

Penggunaan Jaringan Syaraf Tiruan Backpropagation Untuk Seleksi Penerimaan Mahasiswa Baru Pada Jurusan Teknik Komputer Di Politeknik Negeri Sriwijaya

^a Maria Agustin, ^bToni Prahasto

^a Program Studi Sistem Informasi,
Politeknik Negeri Sriwijaya, Palembang

^c Jurusan Teknik Mesin, Fakultas Teknik
Universitas Diponegoro, Semarang

Abstract

Data availability of new students at the Polytechnic State Sriwijaya high enough, so the need for a method to analyze the data. Artificial neural network is an information processing system that has characteristics similar to biological neural networks, neural networks are used to predict because of the ability of a good approach to non-linear. This study will design the software selection admission of new students with a backpropagation neural network method. From the analysis of backpropagation neural networks with one hidden layer with the number of neurons 50, 1000 iterations and the activation function sigmoid produce regression of 0.4822. Backpropagation neural network with two hidden layers with the number of neurons 50, 4000 iterations with sigmoid activation function, resulting in regression of 0.7944. Backpropagation neural networks with 3 hidden layers with the number of neurons 35, 5000 iterations, resulting in regression of 0.8563. Based on the results of this analysis, backpropagation neural networks quite effectively used for selection of candidates for student admission.

Keywords: Selection, Backpropagation, Regression

1. Pendahuluan

Jaringan syaraf tiruan merupakan salah satu sistem pemrosesan informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan suatu masalah dengan melakukan proses belajar melalui perubahan bobot sinapsisnya. Jaringan syaraf tiruan mampu mengenali kegiatan dengan berbasis masa lalu. Data masa lalu akan dipelajari oleh jaringan syaraf tiruan sehingga mempunyai kemampuan untuk memberi keputusan terhadap data yang belum pernah dipelajari.

Jaringan syaraf tiruan, seperti manusia, belajar dari suatu contoh karena mempunyai karakteristik yang adaptif, yaitu dapat belajar dari data sebelumnya dan mengenal pola data yang selalu berubah, selain itu jaringan syaraf tiruan merupakan sistem yang tak terprogram, artinya semua keluaran atau kesimpulan yang ditarik oleh jaringan didasarkan pada pengalamannya selama mengikuti proses pembelajaran atau pelatihan.

Selama ini permasalahan penerimaan mahasiswa baru dengan cara konvensional sangatlah sulit apalagi hanya dengan melihat nilai akhir atau NEM saja, karena dengan nilai yang tinggi belum tentu bisa menjamin bahwa mahasiswa tersebut dapat diterima atau tidak. Sistem seleksi konvensional merupakan tahap awal pada saat seleksi penerimaan mahasiswa, dengan dibantu komputer dan pendekatan jaringan syaraf tiruan diharapkan dapat membantu seleksi masuk calon mahasiswa. Dengan jaringan syaraf tiruan komputer difungsikan sebagai alat bantu yang dapat memprediksi calon mahasiswa tersebut.

Pada penelitian ini akan dikembangkan metode jaringan syaraf tiruan *backpropagation* untuk seleksi penerimaan mahasiswa baru untuk jurusan teknik komputer di Politeknik

Negeri Sriwijaya. sumber data yang dipergunakan adalah data calon mahasiswa baru yang mendaftar tiap tahunnya. Sumber data 3 tahun dari tahun 2008 sampai dengan tahun 2010.

2. Kerangka Teori

2.1. Jaringan Syaraf Tiruan

Jaringan syaraf tiruan (JST) adalah paradigma pemrosesan suatu informasi yang terinspirasi oleh sistem sel syaraf biologi. JST dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi, dengan asumsi bahwa:

1. Pemrosesan informasi terjadi pada banyak elemen sederhana (*neuron*).
2. Sinyal dikirimkan diantara *neuron-neuron* melalui penghubung-penghubung.
3. Penghubung antar *neuron* memiliki bobot yang akan memperkuat atau memperlemah sinyal.
4. Untuk menentukan *output*, setiap *neuron* menggunakan fungsi aktivasi (biasanya bukan fungsi linier) yang dikenakan pada jumlahan *input* yang diterima. Besarnya *output* ini selanjutnya dibandingkan dengan suatu batas ambang.

Neuron biologi merupakan sistem yang "fault tolerant" dalam 2 hal. Pertama, manusia dapat mengenali sinyal *input* yang agak berbeda dari yang pernah kita terima sebelumnya. Sebagai contoh, manusia sering dapat mengenali seseorang yang wajahnya pernah dilihat dari foto atau dapat mengenali seseorang yang wajahnya agak berbeda karena sudah lama tidak menjumpainya. Kedua, tetap mampu bekerja dengan baik. Jika sebuah *neuron* rusak, *neuron* lain dapat dilatih

untuk menggantikan fungsi *neuron* yang rusak tersebut (Jong, 2004).

Hal yang ingin dicapai dengan melatih JST adalah untuk mencapai keseimbangan antara kemampuan memorisasi dan generalisasi. Yang dimaksud kemampuan memorisasi adalah kemampuan JST untuk mengambil kembali secara sempurna sebuah pola yang telah dipelajari. Kemampuan generalisasi adalah kemampuan JST untuk menghasilkan respon yang bisa diterima terhadap pola-pola yang sebelumnya telah dipelajari. Hal ini sangat bermanfaat bila pada suatu saat ke dalam JST itu di inputkan informasi baru yang belum pernah dipelajari, maka JST itu masih akan tetap dapat memberikan tanggapan yang baik, memberikan keluaran yang mendekati (Puspaningrum, 2006).

Jaringan syaraf tiruan menyerupai otak manusia dalam 2 hal, yaitu :

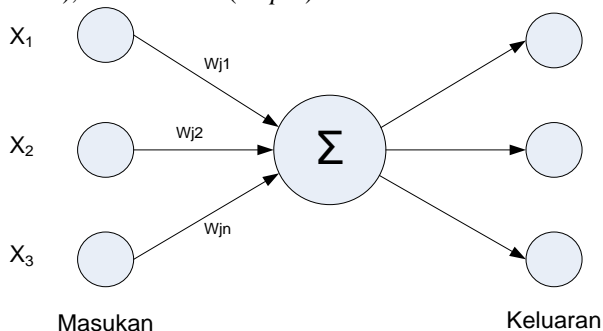
1. Pengetahuan diperoleh jaringan melalui proses belajar.
2. Kekuatan hubungan antar sel syaraf (*neuron*) yang dikenal sebagai bobot-bobot sinaptik digunakan untuk menyimpan pengetahuan.

Jaringan syaraf tiruan ditentukan oleh 3 hal (Jong, 2004) :

1. Pola hubungan antar *neuron* (disebut arsitektur jaringan).
2. Metode untuk menentukan bobot penghubung (disebut metode *training/learning*).
3. Fungsi aktivasi, yaitu fungsi yang digunakan untuk menentukan keluaran suatu *neuron*.

2.2 Model Neuron

Satu sel syaraf terdiri dari 3 bagian, yaitu : fungsi penjumlahan (*summing function*), fungsi aktivasi (*activation function*), dan keluaran (*output*).



Gambar 1. Model Neuron (Hermawan, 2006)

Jika dilihat, *neuron* buatan diatas mirip dengan sel *neuron* biologis. Informasi (*input*) akan dikirim ke *neuron* dengan bobot tertentu. *Input* ini akan diproses oleh suatu fungsi yang akan menjumlahkan nilai-nilai bobot yang ada. Hasil penjumlahan kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap *neuron*. Apabila *input* tersebut melewati suatu nilai ambang tertentu, maka *neuron* tersebut akan diaktifkan, jika tidak, maka *neuron* tersebut akan mengirimkan *output* melalui bobot-bobot *outputnya* ke semua *neuron* yang berhubungan dengannya.

Sehingga dapat disimpulkan bahwa *neuron* terdiri dari 3 elemen pembentuk yaitu :

1. Himpunan unit-unit yang dihubungkan dengan jalur koneksi. Jalur-jalur tersebut memiliki bobot yang berbeda-beda. Bobot yang bernilai positif akan memperkuat sinyal dan yang bernilai negatif akan memperlemah sinyal yang dibawa. Jumlah, struktur, dan

pola hubungan antar unit-unit tersebut akan menentukan arsitektur jaringan.

2. Suatu unit penjumlahan yang akan menjumlahkan *input-input* sinyal yang sudah dikalikan dengan bobotnya.
3. Fungsi aktivasi yang akan menentukan apakah sinyal dari *inputneuron* akan diteruskan ke *neuron* lain atau tidak.

2.3 Konsep Dasar Jaringan Syaraf Tiruan

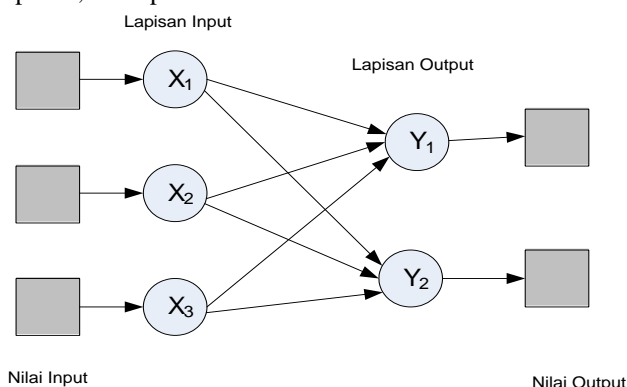
Setiap pola-pola informasi *input* dan *output* yang diberikan kedalam jaringan syaraf tiruan diproses dalam *neuron*. *Neuron-neuron* tersebut terkumpul didalam lapisan-lapisan yang disebut *neuronlayers*. Lapisan-lapisan penyusun jaringan syaraf tiruan tersebut dapat dibagi menjadi 3, yaitu :

1. Lapisan *input*.
Unit-unit di dalam lapisan *input* disebut unit-unit *input*. Unit-unit *input* tersebut menerima pola inputan data dari luar yang menggambarkan suatu permasalahan.
2. Lapisan tersembunyi.
Unit-unit di dalam lapisan tersembunyi disebut unit-unit tersembunyi. Dimana *outputnya* tidak dapat secara langsung diamati.
3. Lapisan *output*.
Unit-unit di dalam lapisan *output* disebut unit-unit *output*. *Output* dari lapisan ini merupakan solusi jaringan syaraf tiruan terhadap suatu permasalahan.

2.4 Arsitektur Jaringan Syaraf Tiruan

Jaringan syaraf tiruan memiliki beberapa arsitektur jaringan yang sering digunakan dalam berbagai aplikasi. Arsitektur jaringan syaraf tiruan tersebut, antara lain (Kusumadewi, 2004).

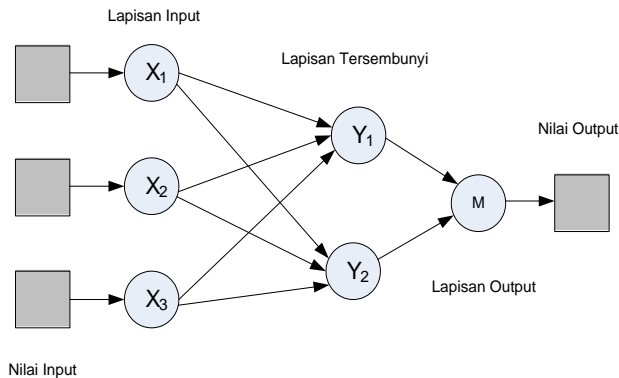
1. Jaringan Layar Tunggal (*single layer network*).
Jaringan dengan lapisan tunggal terdiri dari 1 layer *input* dan 1 layer *output*. Setiap *neuron/unit* yang terdapat di dalam lapisan / layer *input* selalu terhubung dengan setiap *neural* yang terdapat pada layer *output*. Jaringan ini hanya menerima *input* kemudian secara langsung akan mengolahnya menjadi *output* tanpa harus melalui lapisan tersembunyi. Contoh algoritma jaringan syaraf tiruan yang menggunakan metode ini yaitu : ADALINE, Hopfield, Perceptron.



Gambar 2. Arsitektur Layer Tunggal (Hermawan, 2006)

2. Jaringan layar jamak (*multi layer network*).
Jaringan dengan lapisan jamak memiliki ciri khas tertentu yaitu memiliki 3 jenis *layer* yakni *layer input*, *layer output*, layer tersembunyi. Jaringan dengan banyak

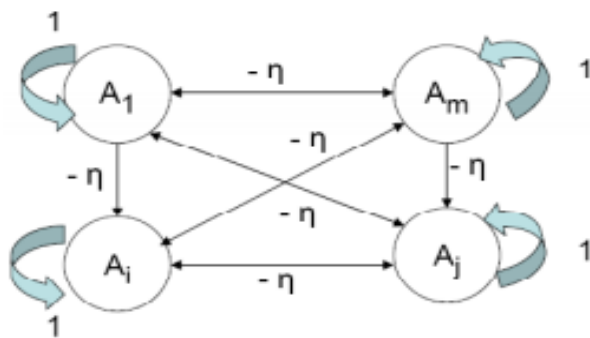
lapisan ini dapat menyelesaikan permasalahan yang kompleks dibandingkan jaringan dengan lapisan tunggal. Namun proses pelatihan sering membutuhkan waktu yang cenderung lama. Contoh algoritma jaringan syaraf tiruan yang menggunakan metode ini yaitu : MADALINE, *backpropagation*, *neocognitron*.



Gambar 3. Arsitektur Layer Jamak (Hermawan, 2006)

3. Jaringan dengan lapisan kompetitif.

Pada jaringan ini sekumpulan *neuron* bersaing untuk mendapatkan hak menjadi aktif. Contoh algoritma yang menggunakan metode ini adalah : LVQ.



Gambar 4. Arsitektur Layer Kompetitif (Hermawan, 2006)

2.5 Metode pelatihan/Pembelajaran Jaringan Syaraf Tiruan.

Cara berlangsungnya pembelajaran atau pelatihan jaringan syaraf tiruan dikelompokkan menjadi 3 yaitu (Puspitaningrum, 2006):

- a. *Supervised Learning* (pembelajaran terawasi).
 Pada metode ini, setiap pola yang diberikan kedalam jaringan syaraf tiruan telah diketahui *output*nya. Selisih antara pola *output* aktual (*output* yang dihasilkan) dengan pola *output* yang dikehendaki (*output* target) yang disebut *error* digunakan untuk mengoreksi bobot jaringan syaraf tiruan sehingga jaringan syaraf tiruan mampu menghasilkan *output* sedekat mungkin dengan pola target yang telah diketahui oleh jaringan syaraf tiruan. Contoh algoritma jaringan syaraf tiruan yang menggunakan metode ini adalah : *Hebbian*, *Perceptron*, *ADALINE*, *Boltzman*, *Hopfield*, *Backpropagation*.
- b. *Unsupervised Learning* (pembelajaran tak terawasi).
 Pada metode ini, tidak memerlukan target *output*. Pada metode ini tidak dapat ditentukan hasil seperti apakah yang diharapkan selama proses pembelajaran. Selama proses pembelajaran, nilai bobot disusun dalam suatu

range tertentu tergantung pada nilai *input* yang diberikan. Tujuan pembelajaran ini adalah mengelompokkan unit-unit yang hampir sama dalam suatu area tertentu. Pembelajaran ini biasanya sangat cocok untuk klasifikasi pola. Contoh algoritma jaringan syaraf tiruan yang menggunakan metode ini adalah : *Competitive*, *Hebbian*, *Kohonen*, *LVQ (Learning Vector Quantization)*, *Neocognitron*.

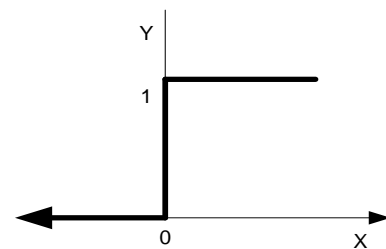
- c. *Hybrid Learning* (pembelajaran hibrida).
 Merupakan kombinasi dari metode pembelajaran *Supervised Learning* dan *Unsupervised Learning*, sebagian dari bobot-bobotnya ditentukan melalui pembelajaran terawasi dan sebagian lainnya melalui pembelajaran tak terawasi. Contoh algoritma jaringan syaraf tiruan yang menggunakan metode ini adalah: algoritma RBF. Metode algoritma yang baik dan sesuai dalam melakukan pengenalan pola-pola gambar adalah algoritma *Backpropagation* dan *Perceptron*. Untuk mengenali teks berdasarkan tipe *font* akan digunakan algoritma *Backpropagation*.

2.6 Fungsi Aktivasi Jaringan Syaraf Tiruan

Dalam jaringan syaraf tiruan, fungsi aktivasi digunakan untuk menentukan keluaran suatu *Neuron*. Argument fungsi aktivasi adalah net masukan (kombinasi linier masukan dan bobotnya).

Beberapa fungsi aktivasi yang digunakan adalah :

- a. *Fungsi Threshold* (batas ambang).
 Fungsi Threshold merupakan fungsi *threshold* biner. Untuk kasus bilangan bipolar, maka angka 0 diganti dengan angka -1. Adakalanya dalam jaringan syaraf tiruan ditambahkan suatu unit masukan yang nilainya selalu 1. Unit tersebut dikenal dengan bias. Bias dapat dipandang sebagai sebuah *input* yang nilainya selalu 1. Bias berfungsi untuk mengubah *threshold* menjadi = 0.



Gambar 5. Fungsi aktivasi Threshold (Kusumadewi, 2004)

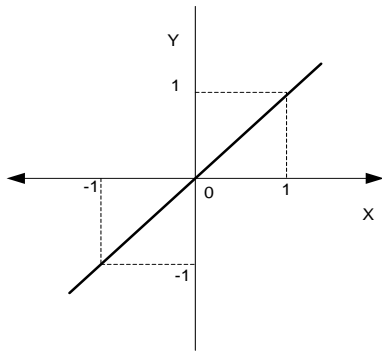
$$F(x) = \begin{cases} 1 & \text{Jika } x \geq a \\ 0 & \text{Jika } x < a \end{cases}$$

- b. *Fungsi Sigmoid*.
 Fungsi ini sering digunakan karena nilai fungsinya yang sangat mudah untuk di diferensiasikan.

$$F(x) = \frac{1}{1+e^{-x}}$$

- c. *Fungsi Identitas*.
 Digunakan jika keluaran yang dihasilkan oleh jaringan syaraf tiruan merupakan sembarang bilangan riil (bukan hanya pada range [0,1] atau [1,-1]).

$$Y = X$$



Gambar 6. Fungsi aktivasi Identitas (Kusumadewi, 2004)

2.7 Algoritma Umum Jaringan Syaraf Tiruan

Algoritma pembelajaran/pelatihan jaringan syaraf tiruan (Puspitaningrum, 2006).

Dimasukkan n contoh pelatihan kedalam jaringan syaraf tiruan, lakukan :

1. Inisialisasi bobot-bobot jaringan. Set $I = 1$.
2. Masukkan contoh ke I (dari sekumpulan contoh pembelajaran yang terdapat dalam set pelatihan) kedalam jaringan pada lapisan *input*.
3. Cari tingkat aktivasi unit-unit input menggunakan algoritma aplikasi
 If \rightarrow kinerja jaringan memenuhi standar yang ditentukan sebelumnya (memenuhi syarat untuk berhenti).
 Then \rightarrow exit.
4. Update bobot-bobot dengan menggunakan aturan pembelajaran jaringan.
5. If $i = n$ then reset $i = 1$
 Else $i = i-1$
 Ke langkah 2.

Algoritma aplikasi/inferensi jaringan saraf tiruan (Puspitaningrum, 2006):

Dimasukkan sebuah contoh pelatihan kedalam jaringan saraf tiruan, lakukan:

1. Masukkan kasus kedalam jaringan pada lapisan *input*.
2. Hitung tingkat aktifasi *node-node* jaringan.
3. Untuk jaringan koneksi umpan maju, jika tingkat aktivasi dari semua unit *outputnya* telah dikalkulasi, maka *exit*. Untuk jaringan dengan kondisi balik, jika tingkat aktivasi dari semua unit *outputnya* menjadi konstan atau mendekati konstan, maka *exit*. Jika tidak, kembali ke langkah 2. Jika jaringannya tidak stabil, maka *exit* dan *fail*.

2.8. Model Jaringan Backpropagation

Model jaringan backpropagation merupakan suatu teknik pembelajaran atau pelatihan *supervised learning* yang paling banyak digunakan. Metode ini merupakan salah satu metode yang sangat baik dalam menangani masalah pengenalan pola-pola kompleks. Didalam jaringan *backpropagation*, setiap unit yang berada di lapisan *input* berhubungan dengan setiap unit yang ada di lapisan tersembunyi. Setiap unit yang ada di lapisan tersembunyi terhubung dengan setiap unit yang ada di lapisan *output*. Jaringan ini terdiri dari banyak lapisan (*multilayer network*). Ketika jaringan ini diberikan pola masukan sebagai pola pelatihan, maka pola tersebut menuju unit-unit lapisan

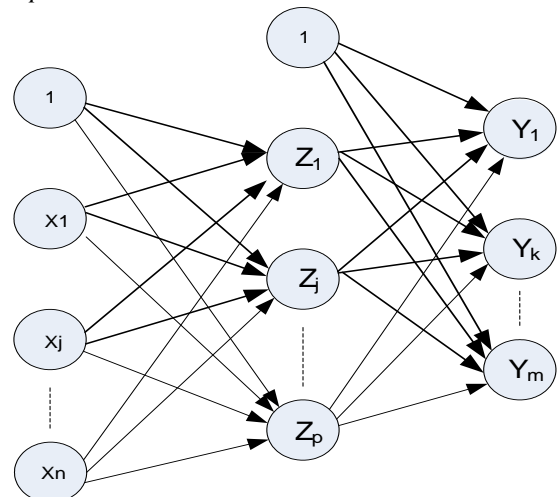
tersembunyi untuk selanjutnya diteruskan pada unit-unit dilapisan keluaran. Kemudian unit-unit lapisan keluaran akan memberikan respon sebagai keluaran jaringan syaraf tiruan. Saat hasil keluaran tidak sesuai dengan yang diharapkan, maka keluaran akan disebarkan mundur (*backward*) pada lapisan tersembunyi kemudian dari lapisan tersembunyi menuju lapisan masukan.

Tahap pelatihan ini merupakan langkah untuk melatih suatu jaringan syaraf tiruan, yaitu dengan cara melakukan perubahan bobot, sedangkan penyelesaian masalah akan dilakukan jika proses pelatihan tersebut telah selesai, fase ini disebut fase pengujian (Puspitaningrum, 2006).

2.9 Arsitektur Jaringan Backpropagation

Setiap unit dari layer *input* pada jaringan *backpropagation* selalu terhubung dengan setiap unit yang berada pada layer tersembunyi, demikian juga setiap unit layer tersembunyi selalu terhubung dengan unit pada layer *output*. Jaringan *backpropagation* terdiri dari banyak lapisan (*multilayer network*) yaitu:

1. Lapisan *input* (1 buah), yang terdiri dari 1 hingga n unit *input*.
2. Lapisan tersembunyi (minimal 1 buah), yang terdiri dari 1 hingga p unit tersembunyi.
3. Lapisan *output* (1 buah), yang terdiri dari 1 hingga m unit *output*.



Gambar 7. Arsitektur jaringan backpropagation (Jong, 2004)

2.10. Pelatihan Jaringan Backpropagation

Aturan pelatihan jaringan *backpropagation* terdiri dari 2 tahapan, *feedforward* dan *backward propagation*. Pada jaringan diberikan sekumpulan contoh pelatihan yang disebut set pelatihan. Set pelatihan ini digambarkan dengan sebuah vektor *feature* yang disebut dengan vektor *input* yang diasosiasikan dengan sebuah *output* yang menjadi target pelatihannya. Dengan kata lain set pelatihan terdiri dari vektor *input* dan juga vektor *output* target. Keluaran dari jaringan berupa sebuah vektor *output* aktual. Selanjutnya dilakukan perbandingan antara *output* aktual yang dihasilkan dengan *output* target dengan cara melakukan pengurangan diantara kedua *output* tersebut. Hasil dari pengurangan berupa *error*. *Error* dijadikan sebagai dasar dalam melakukan perubahan dari setiap bobot dengan mempropagationkannya kembali.

Setiap perubahan bobot yang terjadi dapat mengurangi *error*. Siklus perubahan bobot (*epoch*) dilakukan pada setiap set pelatihan sehingga kondisi berhenti dicapai, yaitu bila mencapai jumlah *epoch* yang diinginkan atau hingga sebuah nilai ambang yang ditetapkan terlampaui.

Algoritma pelatihan jaringan backpropagation terdiri dari 3 tahapan yaitu:

1. Tahap umpan maju (*feedforward*).
2. Tahap umpan mundur (*backpropagation*).
3. Tahap pengupdatean bobot dan bias.

Secara rinci algoritma pelatihan jaringan backpropagation dapat diuraikan sebagai berikut :

- Langkah 0 : Inisialisasi bobot-bobot, konstanta laju pelatihan (α), toleransi *error* atau nilai bobot (bila menggunakan nilai bobot sebagai kondisi berhenti) atau set maksimal *epoch* (jika menggunakan banyaknya *epoch* sebagai kondisi berhenti).
- Langkah 1 : Selama kondisi berhenti belum dicapai, maka lakukan langkah ke-2 hingga langkah ke-9.
- Langkah 2 : Untuk setiap pasangan pola pelatihan, lakukan langkah ke-3 sampai langkah ke-8.
- Langkah 3 : {Tahap I : Umpan maju (*feedforward*)}. Tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya.
- Langkah 4 : Masing-masing unit di lapisan tersembunyi (dari unit ke-1 hingga unit ke-p) dikalikan dengan bobotnya dan dijumlahkan serta ditambahkan dengan biasnya.
- Langkah 5 : Masing-masing unit *output* ($y_k, k=1,2,3,\dots,m$) dikalikan dengan bobot dan dijumlahkan serta ditambahkan dengan biasnya.
- Langkah 6 : {Tahap II : Umpan mundur (*backward propagation*)}. Masing-masing unit *output* ($y_k, k=1,2,3,\dots,m$) menerima pola target t_k sesuai dengan pola masukan/*input* saat pelatihan dan kemudian informasi kesalahan/*error* lapisan *output* (δ_k) dihitung. δ_k dikirim ke lapisan dibawahnya dan digunakan untuk menghitung besarnya koreksi bobot dan bias (ΔW_{jk} dan ΔW_{ok}) antara lapisan tersembunyi dengan lapisan *output*.
- Langkah 7 : Pada setiap unit dilapisan tersembunyi (dari unit ke-1 hingga ke-p; $i=1,\dots,n; k=1,\dots,m$) dilakukan perhitungan informasi kesalahan lapisan tersembunyi (δ_j). δ_j kemudian digunakan untuk menghitung besar koreksi bobot dan bias (ΔV_{ji} dan ΔV_{jo}) antara lapisan *input* dan lapisan tersembunyi.
- Langkah 8 : {Tahap III : Pengupdatean bobot dan bias}. Masing-masing unit *output*/keluaran ($y_k, k=1,2,3,\dots,m$) dilakukan pengupdatean

bias dan bobotnya ($j=0,1,2,\dots,p$) sehingga menghasilkan bobot dan bias baru. Demikian juga untuk setiap unit tersembunyi mulai dari unit ke-1 sampai dengan unit ke-p dilakukan pengupdatean bobot dan bias.

Langkah 9 : Uji kondisi berhenti (akhir iterasi).

3. Metodologi

3.1 Bahan Penelitian

Bahan yang dibutuhkan untuk melakukan penelitian dalam pembuatan aplikasi seleksi penerimaan mahasiswa baru dengan metode jaringan syaraf tiruan *backpropagation* adalah data calon mahasiswa yang mendaftar di Politeknik Negeri Sriwijaya dari tahun 2008-2010. Data yang diproses dipisahkan menjadi 2 bagian, bagian pertama dipergunakan untuk melatih jaringan syaraf tiruan, bagian kedua dipergunakan untuk menguji kinerja dari jaringan syaraf tiruan, yaitu untuk menguji apakah jaringan menghasilkan *output* sesuai dengan yang diinginkan. Pada waktu *input* yang belum pernah dipelajari oleh jaringan. Semakin banyak data yang diperoleh, semakin baik jaringan dapat menyelesaikan masalahnya.

3.2 Alat Penelitian

Alat yang digunakan dalam proses penelitian ini adalah :

1. Pada penelitian ini alat yang digunakan adalah seperangkat komputer dengan spesifikasi ram 1GB, prosesor intel core 2 Dou, HDD 320 Gb.
2. Perangkat lunak yang digunakan untuk membangun sistem ini adalah : Microsoft excel 2007, Matlab 2011.

Matlab singkatan dari MATrix Laboratory merupakan bahasa pemrograman komputer generasi ke empat yang dikembangkan oleh The Mathwork. Inc . Matlab memungkinkan manipulasi matriks, dan data, implementasi algoritma, pembuatan antarmuka dan penghubung antara *user* dengan program dalam bahasa lainnya. Terdapat fitur tambahan yang dapat digunakan sebagai sarana pemodelan simulasi dan analisis dari sistem dinamik dengan menggunakan antarmuka GUI (*graphical user interface*), yang dikenal nama simulink.

Karakteristik Matlab :

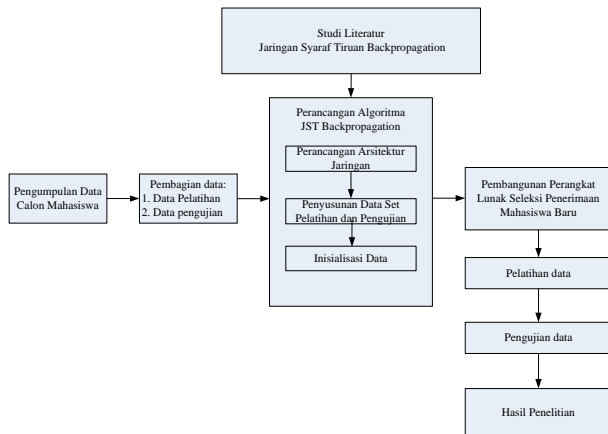
- a. Bahasa pemrogramannya didasarkan pada matrix (baris dan kolom).
- b. Lambat (dibandingkan dengan Fortran atau C) karena bahasanya langsung diartikan. Sebagai contoh, tidak diperlukan *pre-compiled*.
- c. *Automatic memory management*, misalnya kita tidak harus mendeklarasikan arrays terlebih dahulu.
- d. Memiliki waktu pengembangan program yang lebih cepat dibandingkan bahasa pemrograman tradisional seperti Fortran atau C.
- e. Dapat diubah ke bahasa C melalui matlab compiler untuk efisiensi yang lebih baik.
- f. Tersedia banyak toolbox untuk aplikasi-aplikasi khusus.
- g. Bisa dipakai disemua sistem operasi seperti : Unix, Windows atau Macintos.
- h. Fungsi matematikanya lebih lengkap.

3.3 Jalan Penelitian

Proses pembuatan sistem seleksi penerimaan mahasiswa baru di Politeknik Negeri Sriwijaya dengan metode jaringan syaraf tiruan *backpropagation* diterangkan pada penjelasan sebagai berikut yang meliputi kerangka pemikiran dan arsitektur jaringan.

3.3.1 Kerangka Pemikiran.

Kerangka pemikiran seleksi penerimaan mahasiswa baru di Politeknik Negeri Sriwijaya ditunjukkan seperti gambar 3.1



Gambar 8. Kerangka pemikiran seleksi penerimaan mahasiswa baru Politeknik Negeri Sriwijaya

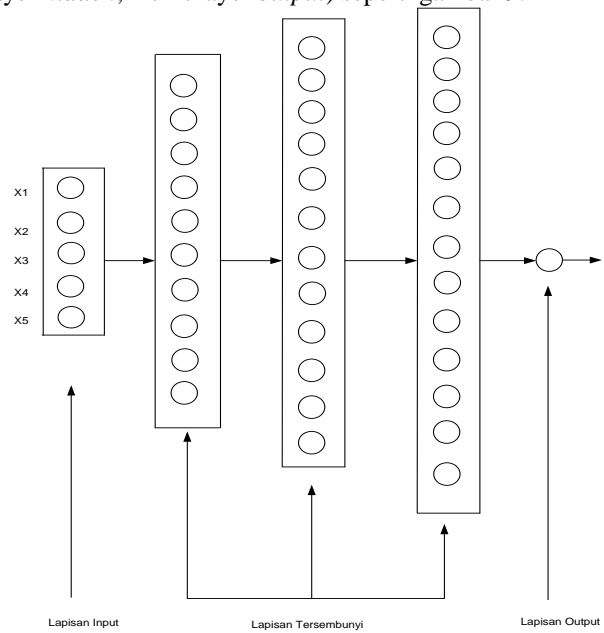
Berikut ini penjelasan kerangka pemikiran seleksi penerimaan mahasiswa baru yang telah digambarkan :

1. Studi literatur merupakan proses mempelajari dan memahami teori-teori tentang jaringan syaraf tiruan *backpropagation*, yang didapat melalui jurnal, buku, artikel, situs internet dan sumber ilmiah lainnya.
2. Data calon mahasiswa diperoleh dari bagian akademik Politeknik Negeri Sriwijaya, data dikumpulkan menjadi satu kesatuan kemudian dilakukan tahap pembagian data, yaitu data pelatihan dan data pengujian. Data pelatihan digunakan ketika melatih jaringan, sedangkan data pengujian digunakan ketika jaringan divalidasi.
3. Perancangan algoritma jaringan syaraf tiruan *backpropagation* terdiri dari beberapa tahap yaitu :
 - a. Perancangan arsitektur jaringan untuk menentukan *node input* (simpul masukan), jumlah lapisan layer tersembunyi dengan jumlah *nodenya* (*simpul hidden*), dan jumlah *node output* (simpul keluaran). Pada saat perancangan arsitektur jaringan ditentukan juga fungsi aktivasi yang akan digunakan. Fungsi aktivasi yang digunakan harus sesuai dengan algoritma yang digunakan dan karakteristik data.
 - b. Menyusun data set yang akan digunakan untuk pelatihan dan pengujian.
 - c. Inisialisasi data meliputi, inisialisasi variabel *learning rate*, maksimum *epoch* dan batas nilai toleransi pelatihan *backpropagation*.

3.3.2 Arsitektur Jaringan Syaraf Tiruan

Model jaringan syaraf tiruan untuk seleksi penerimaan mahasiswa baru menggunakan struktur jaringan syaraf tiruan 5 layer/lapisan. Struktur jaringan syaraf tiruan 5 layer terdiri dari 1 layer *input*, 3 layer *hidden* dan 1 layer *output*. Pada

tesis ini masing-masing *layer* mempunyai net struktur sebagai berikut: net struktur 5- 3- 1 (5 unit layer *input*, 3 unit layer *hidden*, 1 unit layer *output*) seperti gambar 9.



Gambar 9. Struktur Jaringan syaraf tiruan untuk seleksi penerimaan mahasiswa baru

Keterangan:

- X1 = Jurusan
- X2 = Jenis kelamin (1 untuk laki-laki, 0 perempuan).
- X3 = Asal sekolah (diambil 1 digit dari depan jurusan).
- X4 = Usia ijazah
- X5 = Nilai rata-rata

Tabel 1. Kode Jurusan

Kode Jurusan	Nama Jurusan
101	SMA IPA
213	SMK (Teknik Informasi dan Komunikasi)
303	SMK (Teknik Konstruksi Bangunan)
304	SMK (Teknik Perkayuan)
308	SMK (Teknik Mesin Perkakas)
309	SMK (Teknik Mesin Otomotif)
310	SMK (Teknik Mekanik Umum)
314	SMK (Teknik Instalasi Listrik)
317	SMK (Teknik Elektronika Komunikasi)
318	SMK (Teknik Audio Video)
320	SMK (Teknik Pelayaran)
401	MA IPA
500	Lain-lain

3.3.3 Sistem Perangkat Lunak

Sistem perangkat lunak dalam proses seleksi penerimaan mahasiswa baru sebatas fase pembelajaran jaringan (*learning*) dan fase penggunaan (*mapping*). Pada implementasi *software* untuk menjalankan *software* sistem seleksi penerimaan mahasiswa baru pada Politeknik Negeri Sriwijaya menggunakan *software* Matlab 2011.

Fase Pembelajaran JST Backpropagation Error.

Pada fase ini dilakukan proses adaptasi bobot untuk masing-masing bobot antara *layer input* dan *layer hidden*, *layer hidden* dan *layer output*. Adaptasi bobot dilakukan secara terus menerus sampai dicapai keadaan *error* yang paling minimum adapun perhitungan untuk fase pembelajaran adalah bobot awal yang telah di inialisasi, kemudian dilakukan perhitungan seperti berikut :

Hitung harga aktivasi sel-sel pada *layer hidden* dengan persamaan 3.1

$$Z_j = f(V_{0j} + \sum_i^n X_i V_{ij}) \quad (3.1)$$

Dimana V_{ij} adalah sel-sel pada *layer input*, Z_j adalah sel pada *layer hidden* dan X_i adalah bobot interkoneksi antara *layer input* dengan *layer hidden*, selanjutnya hitung harga aktivasi sel-sel pada *layer output* dengan persamaan 3.2

$$Y_k = f(W_{0k} + \sum_j W_{kj} * Z_j) \quad (3.2)$$

Dimana Y_k adalah sel-sel pada *layer output* dan W_{kj} adalah bobot interkoneksi antara *layer hidden* dengan *layer output*. Proses perhitungan harga aktivasi ini berlangsung terus sampai pola berakhir. Kemudian hitung harga δ_k yang akan digunakan untuk menghitung interkoneksi antara *layer hidden* dengan *layer output* dalam jaringan melalui persamaan 3.3

$$\delta_k = [(t_k - y_k) * (1 - y_k) * y_k] \quad (3.3)$$

Dimana t_k merupakan hasil akhir dari jaringan yang diinginkan atau biasa disebut dengan target. Selanjutnya dilakukan pencarian harga δ_j yang akan digunakan untuk menghitung perubahan bobot interkoneksi antara *layer input* dengan *layer hidden* melalui persamaan 3.4

$$\delta_j = \sum_k (\delta_k - W_{kj}) * (1 - Z_j) * Z_j \quad (3.4)$$

Selanjutnya dilakukan adaptasi bobot interkoneksi antara lapisan *hidden* dengan lapisan *output* dengan persamaan 3.5

$$W_{kj}(t + 1) = W_{kj}(t) + \Delta W_{kj}(t) \quad (3.5)$$

Dimana

$$\Delta W_{kj}(t + 1) = \sum (\delta_j * Z_j) + \Delta W_{kj}(t - 1)\Delta$$

adalah kecepatan belajar dan α adalah momentum, kemudian dilakukan lagi adaptasi bobot interkoneksi antara *layer input* dengan *layer hidden* dengan persamaan 3.6

$$V_{ji}(t + 1) = V_{ji}(t) + \Delta V_{ji}(t) \quad (3.6)$$

Tahap selanjutnya adalah perhitungan kesalahan antara target dengan hasil aktivasi jaringan syaraf tiruan. Jika harga kesalahan rata-rata masih besar, proses pembelajaran di ulang dari awal begitu seterusnya sampai dicapai harga kesalahan yang minimum.

4. Hasil dan Pembahasan

4.1. Hasil Penelitian

Data yang digunakan dalam penelitian ini adalah data calon mahasiswa yang sudah ditraining atau dilatih. Fokus penelitian ini adalah seleksi penerimaan mahasiswa baru

sehingga dapat diketahui bahwa calon mahasiswa dapat diterima atau tidak, dengan menggunakan variabel yang telah ditentukan sebelumnya yaitu : jurusan, jenis kelamin, asal sekolah, usia ijasah, dan nilai rata-rata.

4.1.1. Proses Pembelajaran

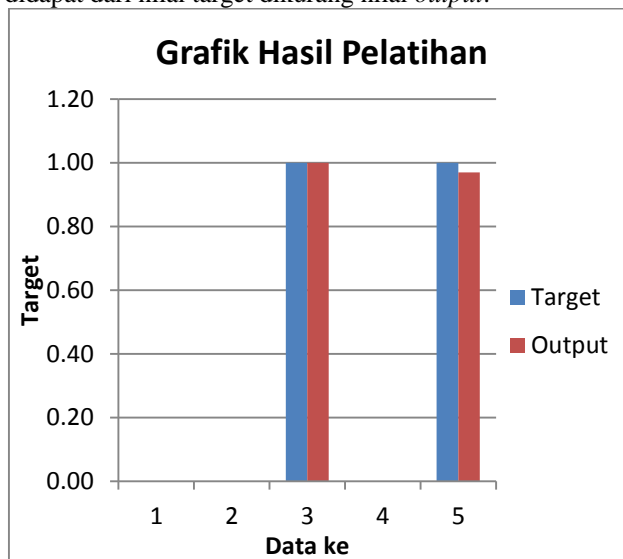
Proses pembelajaran dilakukan dengan menggunakan *software* matlab yang telah menyediakan fungsi-fungsi pembelajaran dan pengujian pada jaringan syaraf tiruan dengan algoritma *backpropagation*. Proses pembelajaran dilakukan untuk mencari konfigurasi terbaik dengan cara mengubah konstanta belajar dan jumlah lapisan tersembunyi secara coba-coba (*trial and error*). Data calon mahasiswa yang digunakan pada proses pembelajaran adalah data calon mahasiswa dalam waktu 2 tahun diambil dari tahun 2008 dan 2009.

Adapun hasil pembelajaran atau pelatihan dapat dilihat pada tabel 2 dibawah ini.

Tabel 2. Hasil pelatihan

Data ke-	Target	Output	Error
1	0.00	-0.00	0.00
2	0.00	0.00	-0.00
3	1.00	1.00	0.00
4	0.00	0.00	-0.00
5	1.00	0.97	0.03

Tabel diatas adalah hasil dari pelatihan atau pembelajaran data calon mahasiswa tahun 2008 dan 2009 Untuk data pertama targetnya adalah 0, *output*nya -0.00, nilai *error*nya 0.00, nilai *error* didapat dari kebalikan nilai *output*, kecuali untuk nilai yang targetnya 1, nilai *error*nya didapat dari nilai target dikurang nilai *output*.



Gambar 10. Grafik hasil pelatihan

Gambar diatas menggambarkan grafik batang hasil pelatihan data calon mahasiswa tahun 2008 dan 2009. Untuk data yang pertama, kedua, dan keempat targetnya adalah 0, *output*nya juga 0 maka grafik batangnya tidak ditampilkan. Untuk data yang ketiga targetnya adalah 1, *output*nya 1, tidak ada *error*nya sehingga grafik batang antara target dan *output* sejajar. Untuk data yang kelima targetnya 1, *output*nya 0.97 dan *error*nya 0.03.

4.2 Pengujian

Pengujian dilakukan melalui dua tahap yaitu, pengujian terhadap data yang dilatih/training dan pengujian data baru yang belum pernah dilatih/training. Berikut resume proses pembelajaran jaringan syaraf tiruan untuk seleksi penerimaan calon mahasiswa baru.

- *Input* : 5 unit
- *Hidden layer* : 3 unit
- *Output layer* : 1 unit
- *Target error* : 0.02
- Pola yang diajarkan : data calon mahasiswa tahun 2008 dan 2009
- Banyak data : 329
- Jumlah iterasi : 5000

Selama proses pembelajaran berlangsung, *error* dari tiap pola dapat ditampilkan, *error* dari tiap pola yang diajarkan pada jaringan yang disimpan dari iterasi 1 sampai dengan 5000. Penyebab lamanya proses pembelajaran pada struktur jaringan yang digunakan adalah pola yang diajarkan sangat banyak, sehingga *looping* dari satu iterasi ke iterasi dipengaruhi oleh banyaknya pola dan struktur data pada pola tersebut, selain itu penyebab *target error* rata-rata jaringan sangat kecil yaitu, 0.02, semakin kecil *target error* jaringan semakin lama proses pembelajaran yang dilakukan oleh jaringan karena untuk memperkecil nilai *error*.

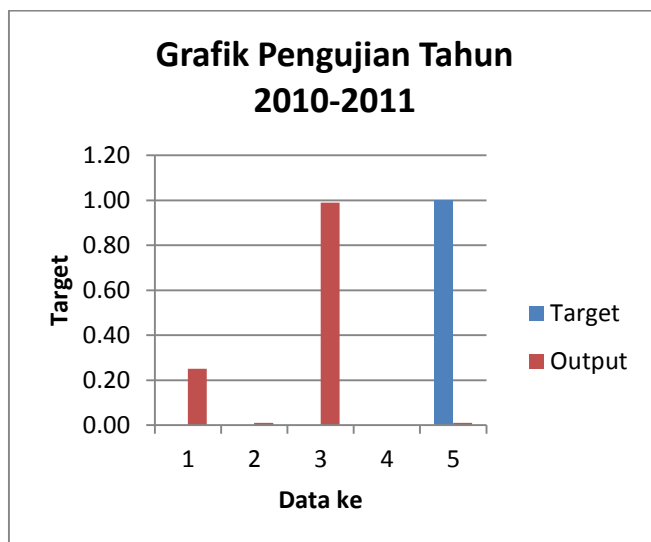
Pada proses pembelajaran, jaringan melakukan proses *backward* atau arus balik dengan cara mengubah nilai bobot jaringan, sebelum *target error* terpenuhi, pembelajaran akan terus dilanjutkan sampai tercapai *target error*. Semakin kecil *target error* yang ditetapkan semakin akurat jaringan mengenali dan menghitung bentuk pola baru yang diujikan kepadanya.

Adapun hasil pengujian terhadap data calon mahasiswa tahun 2010-2011 pada tabel 3 sebagai berikut:

Tabel 3. Hasil pengujian tahun 2010-2011

Data ke	Target	Output	Error
1	0.00	0.25	-0.25
2	0.00	0.01	-0.01
3	0.00	0.99	-0.99
4	0.00	-0.00	0.00
5	1.00	0.01	0.99

Tabel diatas merupakan hasil pengujian dari data calon mahasiswa tahun 2010-2011 Data pertama targetnya 0, *outputnya* 0.25 , *errornya* -0.25. Data kedua targetnya 0, *outputnya* -0.01, *errornya* -0.01. Data ketiga targetnya 0, *outputnya* 0.99, *errornya* -0.99. Data keempat targetnya 0, *outputnya* -0.00, *errornya* 0.00. Data kelima targetnya 1, *outputnya* 0.01, *errornya* 0.99.



Gambar 11. Grafik Pengujian Tahun 2010-2011

Gambar diatas menggambarkan grafik batang hasil pengujian data calon mahasiswa tahun 2010-2011. Dari grafik batang terlihat untuk data pertama, kedua, dan ketiga, yang ditampilkan hanya *outputnya* saja karena data tersebut mempunyai target 0. Untuk data keempat *output* dan target tidak tampil karena data tersebut mempunyai *output* dan targetnya 0. Untuk data kelima targetnya 1 sedangkan *outputnya* 0.01.

4.2.1 Analisa Hasil Pelatihan dengan 1 hidden layer.

Hasil pelatihan data seleksi penerimaan calon mahasiswa baru tahun 2008,2009, dengan menggunakan struktur jaringan 5 layer terdiri dari 1 unit *input*, 3 *hidden layer* dan 1 unit *output* dengan 5000 *epoch*, seperti tabel 4 dibawah ini.

Tabel 4. *Output* JST dengan 1 hidden layer

Jumlah Neuron	Input		Output		
	Iterasi	Regresi	MSE	Waktu (detik)	
30	1000	0.22669	0.11695	31.356	
50	1000	0.48220	0.09400	57.339	
70	1000	0.33978	0.10839	90.200	
80	1000	0.33795	0.10857	119.674	
90	1000	0.23452	0.11573	148.668	
100	1000	0.35111	0.10757	203.606	
150	1000	0.37345	0.10544	474.838	
200	1000	0.38701	0.10414	974.959	

4.2.2 Analisa Hasil Pelatihan Dengan 2 Hidden Layer.

Hasil pelatihan data seleksi penerimaan calon mahasiswa baru tahun 2008, 2009, dengan menggunakan struktur jaringan 5 layer terdiri dari 1 unit *input*, 3 *hidden layer* dan 1 unit *output* dengan 5000 *epoch*, seperti tabel 5 dibawah ini.

Tabel 5. *output* JST dengan 2 *hidden layer*

Input		Output		
Jumlah Neuron	Iterasi	Regresi	MSE	Waktu (detik)
24	5000	0.7165	0.0596	236.0410
25	1000	0.7096	0.0608	35.7420
25	1500	0.6129	0.0765	56.9140
25	3000	0.2816	0.1128	106.7530
30	2500	0.7799	0.0480	140.4130
30	4000	0.6929	0.0637	238.0720
40	1000	0.2146	0.1170	137.8760
40	1500	0.3028	0.1117	216.0210
50	4000	0.7944	0.0452	1439.6030
50	3000	0.7294	0.0573	1112.5790
60	2500	0.7391	0.0556	1686.3270

4.2.3 Analisa Hasil Pelatihan Dengan 3 Hidden Layer.

Hasil pelatihan data seleksi penerimaan calon mahasiswa baru tahun 2008, 2009, dengan menggunakan struktur jaringan 5 layer terdiri dari 1 unit *input*, 3 *hidden layer* dan 1 unit *output* dengan 5000 *epoch*, seperti tabel 6 dibawah ini.

Tabel 6. *Output* JST dengan 3 *hidden layer*

Input		Output		
Jumlah Neuron	Iterasi	Regresi	MSE	Waktu (detik)
35	1000	0.2470	0.1150	59.9200
35	2000	0.7496	0.0537	132.4910
35	5000	0.8647	0.0309	436.9130

5. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan dapat diambil beberapa kesimpulan diantaranya adalah :

1. Jaringan syaraf tiruan dengan 1 *hidden layer*, dengan jumlah *neuron* 50, iterasi 1000 dengan fungsi aktivasi *tansig* mampu mendekati regresi 0.4822.
2. Jaringan syaraf tiruan dengan 2 *hidden layer*, dengan jumlah *neuron* 50, iterasi 4000 dengan fungsi aktivasi *tansig* mampu mendekati regresi 0.7944.

3. Jaringan syaraf tiruan dengan 3 *hidden layer*, dengan jumlah *neuron* 35, iterasi 5000 dengan fungsi aktivasi *tansig* mampu mendekati regresi 0.8563.

4. Inisialisasi parameter (*learning rate*, fungsi aktivasi, jumlah iterasi), dan struktur jaringan sangat menentukan dalam proses pembelajaran jaringan syaraf tiruan.

Dari penelitian ini saran yang dapat diberikan adalah :

Dari penelitian ini, masih dapat dilakukan pengembangan arsitektur jaringan syaraf tiruan *backpropagation* sampai pada tahap prediksi, dengan memasukkan variabel yang lain seperti, asal daerah, penghasilan orang tua.

Daftar Pustaka

- Azadeh, A, Ghaderi S.F., Sohrabkhani S. 2008. *Annual Electricity Consumption Forecasting by Neural Network in High Energy Consuming Industrial Sectors*. ScienceDirect.
- Budiyanto, D., 2000. Data Mining Dengan Jaringan Syaraf Tiruan Backpropagation. *Jurnal Teknologi Industri* Vol. IV. No. 1. Januari 2000: 75-82.
- Chaudhari, Ms, J.C., 2010. *Design Of Artificial Back Propagation Neural Network For Drug Pattern Recognition*. International Journal On Computer Science and Enginnering.
- Chrestanti, R., Santoso A.J., Ernawati L., 2002. Implementasi Backpropagation Dalam Memprediksi Kebangkrutan Bank Di Indonesia. *Jurnal Teknologi Industri*. Vol. VI, No. 4, Oktober 2002.
- Hartanto, T., Prasetyo, Y. *Analisis dan Desain Sistem Kontrol Dengan Matlab*. Penerbit Andi. Yogyakarta.
- Hermawan, A., 2006. *Jaringan Syaraf Tiruan Teori dan Aplikasi*. Penerbit Andi. Yogyakarta.
- Jayalakshmi T., Santhakumaran., 2011. Statistical Normalization and Back Propagation for Classification. *Internasional Journal of Computer Theory and Engineering*. Vol. 3, No. 1, February, 2011.
- Jong S.J. 2004. *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*. Penerbit Andi. Yogyakarta.
- Kusumadewi, S., 2004. *Membangun Jaringan Syaraf Tiruan Menggunakan Matlab & Excel Link*. Penerbit Graha Ilmu.
- Nuraeni, Y. 2009. Penerapan Jaringan Syaraf Tiruan Untuk Mengukur Tingkat Korelasi Antara NEM Dengan IPK Kelulusan Mahasiswa. *Telkomnika*. Vol. 7, No. 3, Desember 2009.
- Sutikno, T., Pujianta A., Supanti Y.T., 2007. Prediksi Resiko Kredit Dengan Jaringan Syaraf Tiruan Backpropagation. *Seminar Nasional Aplikasi Teknologi Informasi 2007*.
- Puspitaningrum, D., 2006. *Pengantar Jaringan Syaraf Tiruan*. Penerbit Andi Jogjakarta.
- Widodo, P.P. Handayanto, R.T., 2009. *Penerapan Soft Computing Dengan Matlab*. Penerbit. Rekayasa Sains.