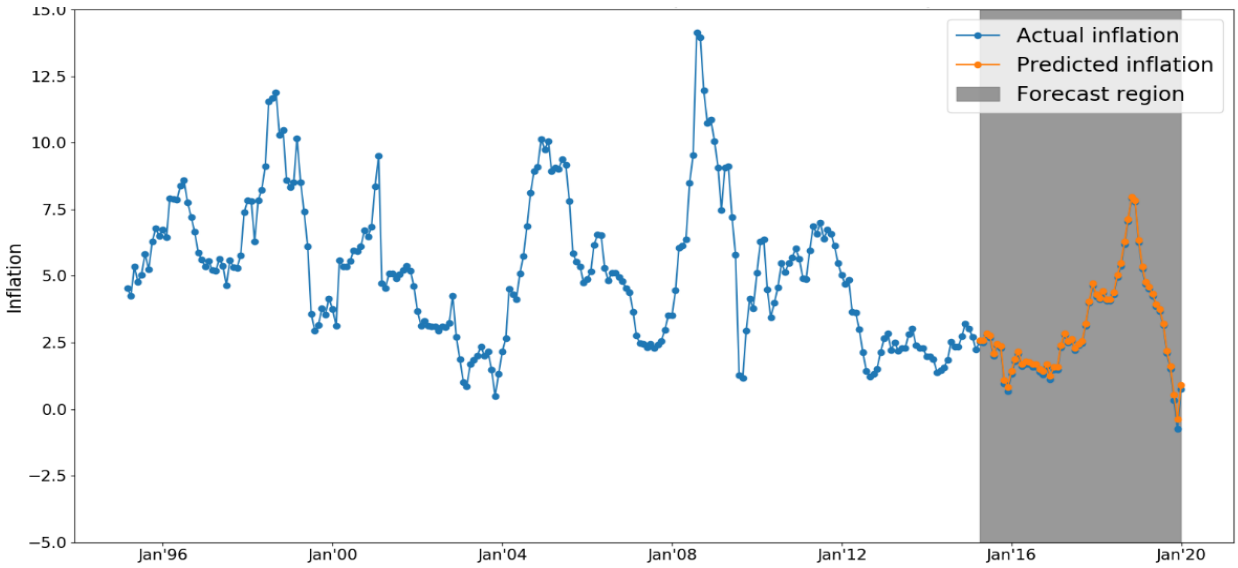


Figure 1.26. ANN models for Region 11

(i) Univariate



(ii) Multivariate

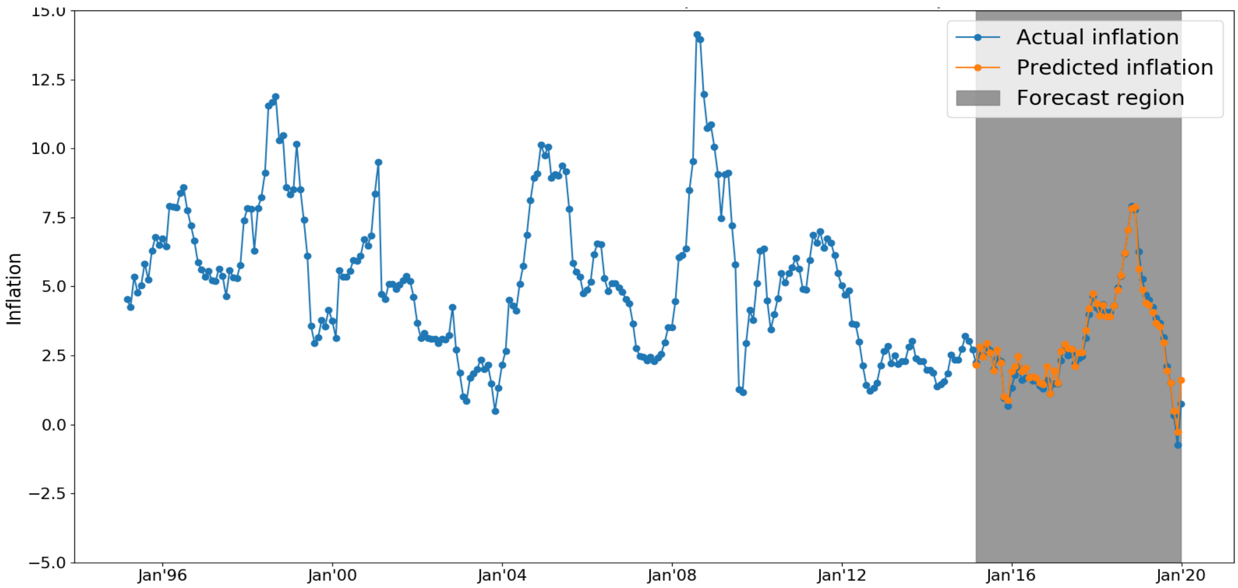
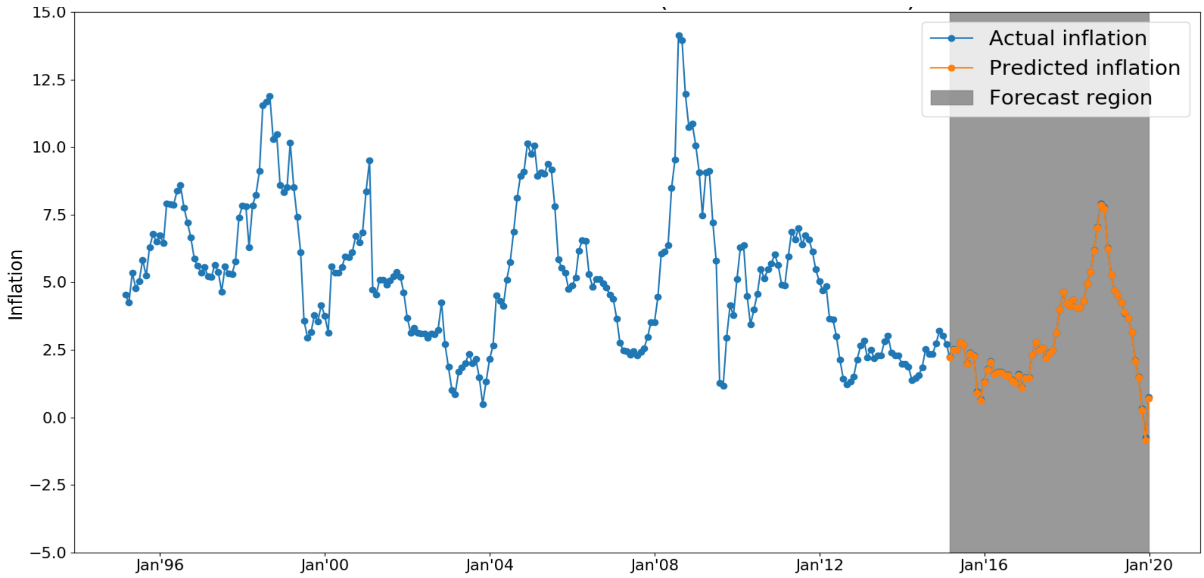


Figure 1.27. LSTM models for Region 11

(i) Univariate



(ii) Multivariate

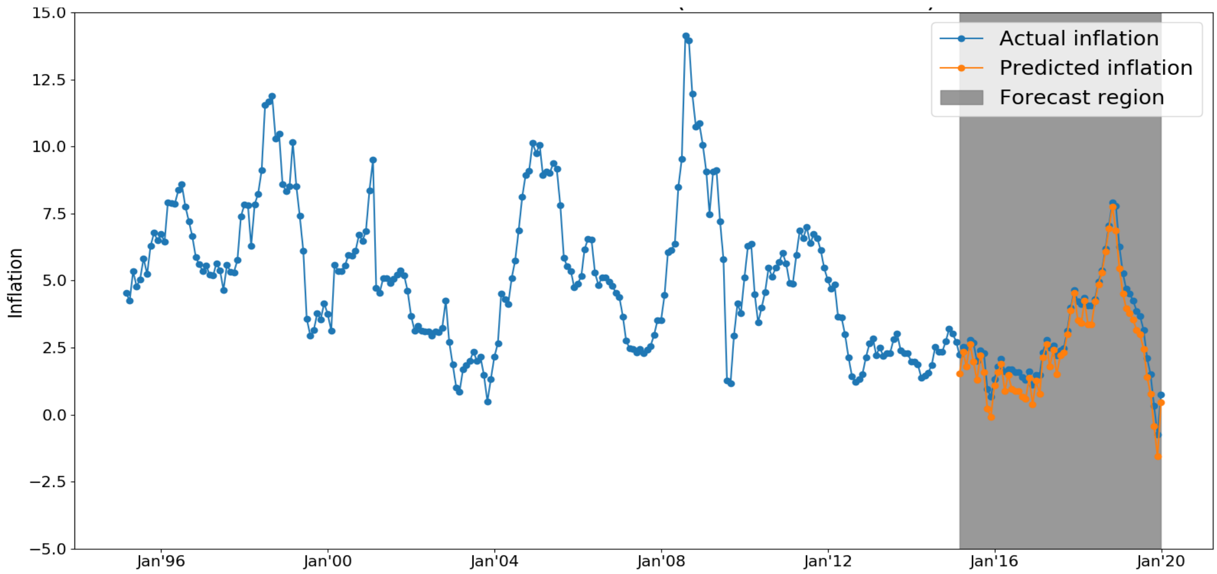
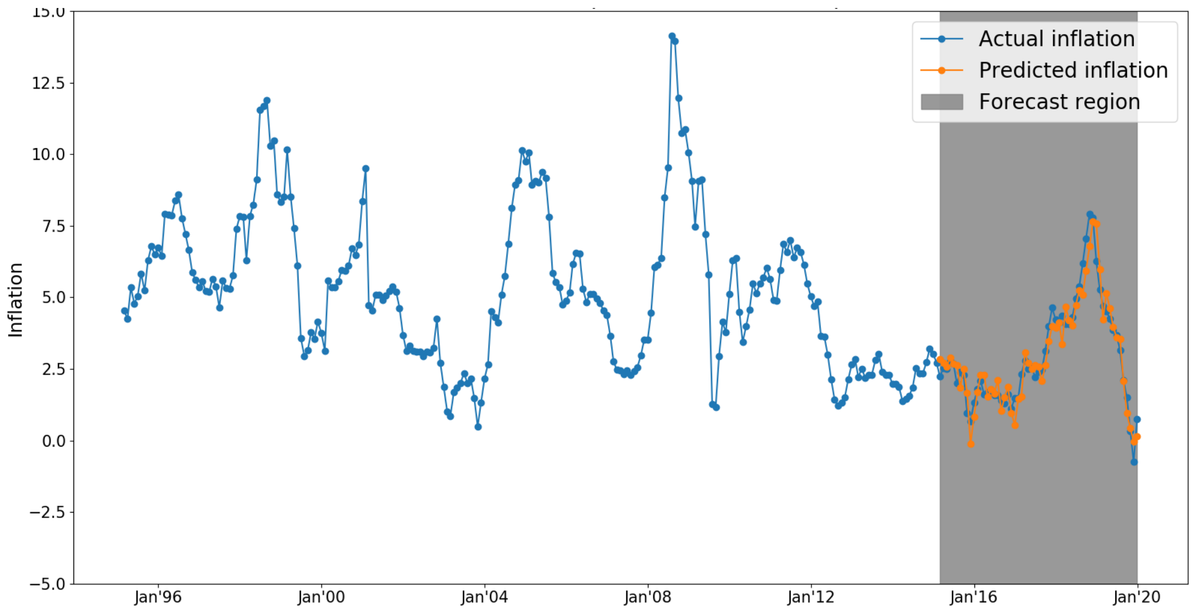


Figure 1.28. ARIMA model for Region 11

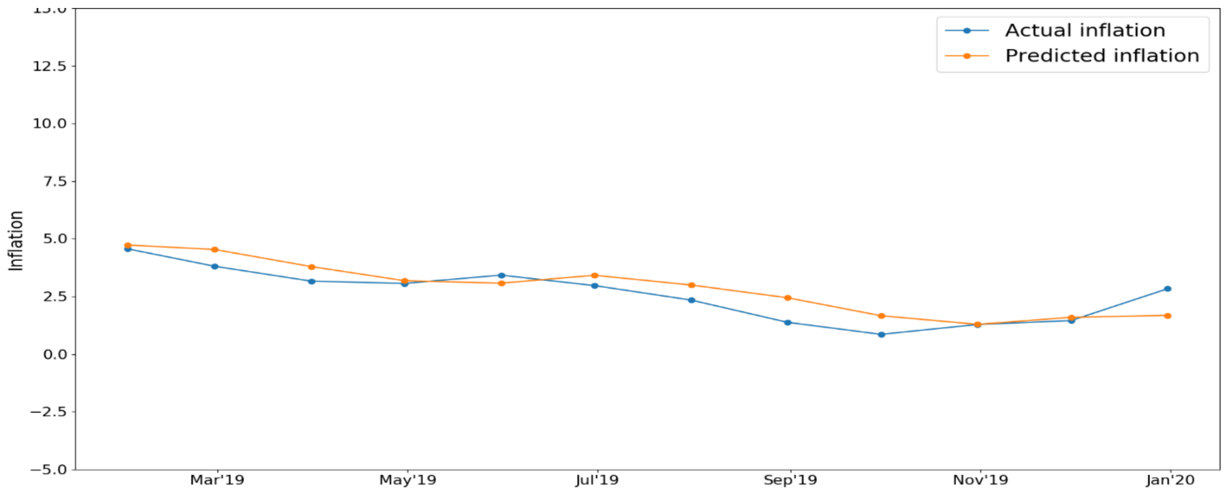


II. 12-month ahead forecasts

A. NCR

Figure 2.1. SVR models for NCR

(i) Univariate



(ii) Multivariate

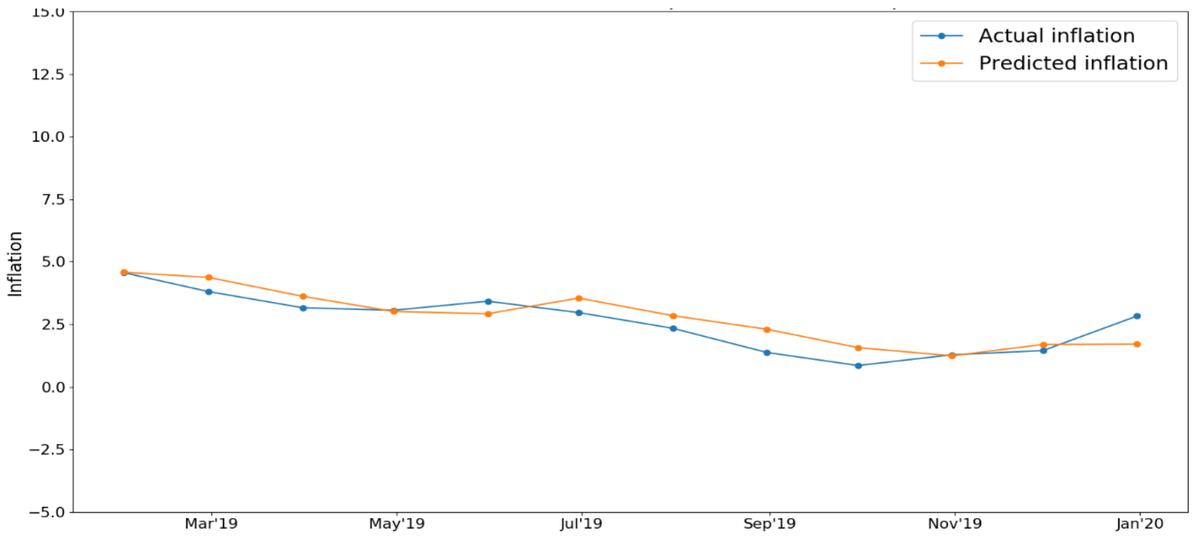
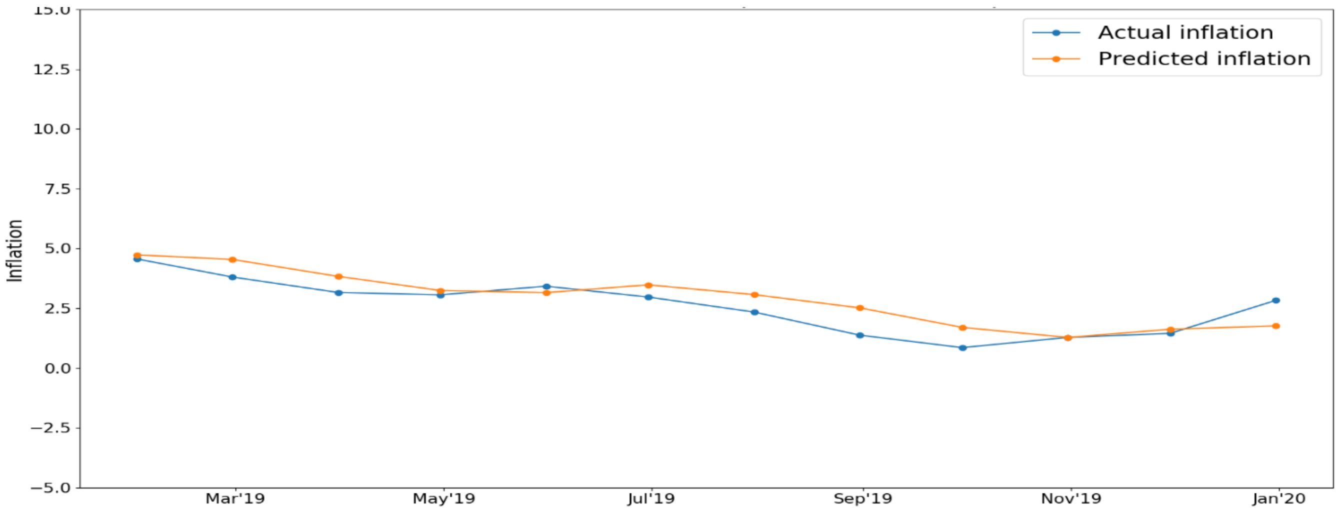


Figure 2.2. ANN models for NCR

(i) Univariate



(ii) Multivariate

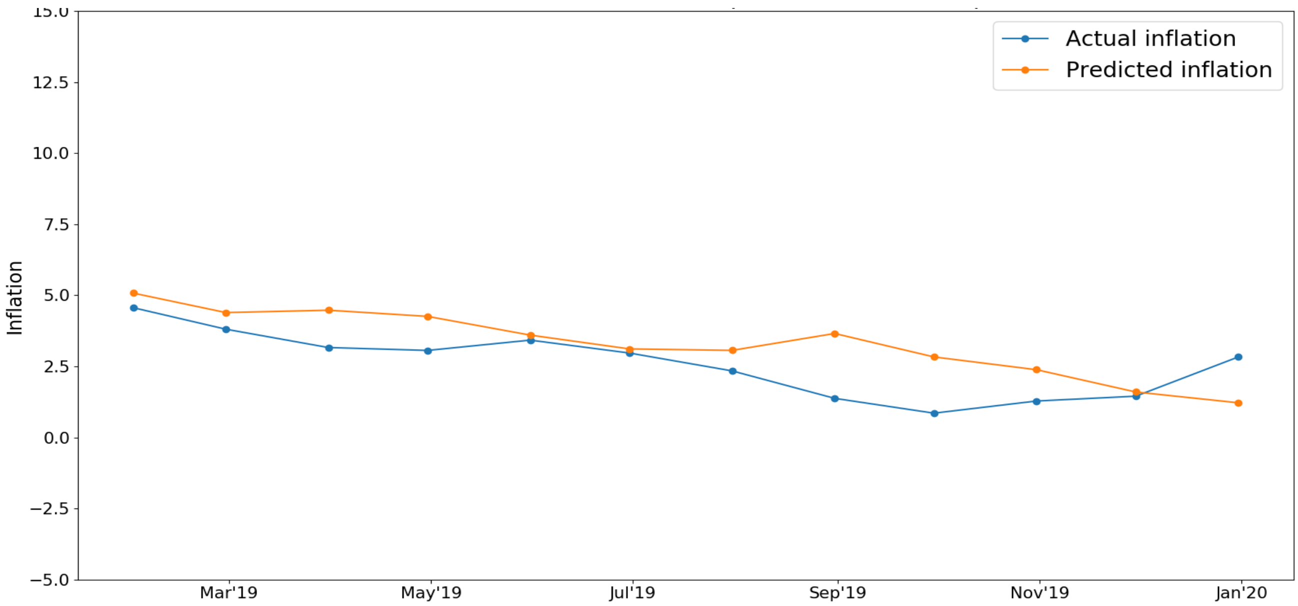
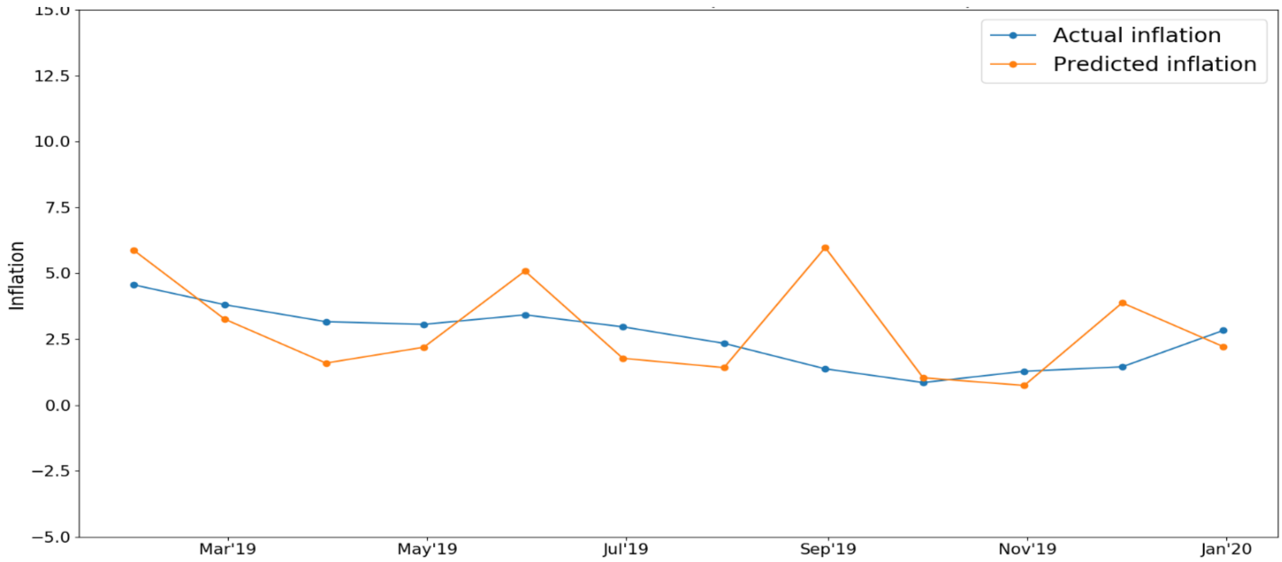


Figure 2.3. LSTM models for NCR

(i) Univariate



(ii) Multivariate

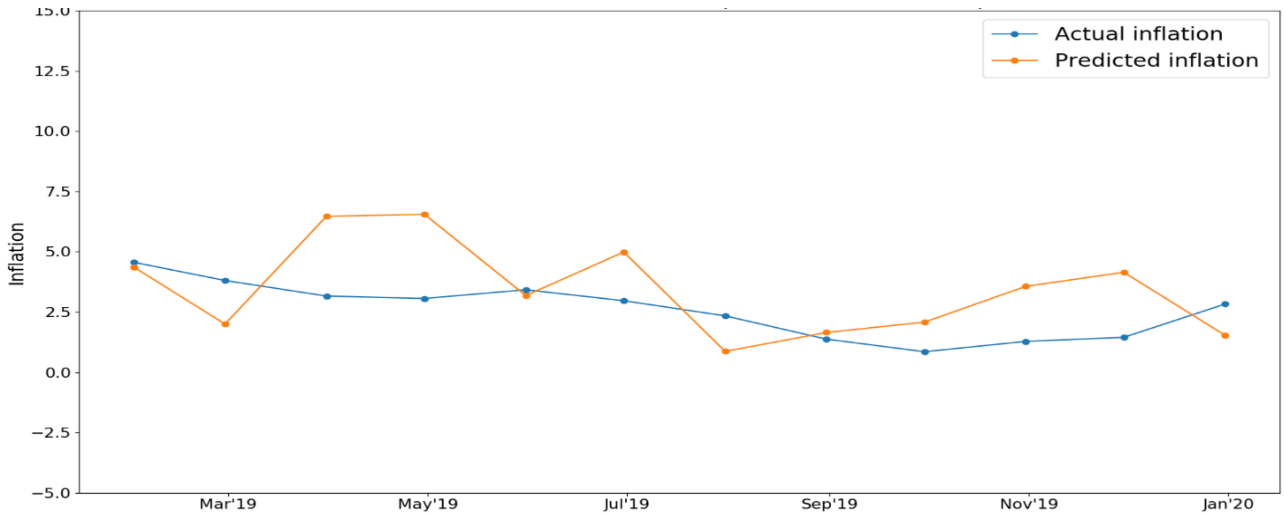
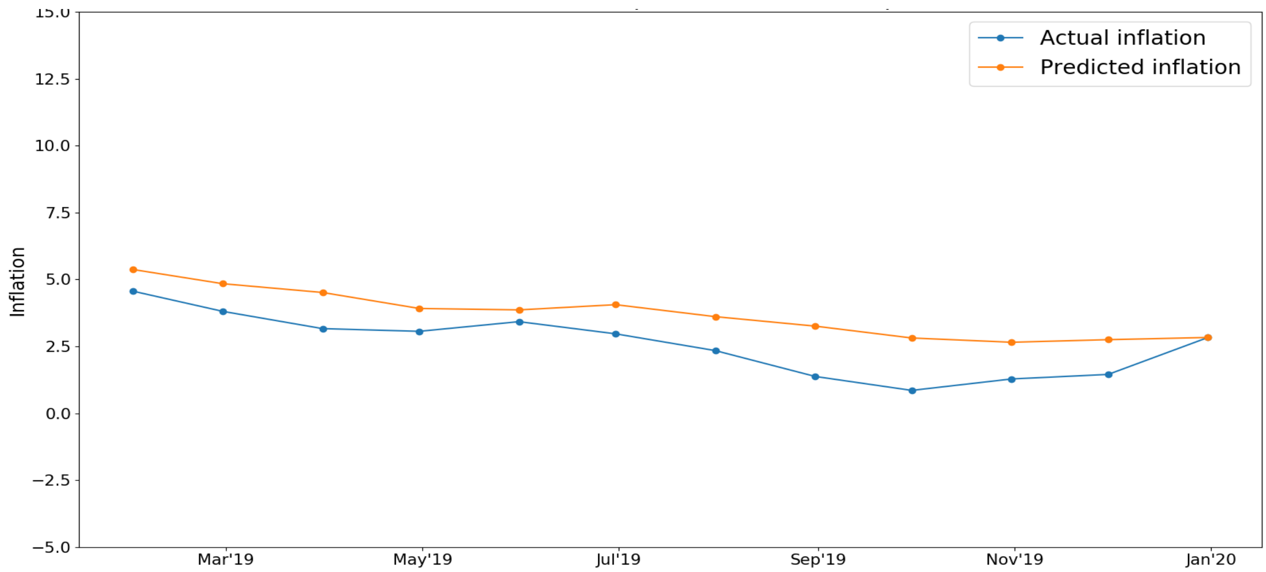


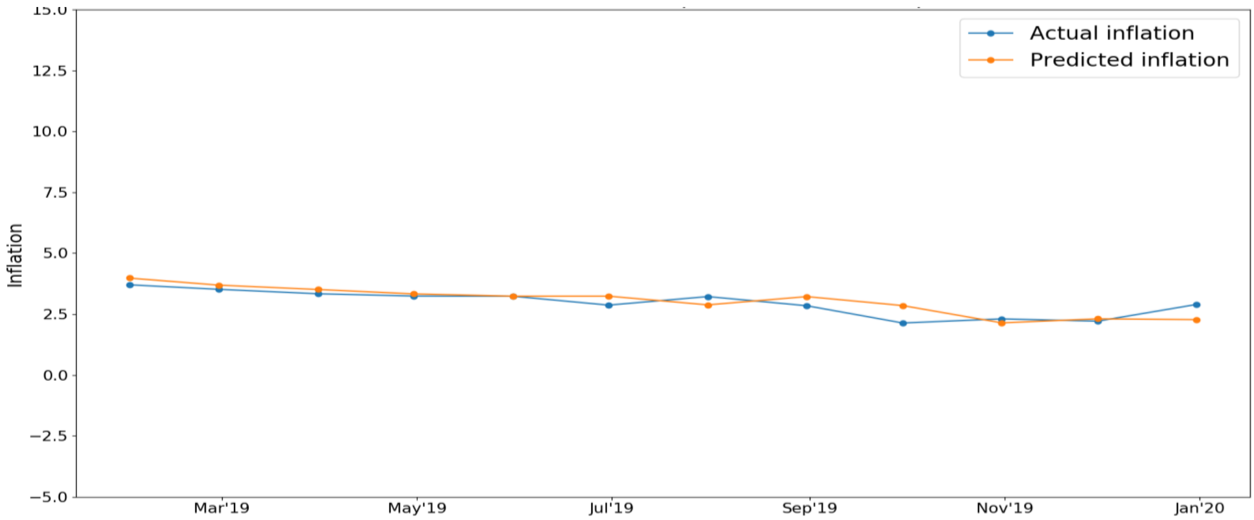
Figure 2.4. ARIMA model for NCR



B. Region 3

Figure 2.5. SVR models for Region 3

(i) Univariate



(ii) Multivariate

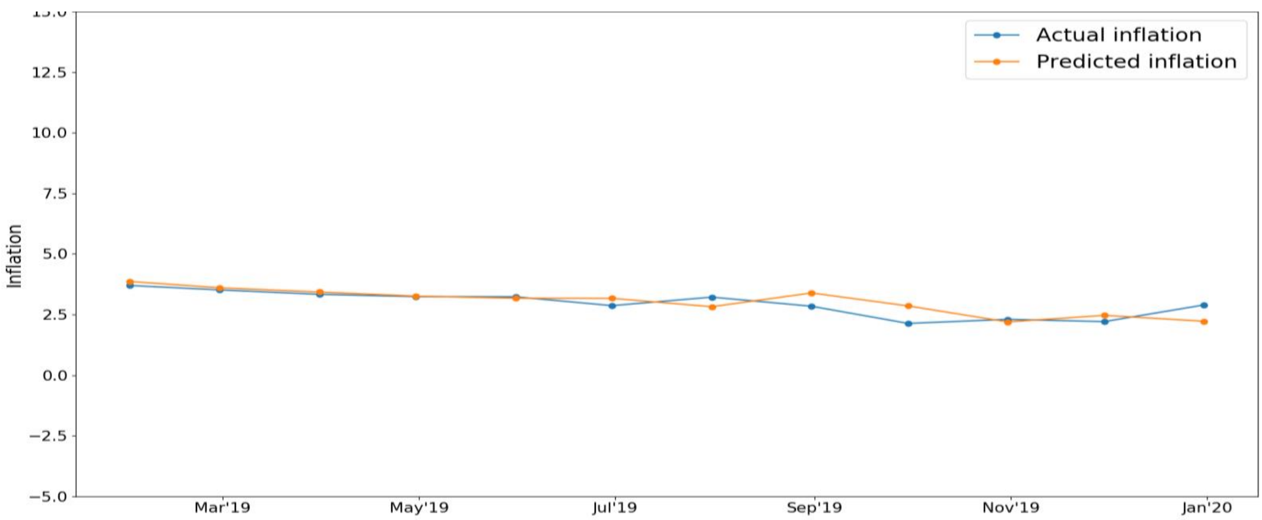
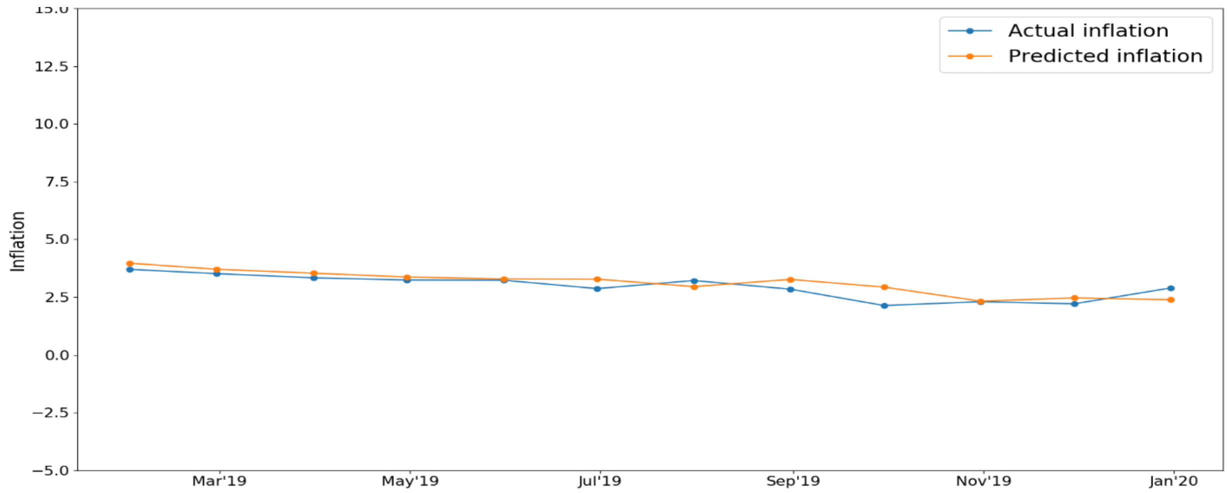


Figure 2.6. ANN models for Region 3

(i) Univariate



(ii) Multivariate

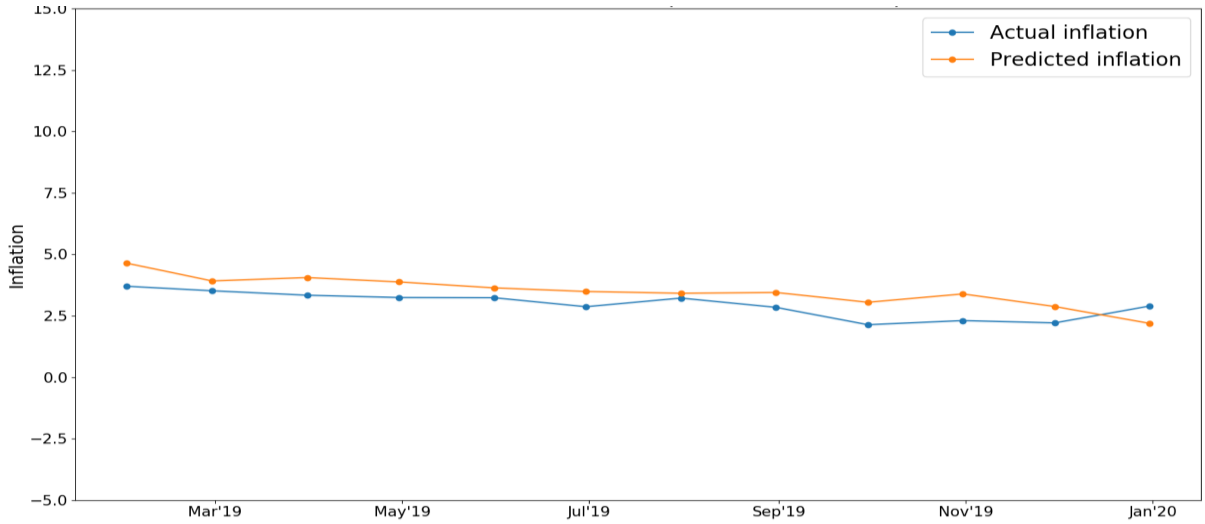
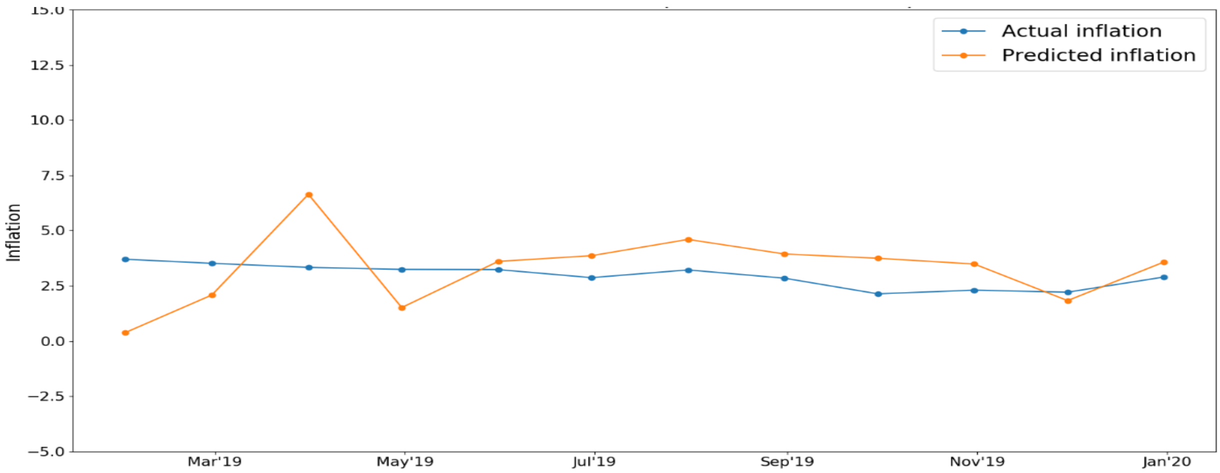


Figure 2.7. LSTM models for Region 3

(i) Univariate



(ii) Multivariate

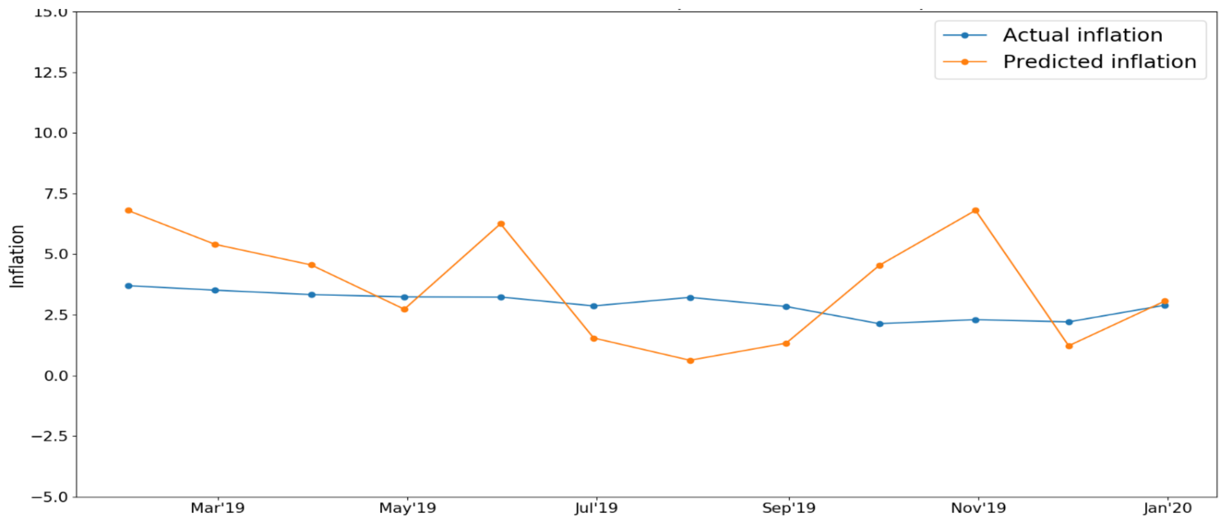
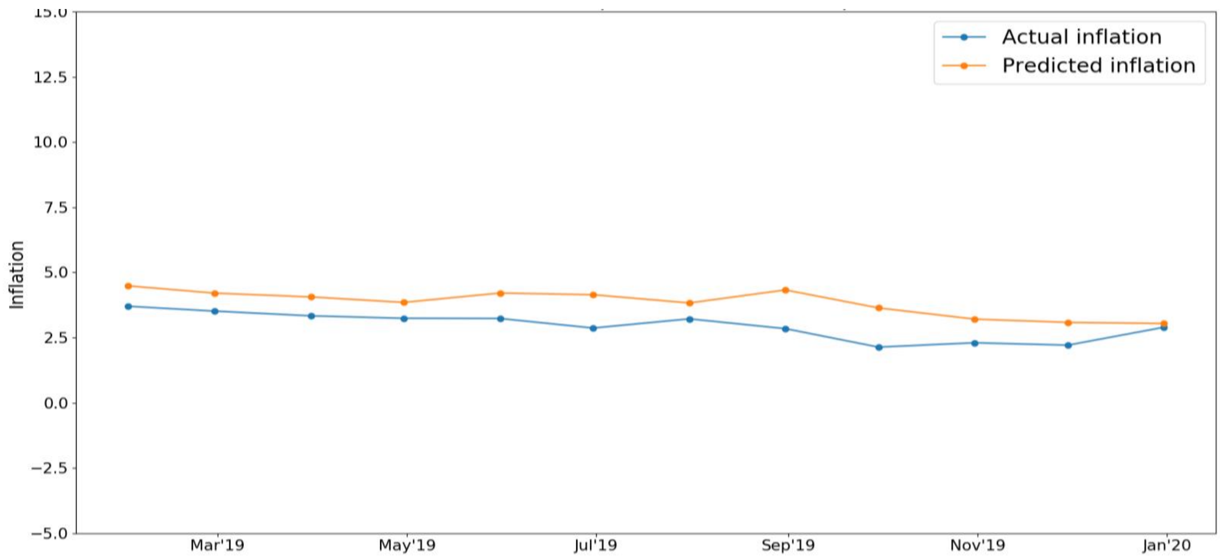


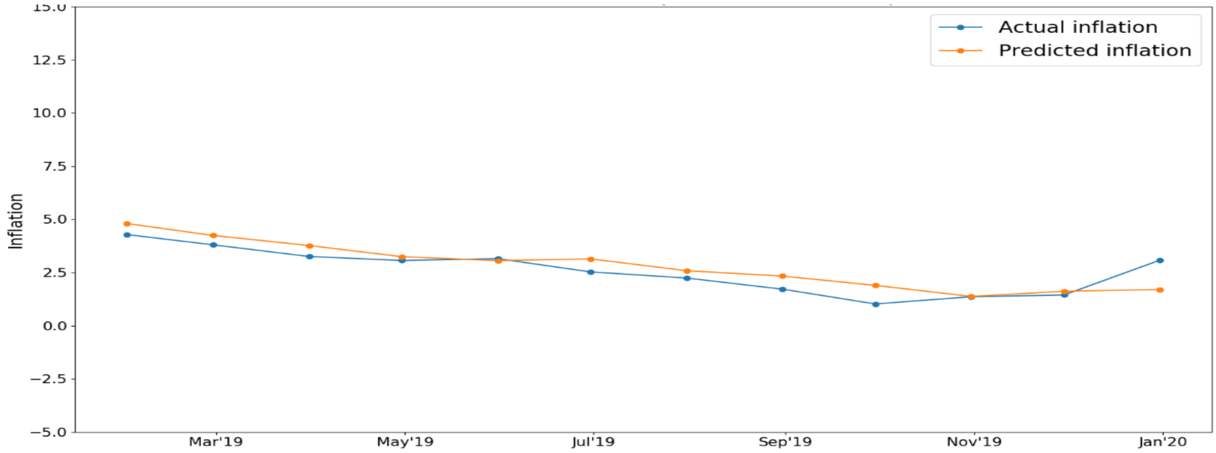
Figure 2.8. ARIMA model for Region 3



C. Region 4-A

Figure 2.9. SVR models for Region 4-A

(i) Univariate



(ii) Multivariate

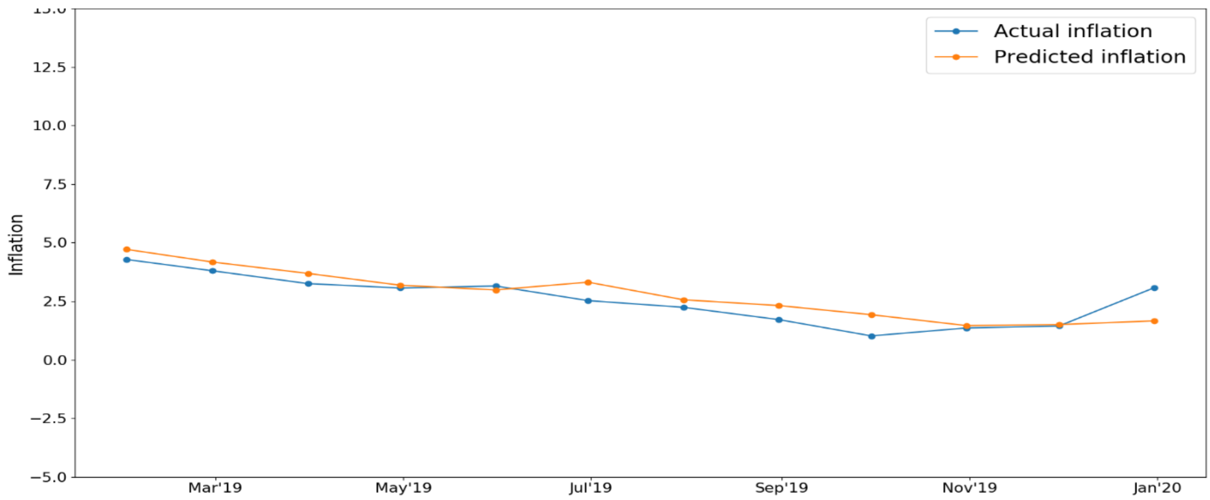
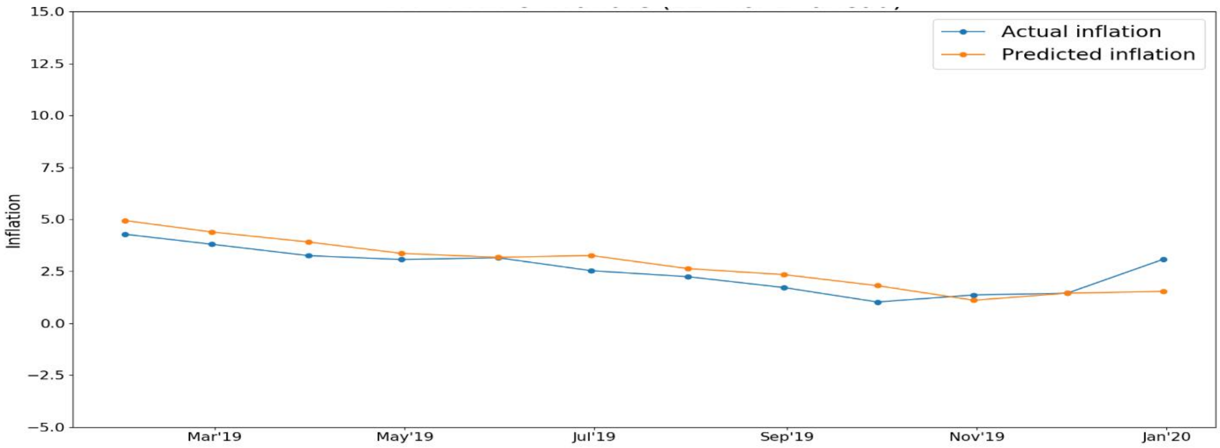


Figure 2.10. ANN models for Region 4-A

(i) Univariate



(ii) Multivariate

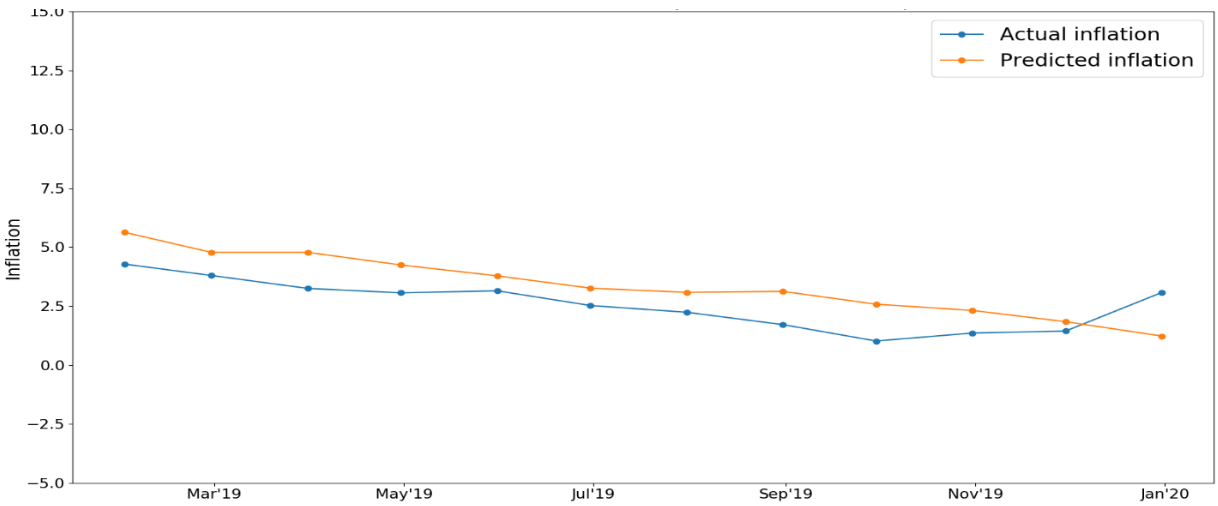
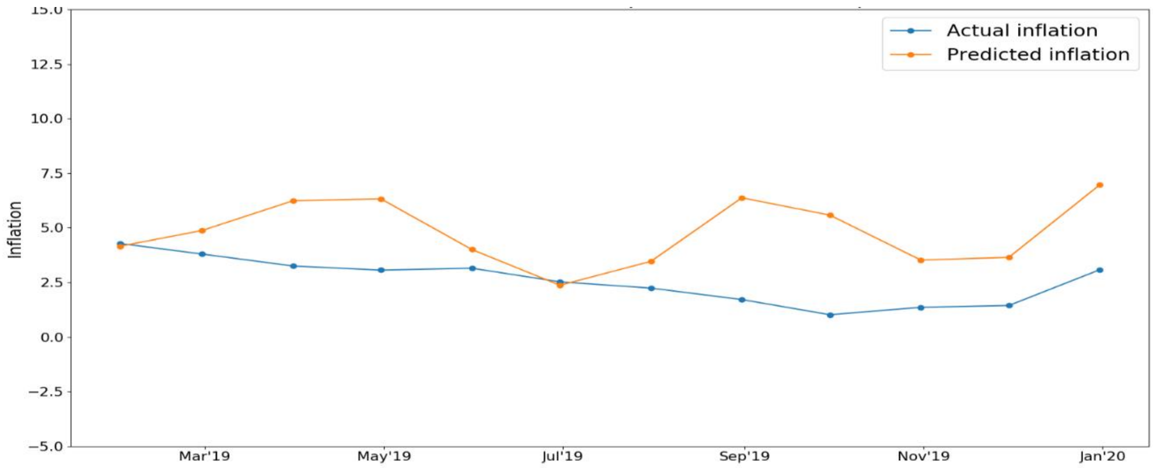


Figure 2.11. LSTM models for Region 4-A

(i) Univariate



(ii) Multivariate

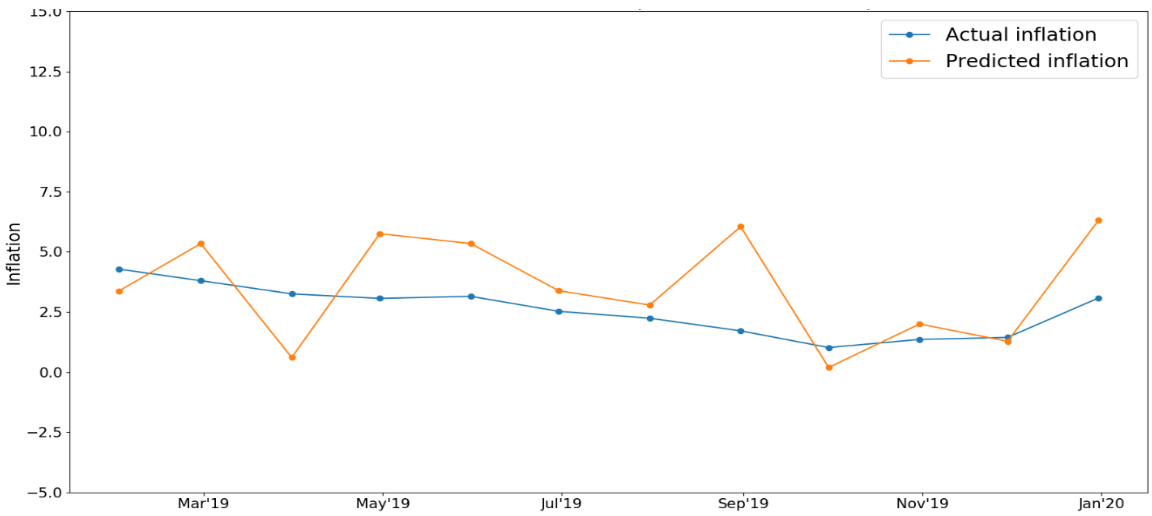
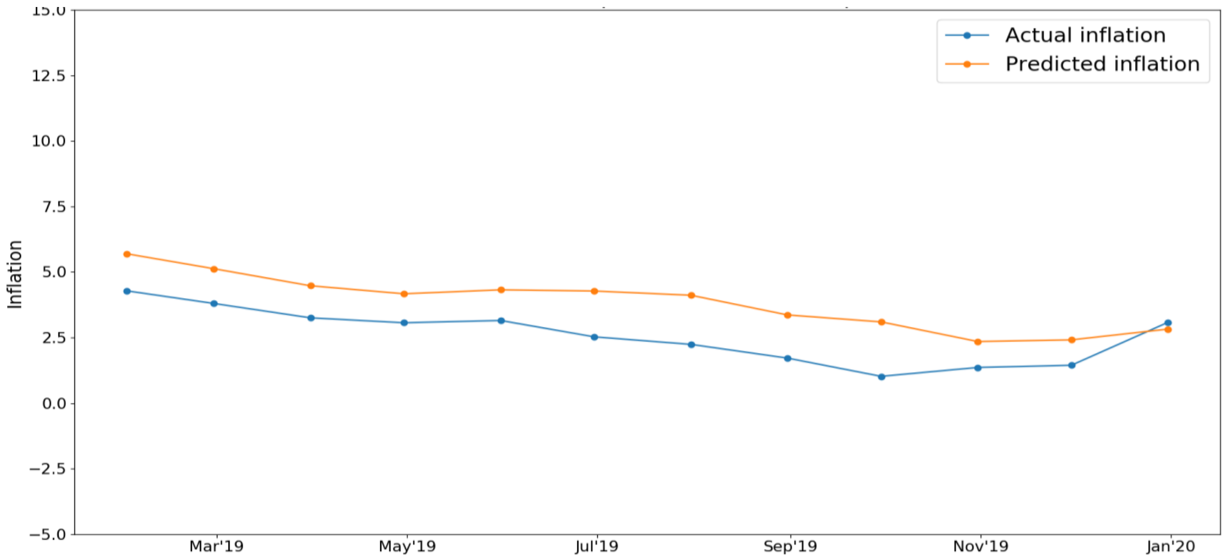


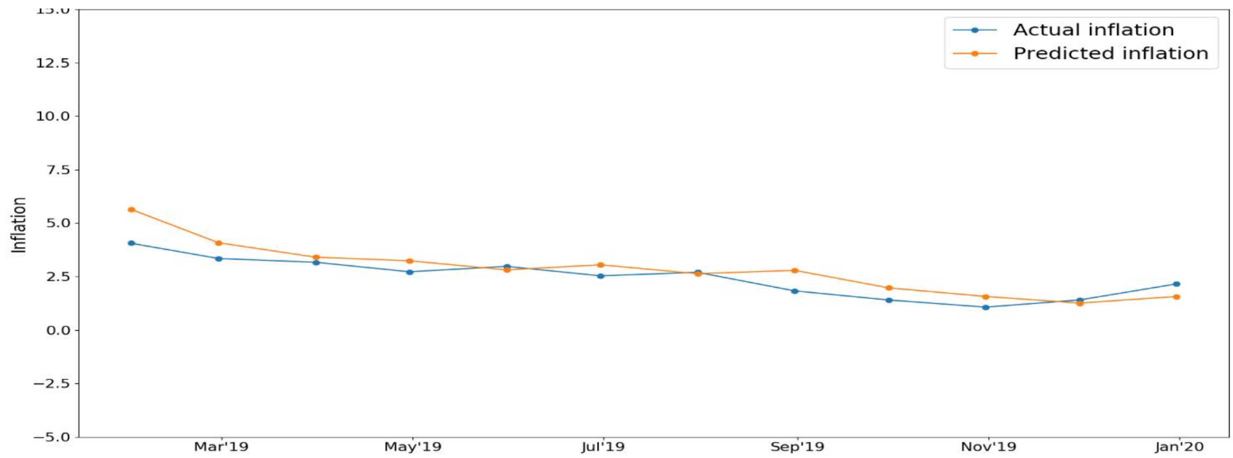
Figure 2.12. ARIMA model for Region 4-A



D. Region 6

Figure 2.13. SVR models for Region 6

(i) Univariate



(ii) Multivariate

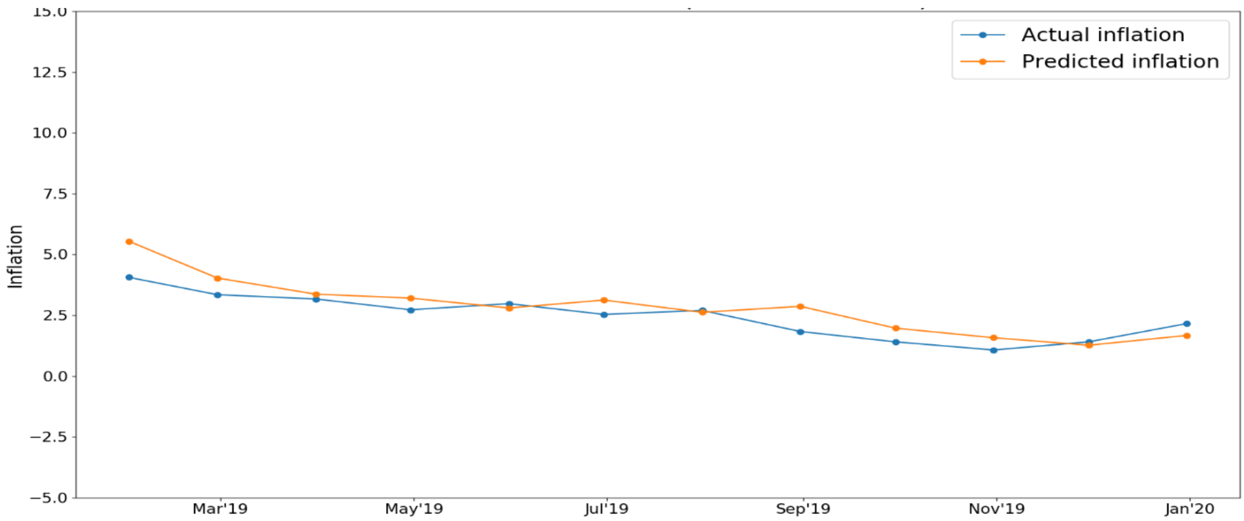
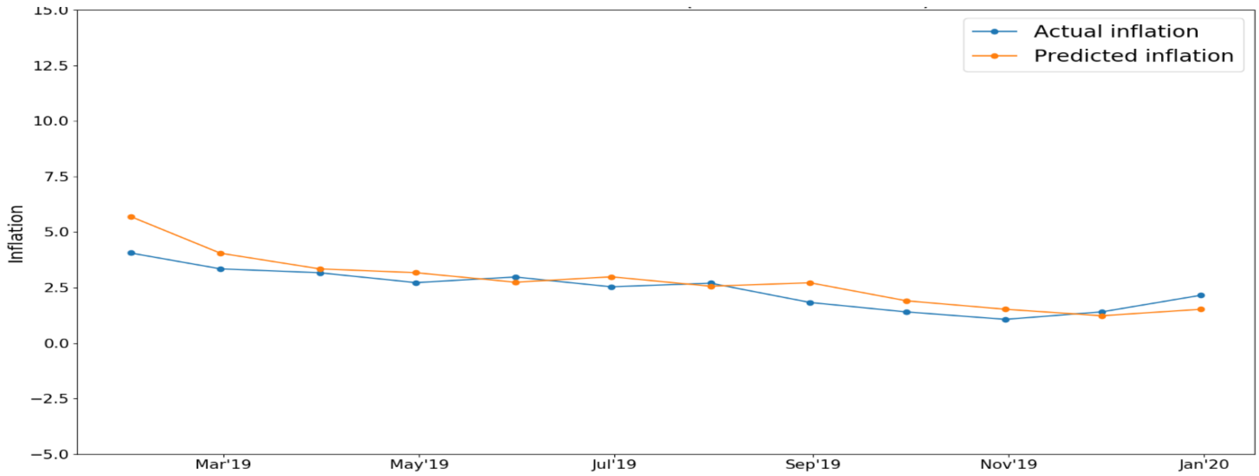


Figure 2.14. ANN models for Region 6

(i) Univariate



(ii) Multivariate

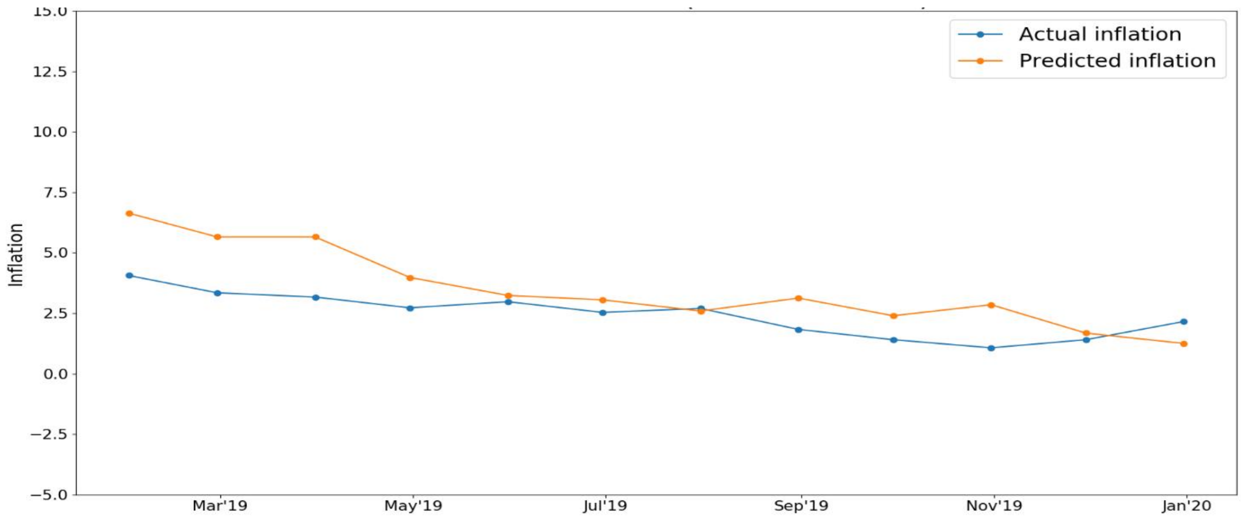
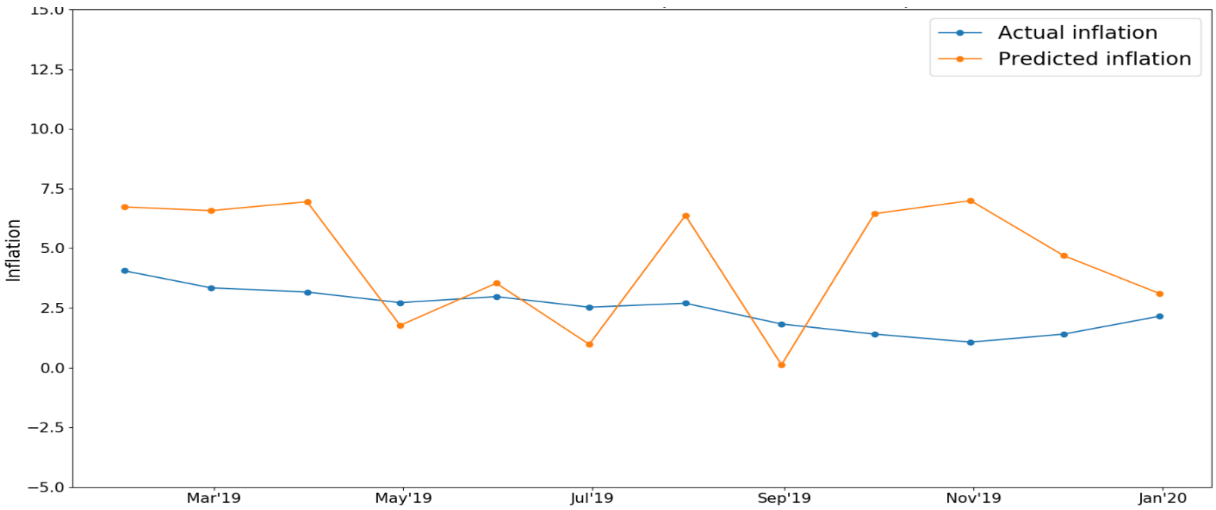


Figure 2.15. LSTM models for Region 6

(i) Univariate



(ii) Multivariate

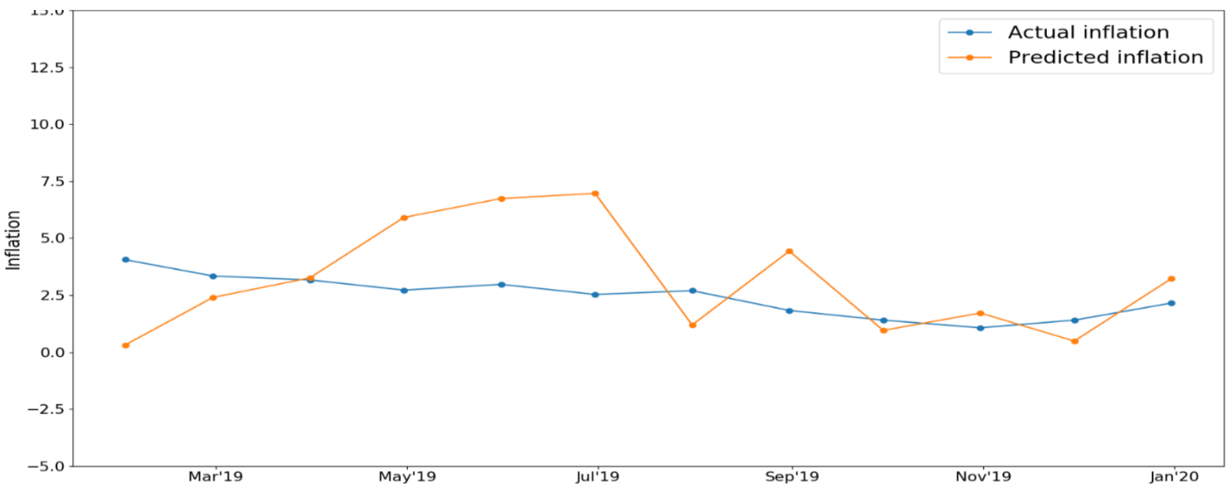
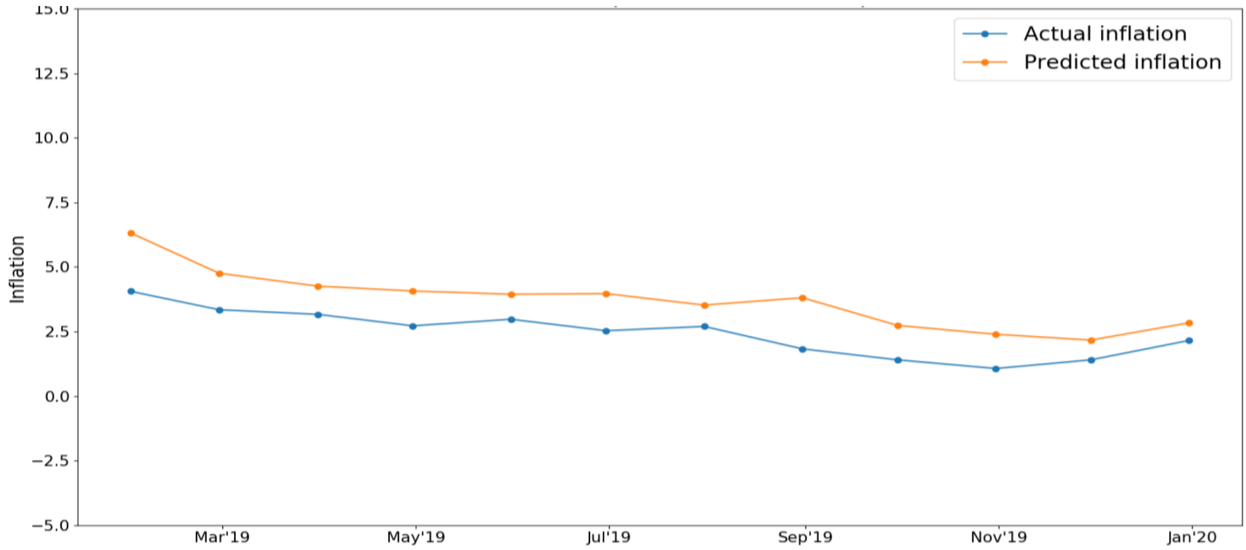


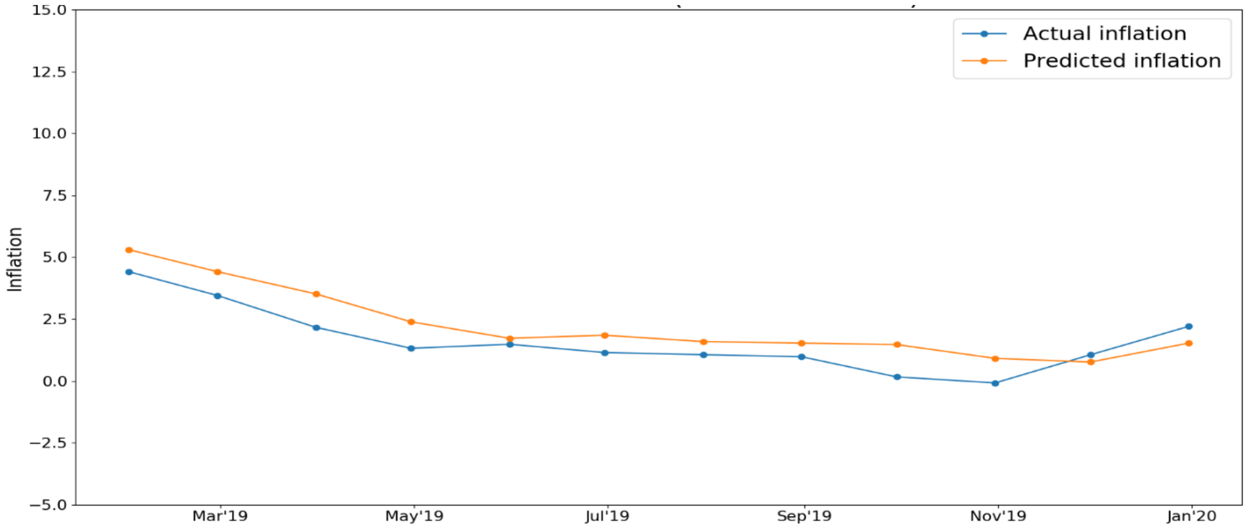
Figure 2.16. ARIMA model for Region 6



E. Region 7

Figure 2.17. SVR models for Region 7

(i) Univariate



(ii) Multivariate

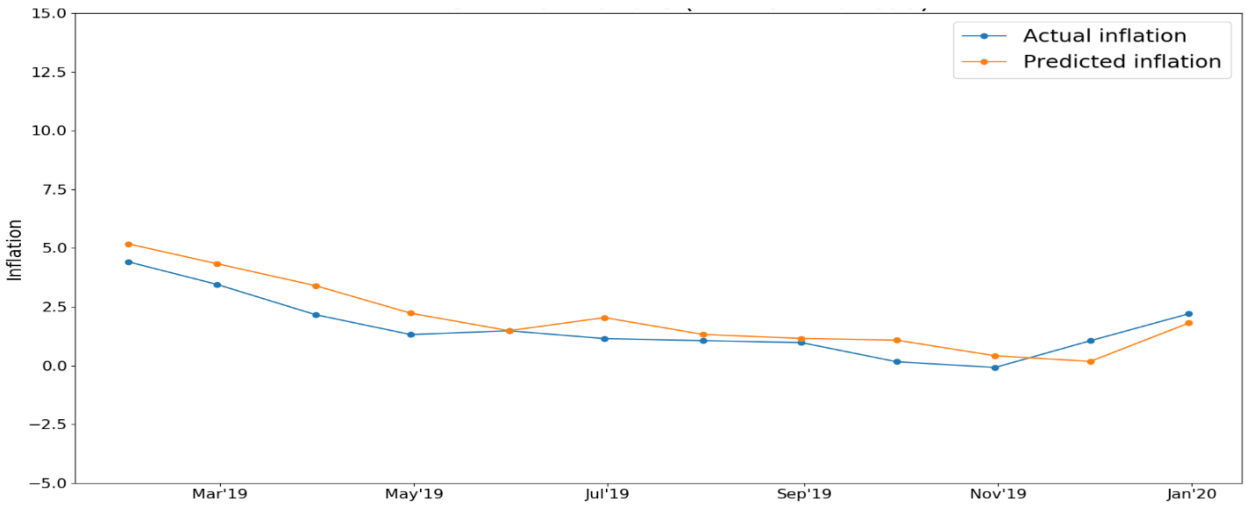
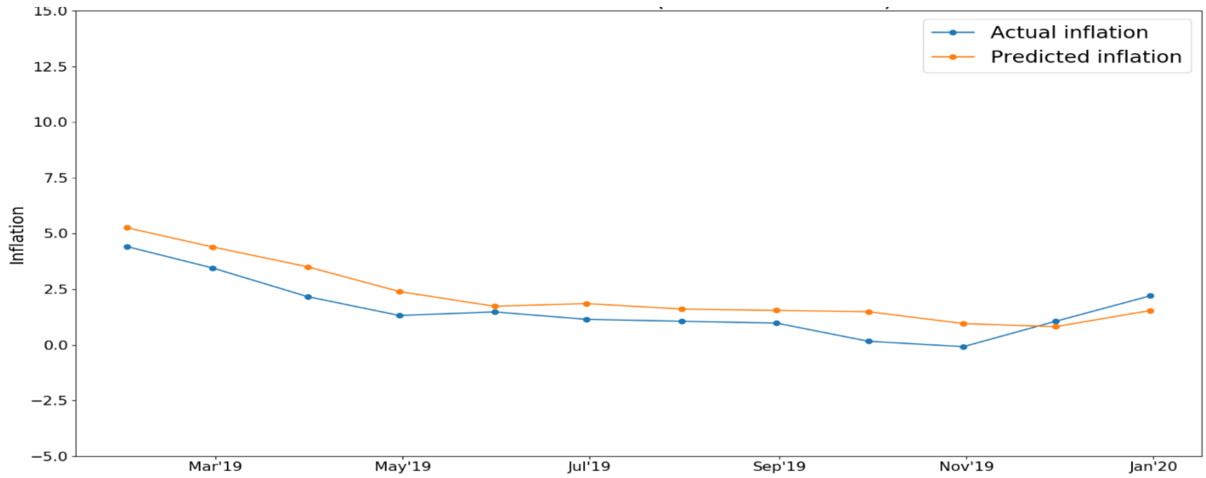


Figure 2.18. ANN models for Region 7

(i) Univariate



(ii) Multivariate

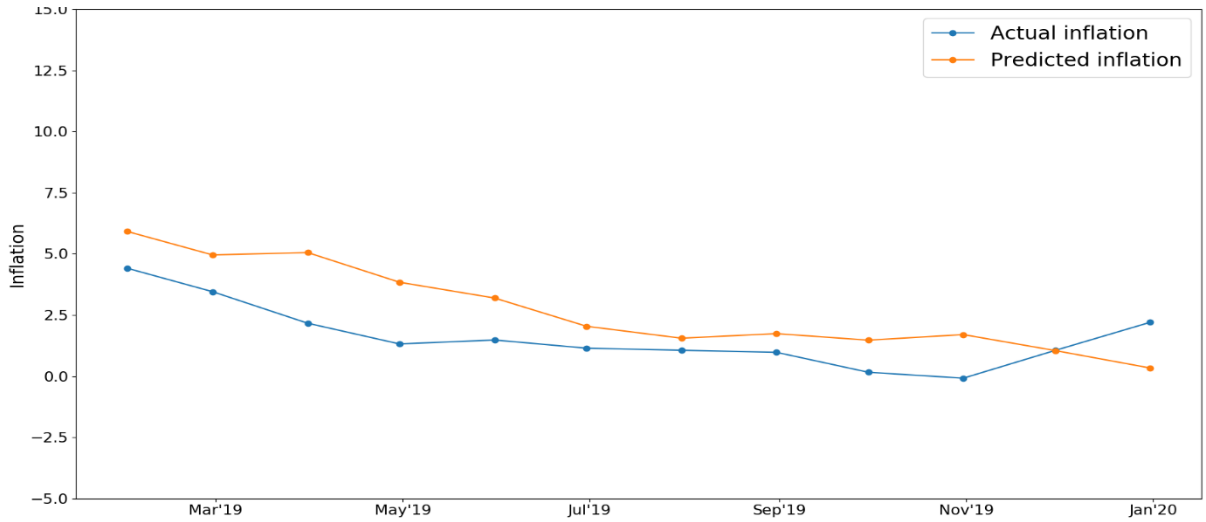
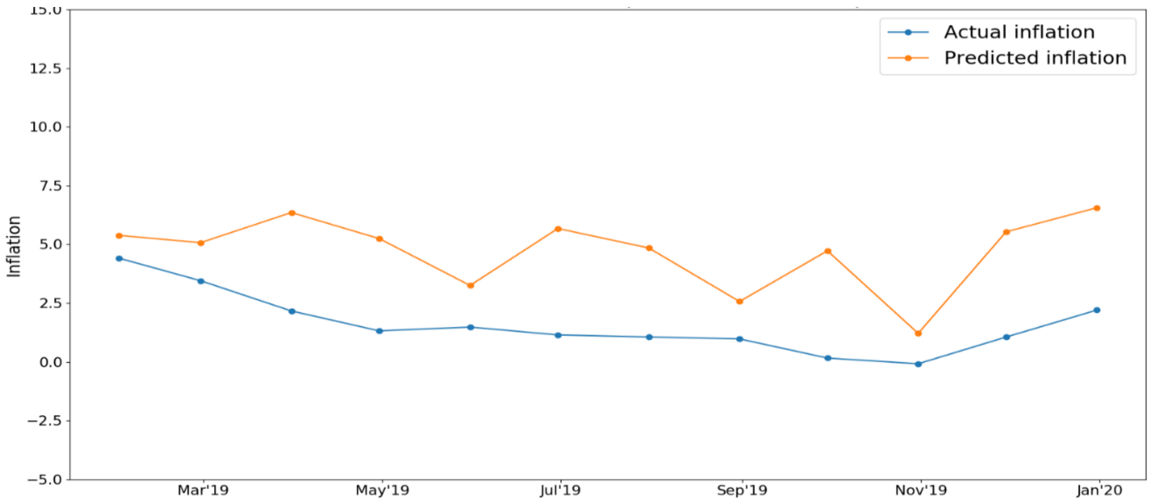


Figure 2.19. LSTM models for Region 7

(i) Univariate



(ii) Multivariate

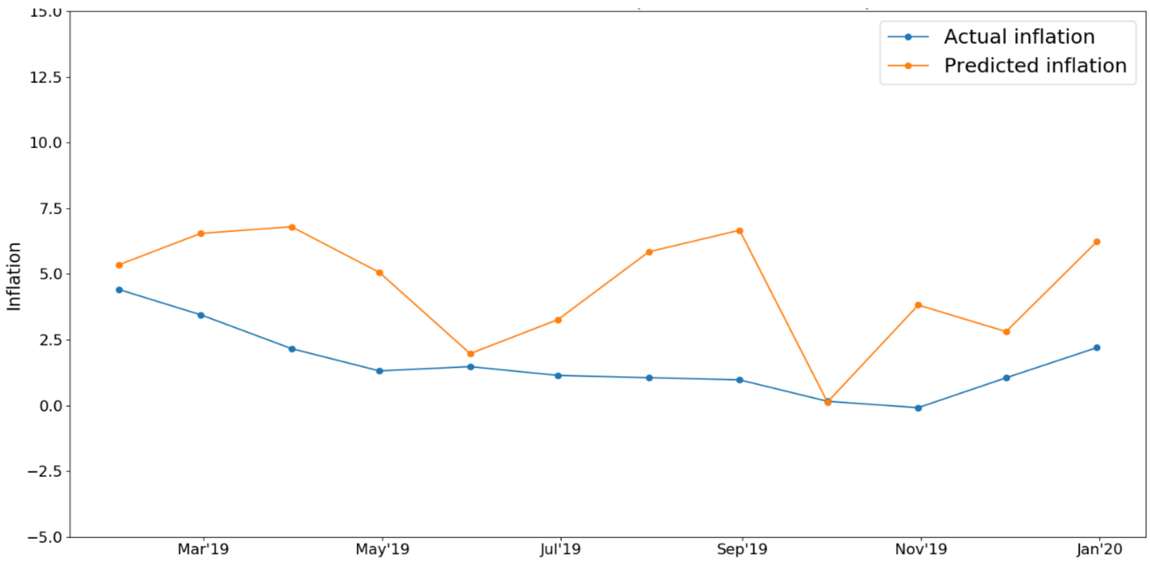
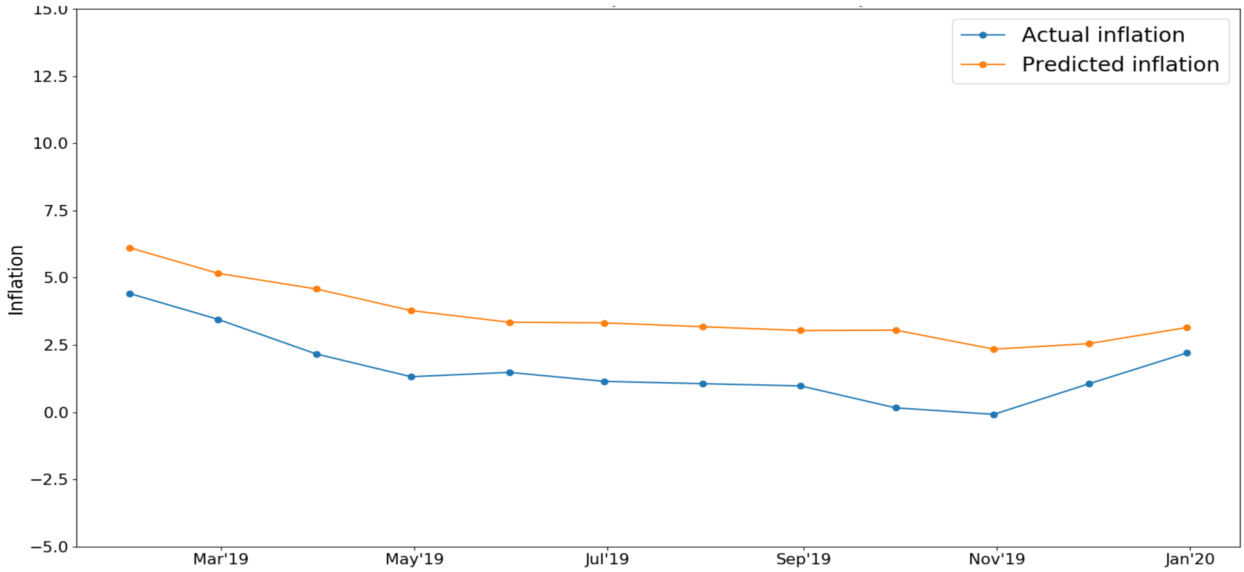


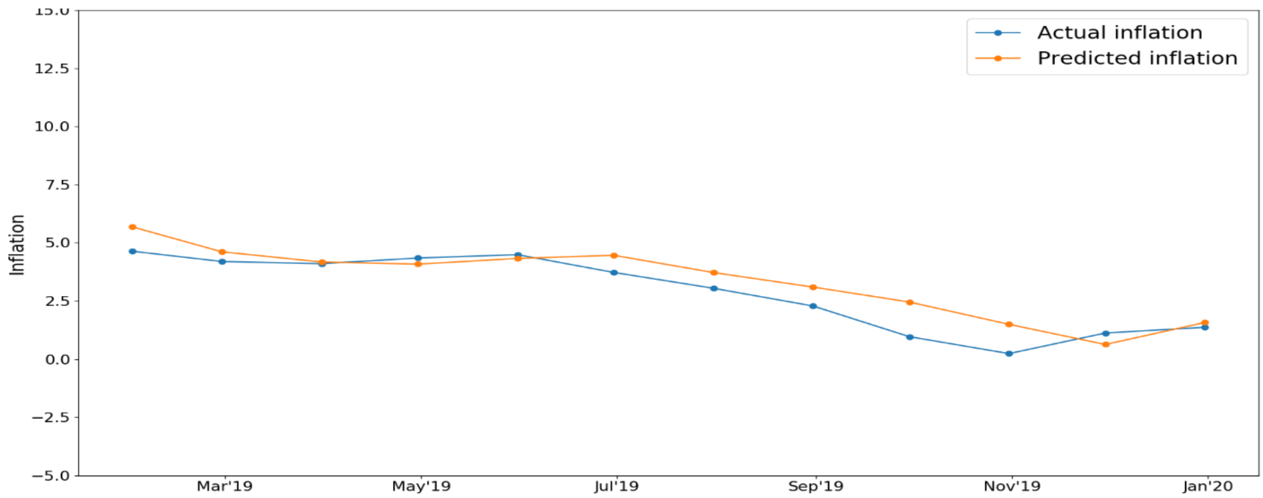
Figure 2.20. ARIMA model for Region 7



F. Region 10

Figure 2.21. SVR models for Region 10

(i) Univariate



(ii) Multivariate

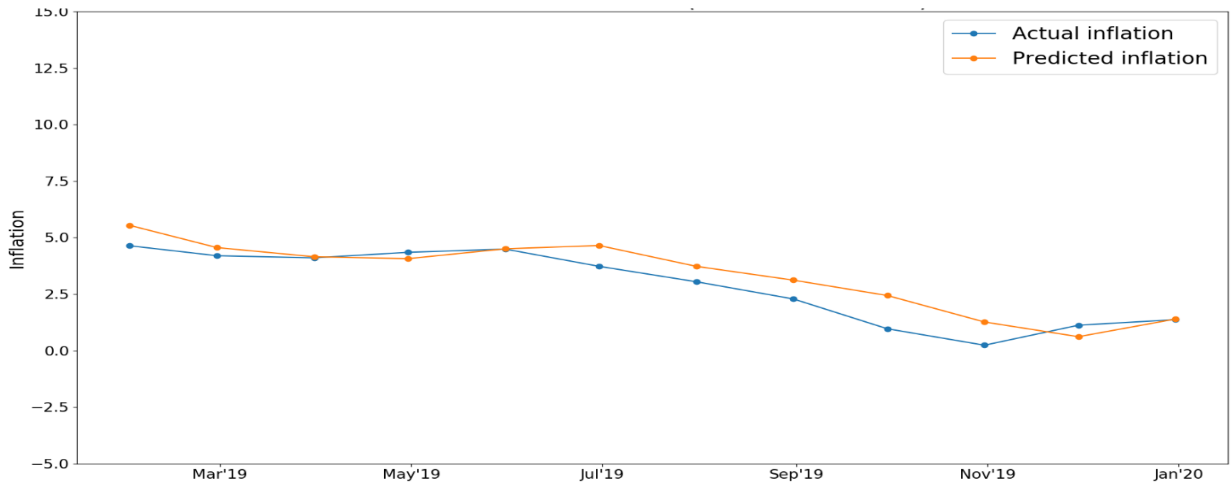
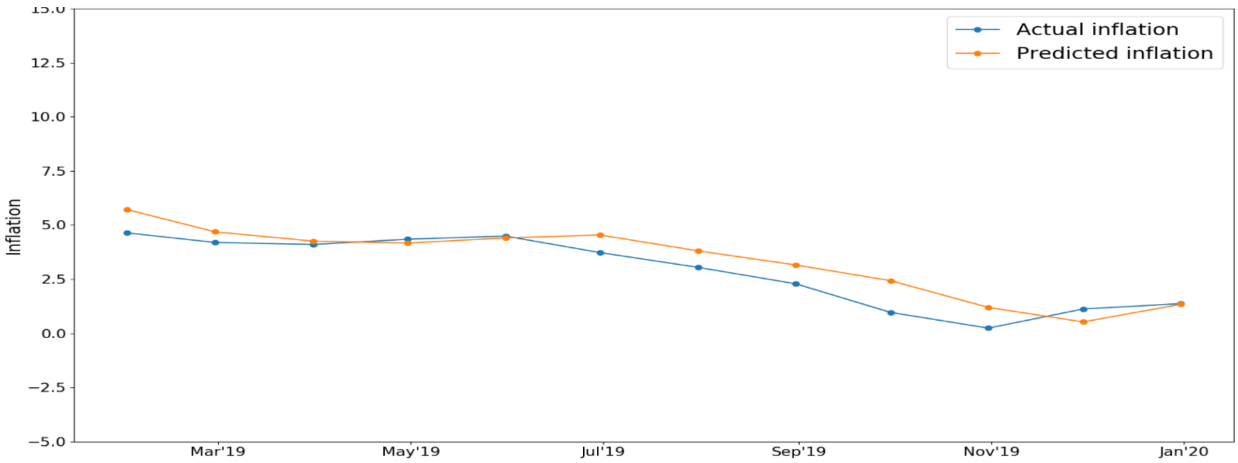


Figure 2.22. ANN models for Region 10

(i) Univariate



(ii) Multivariate

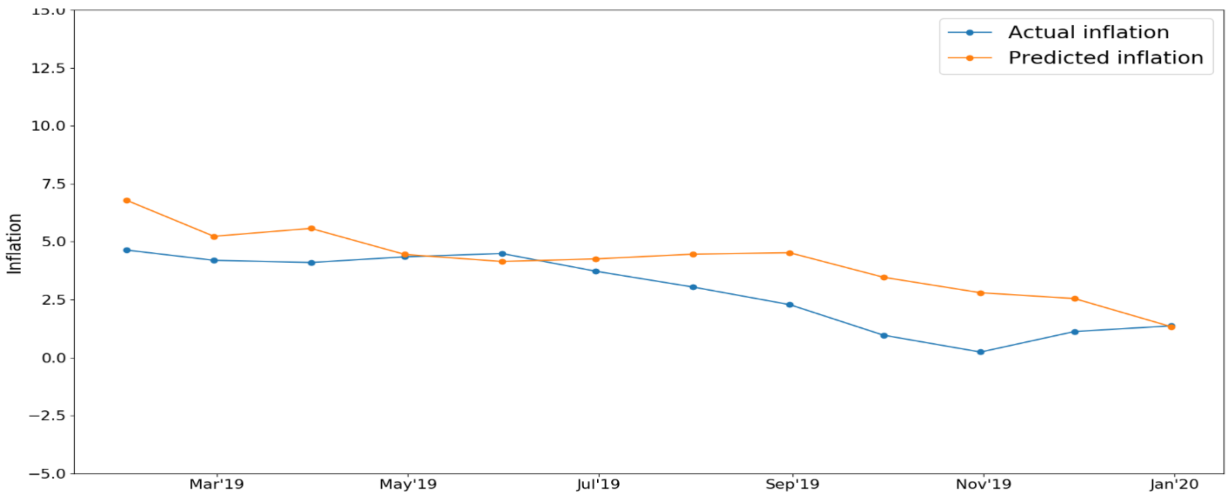
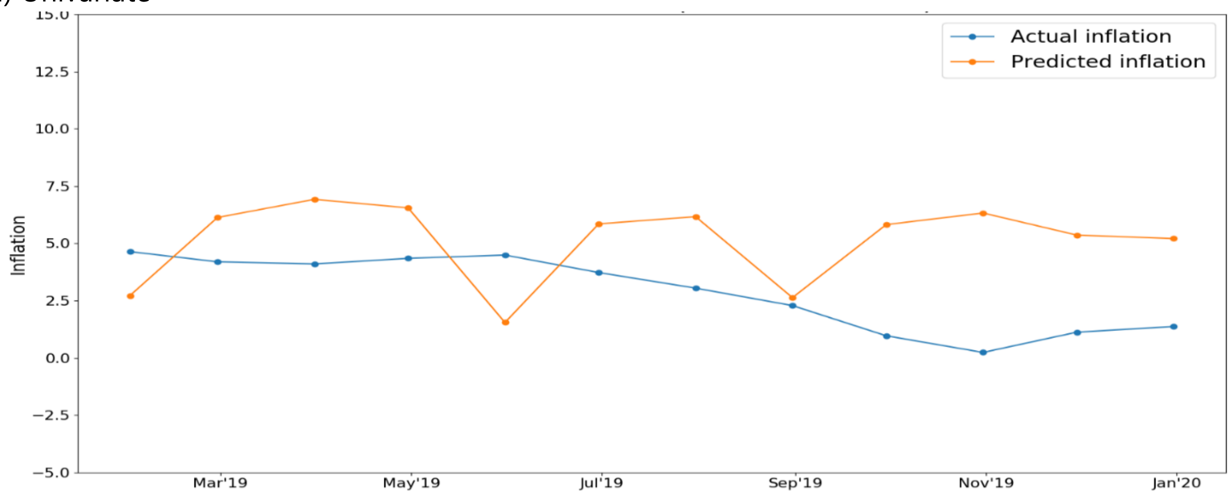


Figure 2.23. LSTM models for Region 10

(i) Univariate



(ii) Multivariate

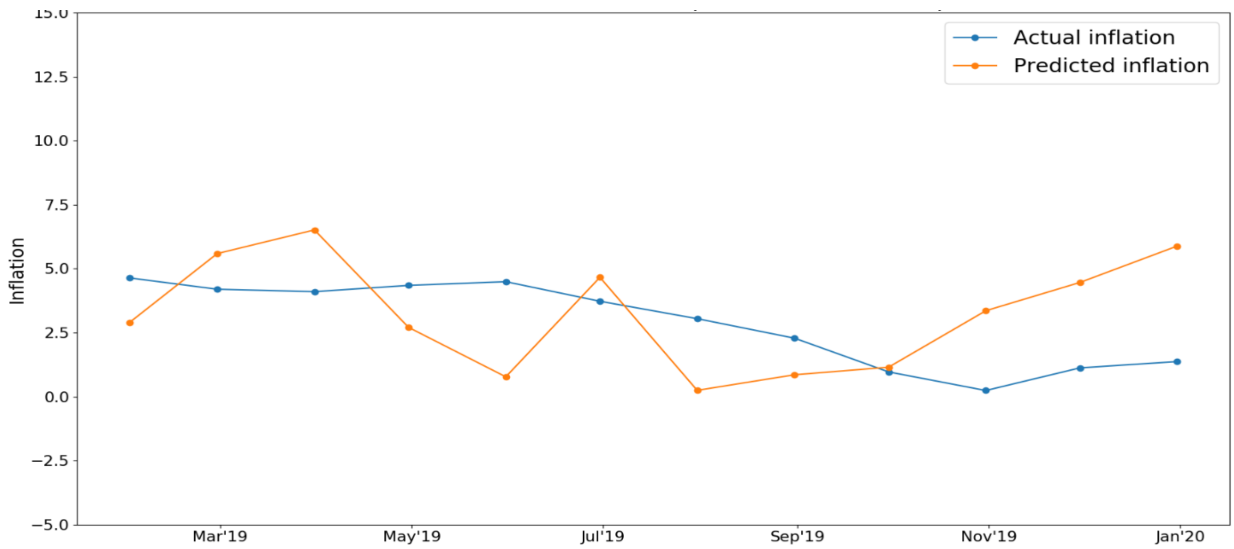
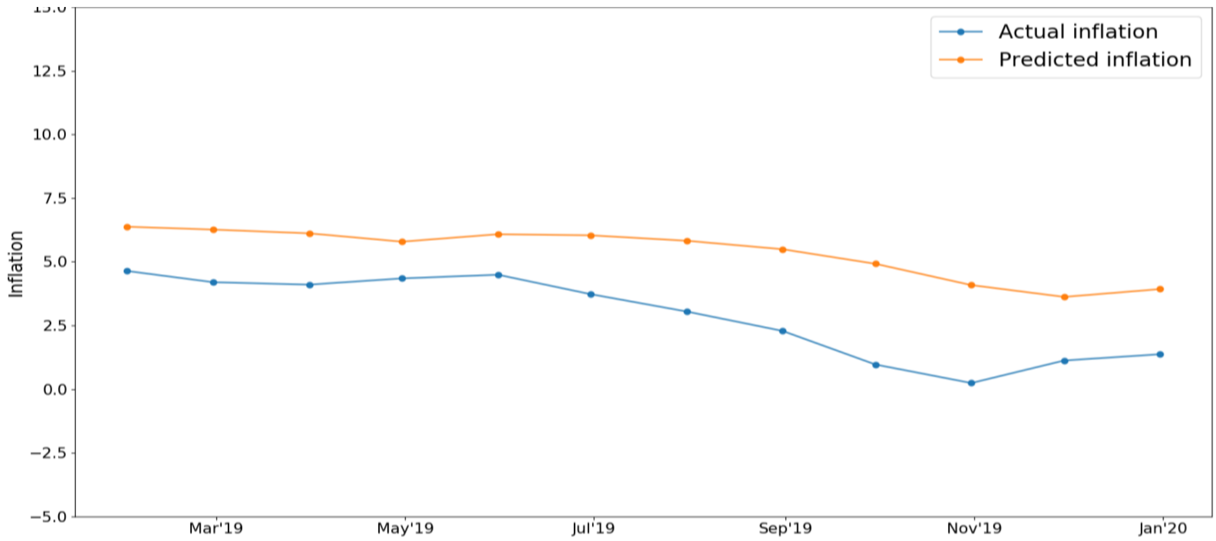


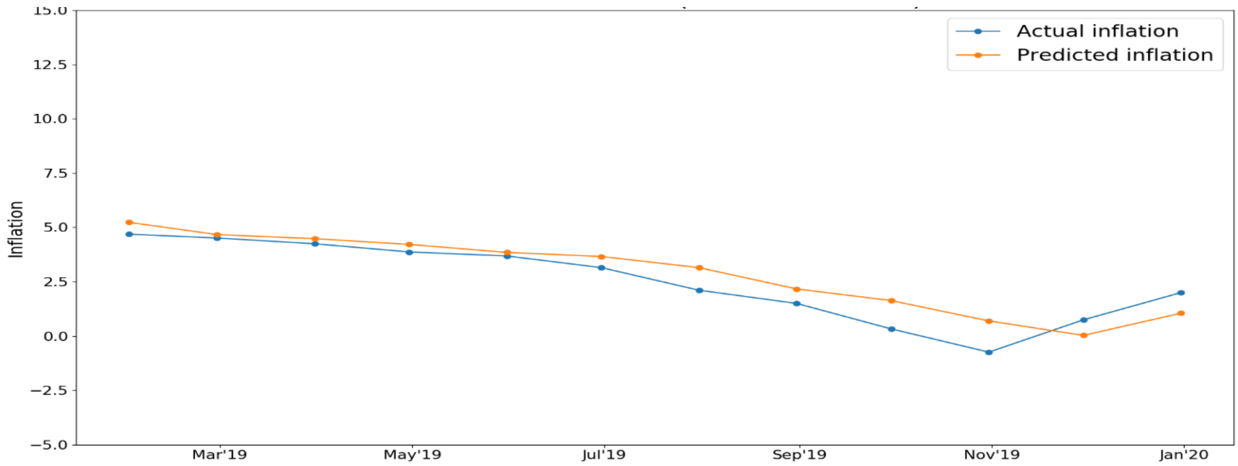
Figure 2.24. ARIMA model for Region 10



G. Region 11

Figure 2.25. SVR models for Region 11

(i) Univariate



(ii) Multivariate

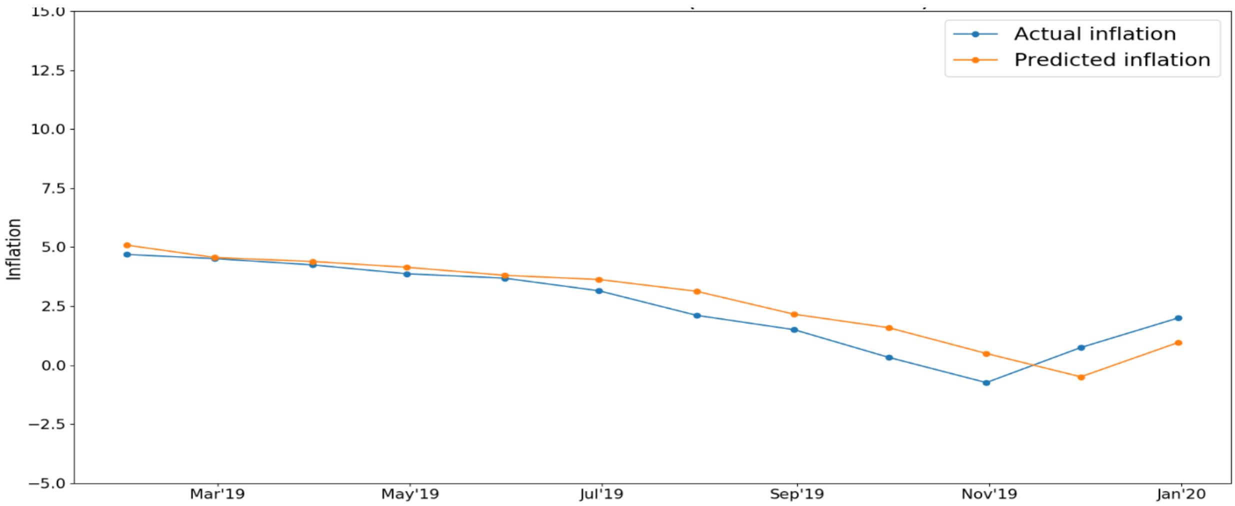
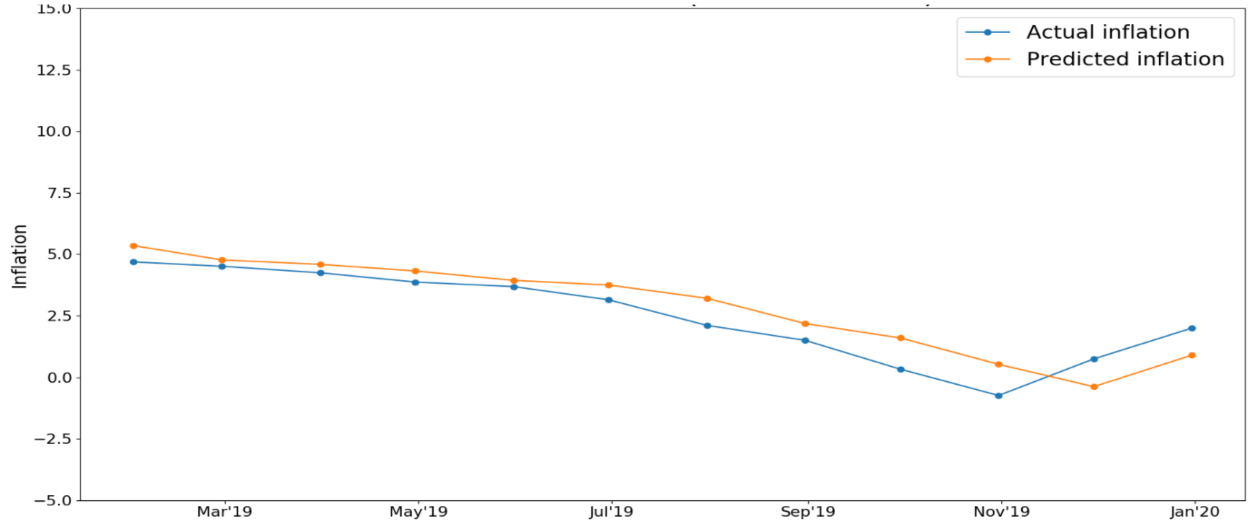


Figure 2.26. ANN models for Region 11

(i) Univariate



(ii) Multivariate

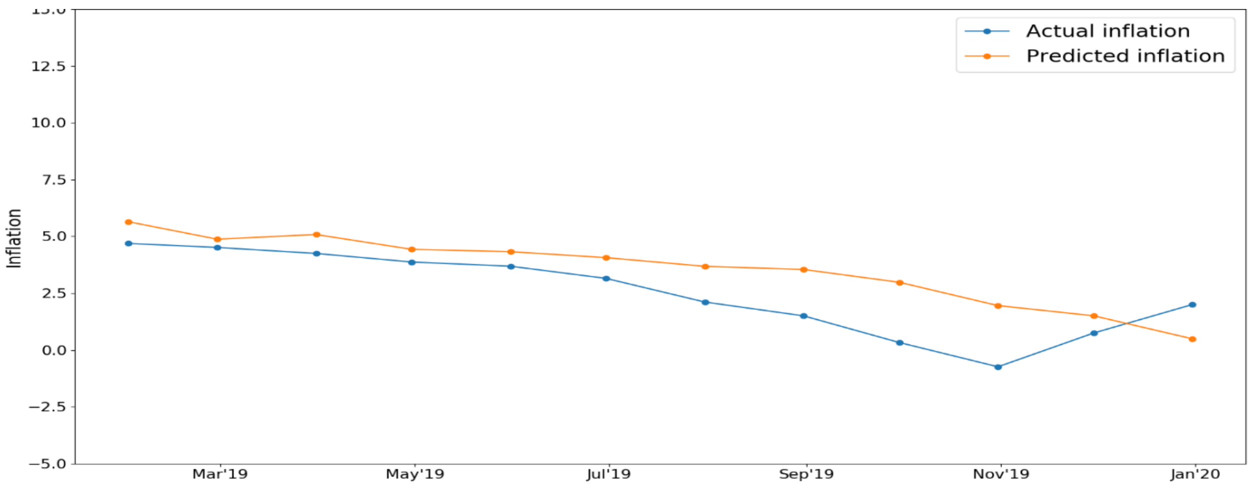
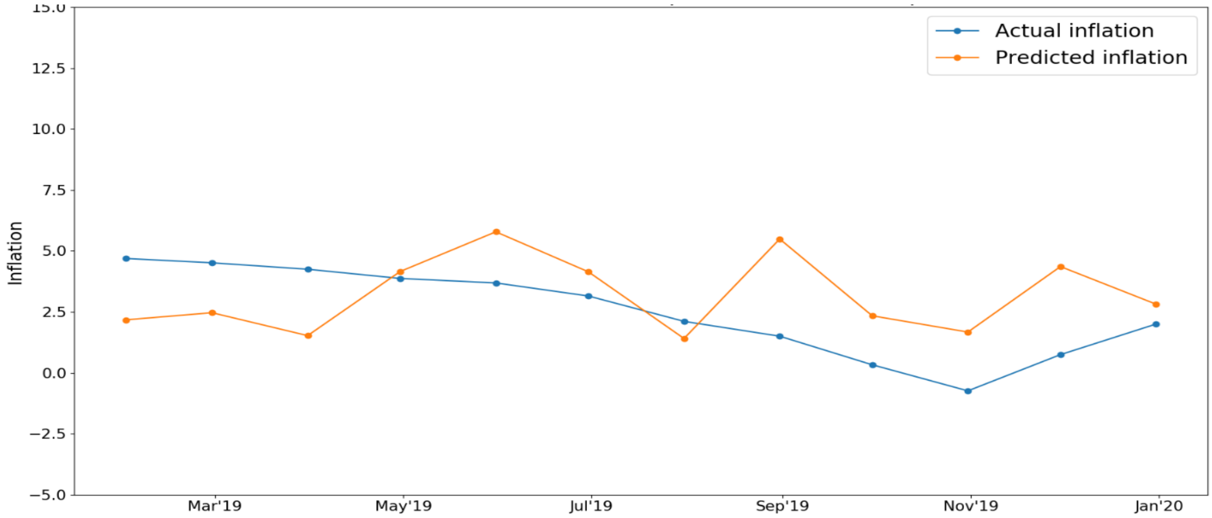


Figure 2.27. LSTM models for Region 11

(i) Univariate



(ii) Multivariate

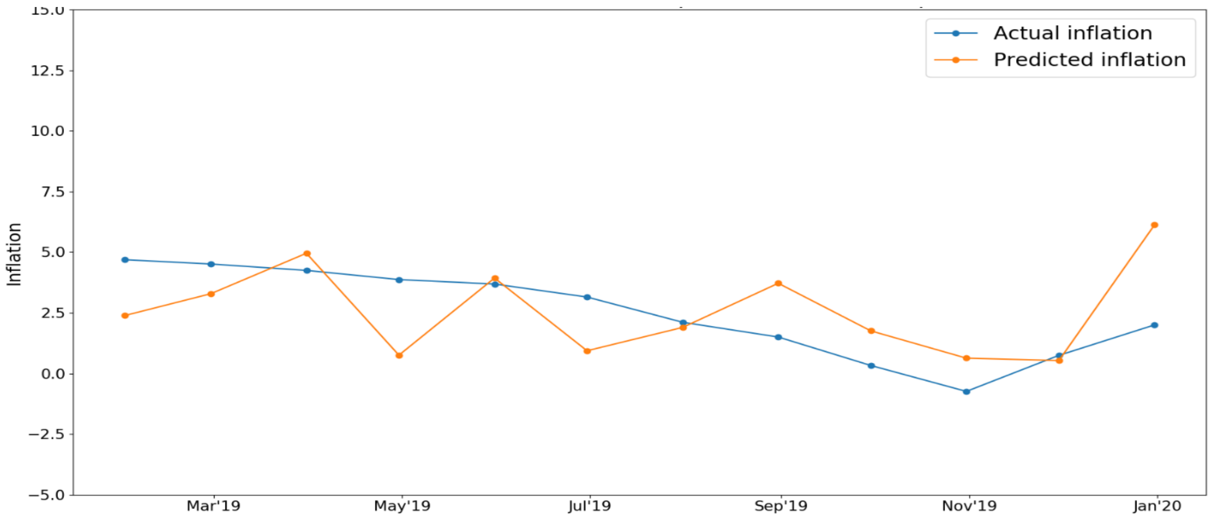
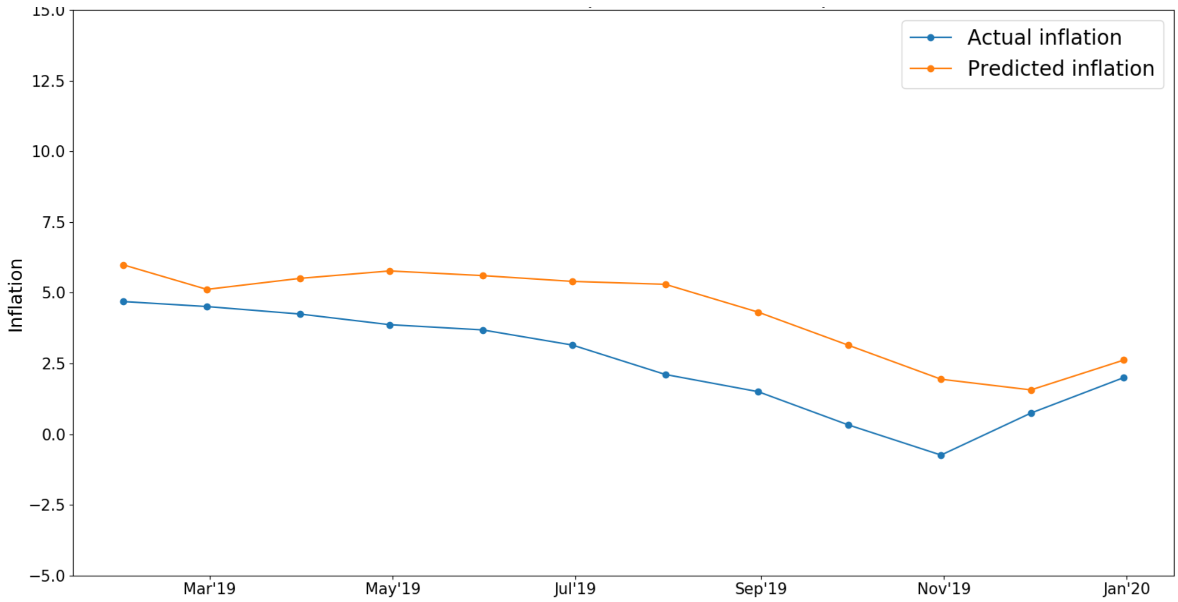


Figure 2.28. ARIMA model for Region 11



BANGKO SENTRAL NG PILIPINAS
BSP Working Paper Series

Scope: The Bangko Sentral ng Pilipinas (BSP) Working Paper Series constitutes studies relevant to central banking, which are conducted by BSP researchers and occasionally, in collaboration with external contributors. The topics may include monetary policy framework and operations, bank supervision, financial markets, macro-financial risks, payments and settlements system, digitalization, big data management and analytics for central banks, central bank communication, central bank governance and legal frameworks, among others.

Peer-reviewed: The BSP working papers are reviewed by the Editorial Committee, led by the Deputy Governor of the Monetary and Economics Sector. Completed working papers are published for the purpose of soliciting comments and discussion for further refinements. The views and opinions expressed are those of the author(s) and do not necessarily reflect those of the BSP.

Submissions: Authors may submit their manuscripts to the following addresses below:

- BSP.Working.Paper@bsp.gov.ph with the subject line that reads BSP Working Paper Series
- The Managing Editor, BSP Working Paper Series, Center for Monetary and Financial Policy, Room 402, 4/F, 5-Storey Building, BSP Main Complex, Malate, Manila

Editorial Guidelines:

- The title page of the manuscript must include the following: title, author's name, abstract describing the main arguments and conclusions of the article, affiliation, corresponding author, 3 - 5 keywords, JEL classification
- Manuscripts must be written in English and in MS Word format, text-aligned with 1.5 line spacing, 1" margins, font Segoe UI, font size 11.
- All graphs, tables, and footnotes must be in font Segoe UI, font size 9.
- Tables must contain only essential data and hence, must be kept to a minimum. Each figure and table must be given an Arabic numeral, followed by a heading.
- All diagrams, charts, and graphs must be referred to as figures and consecutively numbered.
- All figures and tables must be cited in the text.
- Headings and sub-headings must be clearly marked.
- References must be consistent with in-text citations.
- Manuscripts must adopt the Harvard referencing system or APA referencing system.

Authors must include in their submission all graphs and tables in Excel format. They must also ensure that their manuscripts are consistently referenced and free from typographical and presentation errors. The Editorial Committee and the Editorial Staff will not undertake any retyping of manuscripts before publication.

Website: http://www.bsp.gov.ph/publications/regular_wps.asp

List of BSP Working Paper Series

- Available at https://www.bsp.gov.ph/Pages/MediaAndResearch/PublicationsAndReports/BSPWorkingPaperSeries.aspx

No	Author	Title	Date
2020-10	Vidal Marvin C. Gabriel, Dennis M. Bautista, and Cherrie R. Mapa	Forecasting regional inflation in the Philippines using machine learning techniques: A new approach	October 2020
2020-09	Nickson J. Cabote and Justin Ray Angelo J. Fernandez	Distributional Impact of Monetary Policy: Evidence from The Philippines	October 2020
2020-08	Eloisa T. Glindro, Jean Christine A. Armas, V. Bruce J. Tolentino, and Lorna Dela Cruz-Sombe	Heterogenous Impact of Monetary Policy on the Philippine Rural Banking System	September 2020
2020-07	Jean Christine A. Armas and Pamela Kaye A. Tuazon	Revealing investors' sentiment amid COVID-19: the Big Data evidence based on internet searches	September 2020
2020-06	V. Bruce J. Tolentino and Beulah Maria de la Pena	Deregulation and Tariffication At Last: The Saga of Rice Sector Reform in the Philippines	July 2020
2020-05	Eloisa T. Glindro, Hazel C. Parcon-Santos, Faith Christian Q. Cacnio, and Marites B. Oliva	Shifting macroeconomic landscape and the limits of the BSP's pandemic response	June 2020
2020-04	Zernan C. Talabong	Do Prudential Regulations Affect Bank Lending Rates? Insights from Philippine Banks Using an Accounting-Based Approach	June 2020
2020-03	Veronica B. Bayangos, Rafael Augusto D. Cachuela and Fatima Lourdes E. Del Prado	Impact of Extreme Weather Episodes on the Philippine Banking Sector: Evidence Using Branch-Level Supervisory Data	June 2020
2020-02	Joselito R. Basilio and Faith Christian Q. Cacnio	Relative price changes, asymmetric adjustments and aggregate inflation: Evidence from the Philippines	June 2020
2020-01	Eloisa T. Glindro, Hazel C. Parcon-Santos, Faith Christian Q. Cacnio, Marites B. Oliva, and Laura L. Ignacio.	COVID-19 Exit Strategies: How Do We Proceed?	May 2020
2019-04	Charday V. Batac, Eduard Joseph D. Robleza I, Jan Christopher G. Ocampo, and Cherrie F. Ramos	BSPeak: A Text Analysis of BSP's Communications	November 2019
2019-03	Ramon Moreno, Hazel Parcon-Santos, and John Michael Rennie Hallig	A Preliminary Assessment of Drivers of Philippine FX Market Liquidity	October 2019
2019-02	Veronica B. Bayangos and Jeremy L. De Jesus	Have Domestic Prudential Policies Been Effective: Insights from Bank-Level Property Loan Data	March 2019

No	Author	Title	Date
2019-01	Cherry Wyle G. Layaoen and Vernalin Grace F. Domantay	Do Capital Regulations Influence Banks' Holding of "Excess Capital"	March 2019
2018-01	Hazel C. Parcon-Santos	Foreign Exchange Interventions, Capital Outflows, and Financial Vulnerabilities in Selected Asian Emerging Economies	November 2018
2018-01	Roberto S. Mariano, Suleyman Ozmucur, Veronica B. Bayangos, Faith Christian Q. Cacnio, and Marites B. Oliva	Review of the Potential Output and Output Gap Estimation Models of the Bangko Sentral ng Pilipinas	October 2018
2017-01	Veronica B. Bayangos	Capital Flow Measures and Domestic Macro Prudential Policy in Asian Emerging Economies: Have These Been Effective?	June 2017
2016-02	Eufrocinio M. Bernabe, Jr., Hazel C. Parcon-Santos and John Michael Rennie G. Hallig	Spillovers in ASEAN-5 Equity Markets	July 2016
2016-01	Veronica B. Bayangos, Lilia V. Elloso, John Michael Rennie G. Hallig, Jodeth Niña R. Yeung and April Michelle D. Salamatin	The Impact of Foreign Exchange Liberalization Reforms on the Philippine Economy: An Initial Assessment	March 2016
2015-01	Laura L. Ignacio, Hazel C. Parcon-Santos, Teresita B. Deveza, Maria Fatima C. Paule-de Leon, and Jean Christine A. Armas	Reformulating Effective Exchange Rates: Does the Exchange Rate Matter For Trade?	June 2015
2013-01	Francisco G. Dakila, Jr., Veronica B. Bayangos and Laura L. Ignacio	Identifying Sectoral Vulnerabilities and Strengths for the Philippines: A Financial Social Accounting Matrix Approach	July 2013
2012-02	Hazel C. Parcon-Santos and Eufrocinio M. Bernabe, Jr.	The Macroeconomic Effects of Basel III Implementation in the Philippines: A Preliminary Assessment	October 2012
2012-01	Veronica B. Bayangos	Going With Remittances: the Case of the Philippines	July 2012
2010-02	Eloisa T. Glindro and Vic K. Delloro	Identifying and measuring Asset Price Bubbles in the Philippines	June 2010
2010-01	Veronica B. Bayangos and Irene T. Estigoy	A Geometric Price Index for the Philippines: A Preliminary Assessment	March 2010
2009-01	Paul D. McNelis, Eloisa T. Glindro, Ferdinand S. Co, and Francisco G. Dakila, Jr.	Macroeconomic Model for Policy Analysis and Insight (a Dynamic Stochastic General Equilibrium Model for the BSP)	December 2009

No	Author	Title	Date
2008-02	Agnes M. Yap and Cristeta B. Bagsic	Adjustments in the Face of Peso Volatility: Perspective from the Past and Policy Directions	September 2008
2008-01	Haydee L. Ramon	Forecasting the Volatility of Philippine Inflation Using GARCH Models	September 2008
2007-02	Francisco G. Dakila, Jr. and Racquel A. Claveria	Identifying the Determinants of Overseas Filipinos' Remittances: Which Exchange Rate Measure is Most Relevant?	January 2008
2007-01	Paul D. McNelis and Cristeta B. Bagsic	Output Gap Estimation for Inflation Forecasting: The Case of the Philippines	August 2007
2006-02	Cristela Goce-Dakila and Francisco G. Dakila, Jr.	Modeling the Impact of Overseas Filipino Workers Remittances on the Philippine Economy: An Inter-Regional and Economy-Wide Approach	September 2006
2006-01	Cristeta B. Bagsic and Eloisa T. Glindro	Bangko Sentral ng Pilipinas Modernization: A Policy Perpective PDF	August 2006

BSP International Research Conference Volume
Available at
https://www.bsp.gov.ph/Pages/MediaAndResearch/PublicationsAndReports/BSPInternationalResearchConferenceVolume.aspx

- BSP International Research Conference on " Expanding the Boundaries of Central Banking In an Environment of Globalized Finance", 24-25 September 2018
- BSP International Research Conference on "Revisiting Macro-Financial Linkages: Looking Back and Looking Ahead", 20-21 September 2016
- BSP International Research Conference on "The Evolving Role and Limits of Monetary Policy: New Perspectives for Emerging Market Economies", 28-29 October 2014
- BSP International Research Conference on "Contemporary Challenges to Monetary Policy", 28-29 February 2012
- 2010 Central Bank Macroeconomic Modeling Workshop, 19-20 October 2010
- BSP International Research Conference on Remittances, 30-31 March 2009
- Joint BSP-BIS High-Level Conference on Transparency and Communication in Monetary Policy, 01 February 2008