Classification of hypertension disease using Artificial Neural Network (ANN) backpropagation method case study in mitigating health risk: UPT Modopuro Mojokerto Health Center

Eka Mala Sari Rochman [1*], *Nina* Indriawati [1], *Ika Oktavia* Suzanti[1], *Wahyudi* Setiawan[1], *Husni*[1], *Muhammad Izzat Nor* Ma'arof[2], and *Aeri* Rachmad[1]

[1]Departemen of Informatics, Faculty of Engineering, University of Trunojoyo Madura, Kamal, Bangkalan, Indonesia
[2]Faculty of Engineering and Quantity Surveying, INTI International University, Negeri Sembilan 71800, Malaysia

Abstract. Hypertension is a disease caused by increased blood pressure above 140/90 mmHg and is often referred to as "the silent killer" because most sufferers do not realize that they have hypertension, and only realize when complications have occurred. Hypertension is one of the main causes of death worldwide which can be influenced by many factors. In UPT (Integrated Service Unit) PUSKESMAS (Community Health Center) Modopuro, Mojokerto Regency, hypertension is ranked among the top 10 diseases with the most patients. With a fairly high risk of death and an increase in the number of people with hypertension, it is often caused by delays in diagnosis, which must be carried out blood pressure checks by medical personnel at least 2 times with 1 week to establish a diagnosis of hypertension. If hypertension is not treated immediately, it can cause other health conditions such as kidney disease, heart disease, and stroke. Therefore, a system is needed that can be used for the classification of early detection of whether a person has hypertension or not. To overcome these problems, a system was created to classify hypertension using the Backpropagation method. Backpropagation is very effective in helping artificial neural networks learn from mistakes, allowing the system to make more accurate predictions over time. Dataset used in this study is the medical record data of UPT Puskesmas Modopuro patients with 1000 data. The results obtained the best model with a network structure of 7-5-1, learning rate 0.001, and Adam optimizer. With an accuracy of 93.50% and a loss value of 0.0697. While the precision, recall, and f1-score values are 94.00%, 93.00%, and 93.00%, respectively. With good accuracy performance, indicating that the backpropagation model can be applied in hypertension classification.

1 Introduction

Hypertension is one of the main factors of many deaths in the world. According to WHO (World Health Organization) in 2015, stated that approximately 1.13 billion people around the world experience hypertension, in the sense that as many as 1/3 of people in the world are diagnosed with hypertension. Every year the number of people with hypertension continues to grow, it can even be estimated that later in 2025, people who experience hypertension will increase to 1.5 billion, and in each year it is estimated that as many as 9.4 million people die due to hypertension and its complications [1].

The problem of hypertension is always increasing. In 2018, according to the results of Riskesdas (Basic Health Research), the prevalence of hypertensive patients was 34.1% of the total population of Indonesia [2]. Based on the data, the prevalence of hypertension from 2013 to 2018 in East Java Province has increased quite high, which previously in 2007 to 2013 experienced a decrease. [3]. In Mojokerto Regency itself, hypertension is ranked among the top ten diseases in Mojokerto Regency, especially in UPT Puskesmas Modopuro, Mojosari District, Mojokerto Regency [4].

Hypertension is closely related to a patient's blood pressure. Early detection of hypertension is carried out by checking blood pressure by medical personnel at least 2 times with a distance of 1 week to establish a diagnosis of hypertension. [5]. This will certainly require a lot of time, material, and energy. On the other hand, the increase in the number of people with hypertension is often caused by delays in the diagnosis of the disease. Therefore, a low-cost classification system is needed that is able to provide a quick diagnosis in determining whether someone has hypertension or not.

To get an accurate classification, several methods can be used. Among them is classification using the Backpropagation algorithm. Classification using the Backpropagation Artificial Neural Network has been carried out in many studies. As in the research to diagnose hypertension for the sustainability of public health into 4 classes including normal, high, stage 1

* Corresponding author :em_sari@trunojoyo.ac.id

hypertension, and stage 2 hypertension. In this study, 303 data were used with 9 features including age, DxTx (Glucose), FC (Heart Frequency / Pulse), FR (Respiratory Frequency), temperature, systolic, diastolic, smoking, and BMI. And obtained quite high accuracy results of 90%. [6]. Furthermore, other research compared the J48, Naive Bayes, SVM and Backpropagation Neural Network methods in a classification system. The research results show that the Backpropagation Neural Network is superior to other methods used in this research. In his research, the classification of Diabetes was carried out with the accuracy results obtained for the SVM of 81.69%, J48 of 78.26%, Naive Bayes of 78.97% and finally Backpropagation Neural Network method obtained an accuracy of 83.11% [7].

In another study, the application of the Backpropagation method in the classification of diabetes mellitus disease using a case study on the dataset in Pima, India. In this study, the highest accuracy result was 96.35% [8]. In other cases, the Backpropagation method is also used in classifying breast cancer. A dataset of 569 breast cancer patients was used. The results obtained in this research are accuracy of 96.929% with an error of 3.071% with a combination of training parameters with epoch 1000, learning rate 0.01, target 0.001, and hidden layer 5 [9]. Furthermore, the research analyzes the use of Backpropagation Neural Networks to predict heart disease. In this study, using a dataset of 166 using an architecture of 13 input screens, 8-5-2 for the hidden layer and 1 for the output layer, the best accuracy was obtained at 92% [10].

This research proposes a Backpropagation Artificial Neural Network method that works like the human brain. Backpropagation is very effective in helping artificial neural networks learn from mistakes, allowing the system to make more accurate predictions over time. Backpropagation can adapt and learn from the input data generated so that it can map the relationship between input and output. In addition, Backpropagation Artificial Neural Networks can predict outputs based on pre-trained inputs. [10].

In the above research, it is known that the use of the Backpropagation Artificial Neural Network method gets better results than the use of other methods, as mentioned above. And it is known that the addition of momentum can speed up the iteration process, so the training process becomes faster due to jumps approaching the position being sought. [10]. In some studi it is also explained that the use of the Backpropagation algorithm has several advantages, including being able to overcome problems in training classification models with a wide ratio of data used and strong against missing data [8]. Therefore, the application of the Backpropagation Artificial Neural Network method algorithm is carried out. The goal is to classify patients suffering from hypertension or not in the Mojokerto area, East Java.

From several studies described above, the main contribution of this research is to apply the backpropagation model in classifying hypertension in patients at the Modopuro health center. So that patients who have symptoms of hypertension can be treated immediately before potentially developing other diseases.

2 Methods

This chapter will explain the meaning of hypertension, the use of the Backpropagatian method, k-vold cross validation and evaluation using the confusion matrix.

In Fig. 1, the system flow for carrying out the classification process from data input to evaluation is shown. In the first process, hypertension data was obtained from the UPT of the Modopuro Community Health Center, Mojokerto. Next, the data preprocessing stage is carried out, where data cleaning is carried out and data transformation.

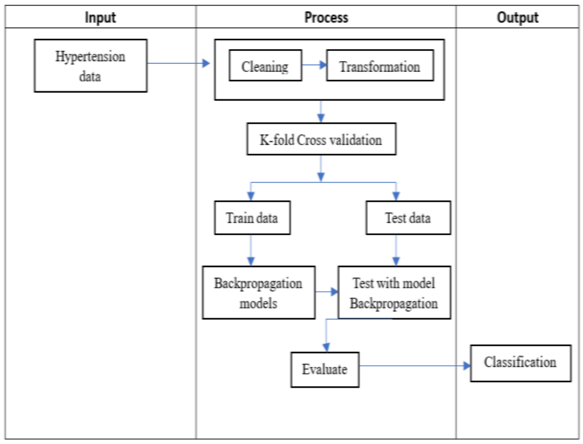

Fig. 1. System Flow.

After that, the data is divided into training data and testing data. Training Data divided using K-Fold Cross Validation, the goal is to validate the training data to obtain the best fold model. Next, modeling is carried out using the Backpropagation algorithm. Then the model is tested using testing data. The next evaluation is carried out using the Confusion Matrix. The output is in the form of classification results using Backpropagation.

2.1 Hypertension

Based on the definition of the World Health Organization (WHO) hypertension (High Blood Pressure) is a condition where there is an increase in systolic blood pressure ≥ 140 mmHg or diastolic ≥ 90 mmHg. Hypertension is commonly called "the silent killer", because most sufferers do not realize they have hypertension, and only realize when complications have occurred. And it is known that of people with hypertension, only one-third or 36.8% of people with hypertension are diagnosed by medical personnel and about only 0.7% take medication. [11].

2.2 Classification

Classification is the grouping of similar things and the separation of dissimilar things to group objects according to their reference parameters. Classification can also be defined as the clear division of an object or concept into classes based on the general similarities of the object and the things that distinguish it. In short, classification is the grouping of objects based on the same characteristics or reference parameters based on certain functions or rules. The model is a mathematical rule or formula [12].

2.3 Data precosessing

Data preprocessing is the stage of processing initial data into a data model that is easy to understand before further use. The purpose of data preprocessing is to correct if there are errors in the raw data. In data preprocessing there is a data validation process that aims to measure the level of data completeness.

2.4 K-Fold cross validation

Cross Validation, also known as cross validation, is a model validation technique that is useful for evaluating how the results of statistical analysis are generalized to an independent data set. K-Fold Cross Validation is used to measure the performance of the algorithm more generally by dividing the dataset into training data and testing data. This can help identify whether the model has a good level of accuracy, as well as help avoid overfitting or underfitting. A larger k value indicates a smaller bias. One of the cross-validation methods is k-fold cross validation. [13].

2.5 Artificial neural network

An Artificial Neural Network (ANN) is an algorithm designed to replicate the functionality of human neural networks. Like the human brain, an ANN is made up of numerous interconnected neurons that transmit signals [14]. In a human nerve cell, the soma (cell body) serves as the central point for receiving and integrating nerve impulses. Dendrites are responsible for gathering incoming messages to be sent to the soma (input and processing), while the axon (neurite) transmits the nerve impulses to other nerve cells (output).

2.6 Backpropagation algorithm

The standard structure of the Backpropagation algorithm includes three layers: the input layer, where data is introduced; the hidden layer, where data is processed; and the output layer, which produces the results based on the processed data [10]. The following is the architecture of Backpropagation in the classification of hypertension.

The artificial neural network architecture in this study uses 7 input neurons adjusted to the number of attributes in the data, then in the hidden layer using several trials starting from 5-25 neurons and one output class. This can be seen in Fig. 2

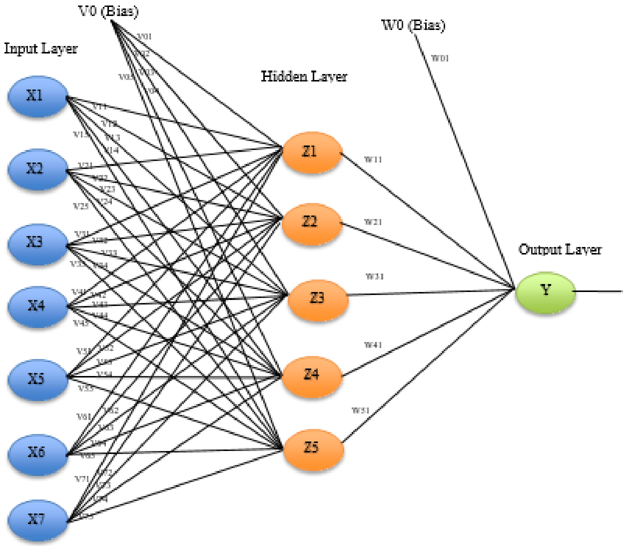

Fig. 2. Backpropagation architecture.

In Fig. 2, there are 2 processes in the Backpropagation algorithm, namely:

1. Training Process

This process aims to recognize data patterns. In this process, it is possible to define iterations in forward propagation and backward propagation, with the aim of obtaining trained weights and biases. The output error in the Backpropagation algorithm serves to change the weight value in backward propagation. To obtain the error value, it must first work on the forward propagation stage.

2. Testing Process

This process is the process that is done on the testing data. At this stage, only the forward propagation process is carried out.

Some of the steps carried out in this research in the forward propagation and backward propagation of the Backpropagation algorithm will be described as follows.

2.6.1 Backpropagation forward propagation stage

1. Set the parameters used such as the number of input layer neurons, hidden layer neurons, output layer neurons, number of iterations, learning rate, and error limit.
2. Random weight initialization
3. Each input neuron (X_0, $X_1, \dots, X_i$) where i is the number of input neurons. Each input neuron gets a weight w_{ij} and goes to each hidden layer neuron.
4. Each hidden neuron ($Y_0, Y_1, \dots, Y_j$) where j is the number of hidden neurons.
5. Implementing a sigmoid activation function to calculate the output signal.
6. Each output signal in the hidden layer is calculated by the weights connecting the hidden to the output layer neurons. (O_k) where k is the number of output neurons
7. Implementing the activation function to calculate the output signal.

2.6.2 Backpropagation backward propagation stage

In the backward propagation stage, the process of updating the weights that connect the output layer and hidden layer is carried out. And also update the weights that connect the hidden layer and input layer. Below are the steps in the backward propagation stage.

8. Each output layer ($O_0, O_1, \ldots, O_k$) where k is the number of output neurons. Obtaining the target pattern from the input
9. Then calculate the weight correction / delta weight between the hidden layer and output layer (w_{jk}).
10. Each hidden layer sums the input deltas (from each unit in the layer above it)
11. Then calculate the weight correction / delta weight between the hidden layer and the input layer
12. Each outer layer neuron The bias and weights are changed
13. Each outer layer neuron The bias and weights are changed
14. Each output is compared with the target pattern t_kdata , to get the overall
15. If the iteration reaches the error limit or meet the maximum epoch limit then the process will stop.

In the Forward Propagation Stage, after obtaining the output activation results on the output layer, the Hypertension data has 2 classes, namely class 0 (no) and class 1 (yes). Therefore, to determine the results of the output classification if it does not reach the yes/no class, the Threshold parameter is given as equation (1) [15] :

$$\begin{array}{l} If\ O_k \leq\ 0.5, then\ O_k =\ 0 \\ If\ >\ O_k 0.5, then\ O_k =\ 1 \end{array} \tag{1}$$

Describes:

O_k = Result of output activation in the kth output layer

In the model training process, an optimizer is used to optimize or minimize the loss function in the model by updating the weights and biases based on the gradient of the loss function. The loss function is used to measure the extent to which the prediction model approaches the actual value.

Learning rate is one of the parameters in the process training. Its function is to calculate the weight correction value during the training process. When the learning rate value is higher, the training process will be faster. The learning rate parameter ranges from 0 to 1. Learning rate value which determines how important the latest information is compared to previous information during the learning process.

The activation function is a useful function in activating or not a neuron in an Artificial Neural Network. There are several activation functions, namely sigmoid, tanh, linear, relu, gaussian, and others.

2.7 Confusion matrix evaluation

Confusion Matrix is a performance measurement for machine learning which will produce 2 classes or more [16, 17]. Confusion matrix is an important performance evaluation tool in data mining classification, providing a comprehensive overview of the model's prediction results. In this matrix, the main elements include True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

Confusion Matrix can also be interpreted as a table with 4 combinations with predicted value results and actual value results. The 4 values that are the reference for the Confusion Matrix are True Positive, True Negative, False Positive and False Negative. The confusion matrix can be seen in Table 1 below [18, 19].

Table 1. Confusion matrix test.

		Actual Class	
		Positive	Negative
Prediction Class	Positive	TP (True Positive)	FP (False Positive)
	Positive	FN (False Negative)	TN (True Negative)

The formula for accuracy is as follows:

1. Accuracy:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{2}$$

2. Precision:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\% \tag{3}$$

3. Recall:

$$\text{Recall} \quad = \frac{\text{TN}}{\text{TN} + \text{FN}} \text{x}\, 100\% \tag{4}$$

4. The F1-measure is the harmonic mean of precision and recall, with a range from 0 to 1:

$$\text{F} - \text{measure} \ = 2\ \text{x}\ \frac{\text{Recall}\ \times\ \text{Precision}}{\text{Recall} + \text{Precision}} \text{x}\, 100\% \tag{5}$$

Describes :

TP (True Positive): Total positive cases classified as positive

FP (False Positive): Total negative cases classified as positive

TN (True Negative): Total negative cases classified as negative

FN (False Negative): Total positive cases classified as negative

3 Result and discussion

3.1 Data collection

In this research, the data used is data from the Modopuro Community Health Center UPT, Mojosari District, Mojokerto Regency, East Java Province. With a period of 5 months, namely May – September 2022 with a total of 500 data for Hypertension and 500 data for non-hypertension. In total, the amount of data used is 1000 data. Then choose the parameters to be used as input data and n as target data. The following is an example of a sample of hypertension data from the Modopuro Community Health Center UPT, Mojokerto.

3.2 Preprocessing stage

The next stage in this research is preprocessing hypertension data. This stage is used to process initial data into a data model that is easy to understand before further use. Several preprocessing stages are as follows:

3.2.1 Data cleaning

In this section, we clean datasets that have no relevance to risk factors for hypertension, such as deleting unnecessary units. Data cleansing helps in identifying and correcting errors that arise while processing data from multiple sources. By removing inconsistencies and inaccuracies, data cleansing can ensure that a data set is reliable. The advantages and benefits of data cleansing are Having clean data will ultimately increase overall productivity and enable high-quality information in your decision making.

3.2.2 Data transformation

Data transformation is the process of modifying data to suit a specific purpose or context. This can involve tasks such as cleansing, standardization, or aggregation, all of which aim to improve the quality and usability of the data for more accurate analysis.

This section there is a process of changing the data to in the form or format that will be processed by the system, with the aim of making it easier to manage the data. Such as handling categorical data in the gender column using one-hot encoding, and min-max normalization to change numerical data into a range of zero to one [0-1] and transformation of the hypertension label YES/NO to 1/0.

3.2.3 Backpropagation classification

This stage is carried out by learning a classification model, namely grouping hypertension disease data based on existing labels, namely classes yes hypertension and no hypertension. Based on the implementation of the Backpropagation classification stage, the output in the form of a confusion matrix on the test data is produced as follows.

Based on the Fig. 3 confusion matrix in the image above, it shows that the total testing data used is 200. From the image above, the confusion matrix values are produced, TP=90, FP=5, TN=97, and FN=8. So based on these results, the accuracy value is obtained using the following (2):

$$Accuracy = \frac{90 + 97}{90 + 5 + 97 + 8} \times 100\% = 93.50\%$$

The accuracy obtained was 93.50%.

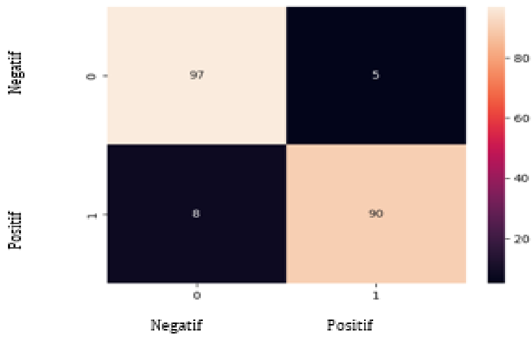

Fig. 3. Confusion matrix.

3.3 Test results

In this research, several trials were carried out to classify hypertension disease data using the Backpropagation Neural Network method. The first test scenario is a learning rate test with the previous parameters and the Adam optimizer. Next, for the second test, namely the number of epoch test, where the parameters used for the test are the best parameters obtained in the previous test scenario.

The third is the activation function trial, where the parameters used for testing are the parameters obtained in the previous trial. The use of the right activation function greatly affects the performance of backpropagation in increasing accuracy values. Likewise with the learning rate value which determines how important the latest information is compared to previous information during the learning process.

And the fourth, namely testing number of hidden layer neurons, is also same as previous trial, namely using the best parameters obtained from the first, second and third trials.

3.3.1 Scenario 1 trial results

In the first trial scenario, learning rate trials were carried out, namely 0.1, 0.01, 0.001. Testing was carried out using predetermined initial parameters, namely hidden layer neurons of 5, epoch 200, sigmoid activation function, and using Adam optimizer. The following are the results of trial scenario 1:

Table 2. Test Results for Scenario 1.

Learning Rate	Loss	Accuracy (%)	Precision (%)	Recall (%)	F1-Score	Time (s)
0.1	0.0612	92.5	93	92	92	150.964
0.01	0.0613	92	92	92	92	103.603
0.001	0.697	93.5	94	93	93	149.292

From the Table 2 above it is known that the highest accuracy was obtained in trials using a learning rate of 0.001. The accuracy results obtained were 93.50%, with a loss value of 0.0697, while the precision, recall and f1-score values were 94.00%, 93.00% and 93.00% respectively with a computing time of 149,292s. The learning rate value used affects the accuracy results.

Learning rate is one of the training parameters to calculate the weight correction value during the training process. The learning rate value is in the range of zero (0) to (1). The greater the learning rate value, the faster the training process will run. The greater the learning rate, the less accurate the network will be, but vice versa, if the learning rate is smaller, the greater the accuracy of the network will be or increase with the consequence that the training process will take longer.

In the Adam optimizer, the best learning rate is 0.001 which is the default learning rate of the Adam optimizer itself. The following is a comparison picture of accuracy, precision, recall, and f1-score obtained from each learning rate.

3.3.2 Scenario 2 trial results

In the second test scenario, epoch trials were carried out, namely 150, 200, 250, 300, and 350. Testing was carried out using predetermined initial parameters, namely hidden layer neurons of 5, sigmoid activation function, Adam optimizer and using parameters The best obtained from the first trial was a learning rate of 0.001. The following are the results of trial scenario 2:

Table 3. Test Results for Scenario 2.

Epoch	Loss	Accuracy (%)	Precision (%)	Recall (%)	F1-Score	Time (s)
150	0.0889	93.5	93	94	93	59.803
200	0.0697	93.5	94	93	93	149.292
250	0.656	93	93	93	93	214.259
300	0.608	93	93	93	93	186.159
350	0.0584	93	93	93	93	288.992

From the Table 3 above it is known that the highest accuracy was obtained in trials using epoch 200. Accuracy results were obtained, namely 93.50%, with a loss value of 0.0697, while the precision, recall and f1-score values were 94.00%, 93.00% and 93.00 respectively. % With computing time of 149,292s.

Larger batch sizes result in faster progress in training, but do not always converge as quickly. Smaller batch sizes train more slowly, but can converge faster. This obviously depends on the problem. In general, models improve with more training periods, up to a certain point. The accuracy will start to decrease when it meets.

3.3.3 Scenario 3 trial results

In the third test scenario, namely testing the activation function namely Sigmoid, RELU (Rectifier Linear Unit), and TANH (Hyperbolic Tangent Function). Testing of this activation function is carried out in combination, where the sigmoid activation function is used as the output activation function, while the hidden activation function uses trials as mentioned above. The use of the sigmoid activation function as the activation function in the output is because the output in this study has a value of 0/1 and the output of the sigmoid ranges from 0-1. Which uses a threshold of 0.5, where if the output is more than 0.5 then it is predicted to be in class 1 (yes hypertension) and if the output is less than 0.5 then it is predicted to be in class 0 (no hypertension). Meanwhile, RELU has a range between -1 to 1 and TANH has a range from 0 to infinity.

Testing was carried out using predetermined initial parameters, namely hidden layer neurons of 5, Adam optimizer, and using parameters The best obtained in the first and second trials previously, namely learning rate 0.001, and epoch 200. The following are the results of trial scenario 3:

Table 4. Test Results for Scenario 3.

Activation function	Loss	Accuracy (%)	Precision (%)	Recall (%)	F1-Score	Time (s)
Sigmoid + Sigmoid	0.0697	93.5	94	93	93	149.292
RELU + Sigmoid	0.0574	93	93	93	93	100.754
TANH + Sigmoid	0.574	92.5	93	92	92	131.278

From the table 4 above it is known that the highest accuracy was obtained in trials by applying the sigmoid activation function to the hidden layer and sigmoid to the output layer. Activation functions help in better training, learning processes, and better generalization capabilities. This function controls the activation of each unit in the layer by working on the weights and the amount of bias. Several types of activation functions that will be used as trials in this study include Sigmoid, ReLu, Tanh. The accuracy results obtained were 93.50%, with a loss value of 0.0697, while the precision, recall and f1-score values were 94.00%, 93.00% and 93.00% respectively with a computing time of 149,292s.

3.3.4 Scenario 4 trial results

In the fourth test scenario, namely testing hidden layer neurons, namely 5, 10, 15, 20 and 25. Testing was carried out using the Adam optimizer and the best parameters that had been obtained in the first, second and third previous trials, namely learning rate 0.001, epoch 200, and sigmoid + sigmoid activation function. The following are the results of trial scenario 4:

Table 5. Test Results for Scenario 2.

Neuron Hidden	Loss	Accuracy (%)	Precision (%)	Recall (%)	F1-Score	Time (s)
5	0.0697	93.5	94	93	93	149.292
10	0.0633	93.5	94	93	93	148.596
15	0.0606	93	93	93	93	96.433
20	0.0590	93	93	93	93	94.362
25	0.0583	93	93	93	93	147.516

From the Table 5 above it is known that the highest accuracy was obtained in trials using hidden 5 neurons. The accuracy results were 93.50%, with a loss value of 0.0697, while the precision, recall and f1-score values were 94.00%, 93.00% and 93.00% with computing time of 149,292s.

In several test scenarios that have been carried out above, it is known that parameter selection influences the accuracy results. In Table 5, from the trials that have been carried out, it is known that the accuracy results using hidden 5 neurons obtained the highest accuracy value of 93.50%, the precision, recall and f1-score values respectively were 94.00%, 93.00% and 93.00% with a computing time of 149.292s. And the number of hidden layer neurons influences the loss value obtained, where the more hidden layer neurons, the smaller the loss value obtained.

Recent research and empirical evidence suggest that using the same number of neurons in all hidden layers often yields equally good results. However, the greater the number of neurons used, the longer the computing time will be. Provide a more detailed analysis of how each parameter (learning rate, epochs, activation functions, number of neurons) affects model performance.

In Table 4, from the trials that have been carried out, it is known that the application of the activation function affects the accuracy results. Where the use of the sigmoid activation function as the activation function in the output is appropriate. Because the output in this study has a value of 0/1 and the output from the sigmoid ranges from 0-1, a threshold of 0.5 is used if the output does not reach the value 0/1. And the highest accuracy is obtained by applying sigmoid + sigmoid activation.

Meanwhile, in Table 3, from the trials that have been carried out, it is known that the highest accuracy is using epoch 150 and 200. However, it is known that the smallest loss is owned by epoch 200, namely a loss value of 0.0697. And the conclusion from Fig. 10 is that the greater the epoch, the smaller the loss value obtained. And it can be seen in Table 3 that the number of epochs affects the computing time. The larger the epoch, the longer the computing time required. Then in Table 2 the best learning rate value is obtained, namely 0.001, where in the previous test the Adam optimizer was found to be the best optimizer.

The way the Adam optimizer works is by combining momentum and learning rate which causes faster convergence, thereby speeding up the training process and producing better scores in a shorter time. Adam's optimizer is also stable over default parameter so it doesn't require much adjustment. This is also the same as previous research where the use of the Adam optimizer produced higher accuracy compared to other optimizers [17] .

Therefore, based on the test results of the best model that has been obtained, it shows that the results of model evaluation using the Backpropagation method with the Adam optimizer and a learning rate of 0.001 and an epoch of 200 show good accuracy results, namely 93.50% with a loss value obtained of 0.0697.

4 Conclusion

Based on the results of this research, several conclusions were obtained in classifying hypertension using the Backpropagation method by applying several parameters to test scenarios.

1. Based on the results of the tests carried out, accuracy results were obtained using the Backpropagation method using the best model, namely producing an accuracy of 93.50% with a loss value of 0.0697. Meanwhile, the precision value is 94.00%, the recall value is 93.00%, and the f1-score value is 93.00%.
2. Test results compare accuracy between the application of several different parameters such as learning rate values, namely 0.1, 0.01, 0.001, 0.0001, and 0.00001.
3. The best model produced in this research was with a learning rate of 0.001 with the optimizer used, namely Adam.
4. Meanwhile, the epoch used in this training are 150, 200, 250, 300, and 350. And the best model obtained is epoch 200 with a computing time of 149,292s, using the sigmoid + sigmoid activation function with the best hidden layer neurons, namely 5 neurons. hidden with an accuracy of 93.50%, precision, recall and f1-score values of 94.00%, 93.00% and 93.00% respectively with a computing time of 149,292s. Therefore, it can be concluded that in this study the learning rate and epoch influence the accuracy values produced by the Backpropagation model.
5. Choosing a learning rate value based on the default optimizer gives better accuracy results than using other learning rate values and the more epoch used, the better a model can learn and the smaller the loss value obtained, however this will result in time longer computing.

The suggestions for developing this study are:

1. Use data with more than 7 attributes other than those mentioned in this study, so that it can be further measured whether this artificial neural network modeling can successfully handle the classification of hypertension cases 2.
2. Please use data with more than 2 classes, so that it will be more varied and complex

References

1. Kemenkes RI P2PTM, *Hari Hipertensi Dunia 2019: 'Know Your Number, Kendalikan Tekanan Darahmu dengan CERDIK.,'* Kementrian Kesehatan Republik Indonesia, 1, (2019), Accessed: Oct. 19, 2022, Available: http://p2ptm.kemkes.go.id/tag/hari-hipertensi-dunia-2019-know-your-number-kendalikan-tekanan-darahmu-dengan-cerdik
2. Kemenkes RI, *Hasil Utama Riskesdas 2018 | Badan Penelitian dan Pengembangan Kesehatan,* (2018) https://www.litbang.kemkes.go.id/hasil-utama-riskesdas-2018/ (accessed Oct. 19, 2022).
3. K. R. Kesehatan Badan Penelitian dan Pengembangan Kesehatan Puslitbang Humaniora dan Manajemen Kesehatan, *HASIL UTAMA RISKESDAS 2018 PROVINSI JAWA TIMUR,* (2018)
4. dr. Nailul Maghfiroh, *Hipertensi di UPT Puskesmas Modopuro*, (2022)
5. Kemenkes RI, *Klasifikasi Hipertensi - Direktorat P2PTM*, **12,** 1, (2018), Accessed: Nov. 02, 2022, Available: http://p2ptm.kemkes.go.id/infographic/klasifikasi-hipertensi
6. J. A. Orozco Torres, A. Medina Santiago, J. M. Villegas Izaguirre, M. Amador García, and A. Delgado Hernández, *Hypertension Diagnosis with Backpropagation Neural Networks for Sustainability in Public Health*, Sensors, **22**, 14, 1–20, (2022), doi: 10.3390/s22145272
7. F. G. Woldemichael and S. Menaria, *Prediction of Diabetes using Data Mining Techniques*, in Proc. 2nd Int. Conf. Trends Electronics Informatics (ICOEI), 414-418, (2018), doi: 10.1109/ICOEI.2018.8553959
8. P. Sarkar and S. Pawar, *Fast and Accurate Prediction of Diabetes Mellitus using Artificial Neural Network*, in International Conference on Recent Advances in Science and Engineering Technology (ICRASET), (2023), doi:10.1109/ICRASET59632.2023.10419972
9. D. Nasien, V. Enjeslina, M. H. Adiya, and Z. Baharum, *Breast Cancer Prediction Using Artificial Neural Networks Back Propagation Method,* in International Conference on Robotic Automation System (ICORAS), (2001), doi: 10.1088/1742-6596/2319/1/012025
10. E. M. S. Rochman, W. Agustiono, N. Suryani, & A. Rachmad, *Comparison between the backpropagation and single exponential smoothing method in sugar production forecasting case,* Commun. Math. Biol. Neurosci., **2021**, (2021)
11. P2PTM Kemenkes RI, *Hipertensi, The Silent Killer - Direktorat P2PTM*, Kementerian Kesehatan Republik Indonesia, (2018). http://p2ptm.kemkes.go.id/infographic-p2ptm/hipertensi-penyakit-jantung-dan-pembuluh-darah/hipertensi-the-silent-killer (Accessed Nov. 01, 2022).
12. A. Tharwat, *Classification assessment methods*, Applied Computing and Informatics, **17**, 1, 168-192, (2021), doi: https://doi.org/10.1016/j.aci.2018.08.003
13. R. Wazirali, *An improved intrusion detection system based on KNN hyperparameter tuning and cross-validation,* Arabian Journal for Science and Engineering, **45**, 10859-10873, (2020), doi: https://doi.org/10.1007/s13369-020-04907-7
14. L. Zajmi, F. Y. H. Ahmed, A. A. Jaharadak, and A. Amril, *Concepts, methods, and performances of particle swarm optimization, backpropagation, and neural networks*, Applied Computational Intelligence and Soft Computing, **2018**, 7, (2018), doi: https://doi.org/10.1155/2018/9547212
15. A. Nabil, M. Seyam and A. Abou-Elfetouh, *Prediction of Students' Academic Performance Based on Courses' Grades Using Deep Neural Networks*, in IEEE Access, **9**, 140731-140746, (2021), doi: 10.1109/ACCESS.2021.3119596
16. I. Arefa, M. S. Alam, I. Siddiquee and N. Siddique, *Performance Analysis of Machine Learning Algorithms for Hypertension Decision Support System*," 2019 IEEE International Conference on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON), 15-20, (2019), doi: 10.1109/RAAICON48939.2019.8
17. A. Rachmad, M. Syarief, J. Hutagalung, S. Hernawati, E.M.S. Rochman, Y.P. Asmara, *Comparison of CNN architectures for Mycobacterium tuberculosis classification in sputum images,* Ingénierie des Systèmes d'Information. **29**, 1, 49-56, (2024), doi: https://doi.org/10.18280/isi.290106
18. A. Rachmad, M. Fuad, E.M.S. Rochman, *Convolutional neural network-based classification model of corn leaf disease,* Mathematical Modelling of Engineering Problems, **10**, 2, 530-536, (2023), doi: https://doi.org/10.18280/mmep.100220
19. A. Rachmad, F. Sonata, J. Hutagalung, D. Hapsari, M. Fuad, E.M.S. Rochman, *An automated system for osteoarthritis severity scoring using residual neural networks,* Mathematical Modelling of Engineering Problems, **10**, 5, 1849-1856, (2023), doi: https://doi.org/10.18280/mmep.100538
20. A. Kumar, S. Sarkar, and C. Pradhan, *Malaria Disease Detection Using CNN Technique with SGD, RMSprop and ADAM Optimizers*, Stud. Big Data, **68**, 211–230, (2020), doi: 10.1007/978-3-030-33966-1_11