

Tingkat Ketepatan Pengenalan Pola Data Algoritma Pelatihan pada Jaringan Syaraf Tiruan Menggunakan Model Neuron 10-16-1 dan 15-29-1

*The Level of Data Pattern Recognition Accuracy on Neural Network
Training Algorithm Using 10-16-1 and 15-29-1 Model Neuron*

Hindayati Mustafidah*, Mu'ammirotus Sholihah

Teknik Informatika, Universitas Muhammadiyah Purwokerto, Indonesia

*corr_author: h.mustafidah@ump.ac.id

ABSTRAK

Algoritma pelatihan merupakan bagian terpenting dalam Jaringan Syaraf Tiruan (JST). Kinerja algoritma ini dipengaruhi oleh beberapa parameter jaringan diantaranya banyaknya neuron dalam lapisan input, banyaknya neuron dalam lapisan tersembunyi, maksimum epoch, *learning rate* (lr). Pada penelitian terdahulu dilakukan pengujian menggunakan model neuron 6-10-1 diperoleh hasil bahwa algoritma pelatihan *Levenberg-Marquardt* (*trainlm*) adalah algoritma yang paling tepat dalam mengenali pola data dengan tingkat rata-rata kesesuaian 87.5%. Pada penelitian ini dilakukan pengujian tingkat ketepatan pengenalan pola data algoritma pelatihan pada jaringan syaraf tiruan menggunakan model neuron 10-16-1 dan 15-29-1, parameter jaringan yang digunakan diantaranya target error = 0.001 (10^{-3}), maksimum epoch = 10000 (10^4), nilai *learning rate* (lr) = 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0. Hasil dari penelitian yang telah dilakukan adalah pada model 10-16-1 algoritma pelatihan JST yang paling optimal tingkat ketepatan pengenalan pola data ditinjau dari rata-rata delta terkecil dan besarnya persentase ketepatan pengenalan pola data yaitu algoritma pelatihan *Levenberg-Marquardt* (*trainlm*) dengan nilai rata-rata delta = 0.00632000000 pada *learning rate* (lr) = 0.9 dengan persentase kecocokan mengenali pola data sebesar 100%. Pada model 15-29-1 algoritma pelatihan JST yang paling optimal tingkat ketepatan pengenalan pola data ditinjau dari rata-rata delta terkecil dan besarnya persentase kecocokan rata-rata delta algoritma pelatihan pada setiap *learning rate* (lr) adalah *Levenberg-Marquardt* (*trainlm*) dengan nilai rata-rata delta = 0.00530500000 pada *learning rate* (lr) = 0.7 dengan persentase ketepatan mengenali pola data sebesar 100%.

Kata-kata Kunci : Algoritma pelatihan, 10-16-1, 15-29-1, pola data, *Levenberg-Marquardt*

ABSTRACT

Training algorithm is the most important part in Artificial Neural Network (ANN). The performance of this algorithm is influenced by several network parameters including the number of neurons in the input layer, the number of neurons in the hidden layer, maximum epoch, learning rate (lr). In a previous study testing use a neuron model 6-10-1 the test results obtained that the Levenberg-Marquardt (trainlm) algorithm is the most appropriate algorithm in recognizing data pattern with an average level of appropriateness of 87.5%. In this study, testing the level of accuracy of data pattern recognition training algorithm on artificial neural network use the neuron model 10-16-1 and 15-29-1, the networks parametes used between error targets = 0.001 (10^{-3}), maximum epoch = 10000 (10^4), the

value of learning rate (lr) = 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0. The result of the research carried out in neuron model 10-16-1ANN training algorithm which is the most optimal level accuracy of data pattern recognition in terms of the smallest delta average and the percentage accuracy of data pattern recognition is Levenberg-Marquardt (trainlm) training with the value the average delta = 0.00632000000 at the learning rate (lr) = 0.9 with a percentage of compatibility recognizing the data pattern of 100%. In the model 15-29-1 ANN training algorithm the most optimal level of accuracy of data pattern recognition in terms of the smallest delta average and the percentage match for the average training algorithm delta at each learning rate (lr) is Levenberg-Marquardt (trainlm) with the average value of data = 0.00530500000 at the learning rate (lr) = 0.7 with the percentage of accuracy in recognition the data pattern of 100%.

Keywords: Training algorithm, 10-16-1, 15-29-1, data patterns, Levenberg-Marquardt.

PENDAHULUAN

Pengenalan pola merupakan salah satu kegunaan dalam Jaringan Syaraf Tiruan (JST). JST merupakan salah satu upaya manusia untuk memodelkan cara kerja atau fungsi sistem syaraf manusia dalam mengerjakan tugas-tugas tertentu. Seperti halnya syaraf manusia mampu mengenali banyak wajah seseorang meskipun tidak setiap wajah memiliki kesamaan, namun tampilan setiap wajah dapat berubah tergantung pada keadaan tertentu hal itu bukanlah pekerjaan yang sulit bagi manusia berbeda dengan komputer. Jaringan syaraf tiruan (*artificial neural network*/JST) adalah model komputasi yang terinspirasi secara biologis, JST terdiri dari beberapa elemen pengolahan (neuron) dan ada hubungan antara neuron (Shanmuganathan & Samarasinghe, 2016).

Bahasan yang dapat dikaji pada JST lebih banyak pada aplikasi dan pengembangan algoritma-algoritma pembelajaran/pelatihan. Algoritma pelatihan merupakan bagian terpenting dalam JST, algoritma ini menentukan kinerja dari JST sebagai alat bantu dalam memecahkan masalah, misalnya untuk keperluan peramalan, diagnosa, dan pengenalan pola. *Backpropagation* merupakan algoritma pembelajaran dalam JST yang terawasi (*supervised learning*) dan biasanya digunakan oleh banyak *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron yang ada pada lapisan tersembunyi (Kusumadewi, 2004). Dalam *supervised learning*, jaringan dipandu oleh sejumlah pasangan data (masukan dan target) yang berfungsi sebagai pembimbing untuk melatih jaringan hingga diperoleh bobot yang terbaik. Pelatihan dilakukan dengan memberikan pasangan pola-pola masukan dan keluaran (Sutojo et al., 2011).

Terdapat 12 algoritma pelatihan jaringan *backpropagation* dalam JST *traincgf*, *traincgp*, *traincgb*, *trainscg*, *traingd*, *traingda*, *traingdm*, *traingdx*, *trainlm*, *trainoss*, dan *trainrp* (Kusumadewi, 2004). Kinerja algoritma dipengaruhi oleh beberapa parameter jaringan seperti yang disampaikan oleh (Kusumadewi, 2004) diantaranya:

- *Maksimum epoch*

Maksimum epoch adalah jumlah epoch maksimum yang boleh dilakukan selama proses pelatihan. Iterasi/langkah akan dihentikan apabila nilai epoch melebihi maksimum epoch.

- *Kinerja tujuan/ target error*

Kinerja tujuan adalah target nilai fungsi kinerja. Iterasi akan dihentikan apabila nilai fungsi kinerja kurang dari atau sama dengan kinerja tujuan.

- *Learning rate (lr)*

Learning rate adalah laju pembelajaran. Semakin besar nilai *learning rate* akan berimplikasi pada semakin besarnya langkah pembelajaran. Jika *learning rate* diset terlalu besar, maka algoritma akan menjadi tidak stabil. Sebaliknya, jika *learning rate* diset terlalu

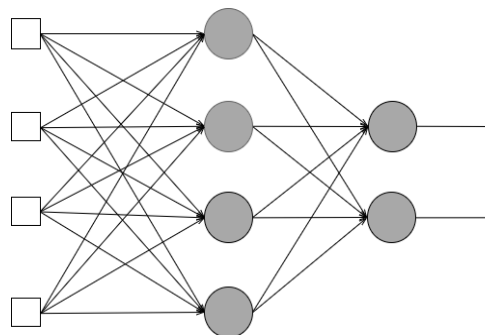
kecil, maka algoritma akan konvergen dalam jangka waktu yang sangat lama. Nilai $lr = 0$ s/d 1.

- *Neuron pada lapisan tersembunyi*

Penggunaan jumlah neuron yang terlalu sedikit pada lapisan tersembunyi akan menyebabkan underfitting. Underfitting terjadi ketika terlalu sedikit neuron di lapisan tersembunyi untuk memadai dalam mendeteksi sinyal kumpulan data yang rumit. Menggunakan terlalu banyak neuron di lapisan tersembunyi dapat menyebabkan overfitting. Overfitting terjadi ketika jaringan saraf memiliki begitu banyak kapasitas pemrosesan informasi sehingga jumlah informasi yang terkandung dalam set pelatihan tidak cukup untuk melatih semua neuron di lapisan tersembunyi (Heaton, 2008).

Penentuan jumlah neuron yang digunakan dalam lapisan tersembunyi mengikuti aturan: 1) jumlah neuron dalam lapisan tersembunyi baiknya di antara ukuran lapisan masukan dan lapisan keluaran, atau 2) jumlah neuron dalam lapisan tersembunyi sebaiknya $2/3$ dari ukuran lapisan masukan ditambah ukuran lapisan keluaran, atau 3) jumlah neuron dalam lapisan tersembunyi sebaiknya lebih kecil atau kurang dari dua kali ukuran neuron dalam lapisan *input* (Heaton, 2008).

Suyanto (2014) menyatakan sel syaraf (neuron) adalah unit pemrosesan informasi yang merupakan dasar dari operasi JST. Pola dimana neuron-neuron pada JST disusun berhubungan erat dengan algoritma pembelajaran yang digunakan untuk melatih jaringan. Arsitektur jaringan dengan banyak lapisan (*multi layer*) merupakan arsitektur jaringan dengan satu atau lebih lapisan tersembunyi (*hidden layer*) dengan *computation nodes* yang berhubungan disebut *hidden neuron* atau *hidden unit* yang ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur jaringan *multi layer feedforward* dengan satu lapisan tersembunyi dan satu lapisan keluaran

Pada Gambar 1 mengilustrasikan jaringan *multi layer feedforward* untuk kasus satu lapisan tersembunyi. Jaringan pada Gambar 1 bisa disebut sebagai jaringan 4-4-2, dalam arti bahwa jaringan tersebut mempunyai 4 neuron pada lapisan input, 4 neuron pada lapisan tersembunyi, dan 2 neuron pada lapisan keluaran.

Taniredja & Mustafidah (2011) menyampaikan bahwa SPSS singkatan dari *Statistical Package For Social Science* yaitu merupakan paket statistika untuk ilmu – ilmu sosial, akan tetapi SPSS banyak juga digunakan untuk bidang – bidang lain yang memang membutuhkan statistika. SPSS berfungsi untuk membantu memproses data-data statistik secara cepat dan tepat, serta menghasilkan berbagai output yang dikehendaki oleh para pengambil keputusan. Statistik dapat diartikan sebagai suatu kegiatan yang bertujuan untuk mengumpulkan data, meringkas atau menyajikan data kemudian menganalisis data dengan menggunakan metode tertentu, dan menginterpretasikan hasil dari analisa tersebut.

Santoso (2017) menyatakan bahwa ANOVA atau uji F digunakan untuk pengujian lebih dari dua sampel yang bertujuan untuk mengetahui apakah ada perbedaan yang

signifikan antara rata-rata hitung beberapa kelompok data. Pengujian statistik dilakukan dengan tujuan untuk mengetahui apakah ada pengaruh atau perbedaan yang signifikan dan seberapa besar jika ada pengaruh terhadap output yang dihasilkan dengan membandingkan tingkat *error*.

Beberapa algoritma pelatihan dalam JST telah diuji dan diterapkan dalam membantu proses pemecahan masalah, namun belum diuji secara menyeluruh diantara algoritma-algoritma yang ada untuk mengetahui algoritma pelatihan dalam JST yang paling optimal tingkat ketelitiannya dalam pengenalan pola data menggunakan model neuron 10-16-1 dan model 15-29-1. Algoritma pelatihan perbaikan metode *batch mode* (*traingda*, *traingdx*, *trainrp*, *trainbfg*, *trainoss* dan *trainlm*) dalam JST telah diuji untuk mengenali pola data kasus prediksi prestasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto pengujian ini menggunakan model neuron 3-10-1 dan menghasilkan algoritma *trainlm* memiliki tingkat kepercayaan 95% dengan rata-rata *error* = 0.0063 (Wibowo, Sugiyanto, & Mustafidah, 2013). Telah menguji 12 algoritma pelatihan dalam kesesuaian mengenali pola data, yaitu *traingd*, *traingdm*, *traingdx*, *trainrp*, *traingda*, *traingcf*, *traingcp*, *traingcb*, *traingcg*, *traingbg*, *traingss* dan *trainlm*. Penelitian ini merupakan hasil dari pengujian dengan menggunakan model neuron 6-10-1 dan menghasilkan pada interval kepercayaan 95% algoritma *trainlm* adalah algoritma yang paling tepat dalam pengenalan pola data dengan tingkat rata-rata ketepatan 87.5% (Mustafidah, Hartati, Wardoyo, & Harjoko, 2014). Pengujian terhadap tingkat *error* terhadap 12 algoritma pelatihan pada jaringan *backpropagation* dengan menggunakan model neuron 5-10-1 menghasilkan algoritma pelatihan *trainlm* memiliki *error* terkecil dengan level $\alpha = 5\%$ dan memberikan *error* = 0.0001986858. Dengan demikian algoritma *trainlm* dapat dianggap sebagai algoritma pelatihan dalam backpropagation untuk aplikasi dalam menyelesaikan masalah (Mustafidah & Suwarsito, 2015a).

Berdasarkan hasil penelitian yang telah dilakukan sebelumnya diperoleh informasi bahwa algoritma *trainlm* adalah algoritma pelatihan yang paling optimal ditinjau dari *error* yang dihasilkan jaringan. Dengan menggunakan 10 neuron pada lapisan tersembunyi. Oleh sebab itu perlu diketahui algoritma pelatihan yang paling optimal tingkat ketelitiannya dalam pengenalan pola data ditinjau dari rata-rata delta terkecil (data hasil keluaran dikurangi dengan data target yang dihasilkan oleh jaringan, selisih antara keduanya disebut dengan delta (Δ)) dan besarnya persentase ketepatan mengenali pola data dengan menggunakan model neuron 10-16-1 (10 neuron pada lapisan masukan, 16 neuron pada lapisan tersembunyi, 1 neuron pada lapisan keluaran) dan model neuron 15-29-1 (15 neuron pada lapisan masukan, 29 neuron pada lapisan tersembunyi, 1 neuron pada lapisan keluaran).

METODE PENELITIAN

Penelitian ini merupakan penelitian metode campuran (*mixed method*) yaitu penelitian pengembangan dengan pengujian kuantitatif dan kualitatif. Pengembangan kode program dilakukan menggunakan bahasa pemrograman MATLAB karena terdapat *toolbox*. Dalam MATLAB tersedia *toolbox* yang dapat digunakan untuk jaringan syaraf tiruan (Tjolleng, 2017). Kemudian data keluaran program dianalisis menggunakan uji statistika inferensi dengan SPSS untuk mendapatkan rata-rata delta terkecil dan statistika deskriptif untuk mendapatkan persentase terbesar tingkat ketelitian dalam pengenalan pola data.

1. Variabel Penelitian

Variabel yang diteliti pada penelitian ini menggunakan beberapa parameter jaringan diantaranya yaitu:

- Algoritma pelatihan : *traingcf*, *traingcp*, *traingcb*, *traingcg*, *traingd*, *traingda*, *traingdm*, *traingdx*, *trainrp*, *trainbfg*, *trainoss*, *trainlm*.

- b. Epoch maksimum : 10000 (10^4)
- c. *Learning rate* (lr) : 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1
- d. *Target error* : 0.001 (10^{-3})
- e. Neuron masukan (X) : 10, 15
- f. Neuron keluaran (Y) : 1
- g. Neuron dalam lapisan tersembunyi (Z) : 16, 29

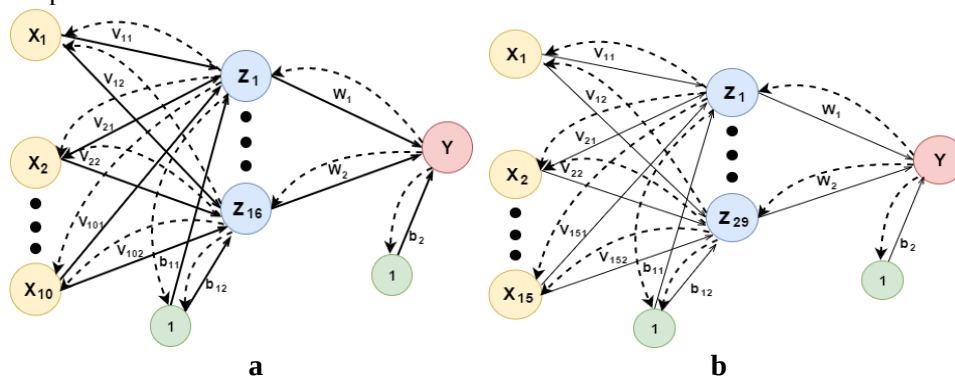
2. Sumber Data

Sumber data yang digunakan untuk mengetahui tingkat ketelitian pengenalan pola data adalah data dokumentasi yang diperoleh dari penelitian (Mustafidah & Suwarsito, 2015b)

3. Desain Arsitektur Penelitian

a. Desain model neuron

Desain arsitektur model neuron yang digunakan dalam penelitian ini adalah struktur jaringan *backpropagation* dengan desain arsitektur model neuron 10-16-1 dan 15-29-1 terlihat pada Gambar 2.



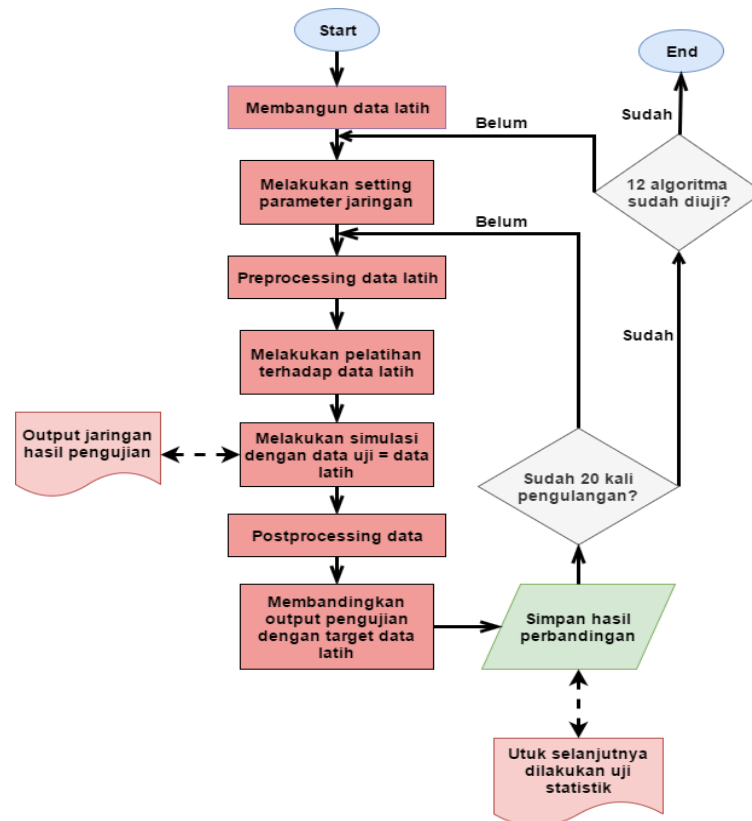
Gambar 2. Arsitektur model neuron: 10-16-1 (a); 15-29-1 (b)

b. Desain Alur Program JST

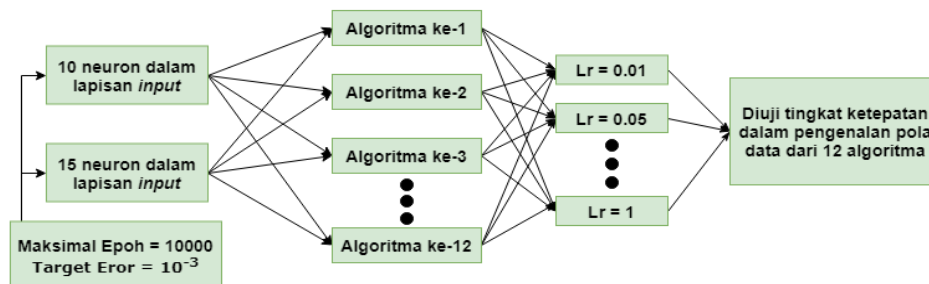
Berikut merupakan desain struktur program JST dari penelitian (Mustafidah & Suwarsito, 2016) seperti tersaji pada Gambar 3.

4. Analisis Data

Data keluaran jaringan yang telah dijalankan oleh setiap algoritma pelatihan yaitu sebanyak sebanyak 12 algoritma terhadap setiap lr, selanjutnya dilakukan uji statistik dengan desain seperti pada Gambar 4.



Gambar 3. Arsitektur alur program JST

Gambar 4. Desain arsitektur pengujian statistik untuk setiap *learning rate*

Analisis data yang digunakan dalam penelitian ini yaitu menggunakan metode analisis inferensi untuk menentukan rata-rata perbedaan dan statistik deskriptif untuk menentukan presentase kesesuaian dalam pengenalan pola data. Kesesuaian pengenalan pola data ini dilakukan dengan cara menentukan selisih antara keluaran jaringan dengan target, yang untuk selanjutnya disebut dengan “delta”. Pengujian statistik inferensia dilakukan menggunakan ANOVA dengan langkah-langkah sebagai berikut:

a. Menentukan hipotesis

H_0 : tidak terdapat perbedaan tingkat pengenalan pola data pada 12 algoritma pelatihan.

H_1 : terdapat perbedaan tingkat pengenalan pola data pada 12 algoritma pelatihan.

b. Menentukan nilai alpha (α)

Dalam penelitian ini digunakan nilai alpha (α) = 5%.

c. Menentukan alat uji

Alat uji yang digunakan yaitu menggunakan uji F sebagai berikut:

$H_0 : \mu_1 = \mu_2 = \dots = \mu_k$ vs $H_1 : \text{minimal 2 mean tidak sama.}$

$$F_{hit} = \frac{MST}{MSE} \sim F_{k-1, N-k}$$

dimana $MST = \text{Mean Square of Treatment}$ dan $MSE = \text{Mean Square of Error}$

Dalam implementasinya pengujian ini dilakukan menggunakan perangkat lunak SPSS.

d. Pengambilan keputusan

Kesimpulan diambil berdasarkan nilai signifikan yang diperoleh (sig.) dengan ketentuan H_0 ditolak jika nilai $\text{sig} < \alpha$.

HASIL DAN PEMBAHASAN

1. Data penelitian

Data penelitian berupa data random baik data masukan jaringan dan data target. Data masukan jaringan dengan 10 neuron pada lapisan *input* ($X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}$) dan Y adalah data target tersaji pada Tabel 1. Data masukan jaringan dengan 15 neuron pada lapisan *input* ($X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}, X_{13}, X_{14}, X_{15}$) dan Y adalah data target tersaji pada Tabel 2.

2. Analisis Data

a. Model neuron 10-16-1

- Analisis Inferensi

Hasil uji setiap algoritma pelatihan JST terhadap *learning rate* menggunakan uji statistik ANOVA dengan tingkat $\alpha (\alpha) = 5\%$ pada model neuron 10-16-1 untuk menentukan algoritma pelatihan yang paling optimal dalam ketepatan pengenalan pola data ditinjau dari nilai rata-rata delta terkecil tersaji pada Tabel 3.

Tabel 1. Data masukan jaringan (X) dan target (Y) dengan 10 neuron pada lapisan input

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	Target (Y)
5.8279	2.0907	4.1537	2.1396	6.8333	4.5142	6.0854	0.8408	1.2105	2.3189	4.3979
4.2350	3.7982	3.0500	6.4349	2.1256	0.4390	0.1576	4.5436	4.5075	2.3931	3.4005
5.1551	7.8333	8.7437	3.2004	8.3924	0.2719	0.1635	4.4183	7.1588	0.4975	3.1422
3.3395	6.8085	0.1501	9.6010	6.2878	3.1269	1.9007	3.5325	8.9284	0.7838	3.6508
4.3291	4.6110	7.6795	7.2663	1.3377	0.1286	5.8692	1.5361	2.7310	6.4082	3.9324
2.2595	5.6783	9.7084	4.1195	2.0713	3.8397	0.5758	6.7564	2.5477	1.9089	5.9153
5.7981	7.9421	9.9008	7.4457	6.0720	6.8312	3.6757	6.9921	8.6560	8.4387	1.1975
7.6037	0.5918	7.8886	2.6795	6.2989	0.9284	6.3145	7.2751	2.3235	1.7390	0.3813
5.2982	6.0287	4.3866	4.3992	3.7048	0.3534	7.1763	4.7838	8.0487	1.7079	4.5860
6.4053	0.5027	4.9831	9.3338	5.7515	6.1240	6.9267	5.5484	9.0840	9.9430	8.6987

Tabel 2. Data masukan jaringan (X) dan target (Y) dengan 15 neuron pada lapisan input

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	...	X_{15}	Target (Y)
9.5013	4.0571	0.1527	5.0281	4.9655	7.0274	1.3652	2.2595		9.3424	6.8732
2.3114	9.3547	7.4679	7.0947	8.9977	5.4657	0.1176	5.7981		2.6445	3.4611
6.0684	9.1690	4.4510	4.2889	8.2163	4.4488	8.9390	7.6037		1.6030	1.6603
4.8598	4.1027	9.3181	3.0462	6.4491	6.9457	1.9914	5.2982		8.7286	1.5561
8.9130	8.9365	4.6599	1.8965	8.1797	6.2131	2.9872	6.4053		2.3788	1.9112
7.6210	0.5789	4.1865	1.9343	6.6023	7.9482	6.6144	2.0907		6.4583	4.2245
4.5647	3.5287	8.4622	6.8222	3.4197	9.5684	2.8441	3.7982		9.6689	8.5598
0.1850	8.1317	5.2515	3.0276	2.8973	5.2259	4.6922	7.8333		6.6493	4.9025
8.2141	0.0986	2.0265	5.4167	3.4119	8.8014	0.6478	6.8085		8.7038	8.1593
4.4470	1.3889	6.7214	1.5087	5.3408	1.7296	9.8833	4.6110		0.0993	4.6077
6.1543	2.0277	8.3812	6.9790	7.2711	9.7975	5.8279	5.6783		1.3701	4.5735
7.9194	1.9872	0.1964	3.7837	3.0929	2.7145	4.2350	7.9421		8.1876	4.5069
9.2181	6.0379	6.8128	8.6001	8.3850	2.5233	5.1551	0.5918		4.3017	4.1222
7.3821	2.7219	3.7948	8.5366	5.6807	8.7574	3.3395	6.0287		8.9032	9.0161
1.7627	1.9881	8.3180	5.9356	3.7041	7.3731	4.3291	0.5027		7.3491	0.0558

Tabel 3 Rata-rata delta algoritma pelatihan pada setiap *learning rate* model 10-16-1

LR	Algoritma Pelatihan	Rata-rata Delta
0.01	<i>Levenberg-Marquardt (trainlm)</i>	0.00786500000
0.05	<i>Levenberg-Marquardt (trainlm)</i>	0.01024000000
0.1	<i>Levenberg-Marquardt (trainlm)</i>	0.00775500000
0.2	<i>Levenberg-Marquardt (trainlm)</i>	0.00972000000
0.3	<i>Levenberg-Marquardt (trainlm)</i>	0.00866000000
0.4	<i>Levenberg-Marquardt (trainlm)</i>	0.00807500000
0.5	<i>Levenberg-Marquardt (trainlm)</i>	0.00670000000
0.6	<i>Levenberg-Marquardt (trainlm)</i>	0.00883000000
0.7	<i>Levenberg-Marquardt (trainlm)</i>	0.00696500000
0.8	<i>Levenberg-Marquardt (trainlm)</i>	0.00995500000
0.9	<i>Levenberg-Marquardt (trainlm)</i>	0.00632000000
1.0	<i>Levenberg-Marquardt (trainlm)</i>	0.00739500000

Berdasarkan Tabel 3 menunjukkan bahwa algoritma pelatihan yang paling optimal ketepatan pengenalan pola data pada model neuron 10-16-1 ditinjau dari rata-rata delta terkecil adalah algoritma *Levenberg-Marquardt (trainlm)* pada *learning rate* (*lr*) = 0.9 dengan nilai rata-rata delta = 0.00632000000.

- Analisis deskriptif

Analisis statistik deskriptif dilakukan untuk mengetahui persentase dari rata-rata delta setiap algoritma pelatihan terhadap *learning rate* dalam ketepatan pengenalan pola data. Persentase rata-rata algoritma pelatihan pada setiap *lr* ditampilkan pada Tabel 4.

Tabel 4. Persentase rata-rata delta algoritma pelatihan pada setiap *learning rate* model 10-16-1

Algoritma	<i>Learning rate</i>												Rata-rata
	0.01	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	
Trainbfg	10	10	0	0	0	10	20	10	0	10	0	0	6
Traincgb	100	100	100	100	100	100	80	100	100	100	100	100	98
Traincgf	100	80	80	100	90	90	100	100	100	80	100	100	93
Traincgp	100	100	100	100	100	100	100	100	100	100	100	100	100
Traingd	100	0	0	0	0	0	0	0	0	0	0	0	8
Traingda	90	10	0	10	30	0	20	10	20	60	0	0	21
Traingdm	80	0	0	0	0	0	0	0	0	0	0	0	7
Traingdx	80	30	0	0	0	10	10	0	0	0	0	0	11
Trainlm	100	100	100	100	100	100	100	100	100	100	100	100	100
Trainoss	10	10	10	0	10	10	10	10	10	10	0	10	8
Trainrp	100	100	100	100	100	100	100	100	100	100	100	100	100
Trainscg	50	100	100	100	80	100	100	100	100	70	100	100	92

Berdasarkan Tabel 4 dapat diketahui bahwa algoritma pelatihan *Polak-Ribière* (*traincgp*), *Levenberg-Marquardt* (*trainlm*) dan *Resilient Backpropagation* (*trainrp*) memiliki presentase kecocokan mengenali pola data sebesar 100%.

b. Model neuron 15-29-1

- Analisis inferensi

Hasil uji setiap algoritma pelatihan JST terhadap *learning rate* menggunakan uji statistik ANOVA dengan tingkat alpha (α) = 5% pada model neuron 15-29-1 untuk menentukan algoritma pelatihan yang paling optimal dalam ketepatan pengenalan pola data ditinjau dari nilai rata-rata delta terkecil tersaji pada Tabel 5.

Tabel 5. Rata-rata delta algoritma pelatihan pada setiap *learning rate* model 15-29-1

LR	Algoritma Pelatihan	Rata-rata Delta
0.01	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00632500000
0.05	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00743000000
0.1	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00805000000
0.2	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00610500000
0.3	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00744500000
0.4	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00607000000
0.5	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00628500000
0.6	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00696000000
0.7	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00530500000
0.8	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00879500000
0.9	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00756500000
1.0	<i>Levenberg-Marquardt</i> (<i>trainlm</i>)	0.00785000000

Berdasarkan Tabel. 5 menunjukkan bahwa algoritma pelatihan yang paling optimal tingkat ketepatan pengenalan pola data pada model neuron 15-29-1 ditinjau dari rata-rata delta terkecil adalah algoritma *Levenberg-Marquardt* (*trainlm*) pada *learning rate* = 0.7 dengan nilai rata-rata delta = 0.00530500000.

- Analisis deskriptif

Analisis statistik deskriptif untuk mengetahui persentase dari rata-rata delta dari setiap algoritma pelatihan terhadap *learning rate* dalam ketepatan pengenalan pola data.

Persentase rata-rata algoritma pelatihan pada setiap *learning rate* ditampilkan pada Tabel 6.

Tabel 6. Persentase rata-rata delta algoritma pelatihan pada setiap *learning rate* model 15-29-1

Algoritma	<i>Learning rate</i>												Rata-rata
	0.01	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	
Trainbfg	13	0	7	20	0	13	0	7	0	0	7	0	6
Traincgb	100	100	100	100	100	100	100	100	100	100	100	93	99
Traincgf	100	100	100	100	100	100	100	100	100	100	100	100	100
Traincgp	100	100	100	100	100	100	100	100	100	100	100	100	100
Traingd	73	0	0	0	0	0	0	0	0	0	0	0	6
Traingda	93	0	0	0	0	0	0	0	0	0	0	0	8
Traingdm	0	0	0	0	0	0	0	0	0	0	0	0	0
Traingdx	0	13	0	0	13	20	13	20	7	13	13	13	10
Trainlm	100	100	100	100	100	100	100	100	100	100	100	100	100
Trainoss	13	40	20	0	13	7	60	20	13	73	20	40	27
Trainrp	100	100	100	100	100	100	100	100	100	100	100	100	100
Trainscg	100	100	100	100	100	100	100	100	100	100	100	100	100

Berdasarkan Tabel 6 dapat diketahui bahwa algoritma pelatihan *Fletcher-Reeves Update (traincgf)*, *Polak-Ribière (traincgp)*, *Levenberg-Marquardt (trainlm)* dan *Resilient Backpropagation (trainrp)* serta *Scaled Conjugate Gradient (trainscg)* memiliki persentase kecocokan mengenali pola data sebesar 100%.

KESIMPULAN

Hasil pengujian yang telah dilakukan dapat disimpulkan bahwa algoritma pelatihan JST yang paling optimal tingkat ketepatan pengenalan pola data ditinjau dari rata-rata delta terkecil dan besarnya persentase kecocokan rata-rata delta algoritma pelatihan pada setiap *learning rate* (*lr*) pada model neuron 10-16-1 adalah *Levenberg-Marquardt (trainlm)* dengan nilai rata-rata delta = 0.00632000000 pada *learning rate* (*lr*) = 0.9 dengan persentase ketepatan mengenali pola data sebesar 100%. Hasil pengujian yang dihasilkan pada model 15-29-1 algoritma pelatihan JST yang paling optimal tingkat ketepatan pengenalan pola data ditinjau dari rata-rata delta terkecil dan besarnya persentase kecocokan rata-rata delta algoritma pelatihan pada setiap *learning rate* (*lr*) adalah *Levenberg-Marquardt (trainlm)* dengan nilai rata-rata delta = 0.00530500000 pada *learning rate* (*lr*) = 0.7 dengan persentase ketepatan mengenali pola data sebesar 100%. Berdasarkan hasil pengujian kedua model neuron maka diperoleh algoritma pelatihan JST *Levenberg-Marquardt (trainlm)* sebagai algoritma yang paling optimal dalam mengenali pola data. Dengan demikian setelah dilakukan penelitian ini, adapun saran untuk pengembangannya yaitu: Pada penelitian ini model neuron yang diuji adalah 10-16-1 dan 15-29-1 oleh sebab itu perlu dilakukan pengujian terhadap besarnya pengaruh penggunaan jumlah neuron pada lapisan tersembunyi yang berbeda untuk mendapatkan algoritma pelatihan JST yang optimal dalam ketepatan pengenalan pola data didasarkan atas rata-rata delta terkecil.

DAFTAR PUSTAKA

Heaton, J. (2008). *Introduction to Neural Network for C#* (2nd ed.). St. Louis: Heaton Research, Inc.

-
- Kusumadewi, S. (2004). *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB dan Excel Link*. Yogyakarta: Graha Ilmu.
- Mustafidah, H., Hartati, S., Wardoyo, R., & Harjoko, A. (2014). Selection of Most Appropriate Backpropagation Training Algorithm in Data Pattern Recognition. *Internasional Journal of Computer Trends and Technology (IJCTT)* ISSN: 2231-2803, 14(2), 92–95.
- Mustafidah, H., & Suwarsito. (2015a). Error Rate Testing of Training Algorithm in Back Propagation Network. *International Journal of Soft Computing and Engineering (IJSCE)* ISSN: 2231-2307, 5(4), 46–50.
- Mustafidah, H., & Suwarsito. (2015b). *Model Parameter Jaringan Syaraf Tiruan untuk Pemilihan Algoritma Pelatihan Jaringan Backpropagation yang Paling Optimal*. Purwokerto.
- Mustafidah, H., & Suwarsito. (2016). Testing Design of Neural Network Parameters in Optimization Training Algorithm. *International Conference of Research and Community Service (IRECOMS)*, 139–146.
- Santoso, S. (2017). *Menguasai Statistik dengan SPSS 24*. Jakarta: PT. Elex Media Komputindo.
- Shanmuganathan, S., & Samarasinghe, S. (2016). *Artificial Neural Network Modelling*. Switzerland: Springer International Publishing.
- Sutojo, T., Edy, M., & Vincent, S. (2011). *Kecerdasan Buatan*. Yogyakarta: Andi Offset.
- Suyanto. (2014). *Artificial Intelligence Searching - Reasoning - Planning - Learning* (2nd ed.). Bandung: Penerbit Informatika.
- Taniredja, T., & Mustafidah, H. (2011). *Penelitian Kuantitatif (Sebuah Pengantar)*. Bandung: Alfabeta.
- Tjolleng, A. (2017). *Pengantar Pemrograman MATLAB*. Jakarta: PT. Elex Media Komputindo.
- Wibowo, F., Sugiyanto, S., & Mustafidah, H. (2013). Tingkat Ketelitian Pengenalan Pola Data pada Algoritma Pelatihan Perbaikan Metode Batch Mode dalam Jaringan Syaraf Tiruan. *JUITA* ISSN: 2086-9398, II(4), 259–264.