

---

# JARINGAN SYARAF TIRUAN MENGGUNAKAN ALGORITMA BACKPROPAGATION UNTUK MEMPREDIKSI NILAI UJIAN AKHIR SEKOLAH (STUDI KASUS DI MAN 2 PADANG)

**Febri Aldi**

Universitas Putra Indonesia YPTK Padang

e-mail : [febri\\_aldi@ymail.com](mailto:febri_aldi@ymail.com)

---

## Abstrak

Munculnya kebijakan tentang penghapusan Ujian Nasional (UN) sebagai penentu kelulusan peserta didik membawa konsekuensi pada meningkatnya kedudukan Ujian Sekolah (US) pada jenjang pendidikan sekolah dasar dan menengah. Hasil Ujian Sekolah juga digunakan sebagai salah satu pertimbangan untuk penentuan kelulusan peserta didik dari suatu satuan pendidikan dan perbaikan proses pembelajaran dalam upaya peningkatan mutu pendidikan. Oleh sebab itu, sebagai salah satu persiapan dalam menghadapinya, penulis membuat sebuah sistem Jaringan Saraf Tiruan menggunakan metode Backpropagation untuk memprediksi nilai Ujian Sekolah siswa sekolah menengah atas. Penelitian ini berbentuk studi kasus, berlokasi di MAN 2 Padang. Variabel penelitian berupa nilai rapor mata pelajaran Matematika, Fisika, Kimia, Biologi, Bahasa Indonesia, Bahasa Inggris, PKn, dan Sejarah. Hasil penelitian menunjukkan bahwa data didapatkan epoch (pengulangan) lebih sedikit yang digunakan untuk memperoleh error minimum, yaitu 5967 dan Mean Square Error (MSE) 0.00999869 sesuai dengan target yang diinginkan. Jaringan Saraf Tiruan backpropagation yang dihasilkan cukup dapat diandalkan untuk melakukan prediksi nilai ujian sekolah siswa sekolah menengah atas.

**Kata kunci:** Jaringan Saraf Tiruan, Backpropagation, Prediksi, Nilai Ujian Sekolah, MSE

---

## 1. PENDAHULUAN

Jaringan Syaraf Tiruan (*artificial neural network*) adalah sistem komputasi yang arsitektur dan operasinya diilhami dari pengetahuan tentang sel saraf biologis di dalam otak. Jaringan Syaraf Tiruan merupakan sebuah paradigma komputasi yang mempekerjakan model sederhana dari sistem *neuron* biologis, meskipun longgar didasarkan pada karakteristik beragam otak manusia. Jaringan Syaraf Tiruan dapat digambarkan sebagai model matematis dan komputasi untuk fungsi aproksimasi non-linear, klasifikasi data *cluster* dan regresi non-parametrik atau sebuah simulasi dari koleksi model Jaringan Saraf Biologi (Febrianto dan Hindayati, 2013). Konsep praktis dari JST untuk memprediksi nilai ujian sekolah dengan cara memasukkan nilai rapor mata pelajaran tertentu sebagai variabel inputan yang dimasukkan ke dalam sistem dilakukan proses pelatihan menggunakan JST dan algoritma pembelajaran *Backpropagation*.

## 2. LANDASAN TEORI

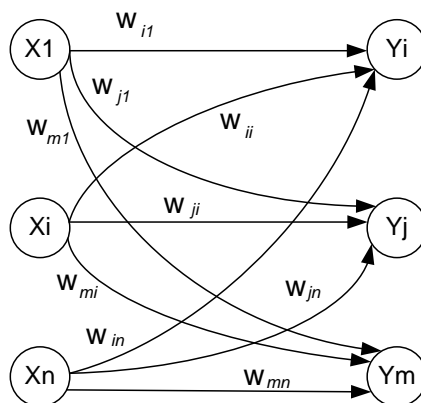
Pada JST, *neuron-neuron* akan dikumpulkan dalam lapisan-lapisan (*layer*) yang disebut dengan lapisan *neuron* (*neuron layers*). *Neuron-neuron* pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya. Informasi yang diberikan pada Jaringan Syaraf akan dirambatkan lapisan ke lapisan, mulai dari lapisan masukan sampai ke lapisan keluaran melalui lapisan tersembunyi (*hidden layer*) (Wuryandari dan Afrianto, 2012).

Arsitektur jaringan menggambarkan susunan jaringan saraf. Memilih arsitektur dari jaringan saraf tiruan adalah keputusan yang sulit, itu karena arsitektur jaringan yang tersedia sangat banyak; masing-masing dengan keuntungan dan kelemahannya. Misalnya, beberapa jaringan memiliki kelebihan dalam hal kecepatan untuk akurasi, sementara beberapa mampu menangani variabel statis dan yang tidak terus-menerus (Sayeki, 2013).

Faktor terpenting dalam menentukan kelakuan suatu *neuron* adalah fungsi aktivasi dan pola bobotnya. Umumnya *neuron-neuron* yang terletak pada lapisan yang sama akan memiliki keadaan yang sama sehingga pada setiap lapisan yang sama *neuron-neuron* memiliki fungsi aktivasi yang sama. Bila *neuron-neuron* pada suatu lapisan (misal lapisan tersembunyi) akan dihubungkan dengan *neuron-neuron* pada lapisan lain (misal lapisan keluaran) maka setiap *neuron* pada lapisan tersebut (lapisan tersembunyi) juga harus dihubungkan dengan setiap *neuron* pada lapisan lainnya (lapisan keluaran). Terdapat 3 macam arsitektur JST, yaitu (Wuryandari dan Afrianto, 2012):

1. Jaringan Layar Tunggal (*singel layer network*)

Dalam jaringan ini, sekumpulan input *neuron* dihubungkan langsung dengan sekumpulan outputnya. Dalam beberapa model hanya terdapat sebuah unit *neuron output*.



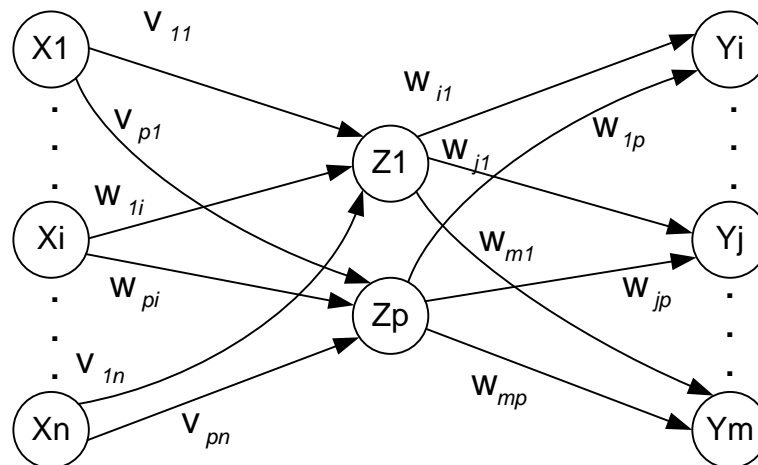
Gambar 1 : Jaringan Layar Tunggal

Gambar 1 menunjukkan arsitektur jaringan dengan  $n$  unit input ( $x_1, x_2, \dots, x_n$ ) dan  $m$  buah unit output ( $y_1, y_2, \dots, y_m$ )

Perhatikan dalam Jaringan Syaraf ini, semua unit input dihubungkan semua unit output, meskipun dengan bobot yang berbeda-beda. Tidak ada unit input yang dihubungkan dengan unit input lainnya. Demikian pula dengan unit outputnya. Besaran  $W_{ji}$  menyatakan bobot hubungan antara unit ke- $i$  dalam input dengan unit ke- $j$  dalam output. Bobot-bobot ini saling independent. Selama proses pelatihan, bobot-bobot tersebut akan dimodifikasi untuk meningkatkan hasil. Model seperti ini tepat digunakan untuk mengenal pola karena kesederhanaannya.

2. Jaringan Layar Jamak (*multi layer network*)

Jaringan *layer* jamak merupakan perluasan dari *layer* tunggal. Dalam jaringan ini selain unit input dan output, ada unit-unit lain (sering disebut *layer* tersembunyi). Sama seperti pada unit input dan output, unit-unit dalam satu *layer* tidak saling berhubungan.



**Gambar 2** : Jaringan Layer Jamak

Pada gambar 2 jaringan dengan  $n$  buah unit input ( $x_1, x_2, \dots, x_n$ ), disebut *layer* tersembunyi yang terdiri dari  $P$  buah unit ( $Z_1, \dots, Z_p$ ), dan  $m$  buah unit outputnya ( $y_1, y_2, \dots, y_m$ ). Jaringan *layer* jamak dapat menyelesaikan masalah yang lebih kompleks dibanding dengan *layer* tunggal, meskipun kadang kala proses pelatihan lebih kompleks dan lama.

3. Jaringan *reccurent*

Model jaringan *reccurent* mirip dengan jaringan *layer* tunggal maupun ganda. Hanya saja ada *neuron* output yang memberikan sinyal pada unit (sering disebut *feedback loop*).

**2.1. Algoritma Perambatan Balik (*BackPropagation*)**

*Backpropagation* adalah algoritma pembelajaran yang terwarisi dan biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan *neuron-neuron* yang ada pada lapisan tersembunyinya. Metode ini merupakan metode yang sangat baik dalam menangani masalah pengenalan pola-pola kompleks (Puji, 2010).

Perambatan galat mundur (*Backpropagation*) adalah sebuah metode sistematis untuk pelatihan *multilayer* Jaringan Syaraf Tiruan. Jaringan *backpropagation* merupakan salah satu algoritma yang sering digunakan dalam menyelesaikan masalah-masalah yang rumit. Algoritma ini memiliki dasar matematis yang kuat dan dilatih dengan menggunakan metode belajar terbimbing. Pada jaringan diberikan sepasang pola yang terdiri atas pola masukan dan pola yang diinginkan. Ketika suatu pola diberikan kepada jaringan, bobot-bobot diubah untuk memperkecil perbedaan pola keluaran dan pola yang diinginkan. Latihan ini dilakukan berulang-ulang sehingga semua pola yang dikeluarkan jaringan dapat memenuhi pola yang diinginkan (Badrul, 2011).

Salah satu model dari JST yang menggunakan *supervised learning* adalah *backpropagation*. *Backpropagation* melakukan pelatihan pada jaringan untuk mendapatkan keseimbangan untuk mengenali pola yang digunakan sebagai pembelajaran dan mengeluarkan hasil keluaran yang sesuai dengan pola masukan yang digunakan. Metode ini memiliki 3 fase dalam pelatihan *backpropagation*, yaitu *feed forward*, *backpropagation*, dan fase untuk memodifikasi bobot. Ketiga fase tersebut akan tetap dijalankan terus, hingga kondisi penghentian terpenuhi (Winata, dkk, 2010).

## 2.2. Langkah-langkah *Backpropagation*

Adapun langkah-langkah dalam pelatihan metode *Backpropagation* ini, memiliki 3 fase, yaitu *feed forward*, *backpropagation*, dan fase untuk memodifikasi bobot. Ketiga fase tersebut akan tetap dijalankan terus, hingga kondisi penghentian terpenuhi. Secara detail langkah-langkah pelatihan *Backpropagation* adalah sebagai berikut (Winata, dkk, 2010) :

- Tahap 0 : Hal pertama yang harus dilakukan adalah melakukan inialisasi bobot, dimana mengambil bobot awal dengan nilai random yang cukup .
1. Menetapkan nilai maksimum epoch, *target error*, dan *learning rate* ( )
  2. Inialisasi : Epoch = 0, MSE = 1.
- Tahap 1 : Melakukan pengerjaan langkah-langkah selanjutnya dengan kondisi selama (epoch < maksimum epoch) dan (MSE > target error), dimana nilai epoch akan terus bertambah sesuai iterasi (epoch = epoch + 1).
- Tahap 2 : Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan langkah-langkah berikutnya.

### Perambatan Maju (*feed forward*)

Tahap 3 : tiap unit masukan ( $x_i, i = 1, \dots, n$ ) menerima sinyal  $x_i$  dan menghantarkan sinyal ini ke semua unit lapisan di atasnya (unit tersembunyi)

Tahap 4 : setiap unit tersembunyi ( $x_i, i = 1, \dots, p$ ) jumlahkan bobot sinyal masukannya.

$$Z_{inj} = V_{oj} + \sum_{i=1}^p z_i v_{ij} \quad (1)$$

Dimana  $V_{oj}$  = bias pada unit tersembunyi  $j$  aplikasi fungsi aktivasinya untuk menghitung sinyal keluarannya,  $Z_j = f(z_{inj})$ , dan kirimkan sinyal ini keseluruh unit pada lapisan di atasnya ( unit keluaran).

Tahap 5 : tiap unit keluaran ( $y_k, k = 1, \dots, m$ ) jumlahkan bobot sinyal masukannya.

$$Z_{ink} = V_{ok} + \sum_{j=1}^n z_j w_{jk} \quad (2)$$

Dimana  $W_{jk}$  = bias pada unit keluaran  $k$  aplikasi fungsi aktifasinya untuk menghitung sinyal keluarannya,  $y_k = f(y_{ink})$ .

### Perambatan mundur (*backpropagation*)

Tahap 6 : tiap unit keluaran ( $y_k, k = 1, \dots, m$ ) menerima pola target yang saling berhubungan pada masukan pola pelatihan, hitung kesalahan informasinya.

$$\delta_k = (t_k - y_k) f^1(y_{netk}) - (t_k - y_k) y_k (1 - y_k) \quad (3)$$

Hitung koreksi bobotnya (digunakan untuk memperbaharui  $w_{jk}$  nantinya)

$$\Delta w_{kj} = \alpha \delta_k z_j ; k = 1, 2, \dots, m ; j = 0, 1, \dots, p \quad (4)$$

Hitung koreksi biasnya (digunakan untuk memperbaharui  $w_{ok}$  nantinya, dan kirimkan  $\delta_k$  ke unit-unit pada lapisan di bawahnya.

Tahap 7 : setiap unit lapisan tersembunyi ( $z_j, j = 1, \dots, p$ ) jumlahkan hasil perubahan masukannya (dari unit-unit lapisan di atasnya).

$$\Delta_{netj} = \sum \delta_k w_{kj} \quad (5)$$

Kalikan dengan turunan fungsi aktivasinya untuk menghitung informasi kesalahannya,

$$\delta_j = \delta_{netj} f^1(z_{netj}) = \delta_{netj} z_j (1 - z_j) \quad (6)$$

Hitung koreksi bobotnya (digunakan untuk memperbaharui  $v_{oj}$  nanti).

Tahap 8 : tiap unit keluaran ( $y_k, k = 1, \dots, m$ ) update bias dan bobotnya ( $j = 0, \dots, p$ ) :

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \tag{7}$$

Tiap unit lapisan tersembunyi ( $z_j, j = 1, \dots, p$ ) update bias dan bobotnya. ( $I = 0, \dots, n$ ) :

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \tag{8}$$

Tahap 9 : Hitung MSE pada tiap epoch.

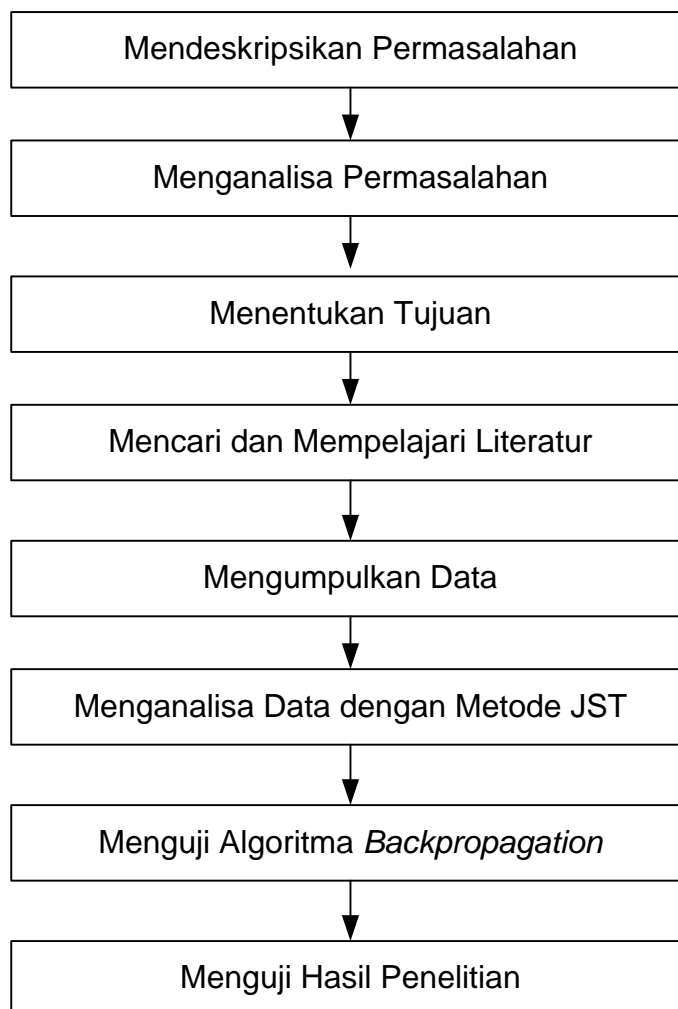
$$MSE = \frac{1}{n_{Poia}} \sum_k^{n_{Poia}} (t_k - y_{out})^2 \tag{9}$$

### 3. METODOLOGI PENELITIAN

Tujuan dari penelitian ini adalah untuk membuat model dalam memprediksi nilai ujian akhir sekolah menggunakan Jaringan Syaraf Tiruan dengan metode *Backpropagation* sebagai pedoman dalam pengambilan keputusan yang lebih baik dengan *output* berupa hasil dari prediksi nilai ujian akhir sekolah.

Pada bagian ini penulis mulai dengan menggambarkan studi kasus Jaringan Syaraf Tiruan metode *Backpropagation* pada sistem dalam memprediksi nilai ujian akhir sekolah pada MAN 2 Padang.

#### Kerangka Kerja Penelitian



Gambar 3 : Kerangka Kerja Penelitian

#### 4. ANALISA DAN PROSES

Data akan dapat dikenali dan sekaligus dapat diproses menggunakan perangkat lunak jika berbentuk matrik dan numerik. Kedelapan data mata pelajaran tersebut diubah kebentuk numerik dengan simbol:  $X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8$  dengan:  $X_1$ = Nilai Matematika,  $X_2$ = Nilai Fisika,  $X_3$ = Nilai Kimia,  $X_4$ = Nilai Biologi,  $X_5$ = Nilai Bahasa Indonesia,  $X_6$ = Nilai Bahasa Inggris,  $X_7$ = Nilai PKn,  $X_8$ = Nilai Sejarah

Dalam pelatihan ataupun pembentukan Jaringan Syaraf Tiruan yang perlu dilakukan pertama kali adalah inisianlisasi bobot awal. Di mana bobot awal ini akan menghubungkan simpul-simpul pada lapisan input dan juga lapisan tersembunyi (*hidden layer*). Bobot awal pada algoritma ini adalah  $V = (V_{11}, V_{21}, V_{12}, V_{22}, V_{13}, V_{23}, V_{14}, V_{24}, V_{15}, V_{25}, V_{16}, V_{26}, V_{17}, V_{27}, V_{18}, V_{28})$ , sedangkan bobot biasanya dipilih secara acak pada simpul-simpul lapisan tersembunyi (*hidden layer*) dan lapisan output ( $Z_1$  dan  $Z_2$ ) dipilih secara acak.

Tahap pelatihan :

- a. Langkah 0: mula-mula bobot diberi nilai acak yang kecil, dari *input layer* ke *hidden layer*.

**Tabel 1 :** Nilai Bobot Dari *Input Layer* ke *Hidden Layer*

	Z1	Z2
X1	-0.5278	1.7366
X2	0.1632	-1.4915
X3	1.1167	1.6959
X4	-2.9605	0.8638
X5	-0.7819	1.6642
X6	0.0173	0.9981
X7	-0.4613	-0.9800
X8	-1.9127	-1.4608
1	4.2612	-0.0308

Berikan nilai bobot ( $W$ ) dari *hidden layer* ke *output layer* dengan nilai acak.

**Tabel 2 :** Nilai Bobot dari *Hidden Layer* ke *Output Layer*

	Y
Z1	0.8709
Z2	0.8338
1	-0.1795

- b. Langkah 1 : Jika kondisi penghentian belum terpenuhi, lakukan langkah 2-8
- c. Langkah 2 : Untuk setiap pasang data pelatihan, lakukan langkah 3-8

Tahap 1 : Umpan Maju (*free forward*)

- d. Langkah 3 : Setiap unit input mengirim sinyal ke unit tersembunyi.
- e. Langkah 4 : Hitung keluaran di unit tersembunyi ( $Z_j$ ).

Pada langkah ini menggunakan rumus algoritma (1) pada bab dua dengan menghitung nilai bobot pada keempat inputan

$$Z_{_1} = 4.2612 + 0.69 (-0.5278) + 0.14 (0.1632) + 0.77 (1.1167) + 0.75 (-2.9605) + 0.69 (-0.7819) + 0.90 (0.0173) + 0.90 (-0.4613) + 0.69 (-1.9127) = -5,148394338$$

$$Z_{_2} = -0.0308 + 0.69 (1.7366) + 0.14 (-1.4915) + 0.77 (1.6959) + 0.75 (0.8638) + 0.69(1.6642) + 0.90(0.9981) + 0.90 (-0.9800) + 0.69 (-1.4608) = 1,864612777$$

$$Z_1 = \frac{1}{1+e^{-1.42438}} = 0,005775178$$

$$Z_2 = \frac{1}{1+e^{-0.2043}} = 0,865833699$$

- f. Langkah 5 : Hitung Keluaran Unit *Outputnya* ( $Y_k$ ).

Langkah ini menggunakan rumus algoritma (2) pada bab dua dengan menghitung nilai bobot pada *hidden layer* ke unit output.

Karena jaringan ini hanya memiliki satu unit *output* y, maka :

$$Y_{_k} = -0.1795 + 0,005775178 (0.8709) + 0,865833699 (0.8338) = 0,547461741$$

$$Y_k = \frac{1}{1+e^{-0.9818}} = 0,6335$$

Tahap II : Umpan Mundur (*Backward Propagation*)

- g. Langkah 6 : Hitung faktor  $\delta$  di unit output  $Y_k$

Pada langkah ke enam menggunakan rumus algoritma (3) pada bab dua. Karena jaringan hanya memiliki satu buah unit output, maka:

$$\delta_k = \delta = (t - y) y (1 - y) = (1 - 0,6335) (0,6335) (1 - 0,6335) = 0,0850778$$

Suku perubahan bobot  $W_{kj}$  (dengan  $a = 0.2$ ) :

$$\Delta W_{10} = 0.2 (0,0850778) (1)$$

$$= 0,01701556$$

$$\Delta W_{11} = 0.2 (0,0850778) (0,005775178)$$

$$= 0,00009827$$

$$\Delta W_{12} = 0.2 (0,0850778) (0,865833699)$$

$$= 0,014732645$$

- h. Langkah 7 : Hitung penjumlahan kesalahan dari unit tersembunyi.

Langkah ini menggunakan rumus algoritma (5) pada bab dua dengan menjumlahkan hasil perubahan masukannya. Karena jaringan hanya memiliki satu buah unit *ouput*, maka:

$$\delta_{_1} = (0,0850778) * (0,8709)$$

$$= 0,074094256$$

$$\delta_{_2} = (0,0850778)(0.8338)$$

$$= 0,070937870$$

Faktor kesalahan  $\delta$  di unit tersembunyi: Faktor kesalahan  $\delta$  ini menggunakan rumus algoritma (6) pada bab dua yaitu dikalikan dengan turunan fungsi aktivasinya untuk menghitung informasi kesalahannya.

$$\delta_1 = 0,074094256 (0,005775178) (1 - 0,005775178) = 0,000425436$$

$$\delta_2 = 0,070937870 (0,865833699) (1 - 0,865833699) = -0,053179851$$

Tahap III : Pembaharuan Bobot Bias

a. Langkah 8 : Hitung semua perubahan bobot.

Langkah ke delapan menggunakan rumus algoritma (7) pada bab dua. Perubahan bobot unit *output* dengan  $W_{jk} = W_{kj}$  :

$$W_{10}(\text{Baru}) = -0.1795 + 0,01701556$$

$$= -0,162484440$$

$$W_{11}(\text{Baru}) = 0.8709 + 0,00009827$$

$$= 0,870998268$$

$$W_{12}(\text{Baru}) = 0.8338 + 0,014732645$$

$$= 0,848532645$$

Hasil percobaan atau pelatihan Jaringan Syaraf Tiruan menunjukkan bahwa *backpropagation* yang sudah dilakukan akan memberikan *training* yang baik yaitu mempunyai hasil ketetapan atau kebenaran yang cukup baik jika diberikan masukan yang serupa (tidak harus sama). Setiap kali membentuk jaringan *backpropagation* selalu diberi nilai bobot dan bisa awal dengan bilangan acak kecil. Bobot dan bias ini akan berubah setiap ada pembentukan jaringan.

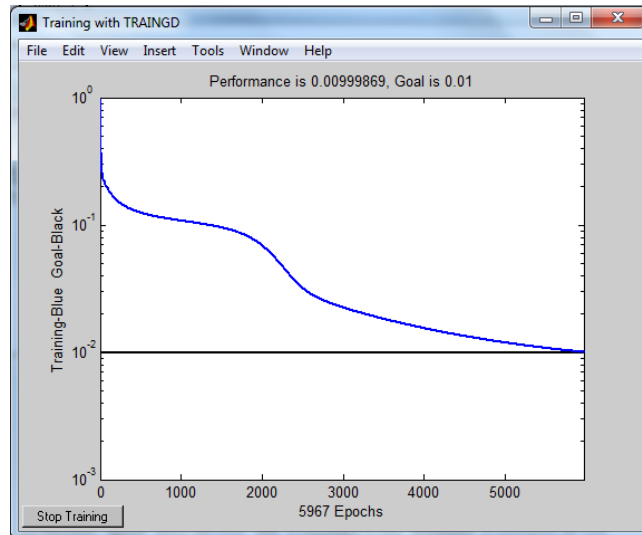
Nilai akurat untuk mengetahui pola predeksi nilai ujian akhir sekolah ini adalah jika nilai *error* ( $e$ ) yang dihasilkan adalah 0 ( nol ). Untuk menghasilkan nilai *error* yang lebih kecil, maka perlu dilakukan terus pelatihan-pelatihan dengan menggunakan bobot awal yang baru, sehingga dihasilkan nilai *error* sama dengan 0 (nol).

Dalam melakukan pelatihan dan pengujian hasil dari pengolahan data nilai siswa, maka pengolahan data tersebut juga akan diujikan ke dalam sistem komputer. *Software* yang akan digunakan yaitu dengan memanfaatkan *software Matlab 6.1*.

Langkah pertama yang harus dilakukan untuk pemrograman *Backpropagation* dengan *Matlab* adalah dengan membuat inisialisasi jaringan. Namun sebelumnya data yang akan diujikan haruslah dibagi menjadi dua bagian, di mana bagian pertama adalah untuk data pelatihan dan bagian kedua adalah untuk data pengujian.

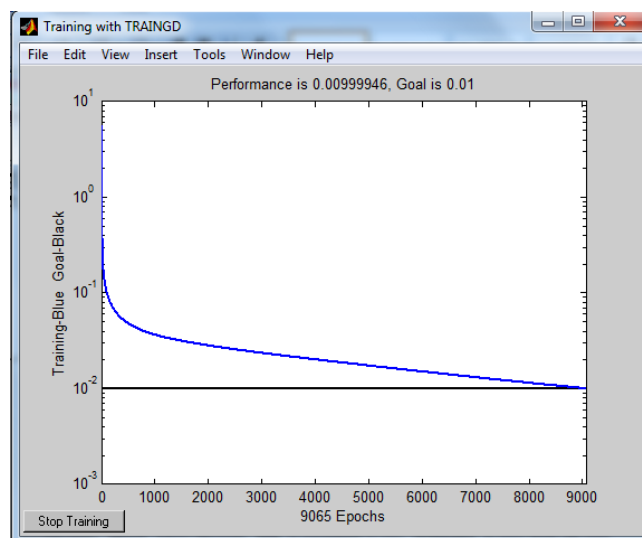
Hasil pelatihan yang terbaik dapat dilihat pada gambar berikut:





**Gambar 4** : Hasil Pelatihan 8-3-1 Sampai dengan 5000000 epochs

Pelatihan dihentikan pada epoch ke 5967 untuk kerja yang diinginkan ( $MSE=0.00999869$ ). Hasil pengujian yang terbaik dapat dilihat pada gambar berikut:



**Gambar 5** : Hasil Pengujian 8-6-1 Sampai dengan 5000000 epochs

Pelatihan dihentikan pada epoch ke 9065 untuk kerja yang diinginkan ( $MSE=0.00999946$ ).

## 5. KESIMPULAN

Setelah melakukan implementasi dan pengujian dengan menggunakan bahasa pemrograman *Matlab*, maka dapat disimpulkan sebagai berikut:

1. Jaringan Saraf Tiruan *backpropagation* yang telah dilatih telah cukup mampu melakukan prediksi nilai ujian sekolah siswa MAN dengan tingkat akurasi yang cukup baik, berdasarkan 8 variabel nilai siswa yang dibutuhkan yaitu: Nilai Matematika, Nilai Fisika, Nilai, Biologi, Nilai Kimia, Nilai Bahasa Indonesia, Nilai PKn dan Nilai Sejarah.
2. Arsitektur yang paling tepat digunakan untuk memprediksi nilai ujian akhir sekolah dengan Jaringan Syaraf Tiruan dengan menggunakan algoritma *Backpropagation* dengan *momentum* adalah 8-2-1 dengan membagi data menjadi 2 bagian yaitu 25 data pelatihan dan 35 data pengujian.

3. Untuk lebih mengetahui kemampuan prediksi nilai ujian sekolah, semakin banyak jumlah data yang dilatih maka akan semakin baik kemampuan prediksi yang dihasilkan oleh Jaringan Syaraf Tiruan, namun akan berdampak pada melambatnya proses pelatihan.

## Referensi

- [1] Dany dan Hindayati M. 2013. ” *Penerapan Jaringan Saraf Tiruan dengan Metode Pembelajaran Backpropagation untuk Mengetahui Tingkat Kualifikasi Calon Siswa pada Sistem Informasi Penerimaan Siswa Baru di MAN 2 Banjarnegara*”. JUITA. Universitas Muhammadiyah. Purwokerto.
- [2] N. Q. Hung. 2009. ” *An artificial neural network model for rainfall forecasting in Bangkok, Thailand*”. HYDROL. EARTH SYST. SCI. Asian Institute of Technology. Thailand.
- [3] Vashty Armila W. 2010. ” *Implementasi Metode Backpropagation Pada Seleksi Penerimaan Anggota Lembaga Kemahasiswaan*”. Teknik Informatika, Program Teknologi Informasi dan Ilmu Komputer. Universitas Brawijaya.
- [4] Puji Pangastuti. 2010. ” *Metode Jaringan Syaraf Tiruan Backpropagation Untuk Mengukur Tingkat Korelasi Prestasi Mahasiswa (Studi Kasus Pada Universitas Dian Nuswantoro Semarang)*”. Program Studi Teknik Informatika. Universitas Dian Nuswantoro. Semarang.
- [5] Ilham Sayekti. 2013. ” *Pengujian Model Jaringan Syaraf Tiruan Untuk Kualifikasi Calon Mahasiswa Baru Program Bidik Misi*”. JTET. Politeknik Negeri Semarang. Semarang.
- [6] Maharani Dessy W, dan Irawan Afrianto. 2012. ” *Perbandingan Metode Jaringan Syaraf Tiruan Backpropagation Dan Learning Vector Quantization Pada Pengenalan Wajah*”. KOMPUTA. Universitas Komputer Indonesia. Bandung.
- [7] Nazla N, Aris S, dan Eko A.S. 2010. ” *Algoritma Back Propagation Neural Network Untuk Pengenalan Pola Karakter Huruf Jawa*”. JURNAL MASYARAKAT INFORMATIKA. Universitas Diponegoro. Semarang.
- [8] Sandro Arif W. 2009. ” *Analisa Prediksi Prestasi Siswa Baru Slta Menggunakan Metode Jaringan Syaraf Tiruan Backpropagation*”. FAKULTAS TEKNIK INFORMATIKA. Universitas Telkom. Bandung.
- [9] Divya. 2013. ” *Diabetes Detection Using Artificial Neural Networks & Back*”. INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH. Bina Nusantara University. Jakarta.
- [10] Dian Pratiwi, Diaz D. Santika dan Bens P. 2013. ” *An Application Of Backpropagation Artificial Neural Network Method for Measuring The Severity of Osteoarthritis*”. IJET-IJENS. Bina Nusantara University. Jakarta.