

Analysis of Backpropagation Method in Predicting Drug Stock

Elisawati^{1)*}, Arie Linarta²⁾ Al Malikul Ikhwanda Putra³⁾, Herris Elvaningsih⁴⁾

¹⁾²⁾⁴⁾STMIK Dumai, ³⁾AMIK Dumai, Riau

¹⁾elisawati06@gmail.com, ²⁾arie.linarta@email.com, ³⁾wanda18.w1@gmail.com, ⁴⁾herris.elvaningsih@gmail.com

Submitted : Jan 19, 2022 | **Accepted** : Jan 25, 2022 | **Published** : Apr 1, 2022

Abstract: Health is one of the needs of the society to be able to carry out activities in daily life. The high public need for health makes the provision of drug stocks being an important aspect for every health service, one of which is the Public Health. To find out how many drugs are received, the Public Health conducts a data collection on the drug stock that is carried out. However, the Public Health could not determine the exact amount of drug stock needed so that some drugs expired due to excess drug stock. With these problems, an analysis using the Backpropagation method was carried out to predict how much drug stock was needed. The purpose of this study is to ensure that the stock of drugs that enter the Public Health is in accordance with the needs of the Public Health so that it can reduce the expired of the drugs. For analysis using the Backpropagation Method, there are two processes used, namely: with the training process to find new weights which will later be used in the second process, namely the testing process. testing is only carried out on the forward propagation process. To carry out the training process and testing process, data from 2019-2020 that has been recapitulated will be used. The results of the analysis using the Backpropagation method show that the highest accuracy results are 88.0356% at epoch 900, learning rate is 0.001, and goal is 0.00001 with the lowest MAPE (Mean Absolute Percentage Error) is 11.964% which shows that the ability of forecasting or analysis models good because how much in the range of 10-20%

Keywords: Backpropagation, MAPE, Public Health, Training, Testing

INTRODUCTION

Public Health as community health service centers also have an important role in the provision of drugs. This makes the community always make the Public Health as a place to get the necessary drugs according to the illness they suffer. For the supply of drug stock, of course, data collection of drug stock is important to determine the amount of incoming medicine. However, the number of incoming drug stocks does not have a definite picture of how much drug stock is needed, causing some medicines to expire.

Seeing these problems, artificial neural network analysis using the backpropagation method is expected to help the Public Health to determine the exact amount of drug stock needed so that there are no expired drugs. The backpropagation method has been widely used in the analysis of artificial neural networks including being implemented in the field of education (Hizham, Nurdiansyah, and Firmansyah 2018), in the medical field (Hidayat, Rejito, and Kusuma 2017), in the economic field (Kurniawan, Wibawanto, and Widodo 2019), in the tourism sector (Aulia 2018), and various other fields.

For this reason, it is necessary to analyze using the backpropagation method so that the amount of incoming drug stock really matches the existing needs. With the aim that the amount of drug stock is in accordance with existing needs, the input entered must be in accordance with the specified criteria. To determine these results, predict how many drug stocks will enter using data from 2019-2020 which will then be processed using the backpropagation method to determine the highest accuracy obtained which will later be used as a reference to be able to predict drug stocks accurately and well.

LITERATURE REVIEW

In predicting drug stocks using the backpropagation method, analysis using the backpropagation method is applied in various research methods such as in research (Guntoro, Costaner, and Lisawita 2019; Indrayati

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Sijabat et al. 2020; Lestari, Albar, and Afwani 2019). Research conducted by (Guntoro et al. 2019) uses the backpropagation method to predict the number of vehicles in Riau Province because it increases the number of immigrants who come to Riau Province and the denser population in Riau. In research (Satria 2021) the backpropagation method is used to predict product sales every month to increase company profits. Research conducted by (Octariadi 2020) uses the backpropagation method for signature pattern recognition, which before the signature pattern is recognized, it will go through preprocessing stages in data processing for pattern recognition. Research uses the backpropagation method by (Saragih, Hartama, and Wanto 2020) to predict how much fresh milk is produced in Indonesia.

In this study to predict the stock of drugs must meet predetermined criteria consisting of 7 input variables to be able to predict the possibility of how much stock of drugs for subsequent receipts.

METHOD

Artificial Neural Network

According to (Yahya 2021), an artificial neural network is a computational system whose architecture and operations are derived from knowledge of biological nerve cells in the brain. An artificial neural network can be described as a mathematical and computational model for the nonlinear approximation function and data classification that is simulated from a collection of biological neural models.

Normalization

According to (Harani and Nugraha 2020:63), normalization is a transformation process to change data values using the Min-Max method to be able to map a value within the range of the new minimum and maximum values. Normalization can also be used to equalize the scale of data attributes into a more specific range such as 0 and 1 where to carry out the normalization process can be done using the following equation:

(Windarto and Dkk 2020:62)

$$x' = \frac{0.8(x - a)}{(b - a)} + 0.1$$

Dimana:

x' = normalization value

x = first value

a = minimum value

b = maximum value

Backpropagation Method

In the journal (Octariadi 2020), backpropagation is an algorithm that can produce a balance between the ability of the artificial neural network to recognize the patterns used during training and testing so that it can provide the correct response to similar input patterns but not the same as the patterns used during training. There are two flow calculations in the backpropagation method, namely forward propagation and backward propagation. The forward propagation used in the backpropagation method uses a binary sigmoid activation function as follows: (Elvaningsih et al. 2021)

$$f(x) = \frac{1}{1 + e^{(-x)}} \quad (1)$$

In the process of forward propagation and backward propagation in the backpropagation method, it is described as follows: (Elvaningsih et al. 2021)

Forward Propagation

The steps that must be passed in forward propagation are as follows:

Calculate the signal to be sent from the input layer to the hidden layer using the equation:

$$z_in_j = v_{oj} \sum_{i=0}^n x_i v_{ij} \quad (2)$$

Calculating the signal to be sent from the hidden layer to the output layer using the equation:

$$y_in_k = w_{ok} \sum_{j=0}^n z_j w_{jk} \quad (3)$$

Reverse Propagation

The steps that must be passed in reverse propagation are as follows::

Updating the weights connecting the output layers and hidden layers using the equation:

$$\delta_k = (t_k - y_k) y_k (1 - y_k) \quad (4)$$

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (5)$$

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (6)$$

Updating the weights connecting the hidden layer and the input layer using the equation:

$$\delta_{in_k} = \sum_{k=1}^n \delta_k w_{jk} \quad (7)$$

$$\delta_j = z_{in_j} z_j (1 - z_j) \quad (8)$$

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (9)$$

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (10)$$

Mean Absolute Percentage Error (MAPE)

In the journal (Abdianto et al. 2021) MAPE is the percentage of error used to see the absolute error in each period which will later be divided by the real observation value. MAPE uses the following equation:

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{P_t - A_t}{A_t} \right| \times 100\%$$

As follows:

n = the data amount

Pt = actual value at time t

At = forecast value at time t

MAPE values are categorized into four classifications as follows: (Abdianto et al. 2021)

TABLE 1 RANGE NILAI MAPE

No	Range	Description
1	<10%	Excellent Prediction Ability
2	10-20%	Good Prediction Ability
3	20-50%	Decent Prediction Ability
4	>50%	Poor Prediction Ability

1. RESULT

Data Collection

Data collection was carried out by direct observation to the Dumai Barat Health Center to obtain the necessary information.

TABLE 2 DATA STOCK OF MEDICINE IN PUBLIC HEALTH DUMAI BARAT

No	Medicine name	First stock	Reception	Usage	Remaining Stock	Distance	Target
1	Klorpromazin HCl tablet salut 100 mg	191	1.000	280	911	2	1.191
2	Fenobarbital tablet 30 mg	465	500	255	710	2	965
3	Glibenklamid tablet 5 mg	2340	3.000	1385	3955	2	5.340
4	Clobazam tablet	10	100	10	100	4	110
5	Clozapine tablet 25 mg	295	300	180	415	2	595
6	Metformin tablet 500 mg	1.340	4.400	1.915	3.825	2	5.740
7	Triheksifenidil Hidroklorida tablet 2 mg	505	500	560	445	2	1.005
8	Haloperidol tablet 5 mg	7	300	60	247	5	307
9	Amlodipin tablet 10 mg	1.609	3.000	1.212	3.397	2	4.609
10	Alopurinol tablet 100 mg	460	1.000	425	1035	2	1.460
11	Klorpromazin HCl tablet salut 100 mg	60	1.000	90	970	2	1.060
12	Fenobarbital tablet 30 mg	480	100	100	480	2	580
13	Glibenklamid tablet 5 mg	30	7250	1125	6155	2	7.280
14	Clobazam tablet	40	100	20	120	4	140

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

No	Medicine name	First stock	Reception	Usage	Remaining Stock	Distance	Target
15	Clozapine tablet 25 mg	296	500	143	653	2	796

In table 2 it can be seen that the data provided has 7 variables, namely the name of the drug, initial stock, receipt, usage, remaining stock, distance, and inventory. It can also be seen in table 2 that the drug name variable is not numeric data or so that in order to be processed, the data contained in the column must be translated into numbers. The following is a table showing the drug name data modeling.

TABLE 3 MODELING OF MEDICINE STOCK DATA INPUT

	X1	X2	X3	X4	X5	X6	y
	1	191	1.000	280	911	2	1.191
	2	465	500	255	710	2	965
	3	2340	3.000	1385	3955	2	5.340
	4	10	100	10	100	4	110
	5	295	300	180	415	2	595
	6	1.340	4.400	1.915	3.825	2	5.740
	7	505	500	560	445	2	1.005
	8	7	300	60	247	5	307
	9	1.609	3.000	1.212	3.397	2	4.609
	10	460	1.000	425	1035	2	1.460
	1	60	1.000	90	970	2	1.060
	2	480	100	100	480	2	580
	3	30	7250	1125	6155	2	7.280
	4	40	100	20	120	4	140
	5	296	500	143	653	2	796
minX	1	7	100	10	100	2	110
maxX	10	2340	7250	1915	6155	5	7280

Normalization Data

Normalization is done so that the data used in the training process on the backpropagation algorithm has a data range between 0 and 1 so that the results obtained are not too large. Calculations are carried out using the following equation:

$$\text{Nilai Normalisasi} = \frac{0,8 \times (\text{Nilai awal} - \text{Nilai minimal})}{\text{Nilai maksimal} - \text{Nilai minimal}} + 0,1$$

The following are the results of data normalization for all data in table 4.

TABLE 4 DATA NORMALIZATION

No	Nama Obat	Stok Awal	Penerimaan	Pemakaian	Sisa Stok	Jarak	target
1	0,1000	0,1631	0,2007	0,2134	0,2072	0,1000	0,2206
2	0,1889	0,2571	0,1448	0,2029	0,1806	0,1000	0,1954
3	0,2778	0,9000	0,4245	0,6774	0,6093	0,1000	0,6835
4	0,3667	0,1010	0,1000	0,1000	0,1000	0,6333	0,1000
5	0,4556	0,1988	0,1224	0,1714	0,1416	0,1000	0,1541
6	0,5444	0,5571	0,5811	0,9000	0,5922	0,1000	0,7282

*name of corresponding author



No	Nama Obat	Stok Awal	Penerimaan	Pemakaian	Sisa Stok	Jarak	target
7	0,6333	0,2708	0,1448	0,3310	0,1456	0,1000	0,1999
8	0,7222	0,1000	0,1224	0,1210	0,1194	0,9000	0,1220
9	0,8111	0,6493	0,4245	0,6048	0,5356	0,1000	0,6020
10	0,9000	0,2553	0,2007	0,2743	0,2235	0,1000	0,2506
11	0,1000	0,1182	0,2007	0,1336	0,2149	0,1000	0,2060
12	0,1889	0,2622	0,1000	0,1378	0,1502	0,1000	0,1524
13	0,2778	0,1079	0,9000	0,5682	0,9000	0,1000	0,9000
14	0,3667	0,1113	0,1000	0,1042	0,1026	0,6333	0,1033
15	0,4556	0,1991	0,1448	0,1559	0,1731	0,1000	0,1765

The data from the normalization results will be divided into 2 parts. The 1st to 10th data will be used as training data. The 11th to 15th data will be used as test data. Training data is data that will be used in the training process, while test data is data that will be used in the testing process after the training process is complete and test data is data that will later be predicted.

Training Process

Initialization of backpropagation neural network parameters as follows:

Number of input neurons : 6

Number of hidden neurons : 6

Number of output neurons : 1

Learning rate (α) : 0,001

Max epoch : 10

Error : 0,1

Initialization of weight V and initial W weight as follows:

bobot V = $(n_{\text{input}} + 1) + n_{\text{hidden}} = 7 + 6$

bobot W = $(n_{\text{hidden}} + 1) + n_{\text{output}} = 7 + 1$

The following are the initial V and W weights used.

TABLE 5 BOBOT V

	Z1	Z2	Z3	Z4	Z5	Z6
0	0,371	0,025	0,825	0,851	0,876	0,578
1	0,515	0,528	0,212	0,294	0,543	0,628
2	0,809	0,952	0,651	0,690	0,456	0,487
3	0,234	0,335	0,110	0,315	0,265	0,134
4	0,065	0,939	0,650	0,535	0,877	0,353
5	0,635	0,500	0,297	0,326	0,317	0,837
6	0,312	0,969	0,033	0,845	0,332	0,132

TABLE 6 BOBOT W

	1
0	0,151
1	0,055
2	0,283
3	0,919
4	0,585
5	0,106

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

	1
6	0,103

After determining the initial weight, the training process with forward propagation and backward propagation can be carried out. This will show the process of forward propagation and backward propagation in a backpropagation neural network.

Forward Propagation Process

Calculation of Neuron Value at Hidden Layer (Z Value)

Calculation of the value of Z1, Z2, Z3, Z4, Z5, Z6 is done using equation (2). The following is a calculation to find the value of Z1, Z2, Z3, Z4, Z5, Z6 for the 1st data.

$$z_{in_j} = v_{0j} \sum_{i=0}^n x_i v_{ij}$$

$$z_{in_1} = 0,371 + (0,515 * 0,1) + (0,809 * 0,1631) + (0,234 * 0,2007) + (0,065 * 0,2134) + (0,635 * 0,2072) + (0,312 * 0,1) = 0,7780$$

$$z_{in_2} = 0,025 + (0,528 * 0,1) + (0,952 * 0,1631) + (0,335 * 0,2007) + (0,939 * 0,2134) + (0,500 * 0,2072) + (0,969 * 0,1) = 0,7011$$

$$z_{in_3} = 0,825 + (0,212 * 0,1) + (0,651 * 0,1631) + (0,110 * 0,2007) + (0,650 * 0,2134) + (0,297 * 0,2072) + (0,033 * 0,1) = 1,1780$$

$$z_{in_4} = 0,851 + (0,294 * 0,1) + (0,690 * 0,1631) + (0,315 * 0,2007) + (0,535 * 0,2134) + (0,326 * 0,2072) + (0,845 * 0,1) = 1,3223$$

$$z_{in_5} = 0,876 + (0,543 * 0,1) + (0,456 * 0,1631) + (0,265 * 0,2007) + (0,877 * 0,2134) + (0,317 * 0,2072) + (0,332 * 0,1) = 1,3439$$

$$z_{in_6} = 0,578 + (0,628 * 0,1) + (0,487 * 0,1631) + (0,134 * 0,2007) + (0,353 * 0,2134) + (0,837 * 0,2072) + (0,132 * 0,1) = 1,0090$$

Calculation of Neuron Value at the Output Layer (Y Value)

The calculation of the value of Y is done using equation (3). The following is a calculation to find the value of Y in the 1st data.

$$y_{in_k} = w_{0k} \sum_{j=0}^n z_j w_{jk}$$

$$y_{in} = 0,151 + (0,6853 * 0,055) + (0,6684 * 0,283) + (0,7646 * 0,919) + (0,7896 * 0,585) + (0,7931 * 0,106) + (0,7328 * 103) = 1,7020$$

$$y_j = f(y_{in_k}) = \frac{1}{1 + e^{(-x)}}$$

$$y = f(y_{in_k}) = \frac{1}{1 + e^{(-y_{in})}} = \frac{1}{1 + e^{(-1,7020)}} = 0,8458$$

After getting the Y value, the forward propagation process ends/completes for the 1st data.

Reverse Propagation Process

Reverse Propagation from Output Layer to Hidden Layer (new W)

The update of the weight W on backward propagation is carried out using equations (4), (5), (6). Here is the calculation for updating the value of the W weights.

$$\delta_k = (t_k - y_k) y_k (1 - y_k)$$

$$\delta_1 = (0,151 - 1,7020) * 1,7020 * (1 - 1,7020) = -0,0906$$

$$\Delta w_{0k} = \alpha \delta_k$$

$$\Delta w_{01} = (0,001 * -0,0906) = -0,0000906206$$

$$\Delta w_{jk} = \alpha \delta_k z_j$$

$$\Delta w_{11} = (0,001 * -0,0906 * 0,6853) = -0,0000620981 \text{ seterusnya hingga}$$

$$\Delta w_{21} = (0,001 * -0,0906 * 0,6684) = -0,0000605746$$

$$\Delta w_{31} = (0,001 * -0,0906 * 0,7646) = -0,0000692871$$

$$\Delta w_{41} = (0,001 * -0,0906 * 0,7896) = -0,0000715515$$

$$\Delta w_{51} = (0,001 * -0,0906 * 0,7931) = -0,0000718734$$

$$\Delta w_{61} = (0,001 * -0,0906 * 0,7328) = -0,0000664096$$

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$$

From the above calculation, the new W weights are obtained as follows:

TABLE 7 WEIGHT W NEW

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

	Wbaru
0	0,1509
1	0,0549
2	0,2829
3	0,9189
4	0,5849
5	0,1059
6	0,1029

Reverse Propagation from Hidden Layer to Input Layer (new V)

The update of the weight V on backward propagation is carried out using equations (7), (8), (9), (10). The following is the calculation for updating the value of the V weight on the 1st data.

$$\delta_{in_k} = \sum_{k=1}^n \delta_k w_{jk}$$

$$\delta_{in_1} = (-0,0906 * 0,055) = -0,004984135$$

$$\delta_{in_2} = (-0,0906 * 0,283) = -0,02564564$$

$$\delta_{in_3} = (-0,0906 * 0,919) = -0,083280365$$

$$\delta_{in_4} = (-0,0906 * 0,585) = -0,053013073$$

$$\delta_{in_5} = (-0,0906 * 0,106) = -0,009605788$$

$$\delta_{in_6} = (-0,0906 * 0,103) = -0,009333926$$

$$\delta_j = \delta_{in_j} f'(z_{in_j}) = \delta_{in_j} z_j (1 - z_j)$$

$$\delta_1 = (-0,004984135 * 0,6853 * (1 - 0,6853)) = -0,001074985$$

$$\delta_2 = (-0,02564564 * 0,6684 * (1 - 0,6684)) = -0,005683777$$

$$\delta_3 = (-0,083280365 * 0,7646 * (1 - 0,7646)) = -0,014990087$$

$$\delta_4 = (-0,053013073 * 0,7896 * (1 - 0,7896)) = -0,008808015$$

$$\delta_5 = (-0,009605788 * 0,7931 * (1 - 0,7931)) = -0,001576099$$

$$\delta_6 = (-0,009333926 * 0,7328 * (1 - 0,7328)) = -0,001827489$$

$$\Delta v_{0j} = \alpha \delta_j$$

$$\Delta v_{01} = 0,001 * -0,001074985 = -1,07499E - 06$$

$$\Delta v_{02} = 0,001 * -0,005683777 = -5,68378E - 06$$

$$\Delta v_{03} = 0,001 * -0,014990087 = -1,49901E - 05$$

$$\Delta v_{04} = 0,001 * -0,008808015 = -8,80801E - 06$$

$$\Delta v_{05} = 0,001 * -0,001576099 = -1,5761E - 06$$

$$\Delta v_{06} = 0,001 * -0,001827489 = -1,82749E - 06$$

$$\Delta v_{ij} = \alpha \delta_j x_i$$

$$\Delta v_{11} = 0,001 * -0,001074985 * 0,1 = -1,07499E - 07$$

$$\Delta v_{12} = 0,001 * -0,005683777 * 0,1 = -5,68378E - 07$$

$$\Delta v_{13} = 0,001 * -0,014990087 * 0,1 = -1,49901E - 06$$

$$\Delta v_{14} = 0,001 * -0,008808015 * 0,1 = -8,80801E - 07$$

$$\Delta v_{15} = 0,001 * -0,001576099 * 0,1 = -1,5761E - 07$$

$$\Delta v_{16} = 0,001 * -0,001827489 * 0,1 = -1,82749E - 07$$

$$\Delta v_{21} = 0,001 * -0,001074985 * 0,1 = -1,75324E - 07$$

So on until the weight Δv_{66}

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

From the above calculation, the new weight V is obtained as follows:

TABLE 8 WEIGHT V NEW

	Vbaru1	Vbaru2	Vbaru3	Vbaru4	Vbaru5	Vbaru6
0	0,3710	0,0250	0,8250	0,8510	0,8760	0,5780
1	0,5150	0,5280	0,2120	0,2940	0,5430	0,6280
2	0,8090	0,9520	0,6510	0,6900	0,4560	0,4870
3	0,2340	0,3350	0,1100	0,3150	0,2650	0,1340

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

4	0,0650	0,9390	0,6500	0,5350	0,8770	0,3530
5	0,6350	0,5000	0,2970	0,3260	0,3170	0,8370
6	0,3120	0,9690	0,0330	0,8450	0,3320	0,1320

After updating the W and V weights, the training process for the 1st data has been completed. Furthermore, the training process for the 2nd data is carried out using the same way as the training on the 1st data. The weights used for the 2nd data training are the latest weights obtained from the 1st data. Likewise for the 3rd data training, the weights used are the latest weights from the 2nd data training results. And so on for the next data.

Because of the maximum epoch used is 10 epochs, the training process is carried out until the 10th epoch. Once all training on all data has been completed, the training process for the 10 epoch iteration cycle has been completed. Next is the process of calculating the average error from the training results in epochs 1-10. The following error is obtained from the training process obtained from the difference between the target output and ANN output.

TABLE 9 TRAINING RESULT ERROR

Data Ke	Prediksi	Target Output	Error
1	0,8612	0,2206	0,6406
2	0,8608	0,1954	0,6654
3	0,8603	0,6835	0,1768
4	0,8598	0,1000	0,7598
5	0,8594	0,1541	0,7053
6	0,8589	0,7282	0,1307
7	0,8584	0,1999	0,6586
8	0,8580	0,1220	0,7360
9	0,8575	0,6020	0,2555
10	0,8612	0,2506	0,6106
Average			0,5339

Based on the data in the table above, the average error from the training results is 0.5339. The average error from the training results from 10 iterations is greater than the set goal or error tolerance, which is 0.1. Therefore, the training will end if the error obtained is less than or equal to the set goal or when the epoch has been completed.

Testing Process

In the testing/prediction process, the artificial neural network only performs the forward propagation process. For the testing process, the normalization of the test data in table 4 is used from the 11th to the 15th data. The following are the steps taken to predict drug stock from the normalization of test data to the 11th test data.

Calculation of Neuron Value at Hidden Layer (Z Value)

Calculation of the value of Z1, Z2, Z3, Z4, Z5, Z6 is done using equation (2). The following is a calculation to find the value of Z1, Z2, Z3, Z4, Z5, Z6 for the 11th data

$$z_{in_j} = v_{0j} \sum_{i=0}^n x_i v_{ij}$$

$$z_{in_1} = 0,3709 + (0,515 * 0,1) + (0,809 * 0,1182) + (0,234 * 0,2007) + (0,065 * 0,1336) + (0,635 * 0,2149) + (0,312 * 0,1) = 0,7413$$

$$z_{in_2} = 0,0246 + (0,5278 * 0,1) + (0,9519 * 0,1182) + (0,3349 * 0,2007) + (0,9389 * 0,1336) + (0,4999 * 0,2149) + (0,9689 * 0,1) = 0,7742$$

$$z_{in_3} = 0,8238 + (0,2114 * 0,1) + (0,6507 * 0,1182) + (0,1098 * 0,2007) + (0,6497 * 0,1336) + (0,2967 * 0,2149) + (0,0327 * 0,1) = 1,1028$$

$$z_{in_4} = 0,8504 + (0,2937 * 0,1) + (0,6898 * 0,1182) + (0,3149 * 0,2007) + (0,5348 * 0,1336) + (0,3259 * 0,2149) + (0,8449 * 0,1) = 1,1795$$

$$z_{in_5} = 0,8759 + (0,543 * 0,1) + (0,456 * 0,1182) + (0,265 * 0,2007) + (0,877 * 0,1336) + (0,317 * 0,2149) + (0,332 * 0,1) = 1,0928$$

*name of corresponding author



$$z_{in_6} = 0.5779 + (0,6279 * 0,1) + (0,487 * 0,1182) + (0,134 * 0,2007) + (0,353 * 0,1336) + (0,837 * 0,2149) + (0,132 * 0,1) = 0,7410$$

Calculation of Neuron Value at the Output Layer (Y Value)

The calculation of the value of Y is done using equation (3). The following is a calculation to find the Y value for the 11th data.

$$y_{in_k} = w_{ok} \sum_{j=0}^n z_j w_{jk}$$

$$y_{in} = 0,1433 + (0,6773 * 0,0491) + (0,6844 * 0,2769) + (0,7508 * 0,9128) + (0,7649 * 0,5784) + (0,7489 * 0,0994) + (0,6772 * 0,0968) = 1,6338$$

$$y_j = f(y_{in_k}) = \frac{1}{1 + e^{(-x)}}$$

$$y = f(y_{in_k}) = \frac{1}{1 + e^{(-y_{in})}} = \frac{1}{1 + e^{(-1,6338)}} = 0,8367$$

From the results of the above calculation, it can be seen that the prediction result of the backpropagation neural network for the 11th test data is 0.8367. The actual target is 0.2060 so the error between the prediction result and the target is 0.6307. The following table shows the prediction results for all test data.

TABLE 10 RESULTS OF TEST DATA PREDICTION

Data Ke	Prediksi	Target	Error
1	0,8367	0,2060	0,6307
2	0,8335	0,1524	0,6810
3	0,8704	0,9000	0,0296
4	0,8437	0,1033	0,7404
5	0,8355	0,1765	0,6589

Calculating MAPE (Mean Absolute Percentage Error)

After getting the error value, then the MAPE value is calculated to determine the percentage of absolute error using the following equation:

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{Pt - At}{At} \right| \times 100\%$$

The following is a table showing the MAPE results for all test data.

TABLE 11 MAPE AND ACCURACY TEST DATA

Data Ke	Prediction	Target	Error	MAPE%	Accurate%
1	0,8367	0,2060	0,6307	306,17	100 – MAPE = -206,17
2	0,8335	0,1524	0,6810	446,76	100 – MAPE = -346,76
3	0,8704	0,9000	0,0296	3,29	100 – MAPE = -96,71
4	0,8437	0,1033	0,7404	716,40	100 – MAPE = -616,40
5	0,8355	0,1765	0,6589	373,25	100 – MAPE = -273,25
Average				369,17	-269,17

Based on the table above, the average MAPE obtained is 369.17% while the accuracy obtained is -273.25%. This shows that the MAPE error percentage obtained is still quite high and is not feasible because it is in the 4th MAPE category, which is > 50%, while the accuracy is very low so it is necessary to carry out the training process again to get better results. For this reason, so that the artificial neural network using the backpropagation method shows the best and accurate results, the data is added to 70 data taken from 2019-2020 data from 10 types of drugs. For training data used 42 data and for test data used 28. The following is an analysis of the results of the classification of test data after analysis using various initialization of artificial neural network parameters.

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

TABLE 12 RESULTS OF CLASSIFICATION OF DRUG STOCK PARAMETERS

No	Epoch	Learning Rate	Goal	Neuron Hidden Layer	MAPE	Akurasi	MSE
1	100	0,001	0,00001	6	52,5760	47,424	0,1058
2	100	0,001	0,00001	7	51,5260	48,474	0,11547
3	100	0,001	0,00001	8	55,4833	44,5167	0,1219
4	100	0,001	0,00001	9	43,0661	56,9339	0,11234
5	100	0,001	0,00001	10	75,4360	24,564	0,12939
6	300	0,001	0,00001	6	18,6821	81,3179	0,11945
7	300	0,001	0,00001	7	12,3610	87,639	0,10779
8	300	0,001	0,00001	8	16,8605	83,1395	0,10657
9	300	0,001	0,00001	9	15,7986	84,2014	0,12619
10	300	0,001	0,00001	10	19,0231	80,9769	0,11268
11	500	0,001	0,00001	6	15,9018	84,0982	0,12469
12	500	0,001	0,00001	7	14,6590	85,341	0,11049
13	500	0,001	0,00001	8	16,9508	83,0492	0,11827
14	500	0,001	0,00001	9	16,4840	83,516	0,12297
15	500	0,001	0,00001	10	17,3720	82,628	0,10967
16	700	0,001	0,00001	6	14,7204	85,2796	0,11004
17	700	0,001	0,00001	7	15,5884	84,4116	0,11693
18	700	0,001	0,00001	8	14,7354	85,2646	0,10922
19	700	0,001	0,00001	9	12,8867	87,1133	0,11686
20	700	0,001	0,00001	10	16,9108	83,0892	0,12078
21	900	0,001	0,00001	6	12,0585	87,9415	0,12
22	900	0,001	0,00001	7	14,9849	85,0151	0,11082
23	900	0,001	0,00001	8	13,3184	86,6816	0,1131
24	900	0,001	0,00001	9	11,964	88,0356	0,10426
25	900	0,001	0,00001	10	13,3439	86,6561	0,11178
26	1000	0,001	0,00001	6	12,7391	87,2609	0,1185
27	1000	0,001	0,00001	7	14,3734	85,6266	0,12761
28	1000	0,001	0,00001	8	14,8787	85,1213	0,12271
29	1000	0,001	0,00001	9	14,9088	85,0912	0,10982
30	1000	0,001	0,00001	10	14,8318	85,1682	0,10814

DISCUSSIONS

The results that have been achieved from 70 drug data taken from 2019-2020 with 42 training data and 28 test data in table 12, namely the lowest MAPE value is 11.964% with the highest accuracy of 88.0356%. From the test results, it can be seen that the highest accuracy and the lowest MAPE are in the number of epochs of 900 with a learning rate of 0.001 and a goal of 0.00001.

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

CONCLUSION

From the results of the analysis, it can be concluded that the lowest error value in predicting the drug stock period can be seen from the Mean Absolute Percentage Error (MAPE) value with the lowest value of 11.964% with an accuracy value of 88.0356%. The error value is obtained by combining training parameters using 9 neuron hidden layers, 900 epochs, 0.001 learning rate and 0.00001 goal. With the MAPE value obtained, this proves that the results of the analysis using 70 data consisting of 42 training data and 28 test data to obtain an artificial neural network with good predictive modeling because the MAPE value of 11.964% is in the range of 10-20% which is a good predictive modeling.

REFERENCES

- Abdianto, D., Elisawati, E., Tawakal, F., & Masrizal, M. (2021). Prediksi Stok Obat Menggunakan Metode Learning Vector Quantization Studi Kasus Puskesmas Dumai Barat. *Prosiding SNST Fakultas Teknik*, 1(1).
- Aulia, R. (2018). Penerapan Metode Backpropagation Untuk Memprediksi Jumlah Kunjungan Wisatawan Berdasarkan Tingkat Hunian Hotel. *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, 4(2), 115-122.
- Elvaningsih, H., Elisawati, E., Tawakal, F., & Masrizal, M. (2021, October). Prediksi Stok Obat Menggunakan Metode Backpropagation (Studi Kasus: Puskesmas Dumai Barat). In *Seminar Nasional Sains dan Teknologi Informasi (SENSASI)* (Vol. 3, No. 1, pp. 228-232).
- Guntoro, G., Costaner, L., & Lisawita, L. (2019). Prediksi Jumlah Kendaraan di Provinsi Riau Menggunakan Metode Backpropagation. *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, 14(1), 51-57.
- Harani, N. H., & Nugraha, F. A. (2020). *Segmentasi Pelanggan Menggunakan Python*. Kreatif.
- Hizham, F. A., & Nurdiansyah, Y. (2018). Implementasi metode backpropagation neural network (bnn) dalam sistem klasifikasi ketepatan waktu kelulusan mahasiswa (studi kasus: Program studi sistem informasi universitas jember). *Berkala Sainstek*, 6(2), 97-105.
- Kurniawan, E., Wibawanto, H., & Widodo, D. A. (2019). Implementasi Metode Backpropagation dengan Inisialisasi Bobot Nguyen Widrow untuk Peramalan Harga Saham. *J. Teknol. Inf. dan Ilmu Komput*, 6(1), 49.
- Lestari, K. T. N., Albar, M. A., & Afwani, R. (2019). Penerapan Metode Backpropagation Dalam Memprediksi Jumlah Kunjungan Wisatawan Ke Provinsi Nusa Tenggara Barat (NTB). *Journal of Computer Science and Informatics Engineering (J-Cosine)*, 3(1), 39-48.
- Octariadi, B. C. (2020). Pengenalan Pola Tanda Tangan Menggunakan Metode Jaringan Syaraf Tiruan Backpropagation. *Jurnal Teknoinfo*, 14(1), 15-21.
- Rejito, J., & Kusuma, S. P. (2016). Metode Back Propagation pada Jaringan Syaraf Tiruan Digunakan untuk Penilaian Kemampuan Bertahan pada Bayi dengan Berat Lahir Rendah. *Jurnal Informatika*, 1(1), 53-60.
- Saragih, J. R., Hartama, D., & Wanto, A. (2020). Prediksi Produksi Susu Segar Di Indonesia Menggunakan Algoritma Backpropagation. *Jurnal Ilmiah Informatika*, 8(01), 59-65.
- Satria, W. (2020). Jaringan Syaraf Tiruan Backpropagation Untuk Peramalan Penjualan Produk (Studi Kasus Di Metro Electronic Dan Furniture). *Djtechno: Jurnal Teknologi Informasi*, 1(1), 14-19.
- Sijabat, P. I., Yuhandri, Y., Nurcahyo, G. W., & Sindar, A. (2020). Algoritma Backpropagation Prediksi Harga Komoditi terhadap Karakteristik Konsumen Produk Kopi Lokal Nasional. *Digital Zone: Jurnal Teknologi Informasi Dan Komunikasi*, 11(1), 96-107.
- Windarto, A. P., Nasution, D., Wanto, A., Tambunan, F., Hasibuan, M. S., Siregar, M. N. H., ... & Nofriansyah, D. (2020). *Jaringan Saraf Tiruan: Algoritma Prediksi dan Implementasi*. Yayasan Kita Menulis.
- Yahya, A. I., & Siahaan, R. F. (2020). Penerapan Jaringan Syaraf Tiruan Untuk Memprediksi Kemampuan Konsumen Dalam Mencicil Kendaraan Roda Dua Di Adira Finance Lubuk Pakam Dengan Metode Backpropagation. *Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI)*, 3(1.1), 40-47.

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.