

# Tingkat Keoptimalan Algoritma Pelatihan pada Jaringan Syaraf Tiruan (Studi Kasus Prediksi Prestasi Belajar Mahasiswa) *Optimization Level of Training Algorithms in Neural Network (Case Studies of Student Learning Achievement Predictions)*

HindayatiMustafidah<sup>1)</sup>, Dimara Kusuma Hakim<sup>2)</sup>, Sigit Sugiyanto<sup>3)</sup>

<sup>1)2)3)</sup>*Teknik Informatika – F. Teknik – Universitas Muhammadiyah Purwokerto*

*Jl. Raya Dukuwaluh Purwokerto 53182*

<sup>1)</sup> h.mustafidah@ump.ac.id

*Abstrak* - Algoritma pelatihan dalam jaringan syaraf tiruan telah diterapkan dalam permasalahan prediksi prestasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto didasarkan atas nilai-nilai dalam mata pelajaran yang diujikan dalam Ujian Nasional saat di SLTA. Algoritma pelatihan yang digunakan adalah *traingd*, namun algoritma ini belum pernah dibandingkan tingkat keoptimalannya dengan algoritma pelatihan yang lain khususnya dalam kasus ini. Oleh karena itu, dalam penelitian ini dilakukan perbandingan tingkat keoptimalan dari algoritma pelatihan *traingd*, *traingdm*, *learnngd*, dan *learnngdm* dalam contoh kasus prediksi prestasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto. Proses penerapan algoritma pelatihan dilakukan dengan pemrograman menggunakan bahasa pemrograman MATLAB. Sedangkan data hasil penerapan algoritma dilakukan uji perbedaan error yang dihasilkan (uji keoptimalan) menggunakan perangkat lunak SPSS 16.0. Berdasarkan hasil uji statistik dari keempat algoritma diperoleh rata-rata masing-masing algoritma yaitu *learnngd*: 0.0215, *learnngdm*: 0.0163, *traingd*: 0.0211, dan *traingdm*: 0.0267. Dengan taraf alpha 5%, diperoleh nilai signifikansi sebesar 0,632. Hasil uji disimpulkan bahwa keempat algoritma pelatihan dengan beberapa parameter jaringan yang meliputi variasi epoch maksimum dan *learning rate*, tidak memberikan pengaruh terhadap tingkat *error* yang dihasilkan secara signifikan. Hal ini berarti tidak ada perbedaan tingkat optimalisasi atau keempat algoritma pelatihan memiliki tingkat optimalisasi sama.

*Kata-kata kunci*- algoritma pelatihan, *error*, epoch, *learning rate*.

*Abstract* - The training algorithm in neural networks have been applied to the problem of student learning achievement predictions of Informatic Engineering Study Program of Muhammadiyah University of Purwokerto that is based on the values in the subjects to be tested in a national exam on high school. The training algorithm used is *traingd*, but this algorithm has never compared the level of its optimization with other training algorithms, especially in this case. Therefore, in this study was done comparing the level of training algorithm optimization those are *traingd*, *traingdm*, *learnngd*, and *learnngdm*, in example of student learning achievement predictions at Informatic Engineering of Muhammadiyah University of Purwokerto. The process of application of the training algorithm was done using MATLAB language programming. The results of data application of the algorithms was tested using SPSS 16.0 to get error difference. Based on the results of the statistical tests of the four algorithms obtained an average of each algorithm are *learnngd*: 0.0215, *learnngdm*: 0.0163, *traingd*: 0.0211, and *traingdm*: 0.0267. By alpha level 5%, obtained the value significance of 0.809. This test results was concluded that the fourth training algorithms with some of parameter of the network that includes the variations maximum epochs and learning rates, not give effect on the resulting error rate significantly. This means there is no difference in the level of optimization of the algorithms or fourth training algorithms has the same optimization level.

*Key words*- training algorithm, error, epoch, learning rate.

## I. PENDAHULUAN

Perkembangan teknologi ilmu komputer saat ini telah menciptakan beberapa teknik pendekatan dalam menyelesaikan suatu masalah yang disebut *soft computing* [1]. *Soft Computing* merupakan bagian dari sistem cerdas

yang merupakan suatu model pendekatan untuk melakukan komputasi dengan meniru akal manusia dan memiliki kemampuan untuk menalar dan belajar pada lingkungan yang penuh dengan ketidakpastian dan ketidaktepatan. Komponen utama pembentuk *soft computing* adalah sistem fuzzy (*fuzzy system*), jaringan syaraf (*neural network*), algoritma evolusioner (*evolutionary algorithm*), dan penalaran dengan probabilitas (*probabilistic reasoning*). Kinerja *soft computing* dalam menyelesaikan permasalahan di berbagai bidang kehidupan telah terbukti manfaatnya. Salah satu metode yang akan digunakan dalam penelitian ini adalah jaringan syaraf tiruan atau *Artificial Neural Network* (ANN) yang merupakan solusi ideal untuk permasalahan-permasalahan yang tidak dapat diformulasikan dengan mudah menggunakan algoritma [2]. Terdapat beberapa algoritma pelatihan yang terdapat dalam jaringan syaraf tiruan. Diantara algoritma-algoritma tersebut adalah yang tergolong dalam *incremental mode (adapt)* yang terdiri dari dua algoritma yaitu algoritma *learnngd* dan *learnngdm*, dan algoritma yang tergolong dalam *batch mode (train)* yaitu algoritma *traingd* dan *traingdm*[3]. Algoritma *learnngdm* merupakan algoritma dasar *gradient descent*. Bobot-bobot jaringan akan diperbaiki sesuai dengan perubahan yang terjadi. Parameter yang berhubungan dengan *learnngd* yaitu learning rate. Semakin besar nilai learning rate mengakibatkan semakin besarnya langkah pembelajaran. Jika learning rate diset terlalu besar, maka algoritma akan menjadi tidak stabil, dan sebaliknya algoritma akan konvergen dalam jangka waktu yang sangat lama. Pada algoritma *learnngdm*, tidak hanya merespon gradien lokal saja, tetapi juga mempertimbangkan kecenderungan yang baru saja terjadi pada permukaan error. Besarnya perubahan bobot ini dipengaruhi oleh suatu konstanta (yang dikenal dengan nama momentum), *mc*, yang bernilai antara 0 sampai 1. Sebagaimana pada algoritma *learnngd* dan *learnngdm*, algoritma *traingd* menggunakan algoritma dasar *gradient descent*, namun untuk pelatihan hanya menggunakan 1 fungsi pelatihan saja (tidak setiap bobot memiliki fungsi pelatihan sendiri-sendiri). Sedangkan algoritma *traingdm* mirip dengan *learnngdm*, hanya saja parameter yang perlu diset adalah fungsi pelatihan menjadi *traingdm*, lainnya sama dengan *traingd*, dan satu lagi parameter momentum. Ada 7 parameter yang harus diatur pada algoritma *traingd* dan *traingdm* yaitu: maksimum epoch, target nilai fungsi kinerja (*error*), laju pembelajaran, maksimum kegagalan, gradien minimum, jumlah epoch yang akan ditunjukkan kemajuannya, dan waktu maksimum yang diijinkan untuk melakukan pelatihan. Di antara

algoritma-algoritma pelatihan ini perlu dicari algoritma yang optimal dalam menyelesaikan masalah.

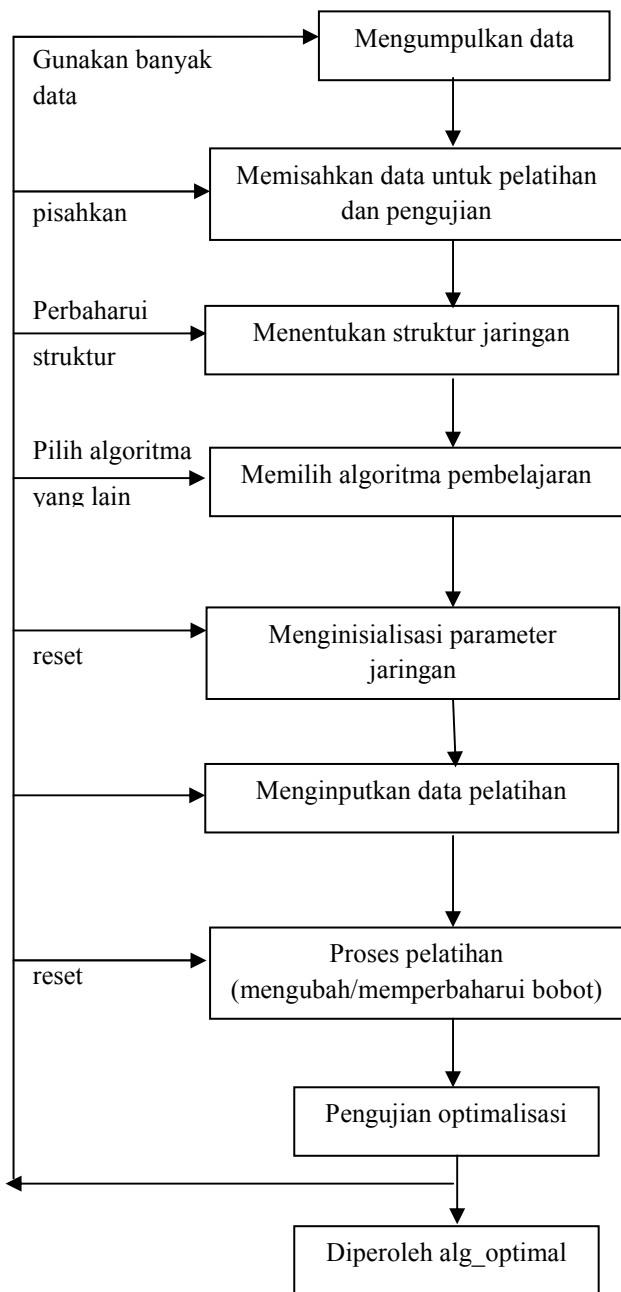
Penggunaan algoritma pelatihan dalam jaringan syaraf tiruan telah dilakukan dalam menyelesaikan permasalahan memprediksi prestasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto didasarkan atas nilai-nilai dalam mata pelajaran yang diujikan dalam Ujian Nasional saat di SLTA [4]. Algoritma pelatihan yang digunakan adalah *traingd* dan menghasilkan tingkat *error* sebesar 0.0664179 dari target *error* 0,05. Penggunaan algoritma ini belum pernah dibandingkan tingkat keoptimalannya (menghasilkan tingkat *error* terkecil) dengan algoritma pelatihan yang lain khususnya dalam kasus ini. Oleh karena itu, dalam penelitian ini akan dilakukan perbandingan tingkat keoptimalan dari algoritma pelatihan *traingd*, *traingdm*, *learnngd*, dan *learnngdm* dalam contoh kasus prediksi prestasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto.

Berdasarkan kajian permasalahan yang telah disampaikan, maka tujuan penelitian ini adalah menerapkan algoritma pelatihan *learnngd*, *learnngdm*, *traingd*, dan *traingdm* dalam contoh kasus prediksi prestasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto untuk mendapatkan tingkat *error* dan menentukan algoritma pelatihan yang paling optimal (yang memberikan *error* terkecil) dari keempat algoritma pelatihan *traingd*, *traingdm*, *learnngd*, dan *learnngdm* dalam contoh kasus prediksi prestasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto.

## II. METODE PENELITIAN

Data yang digunakan dalam penelitian adalah data sekunder yang diambil dari data penelitian [5] berupa data nilai TPA (Tes Potensi Akademik), nilai rata-rata UN (Ujian Nasional) SLTA, dan skor motivasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto.

Penelitian ini merupakan penelitian pengembangan dan penelitian kuantitatif inferensia yaitu membangun program komputer menggunakan beberapa algoritma pelatihan dan kemudian mencari algoritma pelatihan yang paling optimal. Bahasa pemrograman yang digunakan adalah Matlab 7.0. Adapun langkah operasional yang dilakukan dalam penelitian ini adalah sebagai berikut (Gambar 1):

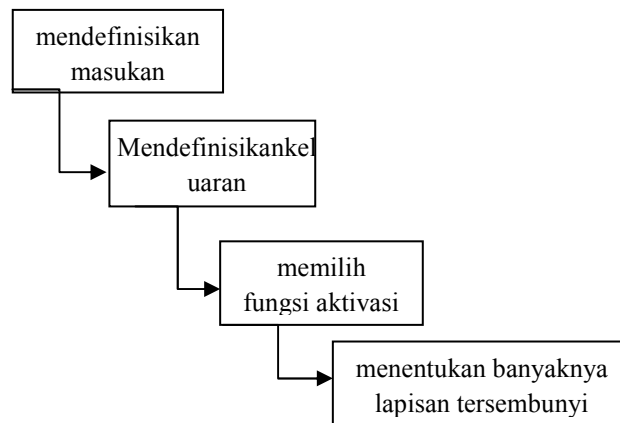


Gambar 1. Tahap Penerapan ANN

**Keterangan:**

- a. Mengumpulkan data  
Data dalam penelitian ini berupa data nilai TPA, nilai rata-rata UN SLTA, dan skor motivasi belajar mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Purwokerto.
- b. Memisahkan data untuk pelatihan dan pengujian  
Data yang terkumpul digunakan sebagai data pelatihan terhadap algoritma-algoritma pelatihan yang akan digunakan. Selanjutnya data ini pula yang digunakan sebagai data pengujian.
- c. Menentukan struktur jaringan dengan tahapan seperti

pada Gambar 2.



Gambar 2. Langkah Membangun Struktur Jaringan Syaraf

- d. Membangun struktur jaringan.
- e. Memilih algoritma pelatihan  
Pada tahap ini digunakan empat algoritma pelatihan yaitu *learnrd*, *learnrdm*, *traingd*, dan *traingdm*.
- f. Menginisialisasi parameter jaringan.  
Parameter jaringan yang digunakan meliputi *learning rate*, jumlah epoch, dan tingkat *error*.
- g. Menginputkan data pelatihan.
- h. Proses pelatihan (mengubah/memperbaharui bobot)
- i. Pengujian

Pengujian dilakukan untuk menentukan optimalisasi dari algoritma-algoritma pelatihan tersebut dengan membandingkan tingkat *error* yang terjadi menggunakan analisis variansi dari keempat algoritma.

Langkah untuk melakukan analisis variansi adalah:

- 1) Uji homogenitas.  
Hipotesis yang digunakan adalah:  
 $H_0$ : data dari keempat algoritma pelatihan homogen  
 $H_1$ : data dari keempat algoritma pelatihan tidak homogen  
 Penolakan  $H_0$  dilakukan jika nilai signifikansi yang diperoleh  $< \alpha$  (dalam hal ini  $\alpha = 0.05$  atau 5%).
- 2) Uji beda rata-rata  
Hipotesis yang digunakan adalah:  
 $H_0$ : varian dari keempat algoritma pelatihan sama  
 $H_1$ : varian dari keempat algoritma pelatihan tidak sama.  
 Penolakan  $H_0$  dilakukan jika nilai signifikansi yang diperoleh  $< \alpha$  (dalam hal ini  $\alpha = 0.05$  atau 5%).

### III. HASIL DAN PEMBAHASAN

#### A. Pengumpulan Data

Data pelatihan yang digunakan adalah data sekunder

berupa data nilai TPA, nilai rata-rata UN SLTA, dan skor motivasi belajar dan IPK mahasiswa yang kemudian disusun kaidah (*rule*) yang berjumlah 100 kaidah seperti pada Tabel 1 berikut:

TABEL I  
DATA PELATIHAN

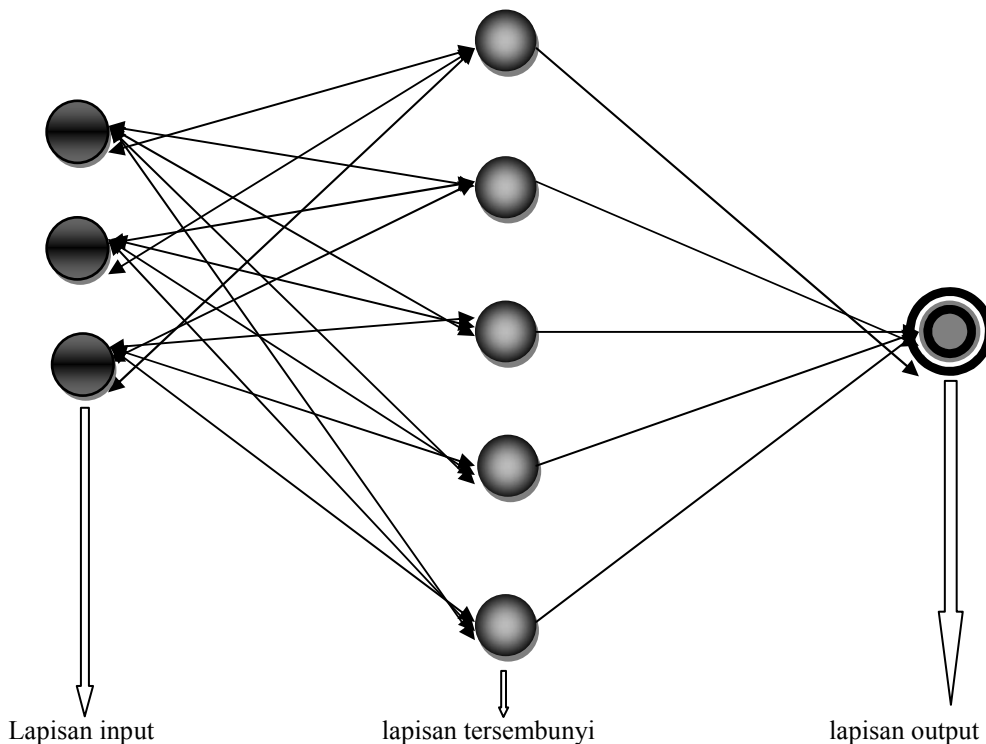
no.	tpa	rata ebtanas	motivasi	IPK	no.	tpa	rata ebtanas	motivasi	IPK
1	0	0	0	0	51	0.5	0.5	0.67	0.5
2	0	0	0.33	0	52	0.5	0.5	1	0.75
3	0	0	0.67	0.25	53	0.5	0.75	0	0.5
4	0	0	1	0.25	54	0.5	0.75	0.33	0.5
5	0	0.25	0	0	55	0.5	0.75	0.67	0.75
6	0	0.25	0.33	0	56	0.5	0.75	1	0.75
7	0	0.25	0.67	0.25	57	0.5	1	0	0.5
8	0	0.25	1	0.5	58	0.5	1	0.33	0.75
9	0	0.5	0	0	59	0.5	1	0.67	0.75
10	0	0.5	0.33	0.25	60	0.5	1	1	1
11	0	0.5	0.67	0.25	61	0.75	0	0	0.25
12	0	0.5	1	0.5	62	0.75	0	0.33	0.25
13	0	0.75	0	0.25	63	0.75	0	0.67	0.5
14	0	0.75	0.33	0.25	64	0.75	0	1	0.5
15	0	0.75	0.67	0.5	65	0.75	0.25	0	0.25
16	0	0.75	1	0.5	66	0.75	0.25	0.33	0.5
17	0	1	0	0.25	67	0.75	0.25	0.67	0.5
18	0	1	0.33	0.5	68	0.75	0.25	1	0.75
19	0	1	0.67	0.5	69	0.75	0.5	0	0.5
20	0	1	1	0.75	70	0.75	0.5	0.33	0.5
21	0.25	0	0	0	71	0.75	0.5	0.67	0.75
22	0.25	0	0.33	0	72	0.75	0.5	1	0.75
23	0.25	0	0.67	0.25	73	0.75	0.75	0	0.5
24	0.25	0	1	0.5	74	0.75	0.75	0.33	0.75
25	0.25	0.25	0	0	75	0.75	0.75	0.67	0.75
26	0.25	0.25	0.33	0.25	76	0.75	0.75	1	1
27	0.25	0.25	0.67	0.25	77	0.75	1	0	0.5
28	0.25	0.25	1	0.5	78	0.75	1	0.33	0.75
29	0.25	0.5	0	0.25	79	0.75	1	0.67	1
30	0.25	0.5	0.33	0.25	80	0.75	1	1	1
31	0.25	0.5	0.67	0.5	81	1	0	0	0.25
32	0.25	0.5	1	0.5	82	1	0	0.33	0.5
33	0.25	0.75	0	0.25	83	1	0	0.67	0.5
34	0.25	0.75	0.33	0.5	84	1	0	1	0.75
35	0.25	0.75	0.67	0.5	85	1	0.25	0	0.5
36	0.25	0.75	1	0.75	86	1	0.25	0.33	0.5
37	0.25	1	0	0.5	87	1	0.25	0.67	0.75
38	0.25	1	0.33	0.5	88	1	0.25	1	0.75
39	0.25	1	0.67	0.75	89	1	0.5	0	0.5
40	0.25	1	1	0.75	90	1	0.5	0.33	0.75
41	0.5	0	0	0	91	1	0.5	0.67	0.75
42	0.5	0	0.33	0.25	92	1	0.5	1	1

no.	tpa	rata ebtanas	motivasi	IPK
43	0.5	0	0.67	0.25
44	0.5	0	1	0.5
45	0.5	0.25	0	0.25
46	0.5	0.25	0.33	0.25
47	0.5	0.25	0.67	0.5
48	0.5	0.25	1	0.5
49	0.5	0.5	0	0.25
50	0.5	0.5	0.33	0.5

no.	tpa	rata ebtanas	motivasi	IPK
93	1	0.75	0	0.5
94	1	0.75	0.33	0.75
95	1	0.75	0.67	1
96	1	0.75	1	1
97	1	1	0	0.75
98	1	1	0.33	0.75
99	1	1	0.67	1
100	1	1	1	1

### B. Struktur Jaringan

Struktur jaringan yang dibangun adalah seperti pada Gambar 3, dimana sesuai dengan arsitektur yang disampaikan oleh [6] seperti berikut:



Ket.:Lapisan input terdiri dari 3 neuron (yaitu nilai TPA, nilai rata-rata UN SLTA, dan skor motivasi belajar mahasiswa), sedangkan lapisan output terdiri dari 1 neuron yaitu IPK mahasiswa).

Gambar 3. Struktur Jaringan Syaraf Tiruan dengan 3 lapisan

### C. Algoritma Pelatihan

Algoritma pelatihan yang digunakan yaitu *learnngd*, *learnngdm*, *traingd*, dan *traingdm*.

### D. Inisialisasi Parameter Jaringan

Parameter jaringan yang digunakan meliputi *learning rate*(lr), jumlah epoch, dan tingkat *error*. *Learning rate* yang digunakan meliputi nilai-nilai 0.01, 0.02, 0.05, 0.1, 0.2, 0.25, dan 0.5. Jumlah epoch digunakan dalam berbagai variasi yaitu 100, 500, 1000, dan 5000.

Sedangkan target error yang digunakan adalah 0.05.

### E. Proses Pelatihan

Proses pelatihan dilakukan terhadap data pelatihan untuk keempat algoritma. Kode sumber untuk proses pelatihan ini dibangun menggunakan bahasa pemrograman MATLAB.

### F. Pengujian

Pengujian dilakukan untuk menentukan optimalisasi dari algoritma-algoritma pelatihan tersebut dengan membandingkan tingkat *error* yang terjadi. Hasil *error* yang diperoleh dari proses pengujian tersaji pada Tabel 2 berikut :

TABEL II  
MEAN SQUARE ERROR (MSE) HASIL PENGUJIAN

epoh	lr	mse learnngd	mse learnngdm	mse traingnd	mse traingndm
100	0.01	0.1392	0.0963	0.0511	0.2350
100	0.02	0.0537	0.0214	0.1078	0.0969
100	0.05	0.0155	0.0126	0.0407	0.0344
100	0.1	0.0224	0.0257	0.0220	0.0147
100	0.2	0.0132	0.0196	0.0167	0.0143
100	0.25	0.0093	0.0188	0.0173	0.0239
100	0.5	0.0105	0.0070	0.0113	0.0185
500	0.01	0.0576	0.0417	0.0443	0.0303
500	0.02	0.0235	0.0165	0.0277	0.0109
500	0.05	0.0122	0.0128	0.0137	0.0125
500	0.1	0.0128	0.0115	0.0086	0.0096
500	0.2	0.0099	0.0093	0.0101	0.0090
500	0.25	0.0116	0.0090	0.0100	0.0114
500	0.5	0.0096	0.0115	0.0095	0.0197
1000	0.01	0.0250	0.0185	0.0250	0.0274
1000	0.02	0.0205	0.0097	0.0205	0.0263
1000	0.05	0.0115	0.0087	0.0115	0.0182
1000	0.1	0.0092	0.0188	0.0092	0.0145
1000	0.2	0.0092	0.0090	0.0092	0.0109
1000	0.25	0.0108	0.0086	0.0108	0.0107
1000	0.5	-	0.0097	-	0.0297

5000	0.01	0.0162	0.0093	0.0162	0.0210
5000	0.02	0.0232	0.0177	0.0232	0.0081
5000	0.05	0.0121	0.0080	0.0121	0.0091
5000	0.1	0.0069	0.0064	0.0069	0.0098
5000	0.2	0.0055	0.0064	0.0055	0.0057
5000	0.25	0.0079	0.0071	0.0079	0.0091
5000	0.5	-	0.0046	-	0.0052

Pengujian homogenitas data dan pengujian beda rata-rata dilakukan menggunakan perangkat lunak SPSS 16.0. Berdasarkan data tersebut dilakukan uji homogenitas data dan diperoleh hasil seperti pada Tabel 3 berikut:

TABEL III  
TEST OF HOMOGENEITY OF VARIANCES

mse	Levene Statistic	df1	df2	Sig.
	1.019	3	104	.387

Berdasarkan hasil signifikansi yang diperoleh ternyata lebih besar dari  $\alpha$  ( $0.387 > 0.05$ ) berarti  $H_0$  diterima sehingga dapat disimpulkan bahwa data bersifat homogen.

Selanjutnya dilakukan analisis variansi yaitu uji beda rata-rata setiap algoritma dan menghasilkan informasi seperti pada Tabel 4 dan Tabel 5 berikut:

TABEL IV  
DESKRIPSI DATA JARINGAN

mse	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
					Lower Bound	Upper Bound		
					learnngd	26		
learnngdm	28	.016293	.0174732	.0033021	.009517	.023068	.0046	.0963
traingnd	26	.021107	.0212911	.0041755	.012507	.029707	.0055	.1078
traingndm	28	.026673	.0443144	.0083746	.009490	.043857	.0052	.2350
Total	108	.021397	.0294057	.0028296	.015787	.027006	.0046	.2350

TABEL 5. HASIL UJI ANOVA

mse	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	.002	3	.001	.576	.632
Within Groups	.091	104	.001		
Total	.093	107			

Hasil ANOVA menunjukkan bahwa signifikansi (0.632)  $> \alpha$  (0.05) sehingga dapat dikatakan bahwa keempat algoritma memiliki variansi *error* yang sama (tidak berbeda). Hal ini juga ditunjukkan dari hasil uji Post hoc menggunakan metode LSD dan Duncan seperti pada Tabel 6berikut:

TABEL VI  
HASIL POST HOC TESTS MULTIPLE COMPARISONS

Dependent Variable:mse							
	(I) algoritma	(J) algoritma	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
						Lower Bound	Upper Bound
LSD	learnigd	learnigdm	.0052071	.0080568	.520	-.010770	.021184
		traingd	.0003931	.0082046	.962	-.015877	.016663
		tringdm	-.0051733	.0080568	.522	-.021150	.010804
	learnigdm	learnigd	-.0052071	.0080568	.520	-.021184	.010770
		traingd	-.0048140	.0080568	.551	-.020791	.011163
		tringdm	-.0103805	.0079062	.192	-.026059	.005298
	traingd	learnigd	-.0003931	.0082046	.962	-.016663	.015877
		learnigdm	.0048140	.0080568	.551	-.011163	.020791
		tringdm	-.0055665	.0080568	.491	-.021543	.010410
tringdm	learnigd	.0051733	.0080568	.522	-.010804	.021150	
	learnigdm	.0103805	.0079062	.192	-.005298	.026059	
	traingd	.0055665	.0080568	.491	-.010410	.021543	

Kedua metode uji LSD dan Duncan masing-masing menunjukkan hal yang sama yaitu keempat algoritma memiliki tingkat *error* yang sama. Pada uji LSD, ditunjukkan bahwa keempat algoritma pada interval konfidensi 95% semua signifikansi di atas 0.05, dan perbedaan rata-rata berada pada daerah negative untuk batas bawah dan positif untuk atas sehingga dikatakan tidak ada yang lebih kecil atau lebih besar.

#### IV. PENUTUP

##### A. Simpulan

Simpulan dari penelitian ini adalah dari keempat algoritma pelatihan pada jaringan syaraf tiruan yaitu *learnigd*, *learnigdm*, *traingd*, dan *traingdm* dengan beberapa parameter jaringan yang meliputi variasi epoch maksimum (100, 500, 1000, dan 5000), variasi *learning rate* (0.01, 0.02, 0.05, 0.1, 0.2, 0.25, dan 0.5) tidak memberikan pengaruh terhadap tingkat *error* yang

dihasilkan dalam contoh kasus prediksi prestasi belajar mahasiswa Teknik Informatika Universitas Muhammadiyah Purwokerto. Hal ini berarti tidak ada perbedaan tingkat optimalisasi dari keempat algoritma pelatihan, artinya keempat algoritma pelatihan tersebut memiliki tingkat optimalisasi yang sama.

##### B. Saran

Selain empat algoritma pelatihan yang telah disebutkan di atas, masih terdapat beberapa algoritma pelatihan yang lain yang perlu diuji keoptimalannya. Algoritma tersebut di antaranya adalah *traingda*, *traingdx*, *trainrp*, dan lainnya. Oleh karena itu disarankan untuk dilakukan pengujian tingkat keoptimalan dari algoritma-algoritma tersebut baik dalam kasus prediksi prestasi mahasiswa ataupun dalam kasus yang lain..

## UCAPAN TERIMA KASIH

1. Rektor Universitas Muhammadiyah Purwokerto yang telah memberikan dana dalam pelaksanaan penelitian ini.
2. Ketua LPPM yang telah memberi dorongan dan persetujuannya, sehingga penelitian ini berjalan dengan baik.
3. Dekan Fakultas Teknik yang telah memberi kesempatan dan fasilitas dalam penelitian ini.

## DAFTAR PUSTAKA

- [1] Kusumadewi, S. dan Hartati, S. 2006. *Neuro-Fuzzy : Integrasi Sistem Fuzzy dan Jaringan Syaraf*. Graha Ilmu. Yogyakarta.
- [2] Jones, M.T. 2008. *Artificial Intelligence A Systems Approach*. Infinity Science Press LLC. New Delhi.
- [3] Kusumadewi, S. 2004. *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB & EXCEL LINK*. Graha Ilmu. Yogyakarta
- [4] Harjono dan Aryanto, D., 2009. Application of Artificial Neural Networks to Predict Student Achievement Study. *SAINTEK ISSN1411-2558, Vol. 5 No. 2*.
- [5] Mustafidah, H. dan Aryanto, D., 2012. Sistem Inferensi Fuzzy untuk Memprediksi Prestasi Belajar Mahasiswa Berdasarkan Nilai Ujian Nasional, Tes Potensi Akademik, dan Motivasi Belajar. *Laporan Penelitian*. UMP. Purwokerto.
- [6] Kartalopoulos, S.V. 2003. *Understanding Neural Networks and Fuzzy Logic : Basic Concepts and Applications*. Third Printing. Prentice-Hall of India. New Delhi.