

Prediksi Jumlah Penjualan Air Mineral Pada Perusahaan “XYZ” Dengan Jaringan Saraf Tiruan

Kusuma Dewangga, S.Kom.
Jurusan Ilmu Komputer
Universitas Gadjah Mada
Jl. Bulaksumur, Yogyakarta
kusumadewangga@gmail.com

ABSTRAK

Propagasi balik adalah sebuah metode pengenalan suatu pola melalui supervisi, dimana diperlukan pembelajaran terlebih dahulu setelah itu tahap uji data. Studi kasus pada penelitian ini terkonsentrasi pada data penjualan air mineral kemesan pada perusahaan yang bertempat di Surabaya. Permasalahan pada perusahaan air mineral ini adalah sulitnya mengalokasikan sumber daya untuk melakukan produksi pada saat permintaan pasar melonjak. Dari permasalahan tersebut, diperlukan suatu prediksi sebagai alat penunjang pengambilan keputusan untuk mengetahui perkiraan permintaan pasar. Pemanfaatan algoritma propagasi balik akan digunakan pada data penjualan air mineral sehingga dihasilkan suatu prediksi atau perkiraan. Berdasarkan prediksi ini perusahaan ingin mengalokasikan sumber daya yang diperlukan untuk memenuhi kebutuhan pasar. Adapun berbagai permasalahan yang dihadapi dalam implementasi algoritma propagasi balik untuk prediksi, yaitu: bagaimana memodelkan Jaringan Saraf Tiruan yang dipakai untuk metode propagasi balik ini, bagaimana data yang digunakan untuk input kedalam Jaringan Saraf Tiruan agar dapat diproses dengan baik, dan bagaimana akurasi prediksi yang dihasilkan dari Jaringan Saraf Tiruan yang sudah diimplementasikan. Kemampuan metode propagasi balik untuk melakukan prediksi cukup efektif namun ada baiknya jika data dari pola ditambahkan.

Kata Kunci: Jaringan Saraf Tiruan, Propagasi Balik, Air Mineral, Prediksi, Data Penjualan

1. PENDAHULUAN

Jaringan Saraf Tiruan(JST) merupakan cabang ilmu yang berusaha memetakan pola pikir manusia kedalam komputer. Jaringan Saraf Tiruan dibentuk dari generalisasi model matematis. Model dari jaringan saraf tiruan ini ada berbagai macam, khusus pada kasus ini digunakan model *Backpropagation*. Objek studi kasus pemanfaatan Jaringan Saraf Tiruan ini digunakan padadata transaksi penjualan Perusahaan Air Minum tertentu yang bertempat di Surabaya. Dengan memanfaatkan Jaringan Saraf Tiruan ini diharapkan pengelola perusahaan mengetahui berapa jumlah sumber daya pada perusahaan yang diperlukan untuk memenuhi permintaan pasar dalam kurun waktu tertentu terhadap produk air minum.

2. METODOLOGI PENELITIAN

A. Jaringan Saraf Tiruan(JST)

Jaringan Saraf Tiruan adalah sistem pemrosesan informasi atau data yang cara kerjanya mirip dengan jaringan saraf biologi. Jaringan saraf tiruan dibentuk sebagai generalisasi model matematika dari jaringan saraf biologi dengan asumsi bahwa:

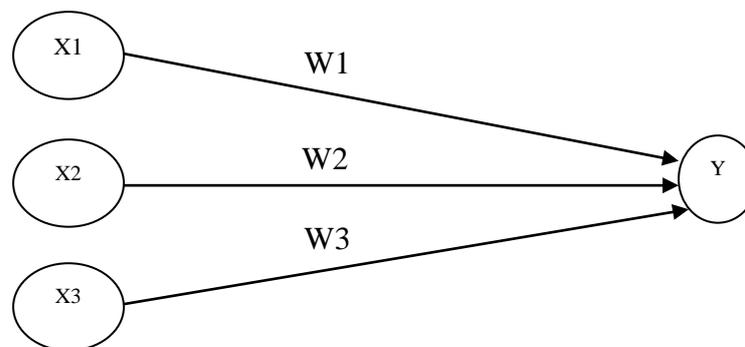
1. Pemrosesan informasi terjadi pada elemen sederhana yang disebut dengan neuron.
2. Sinyal dikirimkan diantara neuron-neuron melalui penghubung-penghubung.
3. Penghubung antar neuron memiliki bobot yang akan dmemerkuat atau memperlemah sinyal.

Untuk menentukan output, setiap neuron menggunakan fungsi aktivasi yang dikenakan pada jumlah inputan yang diterima. Besarnya output ini selanjutnya dibandingkan dengan sinyal output. Karakteristik jaringan saraf tiruan ditentukan oleh tiga hal:

1. Pola penghubung antar neuron disebut dengan arsitektur jaringan.
2. Metode untuk menentukan bobot penghubung disebut dengan metode pelatihan(*training*) danpembelajaran(*learning*).
3. Fungsi aktivasi.

Jaringan saraf tiruan terdiri dari banyak elemen sederhana yang disebut dengan neuron, unit, sel, atau node(*vertex*). Masing-masing neuron ini nantinya terhubung dengan neuron yang lain yang disebut sebagai *edge*. *Edge* ini memiliki nilai bobot

sendiri-sendiri. Bobot tersebut mewakili informasi yang digunakan oleh jaringan saraf tiruan untuk menyelesaikan masalah. Tiap neuron memiliki kondisi awal yang disebut dengan aktivasi yaitu fungsi yang digunakan untuk menerima sinyal inputan. Biasanya neuron mengirimkan aktivasinya sebagai sinyal ke beberapa neuron yang lain. Penting untuk diketahui bahwa neuron hanya bisa mengirimkan satu kali sinyal pada saat yang sama meskipun sinyal tersebut dikirimkan secara *broadcast* ke beberapa neuron yang lain.



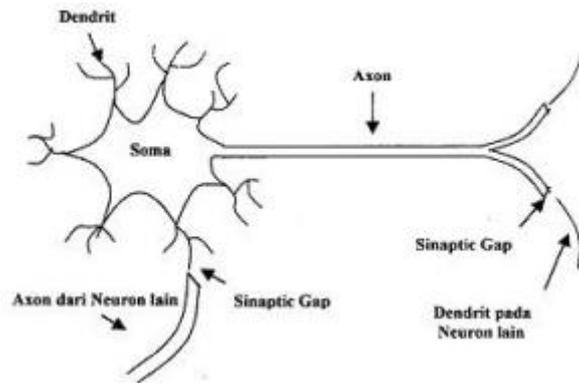
Gambar Contoh Jaringan Saraf Tiruan

Neuron Y akan menerima *input* dari neuron X1, X2, dan X3. Sinyal output neuron tersebut adalah X1, X2, dan X3. Bobot yang dari X1, X2, dan X3 ke neuron Y adalah W1, W2, dan W3. Inputan y_{in} ke neuron Y dapat dinotasikan sebagai berikut.

$$y_{in} = w_1x_1 + w_2x_2 + w_3x_3$$

B. Jaringan Saraf Biologi

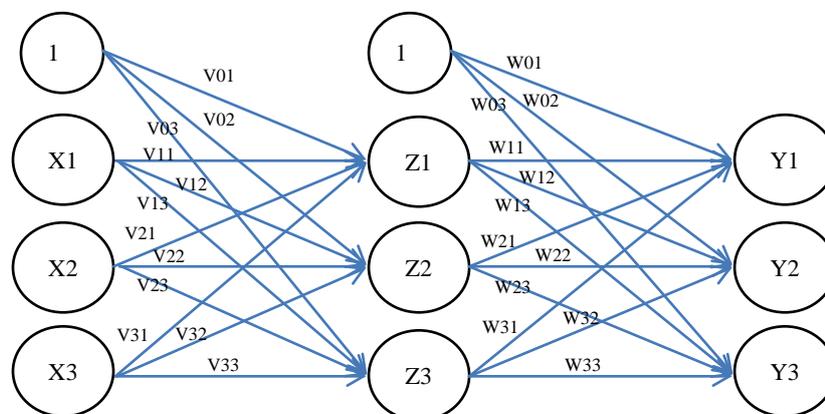
Otak manusia memiliki struktur yang sangat kompleks, karena terdiri dari neuron-neuron dan penghubung yang disebut sinapsis. Neuron bekerja berdasarkan sinyal yang diberikan pada neuron sebelumnya. Kemudian neuron tersebut meneruskannya pada neuron yang lain. Neuron biologi terdiri dari tiga komponen yaitu: *dendrit*, *soma*, dan *axon*. Dendrit menerima sinyal dari neuron-neuron yang lain. Sinyal tersebut adalah sinyal listrik yang dikirimkan melalui sinapsis dengan cara proses kimiawi. Melalui proses kimiawi tersebut, sinyal yang diterima diubah terus menerus hingga melakukan adaptasi sesuai dengan lingkungan yang diperlukan. Proses inilah yang mirip dengan proses perubahan bobot pada jaringan saraf tiruan. Dalam Jaringan saraf tiruan *soma* diperlakukan sebagai *vertex* dan *axon* atau penghubung antar *soma* diperlakukan sebagai *edge*. Berikut ini adalah gambar jaringan saraf biologis pada manusia.



Gambar Jaringan Saraf Biologis

C. Model Jaringan Saraf Backpropagation

Jaringan saraf *Backpropagation* memiliki beberapa unit yang ada dalam satu atau lebih layar tersembunyi(*hidden layer*). Gambar 3 menunjukkan model jaringan backpropagation dengan 3 buah masukan dengan 1 bias, 3 buah hidden layer dengan 1 bias, dan 3 buah ouput layer.



Gambar Model Backpropagation

Notasi V_{ij} merupakan garis bobot dari unit masukan x_i ke unit layar tersembunyi z_j (v_{0j} merupakan garis bobot yang menghubungkan bias di unit masukan ke unit layar tersembunyi z_j). Notasi W_{kj} merupakan garis bobot dari unit layar tersembunyi z_j ke unit keluaran Y_k (w_{k0} merupakan garis bobot dari bias di layar tersembunyi ke unit keluaran Z_k).

D. Fase Training Backpropagation

Fase training backpropagation terdiri dari 3 fase yaitu fase *feedforward*, fase *backpropagation error*, dan fase pembaharuan nilai bobot. Ketiga fase ini adalah fase utama yang terus menerus dilakukan hingga mencapai suatu kondisi berhenti dari batas toleransi yang sudah diberikan.

Fase *feedforward* adalah fase umpan balik data masukan X_i disebarkan kedalam *hidden layer* menggunakan fungsi aktivasi yang telah ditentukan. Hasil perhitungan dari setiap *hidden layer* (Z_j) selanjutnya disebarkan lagi ke layer tersembunyi diatasnya (apabila ada) dan seterusnya sehingga didapatkan hasil output jaringan (Y_k). Selanjutnya output jaringan (Y_k) dibandingkan dengan target yang harus dicapai (t_k). Selisih antara ($t_k - y_k$) adalah error yang terjadi. Jika kesalahan atau error ini lebih kecil dari batas toleransi yang ditentukan maka proses looping dihentikan. Jika Sebaliknya, maka proses perhitungan akan dilanjutkan dengan memodifikasi bobot tiap garis dalam jaringan untuk mengurangi kesalahan yang terjadi.

Sinyal keluaran dari setiap unit pada lapisan keluaran akan dibandingkan dengan targetnya masing-masing yang telah didefinisikan untuk pola masukan tersebut sehingga akan dihasilkan *error* (kesalahan) untuk setiap unitnya. Apabila *error* masih lebih besar daripada *threshold* yang ada maka dilakukan umpan balik dari *output* kedalam *hiddenlayer* hingga input *layer*.

Setelah sampai kedalam *hidden layer* dari propagasi balik maka dilakukan perhitungan nilai bobot yang baru. Bobot ini nantinya mengganti bobot yang lama yaitu V_{ij} dan W_{kj} . Demikian seterusnya hingga memenuhi kondisi *threshold* yang telah diinginkan.

E. Fungsi Aktivasi

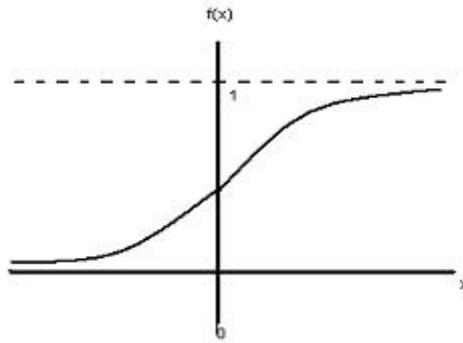
Data penjualan air mineral dalam bulan atau tahun sebelumnya harus ditransformasikan terlebih dahulu sebelum digunakan pada saat pelatihan atau training. Untuk mentransformasikan data penjualan air mineral tersebut digunakan fungsi aktivasi *sigmoid*. Fungsi aktivasi yang dipakai harus memenuhi beberapa syarat yaitu: *continue*, *differensiable*, dan merupakan fungsi yang tidak turun. Salah satu fungsi sigmoid biner dan fungsi sigmoid bipolar. Fungsi sigmoid biner memiliki *range*(0,1) dengan rumus sebagai berikut.

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

Apabila persamaan tersebut diturunkan maka persamaan tersebut menjadi.

$$f'(x) = f(x)(1 - f(x))$$

Apabila digambarkan Grafik fungsi *sigmoid biner* tersebut menjadi.



Gambar Fungsi Sigmoid Biner

Sedangkan fungsi *sigmoid bipolar* memiliki *range*(-1,1) didefinisikan dengan rumus sebagai berikut.

$$f(x) = \frac{2}{1 + e^{-x}} - 1$$

Apabila persamaan tersebut diturunkan maka persamaan tersebut menjadi.

$$f'(x) = \frac{(1 + f(x))(1 - f(x))}{2}$$

Terkadang dalam jaringan saraf tiruan ditambahkan sebuah unit masukan yang nilainya selalu = 1. Unit yang mempunyai sifat demikian itu disebut bias, bias dapat diartikan sebagai sebuah inputan yang nilainya = 1. Bias berfungsi untuk mengubah nilai *threshold* menjadi =0. Jaringan saraf tiruan backpropagation menggunakan bias dalam arsitektur jaringannya baik dalam input layer maupun dalam hidden layer. Jika menggunakan bias maka perhitungan di unit output adalah sebagai berikut:

$$Y_k = b + \sum x_i w_i$$

F. Algoritma Backpropagation

Untuk lebih memperjelas mengenai algoritma Propagasi Balik. Demikian akan dijelaskan mengenai langkah-langkah algoritma dari awal sampai akhir beserta dengan perhitungan matematis setiap langkahnya.

Algoritma Propagasi Balik

1. Inisialisasi semua nilai bobot.
2. Selama belum tercapai kondisi berhenti, lakukan.
3. Untuk setiap input attribut lakukan.
4. Setiap Input Unit ($X_i, i=1, 2, \dots, n$) \leftarrow Nilai Input
5. Setiap Hidden Unit ($Z_j, j=1, 2, \dots, p$) \leftarrow Nilai $X_i (i=1, 2, \dots, n)$
6. Untuk Setiap Hidden Unit
7. Hitung $z_{in} = v_{0j} + \sum_{i=1}^n x_i v_{ij}$

8. Hitung Output dari Hidden Layer $z_j = f(z_in_j)$
9. Setiap Output Unit ($Y_k, k=1, 2, \dots, q$) \leftarrow Nilai $Z_j (i=1, 2, \dots, p)$
10. Untuk Setiap Output Unit
11. Hitung $y_{in} = w_{0k} + \sum_{j=1}^p z_j w_{jk}$
12. Hitung output dari output layer $y_k = f(y_in_k)$
13. Hitung nilai error $\delta_k = (t_k - y_k) f'(y_in_k)$
14. Hitung nilai koreksi bobot $\Delta w_{jk} = \alpha \delta_k z_j$
15. Hitung nilai koreksi bias $\Delta w_{0k} = \alpha \delta_k$
16. $w_{jk}(new) \leftarrow w_{jk}(old) + \Delta w_{jk}$
17. $w_{0k}(new) \leftarrow w_{0k}(old) + \Delta w_{0k}$
18. Untuk Setiap Hidden Unit
19. Hitung $\delta_in_j = \sum_{k=1}^m \delta_k w_{jk}$
20. Hitung Error $\delta_j = \delta_in_j f'(z_in_j)$
21. Hitung Koreksi Bobot $\Delta v_{ij} = \alpha \delta_j x_i$
22. Hitung Koreksi Bias $\Delta v_{0j} = \alpha \delta_j$
23. $v_{ij}(new) \leftarrow v_{ij}(old) + \Delta v_{ij}$
24. $v_{0j}(new) = v_{0j}(old) + \Delta v_{0j}$
25. Periksa apakah sudah memenuhi kondisi berhenti

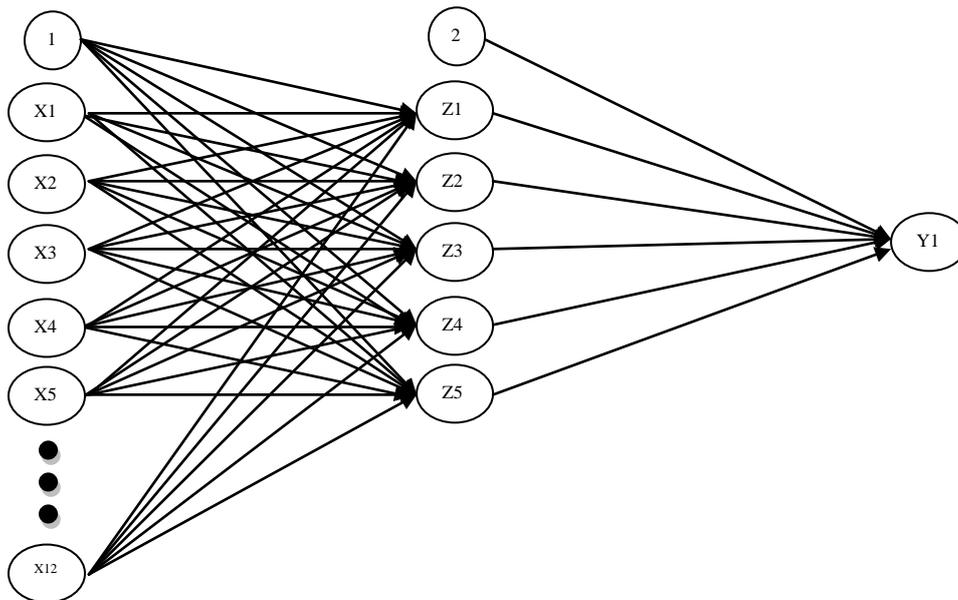
3. HASIL PEMBAHASAN

A. Analisis Model Kebutuhan JST

Dengan algoritma *backpropagation*, record data penjualan selama beberapa bulan atau tahun terakhir akan dipakai sebagai data pelatihan atau *training*, yang mana pada proses *training* ini bobot-bobot pada jaringan *backpropagation* akan melakukan perbaikan untuk mencari nilai yang optimal. Oleh karena itu, sebelumnya perlu ditentukan dulu besarnya periode dimana terdapat data yang fluktuatif. Dalam studi kasus untuk prediksi jumlah penjualan air mineral pada perusahaan “xyz” yang mana penjualan dapat terjadi tiap hari. Periode data dapat diambil selama satu bulan atau satu tahun. Namun, dalam hal ini dipilih data dalam periode satu tahun saja karena penjualan yang akan diperkirakan adalah penjualan pada bulan berikutnya.

Jumlah data dalam satu periode akan digunakan sebagai jumlah unit masukan dalam JST. Sedangkan sebagai target akan diambil data bulan pertama setelah periode berakhir. Pada data penjualan bulanan dengan periode dalam waktu satu tahun, maka masukan *backpropagation* yang akan dipakai terdiri dari 12 unit input dan 1 unit output.

Hidden layer yang dipakai pada JST ini adalah 5 unit. Unit sebagai input yang dipakai adalah 12 mewakili data penjualan untuk setiap bulannya.



Gambar Model Input dan Output JST

B. Prosedur Pengumpulan Data

Dalam *data preparation*, data penjualan air mineral harus ditransformasikan terlebih dahulu sebelum digunakan pada saat pelatihan atau *training*. Untuk melakukan transformasi data penjualan air mineral tersebut digunakan fungsi aktivasi sigmoid. Oleh karena dalam hal ini fungsi aktivasi sigmoid yang digunakan adalah fungsi aktivasi sigmoid(biner). Maka data harus ditransformasikan ke interval antara [0,1], proses ini sering disebut sebagai normalisasi. Untuk mentransformasikan data ke interval [0,1] digunakan transformasi linier.

$$x' = \frac{0,9(x - a)}{b - a} + 0.1$$

a = Nilai minimum penjualan

b = Nilai maksimum penjualan

Untuk lebih mengerti tentang penggunaan rumus diatas dibawah ini akan dijelaskan tabel penjualan air minum dari tahun 2004-2005.

Tabel Penjualan Air Minum

Penjualan Air Mineral Per Bulan												
Thn	Jan	Feb	Mar	Apr	Mei	Jun	Jul	Agu	Sep	Okt	Nop	Des
2004	10.620	7.725	3.284	2.400	1.767	1.463	1.090	1.070	1.355	5.324	7.167	13.780
2005	9.590	5.291	3.081	2.147	2.045	1.696	1.341	1.181	1.613	2.242	6.161	10.430

Data diatas adalah data penjualan air minum untuk setiap bulan pada tahun 2004-2005. Berikut ini adalah data penjualan air minum yang sudah dinormalisasikan menggunakan rumus normalisasi. Untuk lebih jelas mengenai proses perhitungan pada penjualan air minum pada tahun 2004 adalah sebagai berikut.

1. Dibentuk persamaan sebagai berikut

$$x' = \frac{0,9(x - 1070)}{13780 - 1070} + 0,1$$

2. Contoh untuk bulan Januari pada tahun 2004 lakukan perhitungan sebagai berikut.

$$x' = \frac{0,9(10620 - 1070)}{13780 - 1070} + 0,1$$

$$x' = \frac{0,9.9550}{12710} + 0,1$$

$$x' = 0,776$$

Tabel Hasil Proses Transformasi

Penjualan Air Mineral Per Bulan												
Thn	Jan	Feb	Mar	Apr	Mei	Jun	Jul	Agu	Sep	Okt	Nop	Des
2004	0.777	0.571	0.257	0.194	0.149	0.128	0.101	0.1	0.12	0.401	0.532	1
2005	0.703	0.399	0.242	0.176	0.169	0.144	0.119	0.108	0.138	0.183	0.461	0.763

Tabel diatas adalah data penjualan pada bulan ke-1 sampai dengan bulan ke-12. Data tersebut akan digunakan untuk memprediksi penjualan pada bulan ke-13 yaitu Januari 2005. Apabila diinginkan untuk melakukan prediski pada bulan ke-14 maka data yang dipakai sebagai training adalah data pada bulan-2 sampai dengan bulan ke-13. Nantinya akan terdapat 12 pola mewakili setiap bulan data yang akan diprediksi. Berikut ini adalah tabel contoh hasil penggunaan pola untuk melakukan prediksi pada tahun 2004-2005.

Tabel Pola untuk Prediksi (2004-2005)

Pola	Data Masukan(x1,x2,...,x12)							Target
1	0,77688	0,57124	0,25677	0,19418	0,14935	0,12804	0,10142	0,7033
	0,10142	0,1	0,12018	0,40123	0,53173	1		
2	0,57154	0,25677	0,19418	0,14935	0,12804	0,10142	0,39889	
	0,1	0,12018	0,40123	0,53173	1	0,7033		
3	0,25677	0,19418	0,14935	0,12804	0,10142	0,1	0,2424	
	0,12018	0,40123	0,53173	1	0,7033	0,39889		
4	0,19418	0,14935	0,12804	0,10142	0,1	0,12018	0,17626	
	0,40123	0,53173	1	0,7033	0,39889	0,2424		

5	0,14935 0,53173	0,12804 1	0,10142 0,7033	0,1 0,39889	0,12018 0,2424	0,40123 0,17626	0,16904
6	0,12804 1	0,10142 0,7033	0,1 0,39889	0,12018 0,2424	0,40123 0,17626	0,53173 0,16904	0,14433
7	0,10142 0,7033	0,1 0,2424	0,12018 0,2424	0,40123 0,17626	0,53173 0,16904	1 0,14433	0,11919
8	0,1 0,39889	0,39889 0,17626	0,40123 0,17626	0,53173 0,16904	1 0,14433	0,7033 0,11919	0,10786
9	0,12018 0,2424	0,40123 0,17626	0,53173 0,16904	1 0,14433	0,7033 0,11919	0,39889 0,10786	0,13845
10	0,40123 0,17626	0,53173 0,16904	1 0,14433	0,7033 0,11919	0,39889 0,10786	0,2424 0,13845	0,18299
11	0,53173 0,16904	1 0,14433	0,7033 0,11919	0,39889 0,10786	0,2424 0,13845	0,17626 0,18299	0,4605
12	1 0,14433	0,7033 0,11919	0,39889 0,10786	0,2424 0,13845	0,17626 0,18299	0,16904 0,4605	0,76328

C. Implementasi Algoritma Propagasi Balik

Pada bagian ini akan dijelaskan mengenai implementasi dari algoritma propagasi balik yang digunakan untuk melakukan prediksi. Penjelasan dari implementasi ini akan dijelaskan dalam *flowchart* yang mewakili implementasi dari algoritma tersebut.

Flowchart ini mewakili gambar dari bagan propagasi balik pada Gambar input dan output dari JST. Dimana pada bagan tersebut terdapat beberapa atribut yang cukup penting diantaranya adalah 12 node pada lapisan input, 5 node pada lapisan tersembunyi, dan 1 node pada lapisan output. Notasi node untuk masing-masing lapisan ini dinotasikan menjadi:

1. Input Node adalah (x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, x11, dan x12)
2. Node Tersembunyi adalah (z1,z2,z3,z4,dan z5)
3. Output Node adalah Yk

Dari masing-masing node pada input yang menghubungkan kedalam 5 node pada layer tersembunyi. Begitu juga dengan masing-masing node dari 5 node pada layer tersembunyi yang masuk kedalam output node. Garis yang menghubungkan dari input ke hidden layer dikodekan dengan V_{ij} . Sedangkan, garis yang menghubungkan antara layer tersembunyi dengan output node dikodekan dengan W_{kl} .

D. Proses Pelatihan

Pada proses pelatihan ini akan digunakan inisialisasi epoch dan max alpha. Epoch yang digunakan pada saat pelatihan adalah 5000 dan max alpha yang digunakan adalah 0,001. Apabila epoch yang digunakan kurang dari nilai tersebut, maka akan terjadi error rate dari tebakan lebih tinggi.

Setelah itu program akan melakukan algoritma propagasi balik untuk tiap epoch-nya. Nantinya, proses pelatihan ini akan berhenti apabila epoch mencapai nilai 5000 atau max alpha sudah mencapai angka 0,01. Pada proses pelatihan untuk data ini epoch akan berhenti pada nilai 4108 dengan nilai 0,0009995666502555893 dengan nilai alpha adalah 0,001.

E. Proses Testing

Setelah dilakukan proses pelatihan perangkat lunak akan memberikan notifikasi selesainya proses pelatihan dan menampilkan tampilan untuk proses testing. Nantinya pengguna harus memasukan nilai input mulai dari x1 hingga x12 untuk input data yang digunakan sebagai prediksi. Berikut ini adalah contoh input data yang digunakan untuk testing.

Tabel Input untuk Prediksi Tes Pertama

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12
0,12804	0,10142	0,1	0,12018	0,40123	0,53173	1	0,7033	0,39889	0,2424	0,17626	0,16904

Setelah proses input data mesin prediksi akan melakukan perhitungan matematis untuk melakukan prediksi. Prediksi ini akan menghasilkan nilai yang dihasilkan dari tabel pola pada bagian 3. Dari input tersebut dihasilkan nilai 0,14045058069725105, dimana nilai ini mendekati nilai dari pola ke-6. Pola ke-6 bernilai 0,14433 hasil ini cukup mendekati. Perbedaan nilai ini dikarenakan oleh nilai input yang dimasukan tidak sama persis dengan pola yang terdapat pada baris ke-6. Akan tetapi, nilai ini masih bisa diterima.

Proses uji coba kedua akan dicoba menggunakan data input x1 sampai dengan x12 yang lain. Berikut ini adalah tabel input yang digunakan untuk prediksi pada uji coba kedua.

Tabel Input untuk Prediksi Tes Kedua

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12
0,12804	0,10142	0,1	0,12018	0,40123	0,53173	1	0,7033	0,39889	0,2424	0,17626	0,16904

Dari perhitungan dari uji coba yang kedua didapatkan hasil dengan nilai 0,13647602156323518. Nilai ini mendekati nilai pada pola ke-6 dengan nilai error yang jauh lebih besar dari uji coba yang pertama. Akan tetapi hasil ini masih bisa ditoleransi mengingat range dari pola yang digunakan sudah memenuhi pola ke-6. Apabila digunakan input lain yang sama sekali tidak mirip dengan data pada pola, maka output dari tes ini menghasilkan nilai yang sangat jauh dari nilai pada tabel pola.

4. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari pembahasan pada bagian-bagian sebelumnya, maka dapat ditarik beberapa kesimpulan sebagai berikut:

1. Kecepatan Jaringan Saraf Tiruan dalam melakukan prediksi suatu input data dipengaruhi oleh beberapa hal yaitu model input yang digunakan sebagai pelatihan dan pemilihan bobot.
2. Pemilihan bobot sangat mempengaruhi nilai dari fungsi aktivasi. Apabila fungsi aktivasi memiliki nilai yang rendah, maka proses perubahan bobot pada setiap iterasi menjadi sangat kecil dan proses iterasi menjadi lebih panjang untuk melakukan adaptasi terhadap proses pelatihan.
3. Input data pada saat proses testing apabila ditemukan persamaan dengan pola, maka prediksi yang dilakukan akan semakin akurat.
4. Apabila input data pada saat proses testing diinputkan dengan data yang tidak mempunyai kemiripan dengan tabel pola, maka data tersebut tidak akan dikenali oleh JST ini.
5. Penambahan pola dalam JST perlu ditambahkan untuk meningkatkan akurasi dari prediksi yang dihasilkan.

B. Saran

Dari pembahasan pada bagian-bagian sebelumnya, penulis menyarankan perbaikan-perbaikan yang harus dilakukan untuk penelitian lebih lanjut adalah sebagai berikut.

1. Penambahan jumlah hidden unit akan sangat mempengaruhi kinerja JST dalam melakukan prediksi suatu input pada saat testing.

2. Penggunaan Jaringan Saraf Tiruan untuk melakukan prediksi kurang efektif. Hal ini disebabkan oleh variasi pola yang sangat beragam dan tidak mungkin untuk mengenali data penjualan secara akurat.
3. Kecepatan pelatihan dari tiap epoch perlu ditingkatkan apabila pada lapisan input ditambahkan beberapa input node yang baru.

REFERENSI

- [1] Anderson, Dave dan Mcneill. *Artificial Neural Networks Technology*. Kaman Sciences Corp 1992
- [2] Gunawan, 2007. Handout Mata Kuliah ANN Semester Genap 2006-2007. IKADO.
- [3] Kusuma, Sri. *Membangun Jaringan Saraf Tiruan Menggunakan MATLAB dan Excel Link*,. Graha Ilmu Jakarta, 2004.
- [4] Rao, Valluru. *C++ Neural Networks and Fuzzy Logic*. IDG Books Worldwide, Inc.
- [5] Smith, Kate dan Jatinder Gupta, 2002. *Neural Networks in Bussiness: Technique and Application*, Idea Group Publishing .