Accredited SINTA 2 Ranking
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online at: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 8 No. 4 (2024) 466 - 478 e-ISSN: 2580-0760

Comparative Analysis of Gradient Descent Learning Algorithms in Artificial Neural Networks for Forecasting Indonesian Rice Prices

Rica Ramadana[1], Agus Perdana Windarto[2*], Dedi Suhendro[3]
[1,2]Information Systems Study Program, STIKOM Tunas Bangsa, Pematangsiantar, Indonesia
[3]Computerized Accounting Study Program, STIKOM Tunas Bangsa, Pematangsiantar, Indonesia
[1]dhanadhana0312@gmail.com, [2]agus.perdana@amiktunasbangsa.ac.id, [3]dedi.su@amiktunasbangsa.ac.id

Abstract

Artificial Neural Networks (ANN) are a field of computer science that mimics the way the human brain processes data. ANNs can be used to classify, estimate, predict, or simulate new data from similar sources. The commonly used algorithm for prediction in ANN is Backpropagation, which yields high accuracy but tends to be slow during the training process and is prone to local minima. To address these issues, appropriate parameters are needed in the Backpropagation training process, such as an optimal learning function. The aim of this study is to evaluate and compare various learning functions within the Backpropagation algorithm to determine the best one for prediction cases. The learning functions evaluated include Gradient Descent Backpropagation (traingd), Gradient Descent with Adaptive Learning Rate (traingda), and Gradient Descent with Momentum and Adaptive Learning Rate (traingdx). The dataset used is the average wholesale rice price in Indonesia, obtained from the Central Statistics Agency (BPS) website. The evaluation results show that the traingdx learning function with a 5-5-1 architecture model achieves the highest accuracy of 83.33%, representing an 8.3% improvement over the traingd and traingda learning functions, which both achieved a maximum accuracy of 75%. Based on this study, it can be concluded that using various learning functions in Backpropagation yields better accuracy compared to standard Backpropagation.

Keywords: artificial neural network; backpropagation; learning function; accuracy

How to Cite: Rica Ramadana, Agus Perdana Windarto, and Dedi Suhendro, "Comparative Analysis of Gradient Descent Learning Algorithms in Artificial Neural Networks for Forecasting Indonesian Rice Prices", J. RESTI (Rekayasa Sist. Teknol. Inf.), vol. 8, no. 4, pp. 466 - 478, Aug. 2024.
DOI: https://doi.org/10.29207/resti.v8i4.5822

1. Introduction

Artificial Neural Networks (ANN) is a subfield of computer science and technology that employs the principles of neural organisation [1], [2], allowing for rapid problem-solving compared to other techniques. They possess an added capability to benefit from minimal empirical data and can solve complex problems [3]. ANNs are designed for the evaluation of different parameters and can classify, estimate, predict, or simulate new data from comparable or similar sources [4]. One of the most popular ANN models is Backpropagation [5], [6], an algorithm used to systematically train multi-layer networks [7] with the objective of identifying optimal weights by minimizing the model's loss function [8]. In the learning process using Backpropagation, the learning function is crucial for achieving optimal results [9]. Backpropagation offers various learning functions for weights in Matlab, such as Gradient Descent with Momentum (traingdm), Gradient Descent with Adaptive Learning Rate (traingda), Resilient Backpropagation (traingrp), and Gradient Descent with Momentum and Adaptive Learning Rate (traingdx), among others [10]. This algorithm utilizes Gradient Descent (GD) as the standard training algorithm to optimize network performance by finding the local minimum of the function [7]. GD is employed to determine the parameter values (coefficients) of a function that can maximally reduce the loss function [11].

Several related studies utilizing the GD function include the research by Pragana et al. on predicting the average price of rice at the milling level by quality [12], the study by Setiana et al. on predicting goat meat production in Indonesia [13], and the research conducted by Natasya et al. on predicting the wholesale price of rice in Indonesia [14]. In general, these three studies [14] found that the traingd algorithm demonstrated high accuracy but had relatively long

Received: 29-05-2024 | Accepted: 02-08-2024 | Published Online: 07-08-2024

training times [7], [15], [16] and a tendency to get stuck in local minima [17]-[19], which are not optimal [10], [20]. Therefore, alternative learning functions are needed to optimize the performance of Backpropagation [21] and to process the data without compromising the quality of the results [22]. Several variations of the GD algorithm aim to slow down learning when approaching local minima or speed it up as needed to determine the best learning rate at each step [23]. One such variation is GD with Adaptive Learning Rate (traingda), which is used in stochastic optimization algorithms to automatically adjust the learning rate during training based on data characteristics and optimization processes. This allows the algorithm to dynamically change the step size taken toward the optimal solution, thereby improving efficiency and accuracy [23]. Related studies discussing training using gradient descent with adaptive learning rate include the research by Wanto et al. (2020) [7] on analyzing gradient descent with a combination of activation functions in ANN to find the best accuracy. Their results indicated that the traingda function was the best GD algorithm, achieving an accuracy rate of 91% in 3 seconds. Additionally, the study by Ridho et al. (2023) [10] examined the optimization of the traingda, traingdx, and traingdm learning functions for predicting coffee exports to major destination countries. The results indicated that the traingda function successfully enhanced the performance of the neural network in predicting coffee exports, with better optimality, performing 143 iterations and achieving an accuracy rate of 83%. Another variation of GD is GD with Momentum and Adaptive Learning Rate (traingdx), which is useful for accelerating convergence and overcoming local minima [11]. This was demonstrated in the study by Wanto et al. (2018) [22], which compared the standard gradient descent function (traingd) with GD with Momentum and Adaptive Learning Rate (traingdx). The results showed that GD with Momentum and Adaptive Learning Rate was faster and more effective for prediction compared to standard gradient descent (traingd), without significantly affecting the quality of the results, achieving an accuracy rate of 94% and an MSE of 0.0008658637. This study aims to compare the standard GD function (traingd) with the GD function with Adaptive Learning Rate (traingda) and GD with Momentum and Adaptive Learning Rate (traingdx) using the average wholesale rice prices in Indonesia as the test data. Since each previous study claims that the constructed architecture is optimal with the highest accuracy rate [24], the results of this research are expected to develop an accurate architectural model for predicting average wholesale rice prices by comparing various learning functions of the Backpropagation Artificial Neural Network.

2. Research Methods

The dataset used in this study comprises secondary data (data collected indirectly by the researcher, such as data obtained from websites, etc.). The dataset collected in this study consists of the average wholesale rice prices from 2017 to 2023. The research methodology process in this study involves several stages. The stages conducted in this research are illustrated in Figure 1.

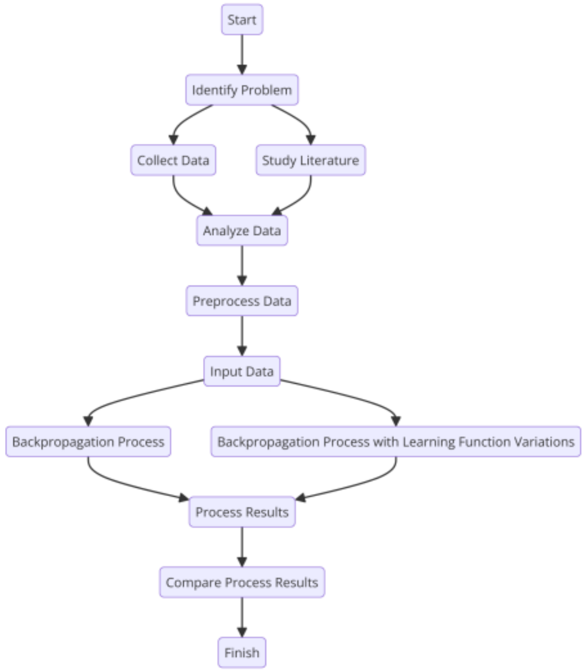

Figure 1. Research Stages

The research methodology begins with the initiation of the project, where the initial step is to identify the research problem or define the research question. Following this, necessary data is collected, which involves sourcing relevant datasets essential for the analysis. Concurrently, a thorough review of existing literature is conducted to understand the current state of knowledge and identify gaps that the research aims to address. Once the data is collected, it undergoes an initial analysis to comprehend its structure, characteristics, and any underlying patterns. This step is followed by data preprocessing, where the data is cleaned and organized to prepare it for modelling, which may involve handling missing values and normalizing the data. The preprocessed data is then fed into the model. At this stage, the research splits into two parallel processes: one involves training the neural network using the standard Backpropagation algorithm, and the other involves training the network using different learning function variations within the Backpropagation algorithm. These processes aim to adjust the model's weights to minimize error through iterative training. After the training processes, the results are processed and analyzed to evaluate the performance of each model in terms of accuracy and error metrics. A comparative analysis is then conducted between the results of the standard Backpropagation process and the Backpropagation process with learning function variations. This comparison helps in identifying the most effective learning function for the predictive model. The research concludes by summarizing the findings and providing recommendations based on the results.

The dataset used consists of the average wholesale rice prices from 2017 to 2023, as shown in Table 1.

Table 1. Average Wholesale Rice Prices in Indonesia (Source :[25])

Month	Year and Rice Price (Rp/Kg)				
	2017	2018	2019	...	2023
January	11579,00	12276,00	12211,09	...	11647,91
February	11571,24	12414,00	12222,00	...	11990,12
March	11494,00	12299,00	12124,00	...	12041,64
April	11449,00	12035,00	12019,00	...	12092,38
May	11465,00	11943,00	12008,00	...	12102,70
June	11465,00	11907,26	12009,00	...	12115,81
July	11448,00	11936,00	12021,00	...	12141,72
August	11411,00	11899,00	12018,00	...	12265,68
September	11481,84	11900,00	12050,00	...	13036,96
October	11552,00	11926,21	12108,00	...	13315,29
November	11665,08	12013,00	12120,00	...	13380,40
December	11838,00	12105,77	12183,03	...	13458,06

The obtained dataset will be divided into a training dataset, covering the years 2017 to 2021 with the target year 2022, and a testing dataset, covering the years 2018 to 2022 with the target year 2023. In this process, the obtained dataset will first be normalized using the normalization formula shown in Equation 1.

$$x' = \frac{0,8(x-a)}{b-a} + 0,1 \quad (1)$$

x' represents the normalized data, x is the original data value to be normalized, a is the minimum value of the entire dataset, and b is the maximum value of the entire dataset. The results of the data division and transformation can be seen in Table 2 and Table 3.

Table 2. Training Dataset Transformation Results

2017	2018	2019	2020	2021	Target
0,41618	0,59564	0,57893	0,61283	0,13167	0,13733
0,41418	0,63118	0,58174	0,61602	0,13296	0,13090
0,39430	0,60157	0,55651	0,61933	0,11880	0,12884
0,38271	0,53359	0,52947	0,62296	0,10129	0,12678
0,38683	0,50990	0,52664	0,60003	0,10875	0,12497
0,38683	0,50073	0,52690	0,58225	0,10850	0,12497
0,38245	0,50810	0,52999	0,57933	0,10257	0,12523
0,37293	0,49858	0,52922	0,57918	0,10026	0,15150
0,39117	0,49883	0,53745	0,57321	0,10000	0,20840
0,40923	0,50558	0,55239	0,57272	0,10412	0,25346
0,43835	0,52793	0,55548	0,57057	0,10618	0,27019
0,48286	0,55181	0,57171	0,57209	0,12008	0,36057

In Training Process stage, the implementation of the designed architecture is carried out. The architecture design will be implemented using Matlab R2012b software. This study employs five architecture models: 5-5-1, 5-15-1, 5-25-1, 5-30-1, and 5-45-1, with both the standard Backpropagation learning function (traingd) and the optimized Backpropagation learning functions (traingda and traingdx).

Subsequently, scripts are inputted to perform the training process and evaluate performance values, followed by the insertion of normalized testing data to simulate the test data based on the training results.

Once all the stages have been completed, the evaluation process is conducted to assess the performance and capability of the best learning function.

Table 3. Training Dataset Transformation Results

2018	2019	2020	2021	2022	Target
0,59565	0,57893	0,61283	0,13167	0,13733	0,43393
0,63118	0,58174	0,61603	0,13296	0,13090	0,52204
0,60157	0,55651	0,61933	0,11880	0,12884	0,53530
0,53359	0,52947	0,62296	0,10129	0,12678	0,54837
0,50991	0,52664	0,60003	0,10875	0,12498	0,55102
0,50070	0,52690	0,58225	0,10850	0,12498	0,55440
0,50810	0,52999	0,57933	0,10258	0,12523	0,56107
0,49858	0,52922	0,57919	0,10026	0,15150	0,59299
0,49883	0,53746	0,57321	0,10000	0,20840	0,79158
0,50558	0,55239	0,57272	0,10412	0,25346	0,86324
0,52793	0,55548	0,57057	0,10618	0,27019	0,88000
0,55182	0,57171	0,57209	0,12008	0,36057	0,90000

3. Results and Discussions

The training dataset used in the training process has an input value of 5, corresponding to the input table derived from the average wholesale rice prices in Indonesia from 2017-2021, with the target year being 2022. The same applies to the testing dataset, which uses the average wholesale rice prices in Indonesia from 2018-2022, with the target year being 2023.

The hidden layer used consists of only one layer but with different numbers of neurons, namely 5, 15, 25, 30, and 45 neurons. The output value is set to 1, based on the target in each data entry that contains only one value.

Once the network architecture is established, the training and testing processes will be conducted based on the predetermined input values. The outcome will be the best network pattern determined by the obtained error values.

3.1 Training and Testing Results, Standard Backpropagation Learning Function (traingd)

Architecture Model 5-5-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output).

The training process will be tested with the following parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingd).

The results of the training and testing conducted using Matlab software can be seen in Figure 2. Figure 2 explains that training using the 5-5-1 architecture model with the traingd learning function resulted in 2602 epochs to achieve the goal. The accuracy results of the training using the 5-5-1 architecture model can be seen in Tables 4 and 5.

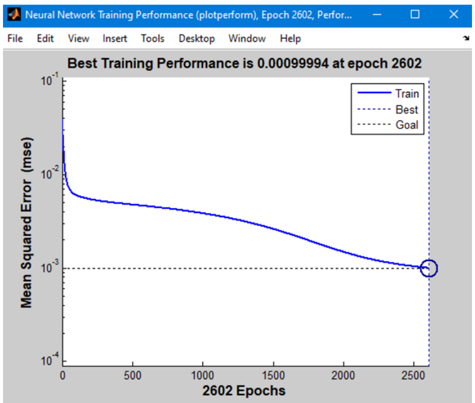

Figure 2. Training Using the 5-5-1 Model (Source: processed data)

Table 4. Training Results using Architecture Model 5-5-1

Training				
No	Target	Output	Error	SSE
1	0,137334	0,1287	0,008634	0,000075
2	0,130897	0,1269	0,003997	0,000016
3	0,128838	0,1211	0,007738	0,000060
4	0,126778	0,0384	0,088378	0,007811
5	0,124975	0,0954	0,029575	0,000875
6	0,124975	0,1633	-0,038325	0,001469
7	0,125233	0,1476	-0,022367	0,000500
8	0,151496	0,1517	-0,000204	0,000000
9	0,208398	0,1951	0,013298	0,000177
10	0,253457	0,2289	0,024557	0,000603
11	0,270193	0,2769	-0,006707	0,000045
12	0,360568	0,3802	-0,019632	0,000385
			Total	0,012015
			MSE	0,001001

Table 5. Accuracy Results of Testing Data using Model 5-5-1

Testing					
No	Target	Output	Error	SSE	Result
1	0,433926	0,9011	-0,467174	0,218252	1
2	0,522038	0,9011	-0,379062	0,143688	1
3	0,535303	0,9011	-0,365797	0,133807	1
4	0,548367	0,9011	-0,352733	0,124421	1
5	0,551024	0,9011	-0,350076	0,122553	1
6	0,554400	0,9010	-0,346600	0,120132	1
7	0,561071	0,9010	-0,339929	0,115552	1
8	0,592988	0,8999	-0,306912	0,094195	1
9	0,791576	0,5923	0,199276	0,039711	0
10	0,863240	0,5826	0,280640	0,078759	1
11	0,880004	0,7001	0,179904	0,032365	0
12	0,900000	0,9321	-0,032100	0,001030	0
			Total	1,224465	75%
			MSE	0,102039	

Architecture Model 5-15-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingd).

The results of the training and testing conducted using Matlab software are shown in Figure3, Tables 6 and 7.

Figure 3 explains that training using the 5-15-1 architecture model with the traingd learning function reached the goal at 731 epochs. The accuracy results of the training using the 5-15-1 architecture model can be seen in Tables 6 and 7.

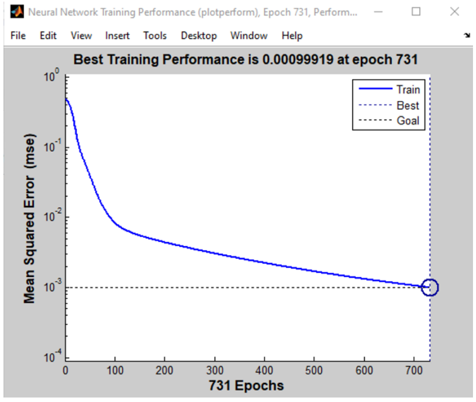

Figure 3. Training Using the 5-15-1 Model (Source: processed data)

Table 6. Training Results using architecture Model 5-15-1

	Training			
No	Target	Output	Error	SSE
1	0,137334	0,1051	0,032234	0,001039
2	0,130897	0,1071	0,023797	0,000566
3	0,128838	0,1144	0,014438	0,000208
4	0,126778	0,1880	-0,061222	0,003748
5	0,124975	0,1288	-0,003825	0,000015
6	0,124975	0,1063	0,018675	0,000349
7	0,125233	0,1140	0,011233	0,000126
8	0,151496	0,1081	0,043396	0,001883
9	0,208398	0,1537	0,054698	0,002992
10	0,253457	0,2746	-0,021143	0,000447
11	0,270193	0,2950	-0,024807	0,000615
12	0,360568	0,3614	-0,000832	0,000001
			Total	0,011990
			MSE	0,000999

Table 7. Accuracy Results of Testing Data using Model 5-15-1

	Testing				
No	Target	Output	Error	SSE	Result
1	0,433926	0,0827	0,351226	0,123360	1
2	0,522038	0,0790	0,443038	0,196283	1
3	0,535303	0,0804	0,454903	0,206937	1
4	0,548367	0,1201	0,428267	0,183413	1
5	0,551024	0,0886	0,462424	0,213836	1
6	0,554400	0,0806	0,473800	0,224486	1
7	0,561071	0,0795	0,481571	0,231911	1
8	0,592988	0,0916	0,501388	0,251390	1
9	0,791576	0,1785	0,613076	0,375862	0
10	0,863240	0,1817	0,681540	0,464497	0
11	0,880004	0,1817	0,698304	0,487628	0
12	0,900000	0,3606	0,539400	0,290952	1
			Total	3,250554	75%
			MSE	0,270880	

Architecture Model 5-25-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingd).

The results of the training and testing conducted using Matlab software are shown in Figure 4, Tables 8 and 9.

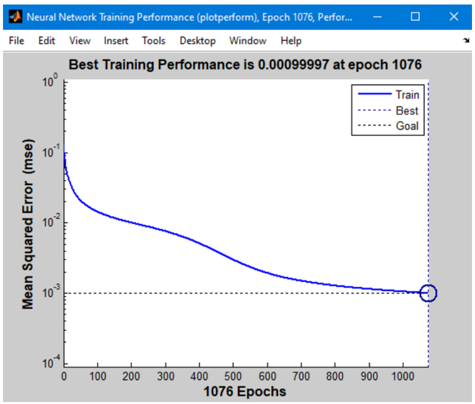

Figure 4. Training Using the 5-25-1 Model (Source: processed data)

Figure 4 explains that training using the 5-25-1 architecture model with the traingd learning function reached the goal at 1076 epochs. The accuracy results of the training using the 5-25-1 architecture model can be seen in Tables 8 and 9.

Table 8. Training Results using architecture Model 5-25-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1170	0,020334	0,000413
2	0,130897	0,1709	-0,040003	0,001600
3	0,128838	0,0792	0,049638	0,002464
4	0,126778	0,1528	-0,026022	0,000677
5	0,124975	0,0533	0,071675	0,005137
6	0,124975	0,1187	0,006275	0,000039
7	0,125233	0,1404	-0,015167	0,000230
8	0,151496	0,1343	0,017196	0,000296
9	0,208398	0,2022	0,006198	0,000038
10	0,253457	0,2457	0,007757	0,000060
11	0,270193	0,2998	-0,029607	0,000877
12	0,360568	0,3478	0,012768	0,000163
			Total	0,011995
			MSE	0,001000

Table 9. Accuracy Results of Testing Data using Model 5-25-1

No	Testing Target	Output	Error	SSE	Hasil
1	0,433926	0,3117	0,122226	0,014939	0
2	0,522038	0,2726	0,249438	0,062219	1
3	0,535303	0,2713	0,264003	0,069698	1
4	0,548367	0,2773	0,271067	0,073477	1
5	0,551024	0,3389	0,212124	0,044997	1
6	0,554400	0,4794	0,075000	0,005625	0
7	0,561071	0,5025	0,058571	0,003431	0
8	0,592988	0,4979	0,095088	0,009042	0
9	0,791576	0,7541	0,037476	0,001404	0
10	0,863240	0,3016	0,561640	0,315439	1
11	0,880004	0,2836	0,596404	0,355698	1
12	0,900000	0,0920	0,808000	0,652864	0
			Total	1,608833	50%
			MSE	0,134069	

Architecture Model 5-30-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output).

The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingd).

The results of the training and testing conducted using Matlab software are shown in Figure 5, Tables 10 and 11.

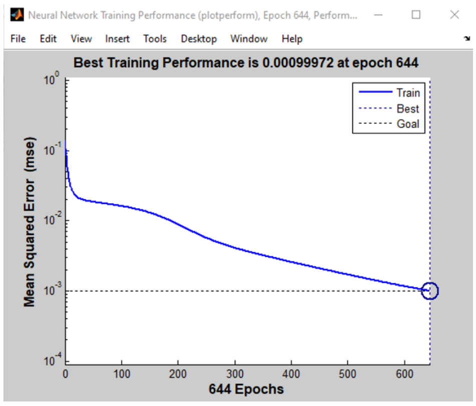

Figure 5. Training Using the 5-30-1 Model (Source: processed data)

Figure 5 explains that training using the 5-30-1 architecture model with the traingd learning function reached the goal at 644 epochs. The accuracy results of the training using the 5-30-1 architecture model can be seen in Tables 10 and 11.

Table 10. Training Results using architecture Model 5-30-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1502	-0,012866	0,000166
2	0,130897	0,1313	-0,000403	0,000000
3	0,128838	0,1061	0,022738	0,000517
4	0,126778	0,1474	-0,020622	0,000425
5	0,124975	0,0930	0,031975	0,001022
6	0,124975	0,1155	0,009475	0,000090
7	0,125233	0,1867	-0,061467	0,003778
8	0,151496	0,1565	-0,005004	0,000025
9	0,208398	0,2275	-0,019102	0,000365
10	0,253457	0,2118	0,041657	0,001735
11	0,270193	0,2169	0,053293	0,002840
12	0,360568	0,3927	-0,032132	0,001032
			Total	0,011996
			MSE	0,001000

Table 11. Accuracy Results of Testing Data using Model 5-30-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,6460	-0,212074	0,044975	1
2	0,522038	0,6464	-0,124362	0,015466	0
3	0,535303	0,6465	-0,111197	0,012365	0
4	0,548367	0,5744	-0,026033	0,000678	0
5	0,551024	0,5881	-0,037076	0,001375	0
6	0,554400	0,6767	-0,122300	0,014957	0
7	0,561071	0,7009	-0,139829	0,019552	0
8	0,592988	0,5854	0,007588	0,000058	0
9	0,791576	0,4446	0,346976	0,120392	1
10	0,863240	0,4373	0,425940	0,181425	1
11	0,880004	0,4372	0,442804	0,196075	1
12	0,900000	0,3609	0,539100	0,290629	1
			Total	0,897947	41,67%
			MSE	0,074829	

Architecture Model 5-45-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer);

Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingd).

The results of the training and testing conducted using Matlab software are seen in Figure 6. Figure 6 explains that training using the 5-45-1 architecture model with the traingd learning function reached the goal at 868 epochs. The accuracy results of the training using the 5-45-1 architecture model can be seen in Tables 12 and 13.

Table 12. Training Results using architecture Model 5-45-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1218	0,015534	0,000241
2	0,130897	0,1259	0,004997	0,000025
3	0,128838	0,1284	0,000438	0,000000
4	0,126778	0,1198	0,006978	0,000049
5	0,124975	0,1738	-0,048825	0,002384
6	0,124975	0,1778	-0,052825	0,002790
7	0,125233	0,1430	-0,017767	0,000316
8	0,151496	0,1348	0,016696	0,000279
9	0,208398	0,1502	0,058198	0,003387
10	0,253457	0,2062	0,047257	0,002233
11	0,270193	0,2677	0,002493	0,000006
12	0,360568	0,3765	-0,015932	0,000254
			Total	0,011964
			MSE	0,000997

Table 13. Accuracy Results of Testing Data using Model 5-45-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,8007	-0,366774	0,134523	1
2	0,522038	0,7750	-0,252962	0,063990	1
3	0,535303	0,7555	-0,220197	0,048487	1
4	0,548367	0,8486	-0,300233	0,090140	1
5	0,551024	0,8507	-0,299676	0,089806	1
6	0,554400	0,8598	-0,305400	0,093269	1
7	0,561071	0,8695	-0,308429	0,095128	1
8	0,592988	0,9475	-0,354512	0,125679	1
9	0,791576	0,9674	-0,175824	0,030914	0
10	0,863240	0,9405	-0,077260	0,005969	0
11	0,880004	0,9390	-0,058996	0,003481	0
12	0,900000	0,9170	-0,017000	0,000289	0
			Total	0,781674	66,67%
			MSE	0,065140	

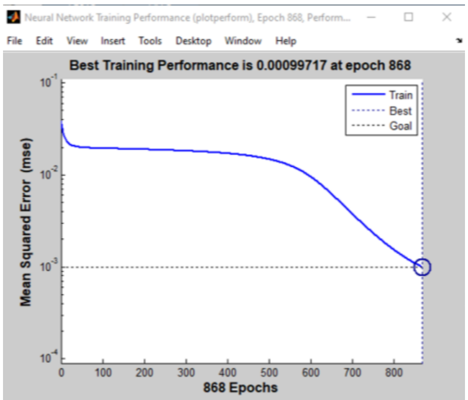

Figure 6. Training Using the 5-45-1 Model (Source: processed data)

3.2 *Gradient Descent Learning Function with Adaptive Learning Rate (traingda)*

Architecture Model 5-5-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingda). The results of the training and testing conducted using Matlab software are shown in Figure 7.

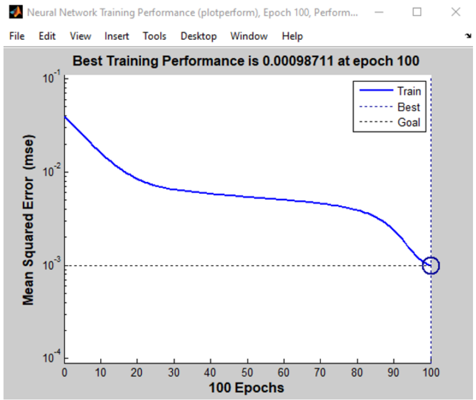

Figure 7. Training Using the 5-5-1 Model with traingda (Source: processed data)

Figure 7 explains that training using the 5-5-1 architecture model with the traingda learning function reached the goal at epoch 100. The accuracy results of the training using the 5-5-1 architecture model can be seen in Tables 14 and 15.

Table 14. Training Results using architecture Model 5-5-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1305	0,006834	0,000047
2	0,130897	0,1287	0,002197	0,000005
3	0,128838	0,1228	0,006038	0,000036
4	0,126778	0,0389	0,087878	0,007723
5	0,124975	0,0964	0,028575	0,000817
6	0,124975	0,1637	-0,038725	0,001500
7	0,125233	0,1480	-0,022767	0,000518
8	0,151496	0,1521	-0,000604	0,000000
9	0,208398	0,1950	0,013398	0,000180
10	0,253457	0,2285	0,024957	0,000623
11	0,270193	0,2767	-0,006507	0,000042
12	0,360568	0,3795	-0,018932	0,000358
			Total	0,011849
			MSE	0,000987

Table 15. Accuracy Results of Testing Data using Model 5-5-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,9000	-0,466074	0,217225	1
2	0,522038	0,9000	-0,377962	0,142855	1
3	0,535303	0,9000	-0,364697	0,133004	1
4	0,548367	0,9000	-0,351633	0,123646	1
5	0,551024	0,9000	-0,348976	0,121784	1
6	0,554400	0,9000	-0,345600	0,119439	1
7	0,561071	0,9000	-0,338929	0,114873	1
8	0,592988	0,8988	-0,305812	0,093521	1
9	0,791576	0,5917	0,199876	0,039950	0
10	0,863240	0,5822	0,281040	0,078983	1
11	0,880004	0,6999	0,180104	0,032437	0
12	0,900000	0,9321	-0,032100	0,001030	0
			Total	1,218749	75%
			MSE	0,101562	

Architecture Model 5-15-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingda). The results of the training and testing conducted using Matlab software are shown in Figure 8. Figure 8 explains that training using the 5-15-1 architecture model with the traingda learning function reached the goal at epoch 75.

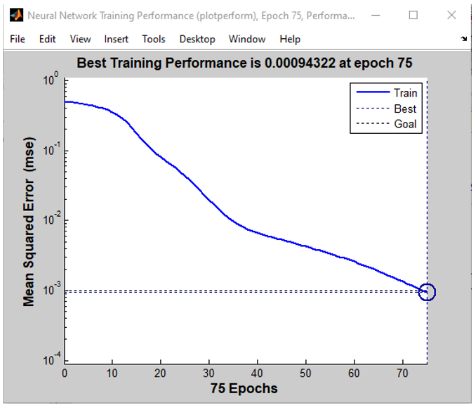

Figure 8: Training Using the 5-15-1 Model with traingda (Source: processed data)

Table 16. Training Results using architecture Model 5-15-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1066	0,030734	0,000945
2	0,130897	0,1087	0,022197	0,000493
3	0,128838	0,1157	0,013138	0,000173
4	0,126778	0,1857	-0,058922	0,003472
5	0,124975	0,1286	-0,003625	0,000013
6	0,124975	0,1074	0,017575	0,000309
7	0,125233	0,1149	0,010333	0,000107
8	0,151496	0,1088	0,042696	0,001823
9	0,208398	0,1550	0,053398	0,002851
10	0,253457	0,2759	-0,022443	0,000504
11	0,270193	0,2952	-0,025007	0,000625
12	0,360568	0,3591	0,001468	0,000002
			Total	0,011316
			MSE	0,000943

Table 17. Accuracy Results of Testing Data using Model 5-15-1

No	Target	Output	Training Error	SSE	Result
1	0,433926	0,0828	0,351126	0,123289	1
2	0,522038	0,0792	0,442838	0,196105	1
3	0,535303	0,0806	0,454703	0,206755	1
4	0,548367	0,1202	0,428167	0,183327	1
5	0,551024	0,0888	0,462224	0,213651	1
6	0,554400	0,0808	0,473600	0,224297	1
7	0,561071	0,0797	0,481371	0,231718	1
8	0,592988	0,0920	0,500988	0,250989	1
9	0,791576	0,1797	0,611876	0,374392	0
10	0,863240	0,1829	0,680340	0,462863	0
11	0,880004	0,1829	0,697104	0,485954	0
12	0,900000	0,3638	0,536200	0,287510	1
			Total	3,240851	75%
			MSE	0,270071	

The accuracy results of the training using the 5-15-1 architecture model can be seen in Tables 16 and 17.

Architecture Model 5-25-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingda). The results of the training and testing conducted using Matlab software are shown in Figure 9.

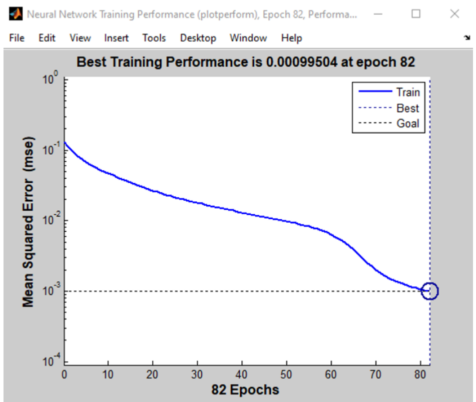

Figure 9. Training Using the 5-25-1 Model with traingda (Source: processed data)

Table 18. Training Results using architecture Model 5-25-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1168	0,020534	0,000422
2	0,130897	0,1705	-0,039603	0,001568
3	0,128838	0,0791	0,049738	0,002474
4	0,126778	0,1533	-0,026522	0,000703
5	0,124975	0,0535	0,071475	0,005109
6	0,124975	0,1190	0,005975	0,000036
7	0,125233	0,1405	-0,015267	0,000233
8	0,151496	0,1345	0,016996	0,000289
9	0,208398	0,2022	0,006198	0,000038
10	0,253457	0,2451	0,008357	0,000070
11	0,270193	0,2994	-0,029207	0,000853
12	0,360568	0,3486	0,011968	0,000143
			Total	0,011938
			MSE	0,000995

Table 19. Accuracy Results of Testing Data using Model 5-25-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,3116	0,122326	0,014964	0
2	0,522038	0,2725	0,249538	0,062269	1
3	0,535303	0,2712	0,264103	0,069750	1
4	0,548367	0,2770	0,271367	0,073640	1
5	0,551024	0,3383	0,212724	0,045252	1
6	0,554400	0,4786	0,075800	0,005746	0
7	0,561071	0,5018	0,059271	0,003513	0
8	0,592988	0,4974	0,095588	0,009137	0
9	0,791576	0,7539	0,037676	0,001419	0
10	0,863240	0,3013	0,561940	0,315777	1
11	0,880004	0,2834	0,596604	0,355936	1
12	0,900000	0,0921	0,807900	0,652702	0
			Total	1,610105	50%
			MSE	0,134175	

Figure 9 explains that training using the 5-25-1 architecture model with the traingda learning function reached the goal at epoch 82. The accuracy results of the training using the 5-25-1 architecture model can be seen in Tables 18 and 19.

Architecture Model 5-30-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer) Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function(traingda). The results of the training and testing conducted using Matlab software are shown in Figure 10.

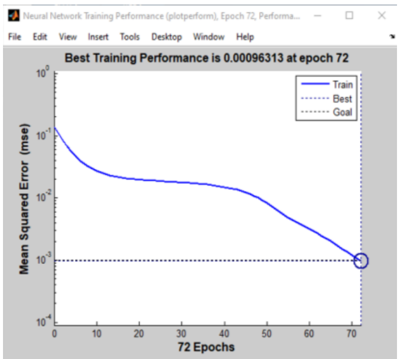

Figure 10. Training Using the 5-30-1 Model with traingda (Source: processed data)

Table 20. Training Results using architecture Model 5-30-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1490	-0,011666	0,000136
2	0,130897	0,1304	0,000497	0,000000
3	0,128838	0,1065	0,022338	0,000499
4	0,126778	0,1484	-0,021622	0,000468
5	0,124975	0,0928	0,032175	0,001035
6	0,124975	0,1148	0,010175	0,000104
7	0,125233	0,1859	-0,060667	0,003680
8	0,151496	0,1558	-0,004304	0,000019
9	0,208398	0,2273	-0,018902	0,000357
10	0,253457	0,2131	0,040357	0,001629
11	0,270193	0,2191	0,051093	0,002610
12	0,360568	0,3924	-0,031832	0,001013
			Total	0,011550
			MSE	0,000963

Table 21. Accuracy Results of Testing Data using Model 5-30-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,6464	-0,212474	0,045145	1
2	0,522038	0,6468	-0,124762	0,015566	0
3	0,535303	0,6469	-0,111597	0,012454	0
4	0,548367	0,5745	-0,026133	0,000683	0
5	0,551024	0,5882	-0,037176	0,001382	0
6	0,554400	0,6762	-0,121800	0,014835	0
7	0,561071	0,7004	-0,139329	0,019413	0
8	0,592988	0,5850	0,007988	0,000064	0
9	0,791576	0,4440	0,347576	0,120809	1
10	0,863240	0,4367	0,426540	0,181936	1
11	0,880004	0,4366	0,443404	0,196607	1
12	0,900000	0,3594	0,540600	0,292248	1
			Total	0,901142	41,67%
			MSE	0,075095	

Figure 10 explains that training using the 5-30-1 architecture model with the traingda learning function reached the goal at epoch 72. The accuracy results of the training using the 5-30-1 architecture model can be seen in Tables 20 and 21.

Architecture Model 5-45-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingda).

The results of the training and testing conducted using Matlab software are shown Figure 11.

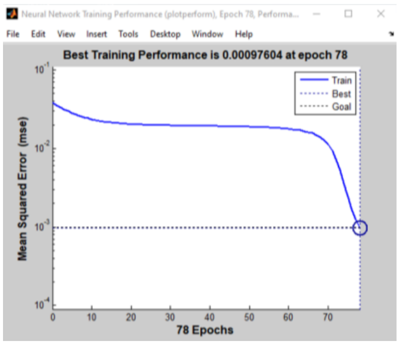

Figure 11. Training Using the 5-45-1 Model with traingda (Source: processed data)

Table 22. Training Results using architecture Model 5-45-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1241	0,013234	0,000175
2	0,130897	0,1287	0,002197	0,000005
3	0,128838	0,1253	0,003538	0,000013
4	0,126778	0,1117	0,015078	0,000227
5	0,124975	0,1654	-0,040425	0,001634
6	0,124975	0,1723	-0,047325	0,002240
7	0,125233	0,1382	-0,012967	0,000168
8	0,151496	0,1299	0,021596	0,000466
9	0,208398	0,1457	0,062698	0,003931
10	0,253457	0,2012	0,052257	0,002731
11	0,270193	0,2621	0,008093	0,000065
12	0,360568	0,3679	-0,007332	0,000054
			Total	0,011709
			MSE	0,000976

Table 23. Accuracy Results of Testing Data using Model 5-45-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,8021	-0,368174	0,135552	1
2	0,522038	0,7770	-0,254962	0,065006	1
3	0,535303	0,7572	-0,221897	0,049238	1
4	0,548367	0,8495	-0,301133	0,090681	1
5	0,551024	0,8515	-0,300476	0,090286	1
6	0,554400	0,8606	-0,306200	0,093758	1
7	0,561071	0,8703	-0,309229	0,095623	1
8	0,592988	0,9482	-0,355212	0,126176	1
9	0,791576	0,9678	-0,176224	0,031055	0
10	0,863240	0,9414	-0,078160	0,006109	0
11	0,880004	0,9400	-0,059996	0,003600	0
12	0,900000	0,9193	-0,019300	0,000372	0
			Total	0,787455	66,67%
			MSE	0,065621	

Figure 11 explains that training using the 5-45-1 architecture model with the traingda learning function reached the goal at epoch 78. The accuracy results of the training using the 5-45-1 architecture model can be seen in Tables 22 and 23.

3.3 Gradient Descent Learning Function with Adaptive Learning Rate and Momentum (traingdx)

Architecture Model 5-5-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output).

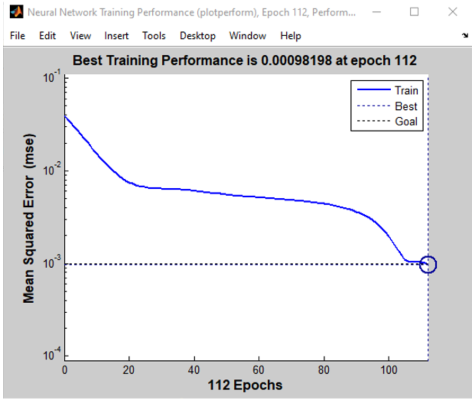

Figure 12. Training Using the 5-5-1 Model with traingdx (Source: processed data)

Table 24. Training Results using architecture Model 5-5-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1584	-0,021066	0,000444
2	0,130897	0,1569	-0,026003	0,000676
3	0,128838	0,1500	-0,021162	0,000448
4	0,126778	0,0497	0,077078	0,005941
5	0,124975	0,1150	0,009975	0,000100
6	0,124975	0,1700	-0,045025	0,002027
7	0,125233	0,1544	-0,029167	0,000851
8	0,151496	0,1570	-0,005504	0,000030
9	0,208398	0,1916	0,016798	0,000282
10	0,253457	0,2233	0,030157	0,000909
11	0,270193	0,2762	-0,006007	0,000036
12	0,360568	0,3546	0,005968	0,000036
			Total	0,011780
			MSE	0,000982

Table 25. Accuracy Results of Testing Data using Model 5-5-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,8799	-0,445974	0,198893	1
2	0,522038	0,8799	-0,357862	0,128065	1
3	0,535303	0,8799	-0,344597	0,118747	1
4	0,548367	0,8799	-0,331533	0,109914	1
5	0,551024	0,8799	-0,328876	0,108159	1
6	0,554400	0,8799	-0,325500	0,105950	1
7	0,561071	0,8799	-0,318829	0,101652	1
8	0,592988	0,8785	-0,285512	0,081517	1
9	0,791576	0,5844	0,207176	0,042922	1
10	0,863240	0,5787	0,284540	0,080963	1
11	0,880004	0,7002	0,179804	0,032329	0
12	0,900000	0,9274	-0,027400	0,000751	0
			Total	1,109863	83,33%
			MSE	0,092489	

The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer) Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1) Learning Function (traingdx).

The results of the training and testing conducted using Matlab software are shown in Figure 12. Figure 12 explains that training using the 5-5-1 architecture model with the traingdx learning function reached the goal at epoch 78.

The accuracy results of the training using the 5-5-1 architecture model can be seen in Tables 24 and 25.

Architecture Model 5-15-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingdx). The results of the training and testing conducted using Matlab software are shown Figure 13.

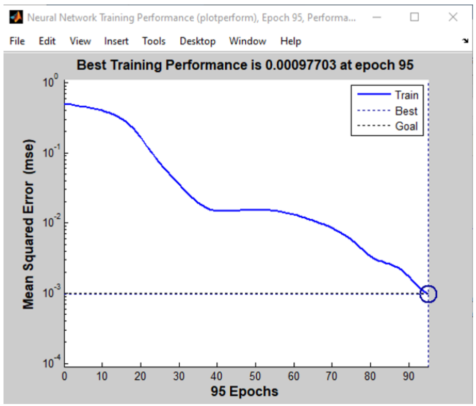

Figure 13. Training Using the 5-15-1 Model with traingdx (Source: processed data)

Figure 13 explains that training using the 5-15-1 architecture model with the traingdx learning function reached the goal at epoch 95. The accuracy results of the training using the 5-15-1 architecture model can be seen in Tables 26 and 27.

Table 26. Training Results using architecture Model 5-15-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1136	0,023734	0,000563
2	0,130897	0,1185	0,012397	0,000154
3	0,128838	0,1156	0,013238	0,000175
4	0,126778	0,1713	-0,044522	0,001982
5	0,124975	0,1206	0,004375	0,000019
6	0,124975	0,1063	0,018675	0,000349
7	0,125233	0,1126	0,012633	0,000160
8	0,151496	0,1071	0,044396	0,001971
9	0,208398	0,1561	0,052298	0,002735
10	0,253457	0,2928	-0,039343	0,001548
11	0,270193	0,3100	-0,039807	0,001585
12	0,360568	0,3384	0,022168	0,000491
			Total	0,011732
			MSE	0,000978

Table 27. Accuracy Results of Testing Data using Model 5-15-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,0805	0,353426	0,124910	1
2	0,522038	0,0773	0,444738	0,197792	1
3	0,535303	0,0786	0,456703	0,208578	1
4	0,548367	0,1121	0,436267	0,190329	1
5	0,551024	0,0856	0,465424	0,216619	1
6	0,554400	0,0788	0,475600	0,226195	1
7	0,561071	0,0779	0,483171	0,233454	1
8	0,592988	0,0869	0,506088	0,256125	1
9	0,791576	0,1574	0,634176	0,402179	0
10	0,863240	0,1600	0,703240	0,494546	0
11	0,880004	0,1601	0,719904	0,518262	0
12	0,900000	0,3313	0,568700	0,323420	1
			Total	3,392410	75%
			MSE	0,282701	

Architecture Model 5-25-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the following parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingdx). The results of the training and testing conducted using Matlab software are shown in Figure 14. Figure 14 explains that training using the 5-25-1 architecture model with the traingdx learning function reached the goal at epoch 96. The accuracy results of the training using the 5-25-1 architecture model can be seen in Tables 28 and 29.

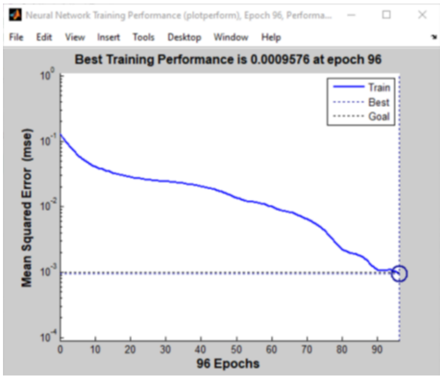

Figure 14. Training Using the 5-25-1 Model with traingdx (Source: processed data)

Table 28. Training Results using architecture Model 5-25-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1144	0,022934	0,000526
2	0,130897	0,1592	-0,028303	0,000801
3	0,128838	0,0810	0,047838	0,002288
4	0,126778	0,1756	-0,048822	0,002384
5	0,124975	0,0694	0,055575	0,003089
6	0,124975	0,1431	-0,018125	0,000329
7	0,125233	0,1541	-0,028867	0,000833
8	0,151496	0,1486	0,002896	0,000008
9	0,208398	0,2065	0,001898	0,000004
10	0,253457	0,2227	0,030757	0,000946
11	0,270193	0,2725	-0,002307	0,000005
12	0,360568	0,3441	0,016468	0,000271
			Total	0,011484
			MSE	0,000957

Table 29. Accuracy Results of Testing Data using Model 5-25-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,2894	0,144526	0,020888	0
2	0,522038	0,2528	0,269238	0,072489	1
3	0,535303	0,2514	0,283903	0,080601	1
4	0,548367	0,2493	0,299067	0,089441	1
5	0,551024	0,3049	0,246124	0,060577	1
6	0,554400	0,4371	0,117300	0,013759	0
7	0,561071	0,4613	0,099771	0,009954	0
8	0,592988	0,4626	0,130388	0,017001	0
9	0,791576	0,7366	0,054976	0,003022	0
10	0,863240	0,2865	0,576740	0,332629	1
11	0,880004	0,2699	0,610104	0,372227	0
12	0,900000	0,0946	0,805400	0,648669	0
			Total	1,721258	41,67%
			MSE	0,143438	

Architecture Model 5-30-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the following parameters: Input (5); Architecture (1 hidden layer); Goal (0.001); Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingdx).

Figure 15 explains that training using the 5-30-1 architecture model with the traingdx learning function reached the goal at epoch 97. The accuracy results of the training using the 5-30-1 architecture model can be seen in Tables 30 and 31

Table 30. Training Results using architecture Model 5-30-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1213	0,016034	0,000257
2	0,130897	0,1076	0,023297	0,000543
3	0,128838	0,1048	0,024038	0,000578
4	0,126778	0,1671	-0,040322	0,001626
5	0,124975	0,0924	0,032575	0,001061
6	0,124975	0,1128	0,012175	0,000148
7	0,125233	0,1854	-0,060167	0,003620
8	0,151496	0,1563	-0,004804	0,000023
9	0,208398	0,2331	-0,024702	0,000610
10	0,253457	0,2245	0,028957	0,000839
11	0,270193	0,2368	0,033393	0,001115
12	0,360568	0,3850	-0,024432	0,000597
			Total	0,011017
			MSE	0,000918

Table 31. Accuracy Results of Testing Data using Model 5-30-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,6424	-0,208474	0,043461	1
2	0,522038	0,6428	-0,120762	0,014583	0
3	0,535303	0,6430	-0,107697	0,011599	0
4	0,548367	0,5735	-0,025133	0,000632	0
5	0,551024	0,5898	-0,038776	0,001504	0
6	0,554400	0,6767	-0,122300	0,014957	0
7	0,561071	0,7005	-0,139429	0,019440	0
8	0,592988	0,5794	0,013588	0,000185	0
9	0,791576	0,4336	0,357976	0,128147	1
10	0,863240	0,4259	0,437340	0,191266	1
11	0,880004	0,4258	0,454204	0,206301	1
12	0,900000	0,3398	0,560200	0,313824	1
			Total	0,945900	41,67%
			MSE	0,078825	

The results of the training and testing conducted using Matlab software are shown in Figure 15.

Figure 15. Training Using the 5-30-1 Model with traingdx (Source: processed data)

Architecture Model 5-45-1: The data used comprises the average wholesale rice prices in Indonesia, with the training data from 2017-2021 serving as the input data and 2022 as the target (output), while the testing data from 2018-2022 serves as the input data and 2023 as the target (output). The training process will be tested with the parameters: Input (5); Architecture (1 hidden layer); Maximum Epoch (2,500,000); Learning Rate (0.1); Learning Function (traingdx). The results of the training and testing conducted using Matlab software are shown in Figure 16.

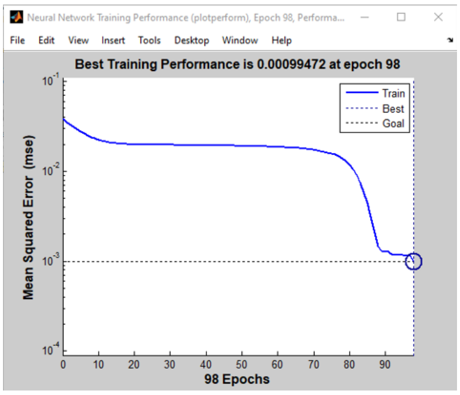

Figure 16. Training Using the 5-45-1 Model with traingdx (Source: processed data)

Figure 16 explains that training using the 5-45-1 architecture model with the traingdx learning function reached the goal at epoch 98. The accuracy results of the training using the 5-45-1 architecture model can be seen in Tables 32 and 33.

Table 32. Training Results using architecture Model 5-45-1

No	Training Target	Output	Error	SSE
1	0,137334	0,1432	-0,005866	0,000034
2	0,130897	0,1486	-0,017703	0,000313
3	0,128838	0,1000	0,028838	0,000832
4	0,126778	0,0669	0,059878	0,003585
5	0,124975	0,1349	-0,009925	0,000099
6	0,124975	0,1807	-0,055725	0,003105
7	0,125233	0,1489	-0,023667	0,000560
8	0,151496	0,1369	0,014596	0,000213
9	0,208398	0,1680	0,040398	0,001632
10	0,253457	0,2423	0,011157	0,000124
11	0,270193	0,3072	-0,037007	0,001370
12	0,360568	0,3522	0,008368	0,000070
			Total	0,011938
			MSE	0,000995

Table 33. Accuracy Results of Testing Data using Model 5-45-1

No	Testing Target	Output	Error	SSE	Result
1	0,433926	0,8527	-0,418774	0,175372	1
2	0,522038	0,8331	-0,311062	0,096760	1
3	0,535303	0,8169	-0,281597	0,079297	1
4	0,548367	0,8898	-0,341433	0,116576	1
5	0,551024	0,8913	-0,340276	0,115788	1
6	0,554400	0,8983	-0,343900	0,118267	1
7	0,561071	0,9058	-0,344729	0,118838	1
8	0,592988	0,9639	-0,370912	0,137576	1
9	0,791576	0,9763	-0,184724	0,034123	0
10	0,863240	0,9577	-0,094460	0,008923	0
11	0,880004	0,9570	-0,076996	0,005928	0
12	0,900000	0,9448	-0,044800	0,002007	0
			Total	1,009454	66,67%
			MSE	0,084121	

3.4 Analysis Results

In general, the analysis results from the training and testing process using the 5 architecture models with the Backpropagation learning function parameters (traingd, traingda, and traingdx) can be seen in Table 34.

Table 34. Recapitulation of Training and Testing Results for Backpropagation Learning Functions

No	Learning Function	Architecture	Training Epoch	Training Time	MSE Training	Testing MSE Testing	Accuracy
1		5-5-1	2602	00:00:05	0,001001264	0,102038718	75%
2		5-15-1	731	00:00:02	0,000999139	0,27087954	75%
3	traingd	5-25-1	1076	00:00:02	0,000999616	0,134069414	50%
4		5-30-1	644	00:00:01	0,000999682	0,074828904	41,67%
5		5-45-1	868	00:00:02	0,000997018	0,065139525	66,67%
6		5-5-1	100	00:00:01	0,000987376	0,101562427	75%
7		5-15-1	75	00:00:00	0,000942998	0,270070912	75%
8	traingda	5-25-1	82	00:00:00	0,000994849	0,134175446	50%
9		5-30-1	72	00:00:00	0,00096253	0,075095181	41,67%
10		5-45-1	78	00:00:00	0,000975774	0,065621281	66,67%
11		5-5-1	112	00:00:01	0,000981655	0,092488591	83,33 %
12		5-15-1	95	00:00:00	0,000978	0,282700804	75%
13	traingdx	5-25-1	96	00:00:00	0,000956999	0,143438157	41,67%
14		5-30-1	97	00:00:00	0,000918062	0,078824961	41,67%
15		5-45-1	98	00:00:00	0,000994815	0,084121202	66,67%

Based on the analysis results presented in Table 5, it can be seen that the traingdx learning function, or Gradient Descent with Adaptive Learning Rate and Momentum, has the highest accuracy value among all architectures for each learning function, achieving 83.33% accuracy with the 5-5-1 architecture. This was attained in 112 epochs with a training time of 1 second and an MSE of 0.09248859.

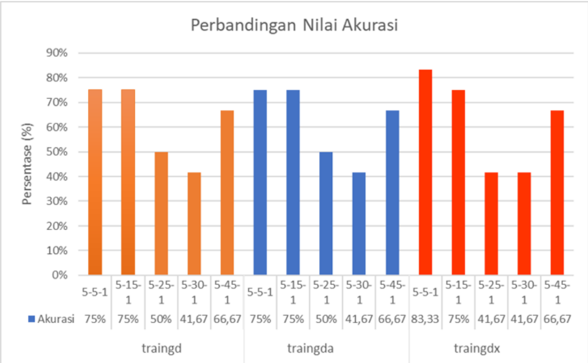

Figure 17. Comparison of accuracy values

Figure 17 presents a comparison of accuracy values between standard Backpropagation (traingd) and Backpropagation using various learning functions. According to Figure 17, the Backpropagation method using the traingdx learning function yields the best results with the 5-5-1 architecture model.

4. Conclusion

This study aims to evaluate and compare various learning functions within the Backpropagation algorithm to forecast the average wholesale rice prices in Indonesia. The dataset used includes the average wholesale rice prices from 2017 to 2023. The results indicate that the Gradient Descent with Adaptive Learning Rate and Momentum (traingdx) learning function with a 5-5-1 architecture model achieved the highest accuracy of 83.33%. This represents an 8.33% improvement over the traingd and traingda learning functions, both of which achieved a maximum accuracy of 75%. Overall, this research concludes that utilizing various learning functions in Backpropagation yields better accuracy compared to standard Backpropagation. This highlights the importance of selecting the appropriate learning function to enhance the performance of predictive models. Quantitative results from this study show that the traingdx learning function with a 5-5-1 architecture model provides a Mean Squared Error (MSE) of 0.09248859 and achieves 83.33% accuracy after 112 epochs with a training time of 1 second. Meanwhile, the traingd and traingda learning functions reach their highest MSEs at 0.102038718 and 0.27087954, respectively, with a maximum accuracy of 75%. Future research can focus on several important aspects, such as testing with larger and more diverse datasets to assess the broader generalization of the model. Additionally, applying and comparing the performance of various neural network architectures can help determine if there are significant improvements in prediction accuracy. Exploring and testing other learning functions not evaluated in this study is also essential to uncover further potential improvements in accuracy. Combining Backpropagation with other optimization techniques, such as Genetic Algorithms or Particle Swarm Optimization, could enhance the performance and efficiency of the model. Besides accuracy, future studies could also focus on training and prediction time efficiency to ensure the model is not only accurate but also fast and efficient.

References

[1] M. Yanto, L. Mayola, and M. Hafizh, "Neural Network Backprogation Identifikasi Pola Harga Saham Jakarta Islamic Index (JII)," *J. RESTI (Rekayasa Sist. Dan Teknol. Informasi)*, vol. 4, no. 1, pp. 90–94, 2020, doi: https://doi.org/10.29207/resti.v4i1.1266.

[2] A. A. Mahmudi, "Optimasi Conjugate Gradient Pada Backpropagation Neural Network Untuk Prediksi Hasil Tangkap Ikan," *Saintekbu*, vol. 12, no. 2, pp. 29–39, 2020, doi: 10.32764/saintekbu.v12i2.1031.

[3] R. N. Singarimbun, O. E. Putra, N. L. W. S. R. Ginantra, and M. P. Dewi, "Backpropagation Artificial Neural Network Enhancement using Beale-Powell Approach Technique," *J. Phys. Conf. Ser.*, vol. 2394, no. 1, 2022, doi: 10.1088/1742-6596/2394/1/012007.

[4] S. Bhardwaj, A. Tarafdar, M. Baghel, T. Dutt, and G. K. Gaur, "Determining Point of Economic Cattle Milk Production through Machine Learning and Evolutionary Algorithm for Enhancing Food Security," *J. Food Qual.*, vol. 2023, 2023, doi: 10.1155/2023/7568139.

[5] H. Mustafidah and I. F. Syarifah, "Tingkat Eror Jaringan Backpropagation pada Model Neuron 15-26-1 dan 15-29-1," in *Seminar Nasional CORISINDO*, 2022, pp. 481–485. [Online]. Available: https://corisindo.stikom-bali.ac.id/penelitian/index.php/semnas/article/view/99

[6] R. Maiyuriska, "Penerapan Jaringan Syaraf Tiruan dengan Algoritma Backpropagation dalam Memprediksi Hasil Panen Gabah Padi," *J. Inform. Ekon. Bisnis*, vol. 4, no. 1, pp. 28–33, 2022, doi: 10.37034/infeb.v4i1.115.

[7] A. Wanto, J. Na'am, Yuhandri, A. P. Windarto, and Mesran, "Analisis Penurunan Gradien dengan Kombinasi Fungsi Aktivasi pada Algoritma JST untuk Pencarian Akurasi Terbaik," *J. Media Inform. Budidarma*, vol. 4, no. 4, pp. 1197–1205, 2020, doi: 10.30865/mib.v4i4.2509.

[8] Putrama Alkhairi and A. P. Windarto, "Classification Analysis of Back propagation-Optimized CNN Performance in Image Processing," *J. Syst. Eng. Inf. Technol.*, vol. 2, no. 1, pp. 8–15, 2023, doi: 10.29207/joseit.v2i1.5015.

[9] F. Damayanti and R. Rismayanti, "Optimasi Fungsi Pembelajaran Backpropagation dalam Mengklasifikasikan Pasien Kanker Paru Pasca Operasi," *J. Unitek*, vol. 15, no. 1, pp. 49–58, 2022, doi: 10.52072/unitek.v15i1.335.

[10] I. I. Ridho, A. A. G. B. Ariana, and A. P. Windarto, "Optimasi Fungsi Pembelajaran Jaringan Saraf Tiruan dalam Meningkatkan Akurasi pada Prediksi Ekspor Kopi Menurut Negara Tujuan Utama," *Build. Informatics, Technol. Sci.*, vol. 4, no. 4, 2023, doi: 10.47065/bits.v4i4.3240.

[11] S. H. Haji and A. M. Abdulazeez, "Comparison Of Optimization Techniques Based on Gradient Descent Algorithm: A Review," *J. Archaeol. Egypt/Egyptology*, vol. 18, no. 4, pp. 2715–2743, 2021.

[12] Dwira Azi Pragana, D. W. Manurung, and Agus Perdana Windarto, "Analisa Metode Backpropagation Pada Prediksi Rata-rata Harga Beras Bulanan Di Tingkat Penggilingan Menurut Kualitas," *J. Comput. Informatics Res.*, vol. 2, no. 3, pp. 77–84, 2023, doi: 10.47065/comforch.v2i3.855.

[13] Rika Setiana, Razalfa Aindi Siregar, Fahry Husaini, and Agus Perdana Windarto, "Analisis Metode Backpropagation Dalam

Memprediksi Jumlah Produksi Daging Kambing di Indonesia," *J. Informatics, Electr. Electron. Eng.*, vol. 2, no. 4, pp. 113–123, 2023, doi: 10.47065/jieee.v2i4.1177.

[14] Natasya, S. Musdalifah, and Andri, "Prediksi Harga Beras Di Tingkat Perdagangan Besar Indonesia Menggunakan Algoritma Backpropagation," *J. Ilm. Mat. Dan Terap.*, vol. 18, no. 2, pp. 148–159, 2021, doi: 10.22487/2540766x.2021.v18.i2.15688.

[15] D. W. Kim, M. S. Kim, J. Lee, and P. Park, "Adaptive learning-rate backpropagation neural network algorithm based on the minimization of mean-square deviation for impulsive noises," *IEEE Access*, vol. 8, pp. 98018–98026, 2020, doi: 10.1109/ACCESS.2020.2997010.

[16] M. Thoriq, "Peramalan Jumlah Permintaan Produksi Menggunakan Jaringan Saraf Tiruan Algoritma Backpropagation," *J. Inf. dan Teknol.*, vol. 4, no. 1, pp. 27–32, 2022, doi: 10.37034/jidt.v4i1.178.

[17] Mesran, S. R. Yahya, F. Nugroho, and A. P. Windarto, "Investigating the Impact of ReLU and Sigmoid Activation Functions on Animal Classification Using CNN Models," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 8, no. 1, pp. 111–118, 2024, doi: https://doi.org/10.29207/resti.v8i1.5367.

[18] N. M. Nawi, F. Hamzah, N. A. Hamid, M. Z. Rehman, M. Aamir, and A. A. Ramli, "An optimized back propagation learning algorithm with adaptive learning rate," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 7, no. 5, pp. 1693–1700, 2017, doi: 10.18517/ijaseit.7.5.2972.

[19] A. R. Bohari, "Meningkatkan Kinerja Backpropagation Neural Network Menggunakan Algoritma Adaptif," *IMTechno J. Ind. Manag. Technol.*, vol. 3, no. 1, pp. 58–63, 2022, doi: 10.31294/imtechno.v3i1.1043.

[20] M. Dasuki, "Optimasi Nilai Bobot Algoritma Backpropagation Neural Network Dengan Algoritma Genetika," *JUSTINDO (Jurnal Sist. dan Teknol. Inf. Indones.*, vol. 6, no. 1, pp. 38–44, 2021, doi: 10.32528/justindo.v6i1.5280.

[21] W. Saputra, A. P. Windarto, and A. Wanto, "Analysis of the Resilient Method in Training and Accuracy in the Backpropagation Method," *IJICS (International J. Informatics Comput. Sci.*, vol. 5, no. 1, p. 52, 2021, doi: 10.30865/ijics.v5i1.2922.

[22] A. Wanto *et al.*, "Analysis Of Standard Gradient Descent With GD Momentum And Adaptive LR For SPR Prediction," *Int. Conf. Comput. Environ. Agric. Soc. Sci. Heal. Sci. Eng. Technol.*, no. 1, pp. 1–9, 2018.

[23] J. Jepkoech, D. M. Mugo, B. K. Kenduiywo, and E. C. Too, "The Effect of Adaptive Learning Rate on the Accuracy of Neural Networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 8, pp. 736–751, 2021, doi: 10.14569/IJACSA.2021.0120885.

[24] S. Syaharuddin, F. Fatmawati, and H. Suprajitno, "Best Architecture Recommendations of ANN Backpropagation Based on Combination of Learning Rate, Momentum, and Number of Hidden Layers," *JTAM (Jurnal Teor. dan Apl. Mat.*, vol. 6, no. 3, p. 629, 2022, doi: 10.31764/jtam.v6i3.8524.

[25] Badan Pusat Statistik, "Rata-Rata Harga Beras di Tingkat Perdagangan Besar (Grosir) Indonesia," 2024. https://www.bps.go.id/id/statistics-table/2/Mjk1IzI=/rata-rata-harga-beras-di-tingkat-perdagangan-besar--grosir--indonesia.html (accessed Mar. 10, 2024).