

Optimasi Bobot *Multi-Layer Perceptron* Menggunakan Algoritma Genetika Untuk Klasifikasi Tingkat Resiko Penyakit Stroke

Nadya Oktavia Rahardiani¹, Wayan Firdaus Mahmudy², Indriati³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹nadya1loktavia@gmail.com, ²wayanfm@ub.ac.id, ³indriati.tif@ub.ac.id

Abstrak

Stroke merupakan salah satu penyakit dengan tingkat kematian yang tinggi di Indonesia. Pendeteksian penyakit stroke dapat dilakukan dengan berbagai cara, salah satunya dengan melakukan cek darah. Umumnya pemeriksaan dilakukan terlebih dahulu kemudian hasil identifikasi tingkat resiko stroke dapat diketahui setelahnya. Jaringan *multi-layer perceptron* (MLP) merupakan salah satu model jaringan syaraf tiruan (JST) yang memiliki bobot acak dari pelatihan *backpropagation* (BP). Pelatihan ini melakukan optimasi untuk mendapatkan bobot yang lebih baik pada jaringan MLP dengan menggunakan algoritma genetika sebagai pelatihan, sehingga diharapkan hasil klasifikasi lebih akurat. Implementasi, pengujian, dan analisis dilakukan pada metode pembelajaran BP dan algoritma genetika untuk membandingkan hasil akurasi untuk pengklasifikasian tingkat resiko penyakit stroke. Pada pengujian BP didapatkan nilai rata-rata MSE sebesar 0.01221 dengan parameter berupa jumlah iterasi = 190, jumlah *neuron* pada *hidden layer* = 10, dan *learning rate* = 0.9. Sedangkan nilai rata-rata MSE metode AG adalah 0.0549 dengan ukuran populasi = 100, banyak generasi = 400, *Cr* = 0.8 dan *Mr* = 0.2. Rata-rata akurasi data yang dihasilkan dengan BP sebesar 88.40% dan rata-rata MSE 0.0122, sedangkan AG menghasilkan rata-rata akurasi data 60.60% dengan rata-rata MSE 0.0549 pada 10 kali percobaan.

Kata kunci: *multi-layer perceptron, jaringan syaraf tiruan, backpropagation, algoritma genetika*

Abstract

Stroke is one of a high mortality disease in Indonesia. A various ways can be done to detect stroke, such as blood test. The result is known just after a few hour. Unfortunately, in some case it took a long time to find out whether a patient at risk of stroke or not. The level of risk can be easily done with a system. Multi-layer perceptron (MLP) network is one of artificial neural network (ANN) model which has a random weight from backpropagation (BP) learning. This study is doing optimization to obtain proper weights, using genetic algorithm (GA) as a training method, so that the classification results are more accurate. Implementation, testing, and analysis are done in BP learning algorithm and GA to compare its accuracy on classifying the risk level of stroke. MSE value obtained in testing phase is 0.0122 with number of iteration = 190, number of neuron in hidden layer = 10, and learning rate = 0.9. While in testing phase of GA obtained 0.0549 with population size = 100, generation size = 400, Cr = 0.8, and Mr = 0.2. In final result, average data accuracy of BP is 88.40% with average MSE value is 0.0122 and GA is 60.60% with average MSE value 0.0549 by 10 times trial.

Keywords: *multi-layer perceptron, artificial neural network, backpropagation, genetic algorithm*

1. PENDAHULUAN

Stroke adalah penyakit pada urutan ke-3 yang paling banyak menyebabkan kematian di dunia setelah penyakit jantung dan kanker. Di Indonesia, jumlah kematian yang diakibatkan stroke pun terus meningkat. Menurut data Riset Kesehatan Dasar pada tahun 2013, tingkat

prevalensi penyakit stroke di Indonesia meningkat dibandingkan 6 tahun kebelakang yaitu dari 8,3% menjadi 12,1%. Bahkan Indonesia adalah negara dengan jumlah penderita stroke paling banyak se-Asia (Yastroki, 2012).

Deteksi gejala stroke harus dilakukan sedini mungkin, mengingat sebagian masyarakat kerap mengabaikan faktor resiko yang mendominasi

terjadinya stroke. Untuk memastikan apakah gejala yang dialami berpotensi menjadi penyakit stroke, harus dilakukan diagnosis secara mendalam oleh dokter spesialis sebagai pakar baik dari segi pemeriksaan, riwayat penyakit atau rekam medis, atau Computed Tomography (CT) Scan. Namun pada praktiknya, analisis gejala stroke ini memerlukan waktu yang tidak singkat dan juga dapat dipengaruhi beberapa faktor lain. Prediksi yang akurat juga bergantung pada pengetahuan pakar maupun literatur yang sudah ada. Dari kendala yang ditemukan tersebut maka diperlukan sebuah solusi yang dapat membantu pihak-pihak terlibat untuk mengidentifikasi tingkat resiko seorang pasien mengalami penyakit stroke agar pasien mendapatkan penanganan sedini mungkin.

Jaringan Syaraf Tiruan (JST) adalah model komputasi yang mensimulasi fungsi dari pengetahuan yang ada pada manusia. Dari berbagai jenis arsitektur JST, *Multi-layer perceptron* (MLP) adalah yang sudah banyak diketahui dan paling sering digunakan (Hamid, Nawi, & Ghazali et al., 2011). Pada dasarnya MLP terdiri dari suatu *input layer*, satu atau lebih *hidden layer*, dan satu *output layer*. MLP menerapkan teknik *supervised learning* yang disebut *Backpropagation* sebagai metode pembelajaran jaringan (Rosenblatt, 1961). Beberapa penelitian telah menerapkan model MLP dengan metode pelatihan *backpropagation* ini, salah satunya penelitian yang dilakukan Sari, Mahmudy, & Wibawa (2016) untuk peramalan tingkat inflasi di Indonesia. Perbandingan menunjukkan bahwa nilai RMSE dengan MLP lebih baik dibandingkan dengan menggunakan Sugeno FIS. MLP juga dapat digunakan untuk pengenalan seperti pengenalan tanda tangan seperti yang dilakukan pada penelitian oleh Rahmi, Wijayaningrum, Mahmudy et al. (2016). Namun ternyata MLP kurang tepat untuk diaplikasikan pada objek ini sehingga akurasi yang dihasilkan rendah.

Meskipun dapat dilakukan untuk berbagai fungsi, metode pembelajaran BP untuk MLP tidak dapat menjamin akan mendapatkan hasil terbaik dan tingkat konvergensinya pun terbilang rendah (Huang, Li, & Xiao, 2014). Maka dari itu diusulkan algoritma lain yaitu algoritma genetika (AG) sebagai pendekatan optimisasi pencarian global untuk menemukan solusi optimal secara cepat pada suatu generasi. Algoritma ini dapat digunakan untuk mengoptimasi bobot pada arsitektur MLP.

Terdapat beberapa penelitian yang

menerapkan AG untuk optimasi bobot pada MLP. Salah satu penelitian dilakukan oleh Sari, Mahmudy, & Wibawa (2016), penerapan MLP digunakan dalam peramalan tingkat inflasi di Indonesia. Root Mean Square Error (RMSE) adalah teknik yang digunakan untuk menguji akurasi pada hasil peramalan dan hasilnya dibandingkan dengan model Sugeno FIS. Perbandingan menunjukkan bahwa nilai RMSE dengan MLP lebih baik dibandingkan dengan menggunakan Sugeno FIS. MLP juga dapat digunakan untuk pengenalan seperti pengenalan tanda tangan seperti yang dilakukan pada penelitian oleh Rahmi, Wijayaningrum, Mahmudy et al. (2016). Namun ternyata MLP kurang tepat untuk diaplikasikan pada objek ini. Karena akurasi yang dihasilkan rendah.

Mengacu pada penelitian yang telah ada sebelumnya, penulis mengajukan penelitian yang membahas tentang optimasi bobot *Multi-layer perceptron* untuk klasifikasi tingkat resiko penyakit stroke dengan menggunakan algoritma genetika. Algoritma genetika digunakan untuk mengoptimasi bobot koneksi arsitektur MLP sehingga diharapkan keluaran yang dihasilkan tersebut akan semakin optimal dan akurat.

2. LANDASAN KEPUSTAKAAN

2.1 Stroke

Stroke adalah penyakit yang disebabkan oleh terhentinya aliran darah yang diperlukan oleh otak akibat aliran darah tersumbat atau bahkan pecah. Hal ini mengakibatkan otak tidak mendapatkan suplai oksigen dan nutrisi yang kemudian menyebabkan rusaknya jaringan pada otak. Terdapat 2 macam golongan faktor penyebab seseorang dapat terkena stroke. Kedua faktor tersebut adalah faktor yang tidak dapat dirubah dan faktor yang dapat dirubah/dikontrol. Faktor yang tidak dapat dirubah yaitu usia, jenis kelamin, dan faktor keturunan. Sedangkan faktor yang dapat dirubah/dikontrol adalah riwayat penyakit jantung, hipertensi, obesitas, diabetes, hiperkolesterol, serta faktor aktivitas fisik dan pola makan (Sutrisno, 2007).

2.2 Jaringan Syaraf Tiruan

Jaringan Syaraf Tiruan (JST) adalah pengolah informasi yang terinspirasi dari cara kerja sistem syaraf secara biologis seperti otak yang memproses informasi. Elemen utamanya adalah struktur dari sistem pengolah informasi itu sendiri yaitu elemen pemrosesan yang saling

terkoneksi (*neuron*) dengan jumlah besar dan bekerja bersama untuk menyelesaikan masalah yang spesifik. JST dipercaya bisa diimplementasikan untuk pengenalan maupun klasifikasi. Salah satu struktur yang biasa digunakan untuk teknik klasifikasi adalah *Multi-layer perceptron* (MLP) (Rahmi, et al., 2016) dengan menggunakan pelatihan *backpropagation* pada metode pembelajarannya. Terdapat 2 tahap perhitungan pada algoritma ini yaitu *feedforward* dan *backpropagation*.

2.3 Algoritma Genetika

Algoritma genetika (AG) merupakan salah satu jenis algoritma evolusi yang paling populer dan berkembang seiring dengan pesatnya perkembangan teknologi informasi. AG mampu menyelesaikan beberapa permasalahan kompleks pada berbagai bidang yang sering menemukan permasalahan optimasi model matematika yang kompleks maupun yang sulit dibangun (Mahmudy, 2015).

3. FORMULASI PERMASALAHAN

Kasus yang diselesaikan pada penelitian ini adalah mengklasifikasi tingkat resiko seorang pasien terkena stroke. Data yang digunakan adalah hasil pemeriksaan darah dari Laboratorium Klinik Sejahtera Kota Probolinggo yang menunjukkan usia, tingkat kolesterol total, HDL, LDL, Trigliserida pada seseorang. Hasil pemeriksaan darah dapat memprediksi kemungkinan seseorang terkena stroke dari faktor-faktor tersebut. Pada penelitian ini akan diidentifikasi berapa tingkat resiko stroke yang mungkin dialami oleh seseorang dengan mengelompokkan menjadi tingkat normal, beresiko, berpotensi tinggi. dapat dilihat pada Tabel 1.

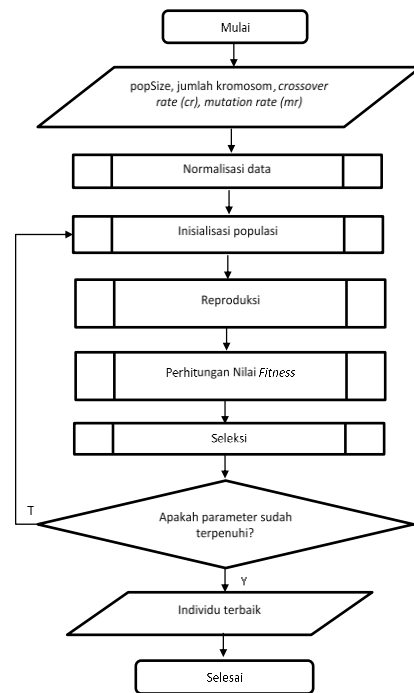
Tabel 1 Data penelitian

Usia	Total Kolesterol	HDL	LDL	Trigliserida	Kelompok
44	174	36,5	123,3	71	1
60	169	36,9	110,1	110	1
23	196	40,3	137,1	93	1
41	213	47,4	143,2	112	1
40	253	46,5	179,3	136	1
55	255	37,8	184,6	163	2
42	191	40,1	117,1	169	2
42	242	42,5	162,5	185	2
25	210	40,1	134,1	179	2
35	197	39,6	119,6	189	2

53	245	45,7	56,5	214	3
44	291	46,7	180,5	319	3

4. PERANCANGAN SISTEM

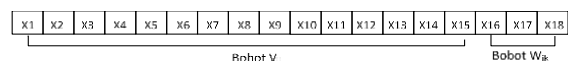
Penelitian ini secara garis besar akan melakukan optimasi bobot pada jaringan MLP. Tahap pertama yang dilakukan adalah inialisasi populasi dari AG yang akan membangkitkan individu secara acak dengan rentang tertentu. Individu inilah yang akan dijadikan bobot pada MLP sebelum dilakukan *feedforward*. Jika sudah diinputkan bobot pada MLP, selanjutnya akan dilatih sampai proses reproduksi. *Feedforward* akan melakukan perhitungan *fitness* untuk seluruh individu. Setelah itu baru menjalankan seleksi sampai mendapatkan populasi terbaik. Siklus algoritma utama sistem dapat dilihat pada Gambar 1.



Gambar 1 Diagram alir utama sistem

4.1 Representasi Kromosom dan Perhitungan Fitness

Representasi kromosom yang digunakan adalah pengkodean real (*real-coded genetic algorithm*). Kromosom direpresentasikan berdasarkan jumlah bobot pada *input layer* ke *hidden layer* (V_{ij}) dan dari *hidden layer* ke *output layer* (W_{jk}). Gambar 2 adalah representasi kromosom yang digunakan.



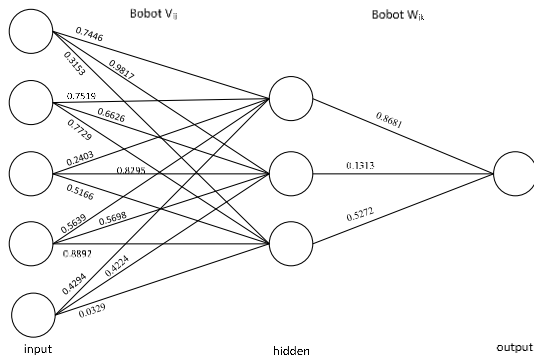
Gambar 2 Struktur Kromosom

Untuk melihat bagaimana AG berjalan, maka akan dihitung nilai *fitness* dengan kromosom yang sudah direpresentasikan. Nilai pada kromosom tersebut dijadikan sebagai nilai pada bobot pada jaringan MLP. Gambar 3 sebagai kromosom awal dan Gambar 4 Representasi bobot pada MLP untuk menunjukkan nilai kromosom sebagai bobot MLP. Nilai yang di inialisasi sudah melalui normalisasi dengan Persamaan 1.

$$x' = \left(0,8 * \frac{x - \text{minval}}{\text{maxval} - \text{minval}}\right) + 0,1 \quad (1)$$

		kromosom																	
		Bobot V _{ij}										Bobot W _{ik}							
indi	vidu	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15			
PS	737	0.6	0.1	0.1	0.5	0.9	0.2	0.5	0.4	0.3	0.9	0.1	0.2	0.5	0.8	0.6	0.8	0.3	0.1
	421	0.4	0.3	0.6	0.9	0.9	0.9	0.5	0.5	0.9	0.9	0.2	0.6	0.3	0.3	0.9	0.6	0.6	0.2
	363	0.1	0.3	0.4	0.4	0.0	0.1	0.4	0.9	0.6	0.0	0.6	0.0	0.7	0.2	0.9	0.1	0.7	

Gambar 3 Representasi kromosom



Gambar 4 Representasi bobot pada MLP

Setelah melakukan representasi bobot maka selanjutnya dilakukan perhitungan *fitness* menggunakan *feedforward*. Sehingga dapat diketahui apakah bobot sudah merupakan bobot terbaik yang dapat digunakan pada klasifikasi tingkat resiko penyakit stroke. Proses perhitungannya dengan representasi kromosom seperti pada Gambar 5.

		kromosom																	
indi	vidu	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15			
PS	737	0.6	0.1	0.1	0.5	0.9	0.2	0.5	0.4	0.3	0.9	0.1	0.2	0.5	0.8	0.6			
	421	0.4	0.3	0.6	0.9	0.9	0.9	0.5	0.5	0.9	0.9	0.2	0.6	0.3	0.3	0.9	0.6	0.6	0.2
	363	0.1	0.3	0.4	0.4	0.0	0.1	0.4	0.9	0.6	0.0	0.6	0.0	0.7	0.2	0.9	0.1	0.7	
	994	0.5	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	594	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	090	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	631	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	654	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	799	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	796	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	270	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	166	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	830	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	407	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
	332	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9

Gambar 5 Representasi kromosom

Kromosom AG diacak pada *range* [-1,1] dengan jumlah populasi awal 10. Dari tahap ini, AG dilakukan mulai dari reproduksi hingga seleksi.

4.2 Reproduksi Crossover

Metode *crossover* yang digunakan adalah *extended intermediate crossover*. Dengan menentukan *crossover rate* (*cr*) dan nilai *alpha*

pada *range* [-0.25,1.25]. Proses *crossover* mengacu pada Persamaan 3, 4, 5.

$$\text{offspring} = pc \times \text{popSize} \quad (3)$$

$$C_1 = P_1 + a(P_2 - P_1) \quad (4)$$

$$C_2 = P_1 + a(P_2 - P_1) \quad (5)$$

dimana

P_1 = kromosom induk pertama

P_2 = kromosom induk kedua

a = nilai alpha yang sudah ditentukan

4.3 Reproduksi Mutasi

Metode mutasi yang digunakan adalah *random mutation* yang mengacu pada Persamaan 6 dan 7 dengan nilai *r* diacak pada interval [-0.1, 0.1].

$$\text{Offspring} = mr \times \text{popsize} \quad (6)$$

$$x' = x' + r (\text{max} - \text{min}) \quad (7)$$

Keterangan:

max = nilai *range* maksimum dari *parents* yang dipilih untuk mutasi

min = nilai *range* maksimum dari *parents* yang dipilih untuk mutasi.

4.4 Perhitungan Nilai Fitness

Dari tahap *feedforward* yang sudah dilakukan pada seluruh individu, akan didapatkan nilai *error* yang digunakan untuk menghitung nilai MSE dengan Persamaan 8.

$$MSE = \frac{1}{k} \sum_{k=1}^n (t_k - y_k)^2 \quad (8)$$

Keterangan:

n = jumlah data

y_k = nilai *data output*

t_k = nilai *data target*

Dari nilai MSE tersebut dapat dihitung *fitness* dengan $\frac{1}{MSE}$.

4.5 Seleksi

Tahap seleksi dilakukan dengan *elitism selection* yang mengurutkan nilai *fitness* dari nilai terkecil pada tiap individu. Jumlah individu yang dipakai di generasi selanjutnya adalah individu terbaik sejumlah populasi yang ditentukan.

5. PENGUJIAN DAN ANALISIS

Terdapat 2 pengujian yang dilakukan yaitu terhadap metode pembelajaran *backpropagation*

dan algoritma genetika yang mengoptimasi bobot MLP. Skenario pengujian BP adalah pengujian *learning rate*, jumlah *neuron* pada *hidden layer*, dan jumlah iterasi. Sedangkan pada AG antara lain adalah pengujian ukuran populasi, ukuran generasi, dan nilai *Cr* dan *Mr*.

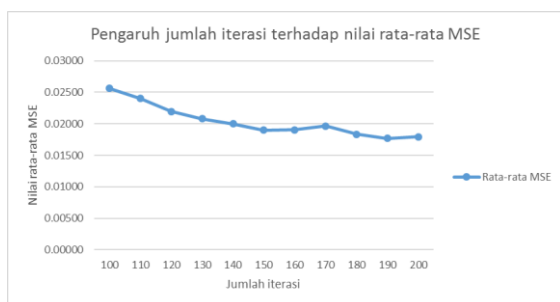
5.1 Pengujian Metode Backpropagation

5.1.1 Pengujian Jumlah Iterasi

Pengujian ini dilakukan untuk mendapatkan nilai MSE terbaik yaitu dengan nilai yang paling rendah. Iterasi yang dilakukan sebanyak 100 sampai 200 iterasi pada 10 kali percobaan. Begitu juga dengan nilai *learning rate* awal yaitu 0.5 serta jumlah *neuron* pada *hidden layer* sebanyak 3 *neuron*. Hasil pengujian dapat dilihat pada Tabel 2.

Tabel 2 Hasil uji jumlah iterasi

Banyak iterasi	Nilai fitness percobaan ke - i			Rata-rata MSE
	1	...	10	
100	0.02262		0.02440	0.02564
110	0.02426		0.02390	0.02401
120	0.02221		0.01895	0.02197
130	0.01914		0.01940	0.02081
140	0.02160		0.01802	0.01998
150	0.02147		0.02056	0.01900
160	0.01936		0.01962	0.01906
170	0.01960		0.01879	0.01965
180	0.01870		0.01984	0.01835
190	0.01657		0.01655	0.01770
200	0.01639		0.01800	0.01795



Gambar 6 Pengaruh jumlah iterasi terhadap nilai rata-rata MSE

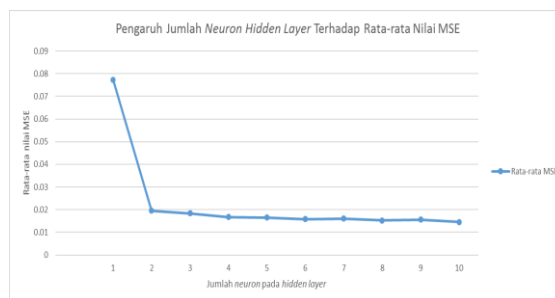
Dari hasil uji coba jumlah iterasi yang ditampilkan pada Gambar 6, dapat diambil kesimpulan bahwa semakin tinggi iterasi yang dilakukan akan menghasilkan nilai rata-rata MSE yang lebih kecil. Nilai MSE terbaik ada pada iterasi ke 190 dengan 0.01769.

5.1.2 Pengujian Jumlah Neuron pada Hidden Layer

Pengujian ini dilakukan pada jumlah 1 sampai 10 *neuron* di *hidden layer* (Huda, et al., 2016). Begitu juga dengan nilai *learning rate* awal yaitu 0.5 serta jumlah iterasi yang mengacu pada nilai MSE terbaik di pengujian sebelumnya yaitu 190. Hasil pengujian dapat dilihat pada Tabel 3.

Tabel 3 Hasil uji jumlah neuron pada hidden layer

Jumlah neuron hidden layer	Nilai fitness percobaan ke - i			Rata-rata MSE
	1	...	10	
1	0.08353		0.07480	0.07726
2	0.01862		0.01959	0.01950
3	0.01850		0.01903	0.01836
4	0.01833		0.01712	0.01677
5	0.01597		0.01610	0.01656
6	0.01544		0.01544	0.01583
7	0.01781		0.01731	0.01605
8	0.01482		0.01466	0.01527
9	0.01499		0.01627	0.01563



Gambar 7 Grafik pengaruh jumlah neuron pada hidden layer terhadap rata-rata nilai MSE

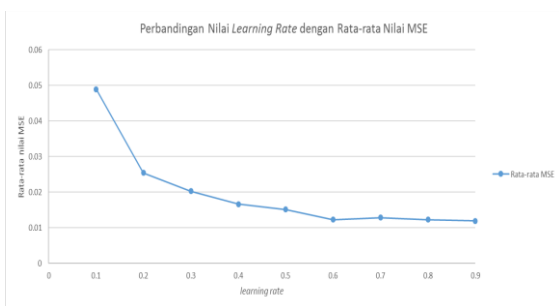
Uji coba perubahan jumlah *neuron* pada *hidden layer* di Gambar 7 menunjukkan bahwa nilai MSE terbaik, 0.01455 didapatkan pada jumlah *neuron* terbanyak yaitu 10. Artinya, nilai MSE ini sangat dipengaruhi oleh jumlah *neuron* pada *hidden layer* yang semakin banyak.

5.1.3 Pengujian Nilai Learning Rate

Pengujian ini dilakukan pada nilai 0.1 sampai 0.9 sebanyak 10 kali percobaan pada masing-masing nilai (Sari, et al., 2016). Dengan menggunakan jumlah iterasi 190 dan jumlah *neuron hidden layer* 10 seperti pada pengujian sebelumnya. Hasil pengujian parameter *learning rate* dapat dilihat pada Tabel 4.

Tabel 4 Hasil uji nilai *learning rate*

Nilai <i>learning rate</i>	Nilai <i>fitness percobaan ke - i</i>			Rata-rata MSE
	1	...	10	
0.9	0.01191		0.01201	0.01189
0.8	0.01219		0.01206	0.01228
0.7	0.01264		0.01293	0.01286
0.6	0.01346		0.01360	0.01229
0.5	0.01501		0.01445	0.01512
0.4	0.01649		0.01768	0.01664
0.3	0.02084		0.02140	0.02027
0.2	0.02405		0.02798	0.02539
0.1	0.05751		0.04628	0.04895



Gambar 8 Grafik pengaruh nilai *learning rate* dengan nilai rata-rata MSE

Berdasarkan Gambar 8 dapat bahwa nilai *learning rate* memiliki pengaruh yang besar terhadap nilai MSE. Semakin tinggi nilai *learning rate* yang digunakan, nilai MSE semakin baik. Rata-rata nilai MSE pun cenderung konstan setelah 0.6. Namun nilai rata-rata MSE yang terbaik ada pada 0.9 dengan nilai rata-rata MSE sebesar 0.01189.

5.2 Pengujian Algoritma Genetika

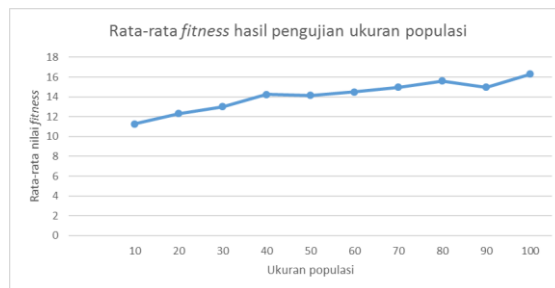
5.2.1 Pengujian Ukuran Populasi

Pengujian ini dilakukan dengan ukuran populasi dari 10 sampai 100. Jumlah generasi yang digunakan yaitu 100 dengan nilai awal *Cr* adalah 0,6 dan *Mr* adalah 0,4. Jumlah data yang digunakan sebanyak 100 data dari 3 kelompok resiko. Hasil pengujian ditampilkan pada Tabel 5.

Tabel 5 Hasil uji banyak populasi

Banyak populasi	Nilai <i>fitness percobaan ke - i</i>			Rata-rata <i>fitness</i>
	1	...	10	
10	11.5569		11.0367	11.2743
20	11.5140		12.9075	12.3096

30	12.3853		12.1851	13.0263
40	13.0334		13.8772	14.2387
50	14.7372		13.5203	14.1469
60	14.8647		13.6757	14.5136
70	14.1954		15.7946	14.9751
80	16.9869		15.0451	15.5934
90	15.5171		17.3550	14.9821
100	17.7894		16.1052	16.3184



Gambar 9 Grafik rata-rata *fitness* hasil pengujian ukuran populasi

Dari hasil uji coba pada Gambar 9, dapat diambil kesimpulan bahwa ukuran populasi yang semakin besar mempengaruhi nilai rata-rata *fitness*. Artinya, semakin besar ukuran populasi maka nilai rata-rata *fitness* akan semakin optimal. Nilai rata-rata *fitness* tertinggi dihasilkan pada 100 populasi. Selain itu jikadilihat dari proses pengujian, semakin besar nilai populasi akan memakan waktu yang lebih lama.

5.2.2 Pengujian Banyak Generasi

Pengujian ini melakukan percobaan banyak generasi dengan menggunakan *popSize* 100 dan kombinasi nilai *Cr* dan *Mr* 0,6 dan 0,4. Percobaan dilakukan dari generasi ke 100 - 1000 dengan hasil yang dapat dilihat pada Tabel 6.

Tabel 6 Hasil uji banyak generasi

Banyak generasi	Nilai <i>fitness percobaan ke-i</i>			Rata-rata <i>fitness</i>
	1	...	10	
100	17.7360		16.2540	15.9474
200	17.9517		17.2830	17.1801
300	16.2838		17.8660	17.8435
400	18.1635		18.2888	18.1958
500	18.2693		18.2179	18.1861
600	18.3063		18.2333	17.2047
700	18.3236		18.2146	17.9802
800	18.1617		16.5226	18.1024
900	15.4513		15.4289	17.5859

1000	18.3592	18.3396	17.2250
------	---------	---------	---------



Gambar 10 Hasil rata-rata *fitness* pengujian ukuran generasi

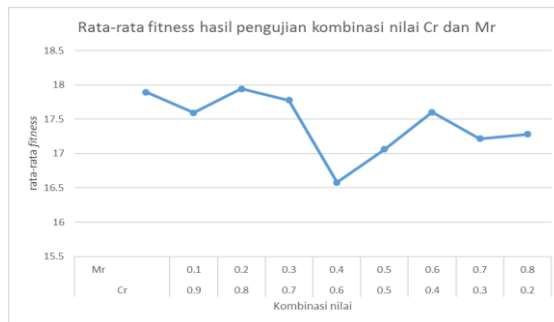
Pada Gambar 10 dapat dilihat bahwa nilai rata-rata *fitness* optimal dihasilkan dari generasi dengan jumlah 400. Generasi ke-120 ini menghasilkan nilai rata-rata *fitness* yang lebih tinggi daripada jumlah generasi dibawahnya, namun diatas angka 400 nilai rata-rata *fitness* cenderung menurun.

5.2.3 Pengujian Kombinasi *Crossover Rate* (*Cr*) dan *Mutation Rate* (*Mr*)

Pengujian ini menggunakan jumlah populasi 100 dengan generasi sebanyak 400 sesuai dengan hasil yang didapatkan pada pengujian sebelumnya. Sedangkan pada uji coba ini akan melakukan 10 kali percobaan pada kombinasi nilai *Cr* dan *Mr* dari 0,1 sampai 0,9 (Taufiq, et al., 2017). Hasil pengujian kombinasi nilai *Cr* dan *Mr* dapat dilihat pada Tabel 7.

Tabel 7 Hasil uji kombinasi nilai *Cr* dan *Mr*

<i>Cr</i>	<i>Mr</i>	Percobaan ke – i			Rata-rata <i>fitness</i>
		1	...	10	
0.9	0.1	18.1152		18.0981	17.8948
0.8	0.2	17.9773		18.3436	17.5921
0.7	0.3	18.3280		18.2898	17.9417
0.6	0.4	18.3321		18.0161	17.7767
0.5	0.5	17.8688		17.6607	16.5768
0.4	0.6	18.3378		17.8817	17.0625
0.3	0.7	18.2322		13.5223	17.6036
0.2	0.8	17.5736		18.2806	17.2181
0.1	0.9	17.2553		17.8767	17.2813



Gambar 11 Grafik hasil uji kombinasi nilai *Cr* dan *Mr*

Pada Gambar 11 dapat diambil kesimpulan bahwa nilai *fitness* terbaik didapatkan dari nilai *Cr* yang lebih besar dari *Mr* yaitu 0,8 dengan *Mr* 0,2. Nilai *fitness* terbaik yang dihasilkan adalah sebesar 17.9417.

5.3 Perbandingan Metode *Backpropagation* dan Algoritma Genetika

Pada tahap ini dilakukan pengujian untuk metode BP dengan metode AG menggunakan ukuran parameter terbaik dari masing-masing metode. Pengujian dilakukan dengan melakukan klasifikasi tingkat resiko penyakit stroke pada data yang sama sebanyak 100 data latih dan 100 data uji, kemudian dibandingkan hasil akurasi dengan hasil MSEnya. Nilai akurasi dan MSE dihasilkan dengan melakukan 10 kali percobaan untuk setiap metode. Tabel 8 merupakan hasil dari perbandingan kedua metode.

Tabel 8 Hasil perbandingan akurasi metode BP dan AG

Percobaan ke-i	Metode <i>backpropagation</i>		Metode AG	
	Akurasi data	MSE	Akurasi data	MSE
1	89.00%	0.012116213	84.00%	0.054523681
2	87.00%	0.012139039	84.00%	0.055856417
3	90.00%	0.012339532	5.00%	0.054579464
4	88.00%	0.012156596	9.00%	0.054605483
5	89.00%	0.012063489	84.00%	0.055527238
6	88.00%	0.012197345	84.00%	0.055057429
7	87.00%	0.012438702	84.00%	0.054483277
8	90.00%	0.011862457	4.00%	0.054826342
9	88.00%	0.012716967	84.00%	0.054465517
10	88.00%	0.012093995	84.00%	0.05507141

Berdasarkan hasil diatas dapat dilihat bahwa nilai akurasi data yang semakin tinggi cenderung memiliki nilai MSE yang semakin kecil atau semakin baik. Jika dilihat pada Tabel 8 metode AG akurasinya turun cukup signifikan pada beberapa kali percobaan dengan nilai MSE

yang stabil. Namun pada metode BP, nilai MSE stabil dengan nilai akurasi data yang juga cukup stabil. Sehingga pada akurasi data, metode BP memiliki hasil yang lebih stabil dan baik untuk klasifikasi.

6. KESIMPULAN

Penelitian dan pengujian yang sudah dilakukan untuk optimasi bobot MLP menggunakan algoritma genetika (AG) untuk klasifikasi tingkat resiko penyakit stroke menghasilkan beberapa kesimpulan, antara lain:

1. Algoritma genetika yang diterapkan untuk optimasi bobot MLP pada pengklasifikasian tingkat resiko penyakit stroke menggunakan range [-1,1] dengan *real coded genetic algorithm* (RCGA) untuk pembangkitan populasi, *extended intermediate crossover*, *random mutation*, serta *elitism selection* untuk tahap seleksi.
2. Pengujian yang dilakukan pada metode pembelajaran BP menghasilkan nilai rata-rata MSE sebesar 0.01188 dengan jumlah iterasi 190, jumlah neuron pada hidden layer sebanyak 10, dan nilai learning rate sebesar 0.9. Sedangkan pada pengujian AG menghasilkan nilai fitness terbaik sebesar 17.9417 dengan ukuran populasi sejumlah 100, banyak generasi 400, serta kombinasi nilai Cr dan Mr 0.8 dan 0.2.
3. Dengan menggunakan parameter terbaik dari BP dan AG, dilakukan klasifikasi tingkat resiko stroke dengan menerapkan fase *feedforward* dari pembelajaran BP. Klasifikasi tersebut menghasilkan nilai akurasi data serta nilai MSE yang dapat dijadikan perbandingan metode mana yang lebih baik dalam melakukan klasifikasi. Metode BP menghasilkan nilai rata-rata akurasi data sebesar 88% sedangkan metode AG menghasilkan nilai sebesar 60.60% dengan percobaan masing-masing 10 kali. Nilai rata-rata MSE metode BP adalah sebesar 0.0122 dan nilai rata-rata MSE metode AG adalah 0.0549.

DAFTAR PUSTAKA

Hamid, N. A. et al., 2011. Accelerating Learning Performance of Back Propagation Algorithm by Using Adaptive Gain Together with Adaptive Momentum and Adaptive Learning Rate on Classification Problems. Malaysia: Universiti Tun Hussein Onn Malaysia.

Huang, K., Shen, L., Chen, K. & Huang, M., 2013. Multilayer Perceptron with Generic Algorithm for Well Log Data Inversion. IEEE.

Huda, F., Mahmudy, W. F. & Tolle, H., 2016. Android Malware Detection Using Backpropagation Neural Network. Indonesian Journal of Electrical Engineering and Computer Science (IJEECS).

Kurnianingtyas, D. & Mahmudy, W. F., 2017. Optimasi Derajat Keanggotaan Fuzzy Tsukamoto Menggunakan Algoritma Genetika Untuk Diagnosis Penyakit Sapi Potong. Jurnal TIHK, 4(1), pp. 8-18.

Mahmudy, W. F., 2015. Dasar-dasar Algoritma Evolusi. s.l.:Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya, Malang.

Mahmudy, W. F., Marian, R. M. & Luong, L., 2014. Hybrid genetic algorithms for part type selection and machine loading problems with alternative production plans in flexible manufacturing system. ECTI-CIT, Volume 8, No.1, pp. 80-93.

Mahmudy, W., Marian, R. & Luong, L., 2013. Modeling and optimization of part type selection and loading problems in flexible manufacturing system using real coded genetic algorithms. International Journal of Electrical, Computer, Electronics and Communication Engineering, Volume 7, No.4, pp. 251-260.

Rahmi, A., Wijayaningrum, V. N., Mahmudy, W. F. & Parewe, A. M. A. K., 2016. Offline Signatur Recognition using Back Propagation Neural Network. Indonesia Journal of electrical Engineering and Computer Science, Volume IV.

Sari, I. P., 2014. Prediksi Data Harga Saham Harian Menggunakan Feedforward Neural Network dengan Pelatihan Algoritma Genetika (Studi kasis pada harga saham harian PT. XL Axiata Tbk). Jurusan Statistika, Fakultas Sains dan Matematika, Universitas Diponegoro .

Sari, N., Mahmudy, W. F. & Wibawa, A., 2016. Backpropagation on Neural Network Method for Inflation Rate Forecasting in Indonesia. IJEECS.

Taufiq, M. N., Dewi, C. & Mahmudy, W. F.,

2017. Optimasi Komposisi Pakan Untuk Peggemukkan Sapi Potong. JTIIK, Volume 1, No.7, pp. 571-582.
- Wibisono, K. & Mahmudy, W. F., 2016. Optimasi Pemilihan Antihipertensi Menggunakan Algoritma Genetika. Prosiding Seminar Nasional Teknologi dan Rekayasa Informasi Tahun 2016.
- Yastroki, 2012. Solusi Sehat Mengatasi Stroke. Tangerang: AgroMedia.
- Zamani, A. M., Amaliah, B. & Munif, A., 2012. Implementasi Algoritma Genetika pada Struktur Backpropagation Neural Network untuk Klasifikasi Kanker Payudara. JURNAL TEKNIK ITS, Volume I.