# CONJUGATE GRADIENT POWELL-BEALE TO IMPROVE BACKPROPAGATION PERFORMANCE IN PREDICTING POPULATION AMOUNTS

Sri Wahyuni[1]*, Medi Hermanto Tinambunan [2]
Dharmawangsa University[1]*, Manado State University [2]

**Keywords :**

backpropagation; conjugate gradient powell-beale restarts; artificial intelligence; population;

***Correspondence Address:**
sriwahyuni15jun@dharmawangsa.ac.id

**Abstract:** One of the parts of artificial intelligence is the backpropagation method. However, this method still has limitations, in which its level of convergence is relatively slow so that it needs to be optimized so that it can work optimally. One of the conjugate Gradient Methods that can be used is the Powell-Beale Restarts Conjugate Gradient which is expected to be able to improve the accuracy of predictions, minimizing processing time and minimizing the range of error predicted results. The prediction was conducted in two steps, namely, by implementing the Backpropagation Standard followed by Conjugate Gradient Powell-Beale Restarts. The results obtained by applying Conjugate Gradient Powel-Beale Restarts into the prediction process is that it can increase the value of accuracy, accelerate the data processing, and minimize the error rate.

## INTRODUCTION

Based on the data obtained from the Central Statistics Agency of Medan, the population of each district in North Sumatra is relatively increasing. This greatly affects the condition of North Sumatra within the next 20 to 30 years. The population increase in each region of North Sumatra province is due to the revolving thoughts in the society that those who have many children have lots of luck. Therefore this population increase becomes the concern of the local government and also the central government in suppressing the rate of population growth. Currently, the total population of North Sumatra is 14.42 million, with an area of 72,981 km$^2$ (North Sumatra Central Statistics

Agency, 2018). If the population growth continues, the area will be more limited, the number of unemployed will increase, and the poverty rate will increase[1]. One of the efforts that the government can do in dealing with this issue is by educating the maximum number of children and families, and improving the quality of human resources in each region so that they can think forward and leave behind their old thoughts. Besides, the government must also be able to undertake the prediction of the population in the following year so that the government can prepare anything that can control the population growth rate. The implementation of these predictions can be performed using artificial neural networks, which in this case is Backpropagation. Backpropagation is a method that applies supervised learning. This method has two phases, namely forward and backward phases[2]. Backpropagation can be said to be part of an artificial neural network that represents the human brain[3]. Backpropagation is often used to make predictions because of the ability to optimize the weight and the bias values used in making predictions[4], [5]. Previous research has been conducted using the Conjugate Gradient Powell-Beale Restarts with the accuracy results obtained an average of 61.5% and with a total of 8 network architecture[6], then research has been done using the conjugate gradient pattern of the ribiere pattern with an accuracy level of 90.91%[7]. Therefore in this research, the writer intended to do further research by combining the Conjugate Gradient Powell-Beale Restarts and Backpropagation to improve prediction accuracy, minimize processing time and minimize error values using 5 network architectures used.

## RESEARCH METHODS

This section discusses the methods used and how they work to complete all the steps until the optimal prediction results were obtained. The data used were normalized before it was processed using the Backpropagation Standard method or Conjugate Gradient Powell-Beale Restarts.

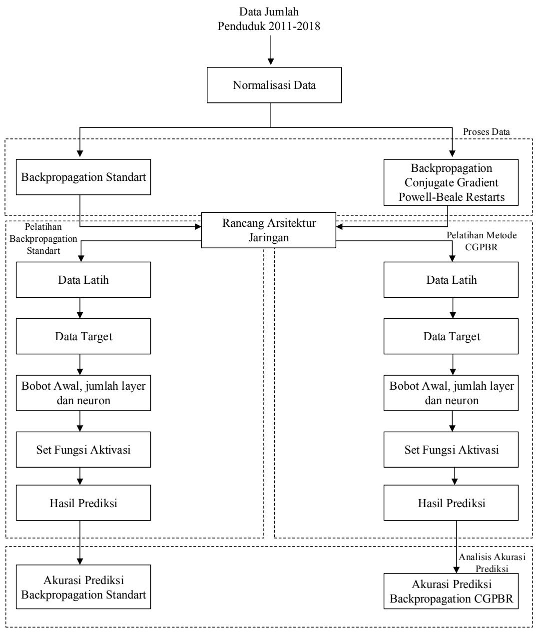


Figure 1. Prediction Optimization Flowchart

In Figure. 1. it can be seen that there are two different paths, however, the source of data used remains the same. The initial step in solving the problem was selecting the data from the North Sumatra Central Statistics Agency website, then the data were prepared for processing using Backpropagation Standard methods and Backpropagation Conjugate Gradient Powell-Beale Restarts. The next step was normalizing the data using the Sigmoid Function, which is an Asymptotic Function, which means that it does not reach 0 or 1, and therefore, it needs to be normalized[8].

In the backpropagation process, it requires an input layer, a hidden layer and an output layer[9]. In determining the number of hidden layers, the Heaton rule was used, which has been explained in the previous chapter, namely; 3-3-1, 3-4-1, 3-5-1. Afterward, the random combinations were taken to add the testing of the combination of methods namely; 3-9-1, 3-12-1, 3-16-1. This technique was performed to get the maximum results in the training process[10]. After completing the training, the same data were tested until the optimal results obtained.

## Network Architecture

Network architecture design was performed to get the predicted value of the population by determining the number of inputs used, the number of hidden layers, and the desired target. The data used were 32 data, namely the data for each region in North

Sumatra from 2011 to 2017. The target output is 2018. Artificial neural networks were applied to get the balance between the memorization and the generalization. The next process was performed until the process reached an error value of the minimum value. Furthermore, it was necessary to determine an acceptable error value so that the training can be stopped. If the error value is equal to or less than the specified limit, then the training can be stopped. The combination used was 3-3-1, 3-4-1, 3-5-1. To meet the rules used, then random samples were taken; 3-9-1, 3-12-1, 3-16-1[11]. The following is one of the network combinations used:

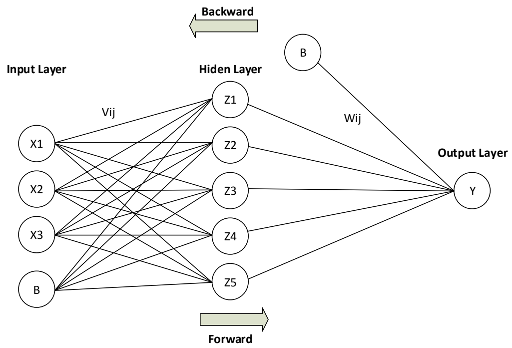


Figure 2. Architecture network, Neural Network Backpropagation

Explanation:

| | |
|---|---|
| X1-X3 | : Data input (input layer) |
| Vij | : Weight to the hiden layer |
| Z1-Z5 | : Hidden layer |
| Y | : Output |
| B | : Bias |
| Wow | : Weight to the output layer |

**Normalization**

In processing the data, it is important to note whether the data is ready to be processed or not. Any data that has not been done is feasible for processing. Therefore data normalization is first performed according to the activation function used. In this case, the functions used were Binary and Bipolar. The Binary Sigmoid Function is a function that has a ranger (0,1) while the Bipolar Sigmoid Function has a ranger (-1,1). Both functions have different uses, in which Bipolar is used for stable data while the Binary function is for unstable data. The output value generated by the activation function never reaches 0 or 1. Therefore the dataset used needed to be normalized into smaller intervals, as can be seen in the following equation:

$$X' = \frac{0.8(x-a)}{b-a} + 0,1 \qquad (1)$$

Explanation:

X ' = Data transformed

x = Initial data before normalized

a = Minimum data value

b = Maximum data value

**Backpropagation**

Backpropagation is an artificial neural network that can be used to study and analyze patterns of existing data in the past much more precisely so that a more accurate output can be obtained[12]. Some steps are carried out in building the backpropagation algorithm, they are:

1. If the termination conditions are still not fulfilled, then do step 2 until step 8.
2. For each training data, do step 3 to step 8.

Phase I: Forward Propagation

3. Each input unit receives a signal and passes it to the hidden unit.
4. Calculate all output from hidden layer Zj.

$$Z_{net_j} = v_{j0} + \sum_{i=1}^{n} x_i \, v_{ji} \qquad (2)$$

5. Count all network output units (yk)

$$y\_net_k = w_{k0} + \sum_{j=1}^{p} z_j w_{kj} \qquad (3)$$

Phase II: Backward Propagation

1. Calculate weight correlation

$$\delta_k = (t_k - y_k) f'(y - net_k) = (t_k - y_k)(1 - y_k) (4)$$

2. Calculate each factor hidden layer unit based on the error in each hidden unit $z_j(j = 1,2,3, \dots \dots \dots n)$

$$\delta\_net_j = \sum_{k=1}^{m} \delta_k w_{kj} \qquad (4)$$

Phase III: Change of Weight

3. Changes in line weights leading to output points

$$W_{kj}(baru) = W_{kj}lama + \Delta W_{kj} \qquad (5)$$

## RESULTS AND DISCUSSION

In conducting this research, population data was needed. Therefore, the North Sumatra region population data from 2011-2018 was taken as can be seen below. The data used were:

Table 1. Population Data for 2011-2018

| Yr / mo | Nias | Mandailing Natal | ... | Padang Sidimpuan | Gunung Sitoli |
|---|---|---|---|---|---|
| 2011 | 132605 | 408731 | ... | 193322 | 127382 |
| 2012 | 132860 | 410931 | ... | 198809 | 128337 |
| 2013 | 133388 | 413475 | ... | 204615 | 129403 |
| 2014 | 133388 | 413475 | ... | 204615 | 129403 |
| 2015 | 136115 | 430894 | ... | 209796 | 135995 |
| 2016 | 141403 | 435303 | ... | 212917 | 137693 |
| 2017 | 142110 | 439505 | ... | 216013 | 139281 |
| 2018 | 142840 | 443490 | ... | 218892 | 140927 |

Based on the data above, data normalization was performed using the function (1), the following results were obtained:

$$X' = \frac{0.8(132605 - 40884)}{2264145 - 40884} + 0{,}1$$

$$X' = \frac{106084 - 32707.2}{2223261} + 0{,}1$$

$$X' = 0.13300$$

The calculation was performed continuously until all data is complete, and the following values were obtained:

| Yr / mo | Nias | Mandailing Christmas | ... | Padang sidimpuan | Gunung Sitoli |
|---|---|---|---|---|---|
| 2011 | 0.133004 | 0.2323630 | ... | 0.1548 | .1311 |
| 2012 | 0.133095 | 0.2331546 | ... | .1568 | .1314 |
| 2013 | 0.133285 | 0.2340700 | ... | 0.1589 | .1318 |
| 2014 | 0.133285 | 0.2340700 | ... | 0.1589 | .1318 |
| 2015 | 0.134267 | 0.2403379 | ... | 0.1607 | 0.1342 |
| 2016 | 0,136169 | 0.2419244 | ... | .1619 | .1348 |
| 2017 | .136424 | 0.2434365 | ... | 0.1630 | .1354 |
| 2018 | 0,136687 | 0.244870 | ... | 0.1640 | 0.1359 |

After the data normalization, we conducted network training. The following are the steps;

Step 0: Initialize any weight and bias used.

Table 3.3 Weight of the Input Screen to the Hidden Layer Vij

| | Z1 | Z2 | Z3 | Z4 | Z5 |
|---|---|---|---|---|---|
| X1 | 0.2 | -0.2 | -0.3 | 0.3 | 0.1 |
| X2 | 0.3 | 0.1 | 0.2 | 0.3 | 0.3 |
| X3 | 0.4 | 0.2 | 0.4 | 0.2 | -0.2 |
| B1 | 0.2 | 0.3 | 0.1 | 0.2 | 0.4 |

Table 3.3 Weight of the Input Screen to the Hidden Layer Vij

| | Y |
|---|---|
| Z1 | 0.5 |
| Z2 | 0.2 |
| Z3 | 0.3 |
| Z4 | -0.2 |
| Z5 | 0.4 |
| B2 | 0.2 |

Step 1 : If the stopping condition is still not fulfilled, then do step 2 to step 8.

Step 2 : For each data that has undergone training, repeat step 3 through step 8.

**Phase I: Forward Propagation**

Step 3 : Each input unit receive a signal and forward it to the hidden unit.

Step 4 : Calculate all output from hidden layer Zj.

$$Z_{net_j} = v_{j0} + \sum_{i=1}^{n} x_i\, v_{ji}$$

$$Z_{net_1} = v_{10} + \sum_{i=1}^{3} x_1\, v_{11} + x_2 v_{12} + x_3 v_{13}$$

$$= 0{,}2 + (0{,}1 * 0{,}2) + (0{,}1 * 0{,}3) + (0{,}1 * 0{,}4)$$

$$= 0{,}2 + 0{,}02 + 0{,}03 + 0{,}04$$

$$= 0{,}29$$

$$Z_{net_2} = v_{20} + \sum_{i=1}^{3} x_1\, v_{11} + x_2 v_{12} + x_3 v_{13}$$

$$= 0{,}3 + (0{,}1 * -0{,}2) + (0{,}1 * 0{,}1) + (0{,}1 * 0{,}2)$$

$$= 0{,}3 - 0{,}02 + 0{,}01 + 0{,}02$$

$$= 0{,}29$$

$$Z_{net_3} = v_{30} + \sum_{i=1}^{3} x_1\, v_{11} + x_2 v_{12} + x_3 v_{13}$$

$$= 0{,}1 + (0{,}1 * -0{,}3) + (0{,}1 * 0{,}2) + (0{,}1 * 0{,}4)$$

$$= 0{,}1 - 0{,}03 + 0{,}02 + 0{,}04$$

$$= 0{,}31$$

$$Z_{net_4} = v_{40} + \sum_{i=1}^{3} x_1\, v_{11} + x_2 v_{12} + x_3 v_{13}$$

$$= 0{,}2 + (0{,}1 * 0{,}3) + (0{,}1 * 0{,}3) + (0{,}1 * 0{,}2)$$

$$= 0{,}2 + 0{,}03 + 0{,}03 + 0{,}02$$

$$= 0{,}28$$

$$Z_{net_5} = v_{50} + \sum_{i=1}^{3} x_1 v_{11} + x_2 v_{12} + x_3 v_{13}$$
$$= 0{,}4 + (0{,}1 * 0{,}1) + (0{,}1 * 0{,}3) + (0{,}1 * -0{,}2)$$
$$= 0{,}4 + 0{,}01 + 0{,}03 - 0{,}02$$
$$= 0{,}42$$

$$Z_j = f\left(Z_{net_j}\right) = \frac{1}{1 + e^{-Z_{net_j}}}$$

$$Z_1 = \frac{1}{1 + e^{-Z_{net_1}}} = \frac{1}{1 + e^{-0{,}29}} = 0{,}572$$

$$Z_2 = \frac{1}{1 + e^{-Z_{net_2}}} = \frac{1}{1 + e^{-0{,}31}} = 0{,}577$$

$$Z_3 = \frac{1}{1 + e^{-Z_{net_3}}} = \frac{1}{1 + e^{-0{,}13}} = 0{,}532$$

$$Z_4 = \frac{1}{1 + e^{-Z_{net_4}}} = \frac{1}{1 + e^{-0{,}28}} = 0{,}56$$

$$Z_5 = \frac{1}{1 + e^{-Z_{net_5}}} = \frac{1}{1 + e^{-0{,}42}} = 0{,}603$$

Step 5 : Calculate all network output units (yk)

$$y\_net_k = w_{k0} + \sum_{j=1}^{p} z_j w_{kj}$$

$$y\_net_1 = 0{,}2 + (0{,}5 * 0{,}572) + (0{,}577 * 0{,}2) + (0{,}53 * 0{,}3) + (0{,}56 * -0{,}2) + (0{,}603 * 0{,}4)$$
$$= 0{,}2 + 0{,}286 + 0{,}1154 + 0{,}159 - 0{,}112 + 0{,}2412$$
$$= 0{,}8896$$

$$y_k = f\left(y_{net_k}\right) = \frac{1}{1 + e^{-y_{net_k}}}$$

$$y_k = \frac{1}{1 + e^{-0{,}8896}} = \frac{1}{1 + 0{,}41} = \frac{1}{1{,}41} = 0{,}709$$

**Phase II: Forward Propagation**

Step 6 : Calculate weight correlation

$$\delta_k = (t_k - y_k) f'(y - net_k) = (t_k - y_k)(1 - y_k)$$

$\delta_1 = (0 - 0{,}709)0{,}709(1 - 0{,}709) = -0{,}709(0{,}291) = -0{,}2063$

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

$\Delta w_{10} = 0{,}01 * (-0{,}2063) * 1 = -0{,}002$

$\Delta w_{11} = 0{,}01 * (-0{,}2063) * 0{,}572 = -0{,}00118$

$\Delta w_{12} = 0{,}01 * (-0{,}2063) * 0{,}577 = -0{,}00119$

$\Delta w_{13} = 0{,}01 * (-0{,}2063) * 0{,}532 = -0{,}00109$

$\Delta w_{14} = 0{,}01 * (-0{,}2063) * 0{,}56 = -0{,}001155$

$\Delta w_{15} = 0{,}01 * (-0{,}2063) * 0{,}603 = -0{,}00124$

Step 7 : Calculate each factor hidden layer unit based on the error in each hidden unit $z_j(j = 1,2,3, \dots \dots \dots n)$

$$\delta\_net_j = \sum_{k=1}^{m} \delta_k w_{kj}$$

$\delta\_net_1 = -0{,}2063 * 0{,}5 = -0{,}103$

$\delta\_net_2 = -0{,}2063 * 0{,}2 = -0{,}041$

$\delta\_net_3 = -0{,}2063 * 0{,}3 = -0{,}061$

$\delta\_net_4 = -0{,}2063 * -0{,}2 = 0{,}041$

$\delta\_net_5 = -0{,}2063 * 0{,}4 = -0{,}082$

Thus the error factor is hidden layer unit:

$\delta_j = \delta\_net_j f'(z_{net_j}) = \delta_{net_{z_j}}(1 - z_j)$

$\delta_1 = (-0{,}103)(0{,}5)(1 - 0{,}5) = -0{,}025$

$\delta_2 = -0{,}041(0{,}2)(1 - 0{,}2) = -0{,}006$

$\delta_3 = -0{,}061(0{,}3)(1 - 0{,}3) = -0{,}012$

$\delta_4 = 0{,}041(-0{,}2)(1 + 0{,}2) = -0{,}009$

$\delta_5 = -0{,}082(0{,}4)(1 - 0{,}4) = -0{,}019$

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

$\Delta v_{10} = 0{,}01 * (-0{,}025) * 1 = -0{,}00025$

$\Delta v_{20} = 0{,}01 * (-0{,}006) * 1 = -0{,}00006$

$\Delta v_{30} = 0{,}01 * (-0{,}012) * 1 = -0{,}00012$

$\Delta v_{40} = 0{,}01 * (-0{,}009) * 1 = -0{,}00009$

$\Delta v_{50} = 0{,}01 * (-0{,}019) * 1 = -0{,}00019$

**Phase III: Change of Weight**

Step 8 : Changes in line weights leading to the output point

$W_{kj}(baru) = W_{kj}lama + \Delta W_{kj}$

$W_{10}(baru) = 0{,}5 + (-0{,}00025) = 0{,}49975$

$W_{11}(baru) = 0{,}2 + (-0{,}00006) = 0{,}19994$

$W_{12}(baru) = 0{,}3 + (-0{,}00012) = 0{,}29988$

$W_{13}(baru) = -0{,}2 + (-0{,}00009) = 0{,}19991$

$W_{14}(baru) = 0{,}4 + (-0{,}00019) = 0{,}39981$

After the calculation using Backpropagation Standard was performed, a combination with the Powell-Beale Restarts Conjugate Gradient method was performed using 0.001 learning rate, maximum epoch 10000, network combination 3-3-1, 3-4-1, 3-5-1, 3-9 -1, 3-12-1, and 3-16-1 and the results obtained can be seen below:

**Iteration Comparison Results**

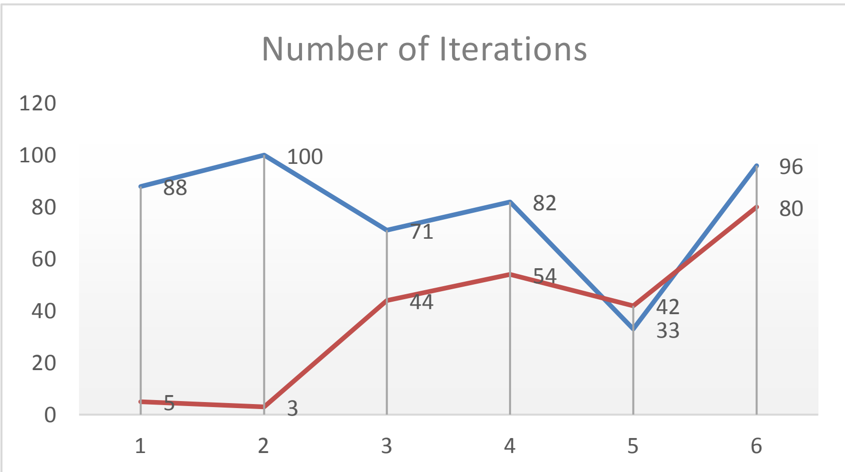


Figure 3. Comparison of the number of Iteration between Backpropagation Standards and Conjugate Gradient Powell-Beale Restarts

**Comparison of Training Accuracy Results**

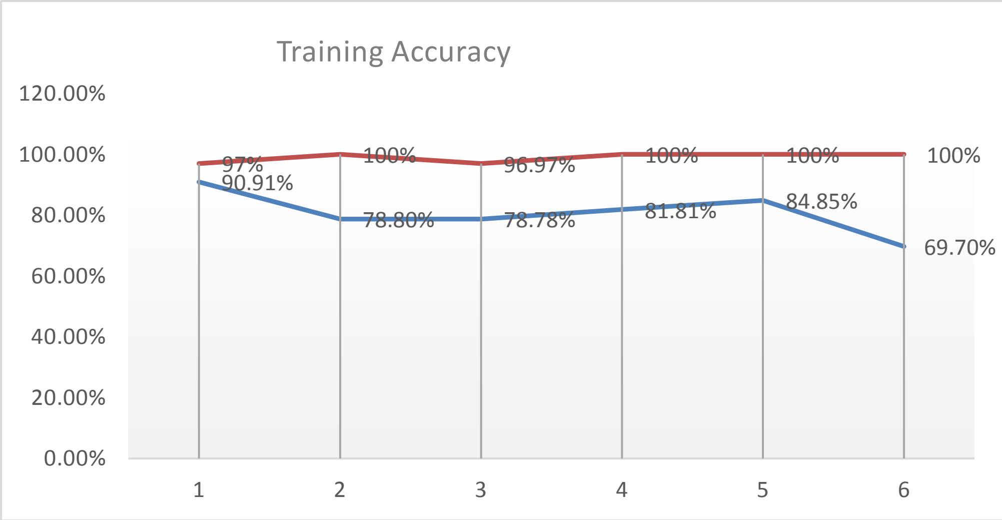


Figure 4. Comparison of training accuracy results between Backpropagation Standards and Conjugate Gradient Powell-Beale Restarts

**Comparison of Testing Accuracy Results**

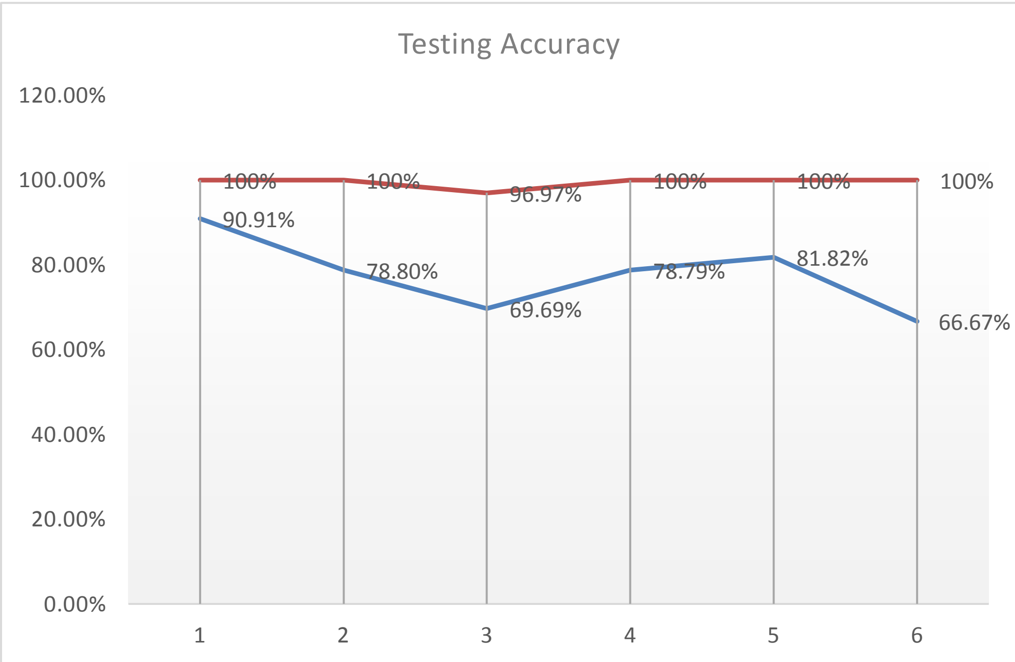


Figure 5. Comparison of testing accuracy results between Backpropagation Standards and Conjugate Gradient Powell-Beale Restarts

## CONCLUSION

After analyzing the network combinations 3-3-1, 3-4-1, 3-5-1, 3-9-1, 3-12-1, and 3-16-1, it can be concluded that the average iterations obtained using the Conjugate Gradient Powell-Beale Restarts are less than the Backpropagation Standard, while the accuracy of the training using the Conjugate Gradient Powell-Beale Restarts is higher by 98.9% compared to the Backpropagation Standard of 80.8%. Further, the testing accuracy of the Conjugate Gradient Powell -Beale is higher by 99.4% compared to Backpropagation Standard of 77.7%.

## REFERENCE

[1] R. Pancasasti and E. Khaerunisa, "Analysis of the Impact Population Growth Rate on Gender Aspects in Urban Areas in Serang City," *Tirtayasa Ekon.*, vol. 13, no. 1, p. 130, 2018.

[2] M. Dhamma, M. Zarlis, and E. B. Nababan, "Multithreading with separate data to improve the performance of Backpropagation method," *J. Phys. Conf. Ser.*, vol. 930, no. 1, pp. 0–6, 2017.

[3] S. A. Fitania, A. Damayanti, and A. B. Pratiwi, "Model identification of dengue fever spreading using firefly algorithm and backpropagation neural network," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 546, no. 3, 2019.

[4] E. Siregar, H. Mawengkang, E. B. Nababan, and A. Wanto, "Analysis of Backpropagation Method with Sigmoid Bipolar and Linear Function in Prediction of Population Growth," *J. Phys. Conf. Ser.*, vol. 1255, no. 1, 2019.

[5] K. Onggrono, T. Tulus, and E. B. Nababan, "Analysis of the Use of Parallel Processing Multithreading in Resilient Backpropagation," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 2, no. 1, pp. 33–40, 2017.

[6] A. Wanto, M. Zarlis, Sawaluddin, and D. Hartama, "Analysis of Artificial Neural Network Backpropagation Using Conjugate Gradient Fletcher Reeves in the Predicting Process," *J. Phys. Conf. Ser.*, vol. 930, no. 1, 2017.

[7] M. H. Tinambunan, E. B. Nababan, and B. B. Nasution, "Conjugate Gradient Polak Ribiere in Improving Performance in Predicting Population Backpropagation," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 835, no. 1, 2020.

[8] A. Wanto, "Optimasi Prediksi Dengan Algoritma Backpropagation Dan Conjugate Gradient Beale-Powell Restarts," *J. Nas. Teknol. dan Sist. Inf.*, vol. 3, no. 3, pp. 370–380, 2018.

[9] I. Sutrisno *et al.*, "Implementation of backpropagation neural network and extreme learning machine of ph neutralization prototype," *J. Phys. Conf. Ser.*, vol. 1196, no. 1, 2019.

[10] E. Irawan, M. Zarlis, and E. B. Nababan, "Analisis Penambahan Nilai Momentum

Pada Prediksi Produktivitas Kelapa Sawit Menggunakan Backpropagation," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 1, no. 2, pp. 84–89, 2017.

[11] J. Nasional, S. Informasi, and A. Wanto, "Optimasi Prediksi Dengan Algoritma Backpropagation Dan Conjugate Gradient Beale-Powell Restarts," vol. 03, no. 2017, pp. 370–380, 2018.

[12] J. T. Hardinata, M. Zarlis, E. B. Nababan, D. Hartama, and R. W. Sembiring, "Modification of Learning Rate with Lvq Model Improvement in Learning Backpropagation," *J. Phys. Conf. Ser.*, vol. 930, no. 1, 2017.