Optimization of the Activation Function for Predicting Inflation Levels to Increase Accuracy Values

Agus Perdana Windarto, Indra Riyana Rahadjeng, Muhammad Noor Hasan Siregar, Muhammad Habib Yuhandri

1STIKOM Tunas Bangsa, Pematangsiiantar, Indonesia
2Universitas Bina Sarana Informatika, Jakarta, Indonesia
3Universitas Graha Nusantara, Padang Sidempuan, Indonesia
4Fakultas Ilmu Komputer, Universitas Putra Indonesia YPTK, Padang, Indonesia

Email: 1. agus.perdana@amiktunasbangsa.ac.id, 2. riyana.irit@bsi.ac.id, 3. noor.siregar@gmail.com, 4. mhabibyuhandri@upiyptk.ac.id

Abstract—This study aims to optimize the backpropagation algorithm by evaluating various activation functions to improve the accuracy of inflation rate predictions. Utilizing historical inflation data, neural network models were constructed and trained with Sigmoid, ReLU, and TanH activation functions. Evaluation using the Mean Squared Error (MSE) metric revealed that the ReLU function provided the most significant performance improvement. The findings indicate that the choice of activation function and neural network architecture significantly influences the model’s ability to predict inflation rates. In the 5-7-1 architecture, the Logsig and ReLU activation functions demonstrated the best performance, with Logsig achieving the lowest MSE (0.00923089) and the highest accuracy (75%) on the test data. These results underscore the importance of selecting appropriate activation functions to enhance prediction accuracy, with ReLU outperforming the other functions in the context of the dataset used. This research concludes that optimizing activation functions in backpropagation is a crucial step in developing more accurate inflation prediction models, contributing significantly to neural network literature and practical economic applications.

Keywords: Backpropagation Algorithms; Artificial Neural Networks; Inflation Prediction; Economic Planning

1. INTRODUCTION

Inflation is one of the most crucial economic indicators, affecting various aspects of social and economic life. An increase in the inflation rate can result in a decrease in people's purchasing power, economic instability and uncertainty in monetary policy [1]–[3]. Therefore, accurate inflation rate predictions are very important for policy makers, economists and market players to make the right decisions. In this context, predictive models that can project inflation levels with high accuracy are urgently needed [4]. Prediction methods in artificial neural networks include a variety of techniques designed to capture and model complex patterns in data. In an effort to predict various phenomena, there are various methods that can be used, both conventional and modern [5], [6]. Conventional methods include simple statistical analysis such as linear regression (7,8), time series methods such as ARIMA (AutoRegressive Integrated Moving Average), and econometric models such as Vector Autoregression (VAR). Meanwhile, modern methods that are increasingly popular include approaches based on Artificial Intelligence[7], [8] and machine learning [9], such as Artificial Neural Networks [8], [10], [11], backpropagation algorithms [3], [12], [13], Random Forest[14], [15], and Support Vector Machines (SVM) based models (8,18,19). In addition, hybrid techniques that combine statistical methods with machine learning models are also widely used to improve prediction accuracy. Each of these methods has its own advantages and disadvantages, as well as varying suitability depending on the type of data and prediction context at hand [16], [17].

Previous research conducted by Atikah and Defri Ahmad [4] in this article, used the Backpropagation method in predicting the inflation rate in Padang City. This method is able to produce fairly accurate forecast results with an MSE value of 0.010689, which shows the possibility of this method as a good alternative for predicting the inflation rate in Padang City. However, it has a weakness in the lack of explanation of potential external factors that could influence the inflation rate in Padang City. Although the article provides a general overview of the fluctuations in the inflation rate over the past few years, there is no in-depth analysis of the external factors that may influence changes in the inflation rate. Meanwhile, in the next research conducted by Yuni Kurniawati and Muhammad Muhajir[18] in this article, the Harmony Search (HS) method was used to optimize the Backpropagation (BP) algorithm in predicting gold prices. This method allows determining relevant input variables and neurons in hidden layers, resulting in more accurate prediction results and faster convergence. However, it still has weaknesses regarding the potential weaknesses or limitations of the Harmony Search (HS) method in optimizing the Backpropagation (BP) algorithm to predict gold prices. And it does not provide an in-depth analysis of factors that can influence the performance of the HS-BP method, such as sensitivity to parameters, stability of convergence, or dependence on training data.

Based on the literature studies that have been described, one approach that is widely used for prediction is artificial neural networks, especially the backpropagation algorithm [19], [20]. Backpropagation is an effective neural network training method for minimizing prediction errors by adjusting network weights based on the resulting output error. Although it has been proven successful in various applications, the performance of backpropagation algorithms is highly dependent on the choice of activation function used in the neural network.
Activation functions play an important role in determining non-linearity and the ability of neural networks to capture complex patterns in data.

Various activation functions such as Sigmoid [22]–[29], ReLU (Rectified Linear Unit) [30]–[36], and TanH (Hyperbolic Tangent) [37]–[46] have different characteristics and can significantly influence model performance [47], [48]. The Sigmoid function, for example, is often used because it produces output in the 0 to 1 range, but tends to suffer from vanishing gradient problems. ReLU, on the other hand, is more efficient in addressing this problem and is frequently used in modern neural networks due to its faster convergence and efficiency in training. The TanH function also offers the advantage of having outputs that are in the -1 to 1 range, allowing for broader data representation [49], [50].

Considering the importance of choosing the right activation function, this research aims to optimize the backpropagation algorithm through a comprehensive evaluation of various activation functions in the context of inflation rate prediction [51], [52]. By using historical inflation data obtained from various trusted sources, this research seeks to determine which activation function provides the best performance in terms of prediction accuracy, Mean Squared Error (MSE) [9], [18], [59]–[64], [22], [23], [53]–[58]. This optimization is expected to make a significant contribution in increasing the accuracy of the inflation prediction model, which can ultimately support more informed and timely economic decision making [12], [65], [66].

Thus, this research not only seeks to improve the technical performance of predictive models, but also aims to provide practical contributions for policy makers and economic actors in anticipating changes in inflation rates. It is hoped that the findings of this research will enrich the literature regarding the application of artificial neural networks in the economic field and open up opportunities for further development in predictive model optimization in various other domains.

2. RESEARCH METHODOLOGY

2.1 Research design

This research uses a quantitative approach with an experimental design to test the effect of various activation functions on the backpropagation algorithm in predicting the inflation rate. Artificial neural network models will be built and tested using historical inflation datasets, with a focus on improving prediction accuracy through activation function optimization. Figure 1 shows the research design to complete this research.

The following is an explanation in Figure 1 regarding this research designed to optimize the backpropagation algorithm with various activation functions in predicting the inflation rate. The following flow diagram explains the main steps in the research process:

1. Dataset: The first step is to collect the dataset that will be used in the research. This dataset contains historical inflation rate data which will be divided into two parts: training data and testing data.
2. Data Division: The dataset is divided into two parts: training data and testing data. Training data is used to train the neural network model, while testing data is used to test the performance of the trained model.
3. Normalization: Data that has been divided is then normalized. Normalization is the process of scaling data so that it falls within a certain range, usually between 0 and 1. This is important to ensure that the neural network model can learn effectively from the data.
4. Architectural Model Selection: After the data is normalized, the neural network architectural model is randomly selected. This includes selecting the number of hidden layers, the number of neurons in each layer, and other parameters to be used in the model.
5. Training: The selected neural network model is then trained using the training data. The training process uses a backpropagation algorithm with momentum-enhanced gradient descent and adaptive learning rate. Different activation functions were also applied in each experiment to see their effect on model performance.
6. Testing: After training, the model is tested using test data. This test aims to evaluate the model's performance in predicting inflation rates based on data that has never been seen before.

7. Evaluation of Training and Testing Results: The results of training and testing are analyzed to measure prediction accuracy. Evaluation metrics such as Mean Squared Error (MSE) are used to determine how well the model predicts the inflation rate.

8. Results Analysis: The results of the model evaluation are then analyzed to understand the influence of the various activation functions and backpropagation parameters used. This analysis helps determine the most optimal model configuration in predicting the inflation rate.

The diagram as a whole depicts the process flow from the beginning of data collection to the analysis of the final results, ensuring that every step is taken to optimize inflation predictions using artificial neural networks.

### 2.2 Data collection

The data collected includes monthly inflation rates, which will be further processed for use in artificial neural network models. This research data is historical data on inflation rates in Indonesia obtained from official sources such as BPS (Central Statistics Agency) of Indonesia. The sample used is monthly inflation data for the last 7 years. This data will be divided into training data (50%) and testing data (50%) to validate the model developed.

The following is a sample of data used in this research:

<table>
<thead>
<tr>
<th>Month</th>
<th>2017</th>
<th>2018</th>
<th>2019</th>
<th>2020</th>
<th>2021</th>
<th>2022</th>
<th>2023</th>
</tr>
</thead>
<tbody>
<tr>
<td>January</td>
<td>3.49</td>
<td>3.25</td>
<td>2.82</td>
<td>2.68</td>
<td>1.55</td>
<td>2.18</td>
<td>5.28</td>
</tr>
<tr>
<td>February</td>
<td>3.83</td>
<td>3.18</td>
<td>2.57</td>
<td>2.98</td>
<td>1.38</td>
<td>2.06</td>
<td>5.47</td>
</tr>
<tr>
<td>March</td>
<td>3.61</td>
<td>3.40</td>
<td>2.48</td>
<td>2.96</td>
<td>1.37</td>
<td>2.64</td>
<td>4.97</td>
</tr>
<tr>
<td>April</td>
<td>4.17</td>
<td>3.41</td>
<td>2.83</td>
<td>2.67</td>
<td>1.42</td>
<td>3.47</td>
<td>4.33</td>
</tr>
<tr>
<td>May</td>
<td>4.33</td>
<td>3.23</td>
<td>3.32</td>
<td>2.19</td>
<td>1.68</td>
<td>3.55</td>
<td>4.00</td>
</tr>
<tr>
<td>June</td>
<td>4.37</td>
<td>3.12</td>
<td>3.28</td>
<td>1.96</td>
<td>1.33</td>
<td>4.35</td>
<td>3.52</td>
</tr>
<tr>
<td>July</td>
<td>3.88</td>
<td>3.18</td>
<td>3.32</td>
<td>1.54</td>
<td>1.52</td>
<td>4.94</td>
<td>3.08</td>
</tr>
<tr>
<td>August</td>
<td>3.82</td>
<td>3.20</td>
<td>3.49</td>
<td>1.32</td>
<td>1.59</td>
<td>4.69</td>
<td>3.27</td>
</tr>
<tr>
<td>September</td>
<td>3.72</td>
<td>2.88</td>
<td>3.39</td>
<td>1.42</td>
<td>1.60</td>
<td>5.95</td>
<td>2.28</td>
</tr>
<tr>
<td>October</td>
<td>3.58</td>
<td>3.16</td>
<td>3.13</td>
<td>1.44</td>
<td>1.66</td>
<td>5.71</td>
<td>2.56</td>
</tr>
<tr>
<td>November</td>
<td>3.30</td>
<td>3.23</td>
<td>3.00</td>
<td>1.59</td>
<td>1.75</td>
<td>5.42</td>
<td>2.86</td>
</tr>
<tr>
<td>December</td>
<td>3.61</td>
<td>3.13</td>
<td>2.72</td>
<td>1.68</td>
<td>1.87</td>
<td>5.51</td>
<td>2.61</td>
</tr>
</tbody>
</table>

### 2.3 Research Procedure

This research was carried out through several stages as follows:

1. Data Pre-processing: The collected inflation data will be cleaned from missing values and outliers. The data is then normalized to ensure optimal performance of the neural network model.
2. Model Development: Artificial neural network models will be developed using various activation functions, including Sigmoid, ReLU, and TanH. Each model will be trained using the backpropagation algorithm.
3. Model Training: Training data is used to train the model using the backpropagation technique. Parameters such as learning rate and number of epochs will be set to ensure model convergence.
4. Model Evaluation: The trained model will be evaluated using testing data. The evaluation metrics used include prediction accuracy, Mean Squared Error (MSE).
5. Analysis and Interpretation: The evaluation results will be analyzed to determine which activation function provides the best performance in the context of inflation prediction. This analysis will be interpreted to provide insight into the effectiveness of each activation function.

### 3. RESULTS AND DISCUSSION

#### 3.1 Results

The dataset used in this research consists of historical data on monthly inflation rates for the last 7 years, namely 2017-2023 based on BPS data. This data is divided into training data (50%) and testing data (50%). After going through the normalization process, this data is ready to be used for training and testing artificial neural network models.

#### 3.1.1 Model Training Results

The artificial neural network model was trained using a backpropagation algorithm with three different activation functions: Sigmoid, ReLU, and TanH. Training parameters include learning rate, momentum, and number of epochs. The following is an image of the training results for each activation function with a 1 hidden layer architectural model:
Figure 2. Architectural Drawing Model 1 Hidden layer

Figure 2 provides training details for three neural network models with a 5-7-1 architecture using different activation functions: Sigmoid, Tanh, and ReLU. Each model employs Gradient Descent with Momentum and Adaptive Learning Rate (LR) as the training algorithm and Mean Squared Error (MSE) as the performance metric. The model with the Sigmoid activation function converges in 89 epochs, achieving a final MSE of 0.00976321 and a gradient of 0.0326, indicating efficient convergence towards the minimum error. The quick convergence time and the low number of validation checks (6) suggest that the model has effectively learned the data patterns. In comparison, the model with the Tanh activation function takes slightly longer, requiring 92 epochs to reach a similar performance level with an MSE of 0.009804 and a gradient of 0.0406. This marginally higher gradient indicates a slower rate of convergence, but the number of validation checks and convergence time are comparable to the Sigmoid model, demonstrating that Tanh is also effective but needs a bit more training time.

The model using the ReLU activation function, however, requires significantly more epochs (119) to meet the performance goal, achieving an MSE of 0.009911 and a much lower gradient of 0.00476, which suggests that the model has nearly reached a plateau in error reduction. The absence of validation checks in the ReLU model implies that it has not encountered validation stop criteria as frequently as the other models, which might be attributed to ReLU’s tendency towards dead neurons if not properly regularized. Comparing these activation functions, the Sigmoid and Tanh functions demonstrate faster convergence and slightly better performance metrics compared to ReLU in this 5-7-1 architecture.

The Sigmoid and Tanh are more efficient in terms of training time and provide marginally better performance, making them more suitable for this neural network architecture. In contrast, ReLU, while achieving comparable results, requires more epochs and has a slower convergence rate, indicating the need for further optimization in learning rate or regularization techniques. Therefore, for this particular model architecture, Sigmoid and Tanh are recommended for faster and more efficient training, while ReLU might benefit from additional tuning to improve its training efficiency.

Figure 3. Architectural Drawing Model 2 Hidden layer
Figure 3 illustrates the training details for three neural network models with a 5-7-3-1 architecture using different activation functions: Sigmoid, Tanh, and ReLU. Each model employs Gradient Descent with Momentum and Adaptive Learning Rate (LR) as the training algorithm and Mean Squared Error (MSE) as the performance metric. The model with the Sigmoid activation function converges in 117 epochs, achieving a final MSE of 0.009914 and a low gradient of 0.00673. This indicates good convergence towards the minimum error, with six validation checks, suggesting that the model has effectively learned the data patterns, although it required more epochs compared to simpler architectures.

In contrast, the model with the Tanh activation function achieves faster convergence in just 71 epochs, with an MSE of 0.009898 and a higher gradient of 0.0248. This higher gradient indicates a faster initial learning rate, stabilizing as training progresses. The Tanh model also required six validation checks, demonstrating efficient learning within a shorter training time. The ReLU activation function, however, requires significantly more epochs (229) to meet the performance goal, with an MSE of 0.009999 and a moderate gradient of 0.0129. Despite eventually reaching the performance goal, the ReLU model's need for more training time and the same number of validation checks as the other models suggest potential challenges, such as susceptibility to dead neurons and the necessity for further fine-tuning.

Comparing these activation functions, Tanh stands out with the fastest convergence and effective performance in the complex 5-7-3-1 architecture. Sigmoid, although requiring more epochs, also shows good convergence and stable performance. ReLU, while achieving similar MSE, requires a significantly higher number of epochs and additional optimization to enhance its training efficiency. Therefore, for complex neural network architectures with multiple hidden layers, Tanh and Sigmoid are recommended for their efficiency and effective learning. ReLU, though capable, may benefit from further tuning to improve its performance and training speed. Selecting the appropriate activation function should consider the balance between model complexity and training efficiency to achieve optimal results. In Figure 4 is a picture of the training data curve from the six training sessions carried out.

Figure 4. Draw the Curves of the six different Model architectures

Figure 4 presents the training curves for six different combinations of neural network architectures and activation functions. In the 5-7-1 architecture with the Sigmoid activation function, the curve shows a steady decrease in the Mean Squared Error (MSE), reaching its best performance at 89 epochs with an MSE of 0.00976321. The Sigmoid function appears to work well, demonstrating effective convergence without clear signs of overfitting or underfitting. Similarly, the 5-7-1 architecture with the Tanh activation function shows a similar pattern of decreasing MSE, achieving its best performance at 92 epochs with an MSE of 0.0098026. This indicates that Tanh also performs well in this architecture, although with a slightly higher MSE compared to Sigmoid. For the same architecture, the ReLU activation function requires more epochs to converge, with the best performance at 119 epochs and an MSE of 0.009914. While ReLU eventually achieves a significant reduction in MSE, this curve indicates that ReLU takes longer to learn in this architecture. In the more complex 5-7-3-1 architecture with two hidden layers, the Sigmoid function requires up to 117 epochs to converge, with the best MSE at 0.0099144. This curve shows a steady decrease in MSE, indicating that the Sigmoid function can handle more complex architectures, albeit requiring more time.

Interestingly, the Tanh function in the 5-7-3-1 architecture achieves faster convergence compared to the single-layer architecture, with the best performance at 71 epochs and an MSE of 0.0098752. This suggests that
Tanh might be more effective in handling more complex architectures. On the other hand, the ReLU function shows very slow convergence in this architecture, needing up to 229 epochs to reach its best MSE of 0.009991. This suggests possible overfitting or the need for further adjustments in training parameters such as the learning rate or regularization.

From this analysis, it can be concluded that the Sigmoid and Tanh activation functions perform well with a steady decrease in MSE in both single-layer and multi-layer architectures. The Tanh function seems to be more effective in more complex architectures, while the ReLU function requires more epochs and parameter adjustments to achieve optimal performance. Therefore, the selection of activation functions and neural network architectures should be tailored to the complexity of the data and the specific needs of the inflation prediction application. The following is Table 2 as a presentation of the values.

### Table 2. Recap of Research Results

<table>
<thead>
<tr>
<th>No</th>
<th>Architecture</th>
<th>Activation Function</th>
<th>Train Epoch</th>
<th>Train MSE</th>
<th>Accuracy</th>
<th>Test Epoch</th>
<th>Test MSE</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5-7-1</td>
<td>Logsig</td>
<td>89</td>
<td>0.00976321</td>
<td>75</td>
<td>50</td>
<td>0.00923089</td>
<td>75</td>
</tr>
<tr>
<td>2</td>
<td>5-7-1</td>
<td>Tanh</td>
<td>92</td>
<td>0.00980454</td>
<td>58</td>
<td>63</td>
<td>0.00983263</td>
<td>75</td>
</tr>
<tr>
<td>3</td>
<td>5-7-1</td>
<td>ReLU</td>
<td>119</td>
<td>0.00991166</td>
<td>50</td>
<td>105</td>
<td>0.00930309</td>
<td>75</td>
</tr>
<tr>
<td>4</td>
<td>5-7-3-1</td>
<td>Logsig</td>
<td>117</td>
<td>0.00991495</td>
<td>67</td>
<td>91</td>
<td>0.00966866</td>
<td>67</td>
</tr>
<tr>
<td>5</td>
<td>5-7-3-1</td>
<td>Tanh</td>
<td>71</td>
<td>0.00987563</td>
<td>67</td>
<td>46</td>
<td>0.00994185</td>
<td>75</td>
</tr>
<tr>
<td>6</td>
<td>5-7-3-1</td>
<td>ReLU</td>
<td>229</td>
<td>0.00999145</td>
<td>50</td>
<td>134</td>
<td>0.00978677</td>
<td>67</td>
</tr>
</tbody>
</table>

The table summarizes (Table 2) the performance metrics for various neural network architectures and activation functions, highlighting the number of epochs, Mean Squared Error (MSE), and accuracy for both training and testing phases. In the 5-7-1 architecture with the Logsig activation function, the model demonstrates consistent performance, achieving 75% accuracy and a low MSE of 0.00976321 during training, and maintaining this performance with 75% accuracy and an MSE of 0.00923089 during testing. This indicates effective convergence and good generalization. The Tanh activation function in the same architecture requires slightly more epochs to converge (92 epochs), showing lower training accuracy at 58%. However, its testing accuracy improves to 75%, with a stable MSE, suggesting that Tanh might need more time to learn but can perform well once adequately trained.

The ReLU function in the 5-7-1 architecture requires even more epochs to converge (119 epochs) and shows lower training accuracy at 50%. During testing, its performance improves significantly to 75% accuracy with a decreased MSE of 0.00930309, indicating that ReLU benefits from extended training despite slower initial convergence. In the more complex 5-7-3-1 architecture, the Logsig function shows moderate performance with an accuracy of 67% for both training and testing, and a relatively low MSE, indicating stable but not optimal performance in handling increased complexity.

The Tanh function, on the other hand, adapts well to the more complex architecture, converging faster (71 epochs) and achieving improved testing accuracy of 75%, with a stable MSE. This suggests that Tanh is effective in complex models. ReLU, however, requires a significantly higher number of epochs to converge (229 epochs) in the 5-7-3-1 architecture, with lower training accuracy at 50%. During testing, its accuracy increases to 67%, but the MSE remains high, indicating potential overfitting or the need for further parameter optimization.

The Logsig and Tanh activation functions demonstrate good generalization and stability across both simple and complex architectures. The Logsig function provides quick convergence and consistent performance, while Tanh shows adaptability to complexity with improved accuracy during testing. ReLU, though effective after extended training, requires careful parameter tuning and longer training periods, especially in more complex models. Therefore, selecting the appropriate activation function and neural network architecture should balance training time and model complexity to achieve optimal performance.

### 3.1.2 Model Testing Results

Training, the model is tested using test data. The test results show that the Sigmoid activation function provides the most accurate prediction results in the 1 hidden layer architecture and in the 2 hidden layers in the Tanh activation function.

### 3.2 Discussion

#### 3.2.1 Activation Function Performance Analysis

This research examines the performance of various activation functions on two different artificial neural network architectures in predicting inflation rates. Based on the results obtained, there are several key findings that can be discussed further:

- **Logsig Activation Function**: On 5-7-1 architecture, Logsig shows quite good performance with lower MSE on test data (0.00923089) and high accuracy (75%). This shows that Logsig is able to handle data well on a simple architecture, providing accurate predictions with fairly fast convergence (89 epochs for training). And on the 5-7-3-1 architecture, Logsig experienced a decrease in performance with a higher MSE (0.00966866) and...
accuracy decreased to 67%. This decrease may be due to increased architectural complexity that is not well handled by the Logsig activation function.

b) Tanh Activation Function: On the 5-7-1 architecture, Tanh shows an increase in accuracy from 58% on training to 75% on testing, although the MSE remains quite high (0.00983263). This shows that Tanh can provide good accuracy but may require more epochs to achieve optimal convergence. On the 5-7-3-1 architecture, Tanh achieved high accuracy (75%) with a lower number of epochs (71 epochs for training and 46 epochs for testing), but with a slightly higher MSE (0.00994185). This shows that Tanh is able to handle more complex architectures well, but still requires better tuning to reduce errors.

c) ReLU Activation Function: On the 5-7-1 architecture, ReLU shows quite stable performance with a relatively low MSE in the test (0.00930309) and high accuracy (75%), although it requires more epochs to achieve that result (119 epochs for training and 105 epochs for testing). This indicates that ReLU has good potential to produce accurate predictions but may require more training time. On the 5-7-3-1 architecture, ReLU requires a very high number of epochs (229) for training, but the test results show a fairly low MSE (0.00978677) with an accuracy of 67%. This indicates possible overfitting or the need for further adjustment of training parameters.

### 3.2.2 Implications of Research Results

These findings have several important implications:

a) Activation Function Selection: ReLU and Logsig activation functions proved effective in a simpler architecture (5-7-1), while Tanh showed more consistent performance in a more complex architecture (5-7-3-1).

b) Parameter Optimization: These results show the importance of optimizing training parameters such as the number of epochs, learning rate, and using regularization techniques to avoid overfitting, especially on more complex architectures.

c) Real World Applications: In the context of inflation rate prediction, more accurate models can help policymakers and economic analysts make more informed decisions. ReLU, with its ability to handle a wider range of data without suffering from vanishing gradients, appears to be a good choice for this application.

### 4. CONCLUSION

The study demonstrates that the choice of activation function and neural network architecture significantly impacts the performance of models predicting inflation rates. The ReLU and Logsig activation functions performed well in the 5-7-1 architecture, while the Tanh function showed consistency in the more complex 5-7-3-1 architecture, despite variations in MSE. Optimizing training parameters and further exploring other activation functions can enhance the accuracy and stability of inflation prediction models. This research highlights the importance of selecting the right activation function and optimizing parameters in developing more accurate predictive models, which can support better and more timely economic decision-making. The results also provide significant contributions to neural network literature and open avenues for further advancements in predictive model optimization across various domains.

### REFERENCES


