

Peramalan Penjualan Mobil Menggunakan Jaringan Syaraf Tiruan dan *Certainty Factor*

Fachrudin Pakaja, Agus Naba dan Purwanto

Abstrak—Prediksi penjualan adalah salah satu cara untuk meningkatkan laba perusahaan, peramalan diperlukan untuk menyelaraskan antara perbedaan waktu yang sekarang dan yang akan datang terhadap kebutuhan, Jaringan Syaraf Tiruan (JST) dapat mengaplikasikan dengan baik metode peramalan.

Pendekatan peramalan kuantitatif dengan metode *times series* akan menentukan nilai data masukan dari sekumpulan data serial atau berkala dari transaksi pada suatu jangka waktu tertentu. Data dibagi menjadi data pelatihan, pengujian dan validasi. Proses peramalan menggunakan metode *certainty factor* (CF) sebagai nilai pembanding pada bobot koreksi yang telah di latih dalam jaringan *backpropagation* untuk prediksi yang optimal. Simulasi program peramalan penjualan mobil honda tahun 2015 dengan variabel input data penjualan daerah 30,000 unit, penjualan *dealer* 25.000, penjualan tunai 25.000, CF = 0.5 dan kredit 19.000 menghasilkan ramalan penjualan sebanyak 29579 unit dengan target error 4,205 %.

Kata Kunci—Peramalan, *Time series*, *Certainty Factor*, JST, *Backpropagation*.

I. PENDAHULUAN

Penjualan merupakan salah satu indikator paling penting dalam sebuah perusahaan, bila tingkat penjualan yang dihasilkan oleh perusahaan tersebut besar, maka laba yang dihasilkan perusahaan itu pun akan besar pula sehingga perusahaan dapat bertahan dalam persaingan bisnis dan bisa mengembangkan usahanya. Prediksi penjualan/ *sales forecasting* adalah salah satu cara yang efektif untuk dapat meningkatkan laba perusahaan.

Data dan informasi penjualan sangat penting bagi perusahaan untuk merencanakan penjualan yang akan datang, misalnya: data pelanggan, jumlah kendaraan, harga mobil, suku cadang, jenis kendaraan dan yang tidak kalah pentingnya adalah kebijakan pemerintah dalam memberikan pajak kendaraan serta subsidi bahan bakar kendaraan.

Salah satu bidang dalam jaringan syaraf tiruan dapat diaplikasikan dengan baik untuk melakukan peramalan (*forecasting*). Runtut waktu data (*time series*) sering digunakan sebagai data input dan output untuk melakukan proses pelatihan.

Fachrudin Pakaja adalah mahasiswa magister Teknik Elektro di Universitas Brawijaya, Malang, Indonesia (Telp.085233050105; email igo_poonya@yahoo.com)

Agus Naba adalah dosen di Fakultas MIPA Universitas Brawijaya, Malang, Indonesia (Telp.08123301353). ; email anaba@ub.ac.id

Purwanto adalah dosen di Jurusan Teknik Elektro Universitas Brawijaya, Malang, Indonesia (Telp.08123262412); email purwanto@ub.ac.id

II. PERAMALAN DAN JARINGAN SYARAF TIRUAN

A. Peramalan

Peramalan adalah proses untuk memperkirakan berapa kebutuhan dimasa yang akan datang yang meliputi kebutuhan dalam ukuran kuantitas, kualitas, waktu dan lokasi yang dibutuhkan dalam rangka memenuhi permintaan barang ataupun jasa (Nasution, 1999). Peramalan permintaan merupakan tingkat permintaan produk-produk yang diharapkan akan terealisasi untuk jangka waktu tertentu pada masa yang akan datang. Pada dasarnya pendekatan peramalan dapat diklasifikasikan menjadi dua pendekatan, yaitu (Makridakis, *et.al.*, 1995) :

1. Pendekatan kualitatif
2. Pendekatan kuantitatif

B. Pola Data Peramalan Time Series

Ada 4 jenis pola data dalam peramalan (Makridakis, *et.al.*, 1995) yaitu :

1. *Trend* : Pola data tren menunjukkan pergerakan data cenderung meningkat atau menurun dalam waktu yang lama
2. *Seasonality* (musiman) : Pola data musiman terbentuk karena faktor musiman, seperti cuaca dan liburan.
3. *Cycles* (Siklus) : Pola data siklus terjadi jika variasi data bergelombang pada durasi lebih dari satu tahun dipengaruhi oleh faktor politik, perubahan ekonomi (ekspansi atau kontraksi) yang dikenal dengan siklus usaha.
4. *Horizontal/Stasionary/Random variation* : Pola ini terjadi jika data berfluktuasi di sekitar nilai rata-rata secara acak tanpa membentuk pola yang jelas seperti pola musiman, trend ataupun siklus.

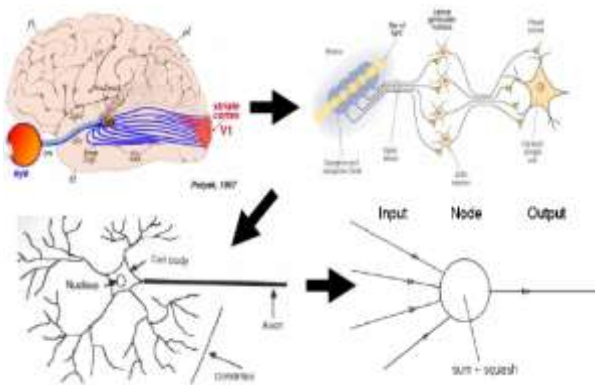
C. Jaringan Syaraf Tiruan (Artificial Neural Network)

Suatu jaringan saraf tiruan memproses sejumlah besar informasi secara paralel dan terdistribusi, hal ini terinspirasi oleh model kerja otak biologis.

Hecht-Nielsen (1988) mendefinisikan sistem syaraf buatan adalah : suatu struktur pemroses informasi yang terdistribusi dan bekerja secara paralel, yang terdiri atas elemen pemroses (yang memiliki memori lokal dan beroperasi dengan informasi lokal) yang diinterkoneksi bersama dengan alur sinyal searah yang disebut koneksi. Setiap elemen pemroses memiliki koneksi keluaran tunggal yang bercabang (*fan out*) ke sejumlah koneksi kolateral yang diinginkan (setiap koneksi membawa sinyal yang sama dari keluaran elemen pemroses tersebut). Keluaran

dari elemen pemroses tersebut dapat merupakan sebarang jenis persamaan matematis yang diinginkan. Seluruh proses yang berlangsung pada setiap elemen pemroses harus benar-benar dilakukan secara lokal, yaitu keluaran hanya bergantung pada nilai masukan pada saat itu yang diperoleh melalui koneksi dan nilai yang tersimpan dalam memori lokal.

Struktur pada gambar 1 adalah bentuk standar dasar satuan unit jaringan otak manusia yang telah disederhanakan. Jaringan otak manusia tersusun dari 10^{13} neuron yang terhubung oleh sekitar 10^{15} dendrite. Fungsi *dendrite* adalah sebagai penyampai sinyal dari neuron tersebut ke neuron yang terhubung dengannya. *Nucleus* merupakan inti dari suatu neuron, *axon* berfungsi sebagai saluran keluaran dari neuron, dan *synapsis* yang mengatur kekuatan hubungan antar neuron



Gambar 1. Struktur dasar jaringan syaraf tiruan

Jaringan neuron buatan terdiri atas kumpulan grup neuron yang tersusun dalam lapisan

- Lapisan input (*Input Layer*): berfungsi sebagai penghubung jaringan ke dunia luar (sumber data).
- Lapisan tersembunyi (*hidden Layer*): Suatu jaringan dapat memiliki lebih dari satu *hidden layer* atau bahkan bisa juga tidak memilikinya sama sekali.
- Lapisan Output (*Output Layer*): Prinsip kerja neuron-neuron pada lapisan ini sama dengan prinsip kerja neuron-neuron pada lapisan tersembunyi (*hidden layer*) dan di sini juga digunakan fungsi *Sigmoid*, tapi keluaran dari *neuron* pada lapisan ini sudah dianggap sebagai hasil dari proses

Secara umum, terdapat tiga jenis *neural network* yang sering digunakan berdasarkan jenis *network*-nya, yaitu :

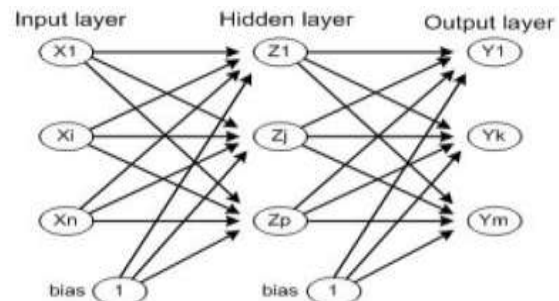
1. *Single-Layer Neural* adalah jaringan syaraf tiruan yang memiliki koneksi pada inputnya secara langsung ke jaringan output.
2. *Multilayer Perceptron Neural Network* adalah jaringan syaraf tiruan yang mempunyai layer yang dinamakan "*hidden*", ditengah layer input dan output. *Hidden* ini bersifat variabel, dapat digunakan lebih dari satu *hidden layer*.
3. *Recurrent Neural Networks Neural network* adalah jaringan syaraf tiruan yang memiliki ciri, yaitu adanya koneksi umpan balik dari *output* ke *input*.

D. Jaringan Syaraf Tiruan Propagasi Balik (*Neural Network Backpropagation*)

Backpropagation merupakan salah satu metode pelatihan

dari Jaringan Syaraf Tiruan. *Backpropagation* menggunakan arsitektur *multilayer* dengan metode pelatihan *supervised training*.

Model Propagasi Balik memiliki beberapa unit yang ada dalam satu atau lebih layer tersembunyi. Gambar 2.3. adalah arsitektur model Propagasi Balik dengan n buah masukan (ditambah satu bias), sebuah layer tersembunyi yang terdiri dari p unit (ditambah sebuah bias) serta m buah unit keluaran.



Gambar 2. Arsitektur Model Propagasi Balik

E. Backpropagation untuk Peramalan

Selama tahap *feed-forward* (maju), setiap unit masukan (X_i) menerima sinyal masukan dan mengirim sinyal ini ke setiap unit tersembunyi Z_1, \dots, Z_p . Setiap unit tersembunyi menghitung aktifasinya dan mengirim sinyalnya (z_j) ke setiap unit keluaran. Setiap unit keluaran (Y_k) menghitung aktifasinya (y_k) untuk menunjukkan respon jaringan terhadap pola masukan yang diberikan.

Selama pelatihan, untuk setiap unit keluaran dibandingkan aktifikasi y_k dengan targetnya t_k untuk menentukan galat antara pola masukan dengan unit keluaran tersebut. Setelah didapat galat, faktor δ_k ($k=1, \dots, m$) dihitung δ_k yang digunakan untuk mendistribusikan galat pada unit keluaran Y_k kembali ke seluruh unit pada lapis sebelumnya (unit tersembunyi yang terhubung dengan Y_k). Kemudian galat ini dipakai untuk mengubah bobot antara keluaran dengan lapisan tersembunyi. Dengan cara yang sama, faktor δ_j ($j=1, \dots, p$) dihitung untuk setiap unit Z_j . Faktor δ_j digunakan untuk mengubah bobot antara lapisan tersembunyi dengan lapisan input.

Setelah seluruh faktor δ ditentukan, bobot untuk seluruh lapisan langsung disesuaikan. Penyesuaian bobot W_{jk} (dari unit tersembunyi Z_j ke unit keluaran Y_k) didasarkan pada faktor δ_k dan aktifasi dari unit Z_j , yaitu z_j . Penyesuaian bobot v_{ij} (dari unit masukan X_i ke unit tersembunyi Z_j) adalah didasarkan pada faktor δ_j dan aktifasi unit masukan x_i .

Fungsi aktifasi yang biasanya dipakai untuk melatih JST propagasi balik adalah fungsi sigmoid, baik biner maupun bipolar. Berikut algoritma pelatiahannya (Fausett, Laurene, 1994) :

- **Langkah 0.** Inisialisasi bobot (menentukan suatu nilai random kecil)
- **Langkah 1.** Selama kondisi berhenti bernilai salah, dilakukan langkah 2-9
- **Langkah 2.** Untuk setiap pasangan pelatihan, dilakukan langkah 3-8 (*Feedforward*)
- **Langkah 3.** Setiap unit masukan ($X_i, i=1, \dots, n$)

menerima sinyal masukan x_i dan mengirim sinyal inike seluruh unit pada lapisan berikutnya (lapisan tersembunyi).

- **Langkah 4.** Untuk setiap unit tersembunyi (Z_j , $j = 1, \dots, p$), sinyal masukan terboboti di jumlahkan ($z_in_j = v_{oj} +$) dan diterapkan fungsi aktifasi untuk menghitung keluarannya ($z_j = f(z_in_j)$) dan mengirim sinyal ini ke seluruh unit lapisan berikutnya (lapisan keluaran).
- **Langkah 5.** Untuk setiap unit keluaran (Y_k , $k = 1, \dots, m$) sinyal masukan terboboti di masukkan ($y_in_k = w_{oj} +$)

Propagasi balik dari galat :

- **Langkah 6.** Untuk setiap unit keluaran (Y_k , $k = 1, \dots, m$) menerima sebuah pola target yang bersesuaian dengan pola masukan, dihitung galatnya ($\delta_k = (t_k - y_k) f'(y_in_k)$) dan dihitung koreksi bobotnya ($\Delta w_{jk} = \alpha \delta_k z_j$) dan dihitung juga koreksi biasnya ($\Delta w_{ok} = \alpha \delta_k$) dan mengirimkan δ_k ke unit pada lapisan sebelumnya.
- **Langkah 7.** Untuk setiap unit yang tersembunyi (Z_j , $j = 1, \dots, p$), fungsi delta dijumlahkan ($\delta_in_j =$) kemudian dikalikan dengan fungsi aktifasinya untuk menghitung galatnya ($\delta_j = \delta_in_j f'(z_in_j)$), dihitung koreksi bobotnya ($\Delta v_{ij} = \alpha \delta_j x_i$) dan koreksi biasnya ($\Delta v_{oj} = \alpha \delta_j$).
- **Langkah 8.** Untuk setiap unit keluaran (Y_k , $k = 1, \dots, m$) bias dan bobotnya diubah ($j = 1, \dots, p$) ($w_{jk}(\text{new}) = w_{jk}(\text{old}) + \Delta w_{jk}$). Untuk setiap unit tersembunyi (Z_j , $j = 1, \dots, p$) bobot dan biasnya diubah ($i = 1, \dots, p$) ($v_{ij}(\text{new}) = v_{ij}(\text{old}) + \Delta v_{ij}$).

F. Certainty Factor / Faktor Kepastian

Certainty factor (CF) diperkenalkan oleh *Shortlife Buchanan* dalam pembuatan MYCIN. CF merupakan nilai parameter klinis yang diberikan MYCIN untuk menunjukkan besarnya kepercayaan. CF memperkenalkan konsep *belief* (keyakinan) dan *disbelief* (ketidakyakinan). Konsep ini kemudian diformulasikan dalam rumusan dasar sebagai berikut (Giarattano dan Riley, 1994) :

$$CF(H, E) = MB(H, E) - MD(H, E) \quad (1)$$

III. DESAIN SISTEM

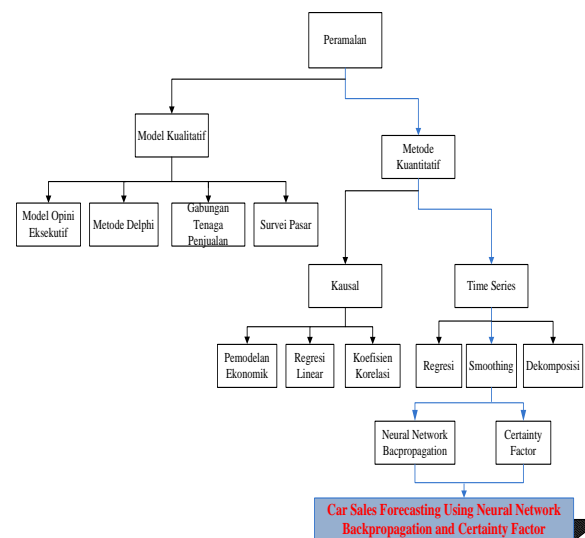
Peramalan penjualan untuk mendapatkan keuntungan maksimal menggunakan metode propagasi balik dan faktor kepastian. Kondisi ini dibangun melihat kondisi pasar dimana semakin besarnya persaingan antara *dealer* mobil dan makin maraknya jenis kendaraan baru maupun lama yang beredar di Indonesia.

A. Desain Aplikasi Jaringan Syaraf Tiruan

Langkah-langkah membangun struktur jaringan untuk peramalan sebagai berikut :

1. Transformasi data: dilakukan agar terjadi kestabilan taburan data dicapai dan juga untuk menyesuaikan nilai data dengan *range* fungsi aktivasi yang digunakan dalam jaringan. Data ditransformasikan ke interval (0.1)
2. Pembagian data dilakukan dengan membagi data penelitian menjadi data pelatihan dan pengujian

3. Perancangan arsitektur jaringan yang optimum
4. Memilih dan menggunakan arsitektur jaringan yang optimum
5. Pemilihan jaringan optimum dan penggunaannya untuk peramalan.



Gambar 3. Flowchart Metodologi Penelitian

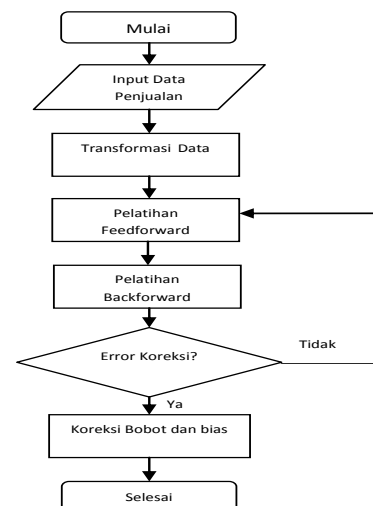
B. Pengujian

Dari hasil perancangan sistem sebelumnya akan dilakukan pengujian data untuk menguji tingkat validasi dan hasil proses dalam sistem. Hal ini dilakukan dengan memasukkan seluruh data yang diperoleh dari proses pengumpulan data ke dalam sistem. Dilakukan proses transformasi data untuk dilakukan pelatihan untuk menguji nilai keakuratan data dan mencari jaringan yang optimal dari JST.

IV. APLIKASI SISTEM PERAMALAN

A. Perencanaan Sistem

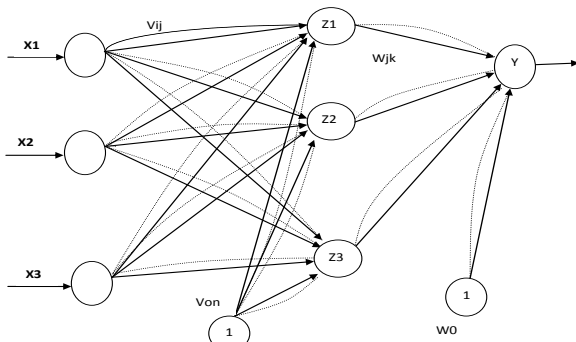
Arsitektur jaringan syaraf tiruan yang dibangun terdiri dari beberapa lapisan, yaitu : lapisan masukan (layer input), satu lapisan tersembunyi (layer hidden), dan lapisan keluaran (layer output). Penghubung setiap lapisan adalah bobot.



Gambar 4. Alur flowchart jaringan syaraf tiruan

Proses pembelajaran sendiri diawali dari proses *feedforward*, dan kemudian dilanjutkan pada proses *backpropagation*. Setelah proses *backpropagation*, akan dilakukan pengecekan apakah nilai target *error* telah dicapai, jika target *error* telah dicapai, maka proses pembelajaran selesai, yang menghasilkan koreksi dari bobot jaringan. Jika tidak maka akan kembali ke proses *feedforward*. Hal ini akan terus berlangsung sampai menemukan nilai *epoch* maksimum

Unit input dilambangkan dengan variabel x , hidden variabel z dan nilai output dilambangkan dengan variabel y , sedangkan nilai bobot antara x dan z dilambangkan dengan variabel v dan nilai bobot antara z dan y dilambangkan dengan variabel w .



Gambar 5. Rancangan jaringan syaraf tiruan *backpropagation*

B. Prosedur Perancangan

Terdapat 2 macam prosedur perancangan, yaitu : prosedur pelatihan (*training*) dan validasi.

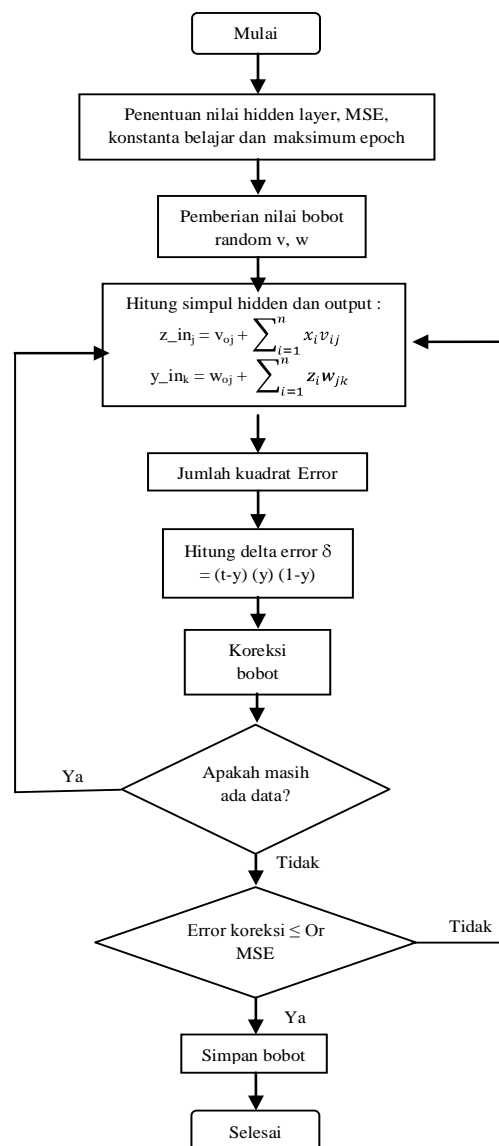
1. Prosedur *training* adalah prosedur untuk melakukan pembelajaran terhadap pola-pola yang akan dikenali. Proses ini dilakukan dengan menggunakan data *training*. Proses ini berhenti jika MSE lebih kecil dari *error* yang ditetapkan atau *epoch* yang telah ditentukan *user* telah tercapai sehingga didapatkan bobot-bobot neuron yang diharapkan. *Epoch* (iterasi) adalah satu set putaran vector-vektor pembelajaran. Beberapa *epoch* diperlukan untuk pembelajaran sebuah *backpropagation* sehingga kesalahan mendekati 0 (nol).
2. Prosedur validasi dengan menggunakan faktor kepastian. Proses ini menggunakan bobot JST yang diperoleh dari proses *training* untuk menguji data testing yang ada, setelah data dilakukan prosedur peramalan, hasil dihitung dengan faktor kepastian untuk mengetahui tingkat kepercayaan terhadap data yang diolah.

C. Penyusunan Pola

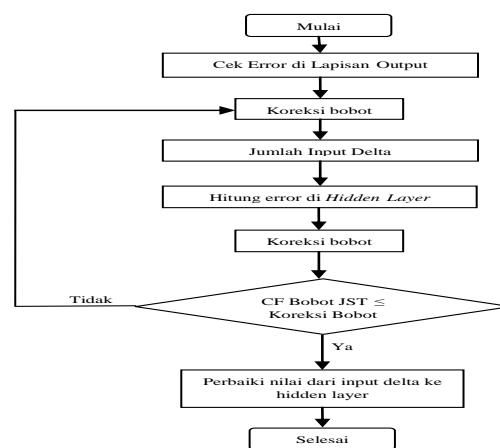
Penyusunan pola data peramalan dengan memasukkan data-data literatur yang digunakan. Data literatur yang digunakan adalah laporan data penjualan mobil secara nasional yang di peroleh dari data gabungan industri kendaraan Indonesia (GAIKINDO), sedangkan untuk jumlah penjualan *dealer* menggunakan data yang bersumber dari astra auto 2000 cabang Depok

Data yang telah di input selanjutnya akan dilakukan proses transformasi agar terjadi kestabilan taburan data dicapai dan juga untuk menyesuaikan nilai data dengan *range* fungsi aktivasi *sigmoid* biner yang digunakan dalam

jaringan. Dari data transformasi ini data terkecil menjadi 0.1 dan data terbesar menjadi 0.9.



Gambar 6. Proses *training* JST *backpropagation*



Gambar 7. Diagram alur peramalan proses JST

Laporan data penjualan *dealer* untuk wilayah kota Depok dan sekitarnya periode waktu 2005 sampai 2010.

TABEL1
PENJUALAN KENDARAAN DOMESTIK INDONESIA PERIODE TAHUN
2005 – 2010 DALAM KATEGORI MEREK KENDARAAN

Tahun	Toyota	Daihatsu	Mitsubishi	Suzuki	Honda	Nissan
2005	182.765	48.762	89.158	87.274	53.750	10.551
2006	123.896	33.021	47.023	44.760	30.000	4.006
2007	150.631	51.957	61.547	58.095	40.000	19.030
2008	211.909	78.041	87.524	73.067	52.500	31.879
2009	186.687	77.513	61.735	44.689	39.570	81.338
2010	280.680	118.591	108.483	71.210	21.440	37.242

Sumber : Gaikindo

TABEL2
DATA PENJUALAN DEALER UNTUK WILAYAH
DEPOK UNTUK PERIODE TAHUN 2005 SAMPAI 2010

Merk Mobil	2005	2006	2007	2008	2009	2010
Toyota	8.955	5.799	6.642	8.490	8.577	11.752
Daihatsu	2.855	1.864	2.625	4.035	3.693	118.591
Mitsubishi	6.953	2.867	1.337	1.480	987	108.483
Suzuki	2.531	1.534	3.051	3.619	2.071	71.210
Honda	2.644	876	1.963	2.333	1.697	81.338
Nissan	335	145	860	1.276	1.010	37.242

Sumber : Astra 2000 Depok



Gambar 8. Tampilan program data yang di inputkan ke sistem



Gambar 9. Tampilan program transformasi data

D. Pelatihan Sistem

Proses pelatihan data dilakukan dengan jumlah data yang digunakan sebanyak 90% dari data sebaran dan melakukan percobaan tersebut berulang ulang dengan data input yang berbeda. Parameter yang digunakan tetap dan telah di *setting* dalam program

1. Fungsi aktivasi *sigmoid* biner
2. Konstanta belajar (α) = 0.1

3. Lapisan Tersembunyi = 6

4. Besarnya galat = 0.1

Dari percobaan yang telah dilakukan pada jumlah iterasi ke 8520 diperoleh jumlah target error (tse) = 0.13080115458486 dengan nilai bobot output seperti pada table nilai output pelatihan.



Gambar 10. Tampilan proses pelatihan data

TABEL 3
NILAI BOBOT OUPUT PELATIHAN BERDASARKAN
MERK KENDARAAN

Suzuki	Mitsubishi	Toyota	Honda	Daihatsu	Nissan
0.1903	-0.2746	-0.1884	-0.1883	-0.1884	-0.1908

E. Pengujian dan Validasi Sistem

Pada pengujian validasi ini terdapat 3 proses pelatihan yaitu menjalankan program pelatihan dengan mengubah nilai-nilai masukan dari parameter pelatihan. Nilai parameter diacak dengan tujuan untuk mencari perbandingan nilai dan mencari nilai optimal dari bobot.

1. Pengujian dengan mengubah nilai dari parameter konstanta belajar, nilai *hidden* layer 6 dan besarnya galat 0.1 tampak pada tabel 4.

TABEL 4
PENGUJIAN DAN VALIDASI 1

Konstanta Belajar	Jumlah Iterasi	TSE
0.05	4758	0.0363271
0.07	8982	0.4868024
0.2	1854	0.0448234
0.5	3438	0.0459183

2. Pengujian dengan mengubah nilai parameter lapisan tersembunyi, nilai konstanta belajar tetap = 0.1 dan besarnya galat tetap 0.1 tampak pada tabel 5

TABEL 5
PENGUJIAN DAN VALIDASI 2

Lap.Tersembunyi	Jumlah Iterasi	TSE
2	72	0.024050
4	2052	0.011262
7	3174	0.012707
9	3900	-0.036728

3. Pengujian dengan mengubah nilai parameter dari besar galat, nilai konstanta belajar tetap = 0.1 dan jumlah lapisan tersembunyi = 6 tampak pada tabel 6

TABEL 6
PENGUJIAN DAN VALIDASI 3

Besar Galat	Jumlah Iterasi	TSE
0.05	12150	-0.059183
0.09	8124	0.043311
0.5	6	0.302812
0.9	6	-0.045918

F. Proses Peramalan

Data masukan di input secara manual dan acak untuk mencari nilai maksimal dari peramalan dengan menggunakan CF sebagai pembanding dari nilai *target error* target peramalan yang diinginkan.

Contoh penerapan data pada program :

1. Simulasi tahun 2015
2. Tipe produk : Honda
3. Penjualan Tertinggi daerah : 30.000 unit
4. Jumlah Penjualan dealer :25.000 unit
5. Jumlah Penjualan tunai : 19.000 unit
6. Jumlah penjualan kredit : 16.000 unit
7. Keakutan CF : 0.5



Gambar 11. Tampilan program peramalan honda

Dari *input* data yang dimasukkan di peroleh hasil peramalan pada tahun 2015 akan terjual mobil Honda sebanyak 29579 unit dengan nilai *target error* sebesar 4,205%.

V. KESIMPULAN

A. Kesimpulan

Berdasarkan uraian pada bab-bab sebelumnya dan berdasarkan hasil simulasi, maka pada penelitian ini dapat ditarik beberapa kesimpulan sebagai berikut :

1. Metode Jaringan Syaraf Tiruan mempunyai sifat yang adaptif yaitu jaringan berusaha mencapai kestabilan data untuk mencapai nilai *output* yang diharapkan.

2. Kemampuan Jaringan Syaraf Tiruan dapat diterapkan pada bentuk fungsi hubungan antara variabel-variabel bebas terhadap variabel terikat walaupun hubungannya tidak diketahui dengan baik atau sulit diketahui.
3. Penentuan parameter-parameter jaringan sangat berpengaruh terhadap lamanya proses pelatihan.
4. Jumlah iterasi tidak dapat ditentukan oleh besarnya ketelitian pengenalan pola yang diinginkan tetapi ditentukan oleh parameter-parameter jaringan yang digunakan, kondisi awal dari jaringan dan karakteristik data-data masukan.
5. *Certainty factor* digunakan sebagai nilai pembanding dengan koreksi bobot untuk menghasilkan peramalan yang optimal.
6. Proses peramalan penjualan mobil yaitu dengan memasukkan data perkiraan penjualan di masa depan, untuk di olah menggunakan *JST backpropagation* untuk menghasilkan data yang diinginkan.

B. Saran

Penelitian ini sebaiknya dilakukan test peramalan secara langsung oleh perusahaan untuk mengetahui kemampuan model jaringan syaraf tiruan yang telah dihasilkan pada tahap meramalkan penjualan mobil di masa yang akan datang.

Pakar penjualan sangat diperlukan dalam melakukan analisis *certainty factor*, sehingga untuk pengembangan penelitian ini dapat melibatkan keahlian seorang pakar untuk membantu memberikan kemungkinan kepastian dan keakuratan hasil penjualan sehingga proses peramalan ini dapat berjalan dengan optimal.

DAFTAR PUSTAKA

- [1] Zheng; Zhong. 2011. **Time series forecasting using a hybrid RBF neural network and AR model based on binomial smoothing**. World Academy of Science, Engineering and Technology 75.
- [2] Shih; Chung. 2008. **The Development Of A CFM Hybrid Artificial Sale Forecasting Model**. International Journal of Electronic Business Management, Vol. 6, No. 4.
- [3] Setiawan. 2008 . **Prediksi Harga Saham Menggunakan Jaringan Syaraf Tiruan Multilayer Feedforward Network dengan Algoritma Bacpropagation**. Konferensi Nasional Sistem dan Informatika, Bali (KNS&I08-020).
- [4] Rohman; Fauziah. 2008. **Rancang Bangun Aplikasi Sistem Pakar Untuk Menentukan Jenis Gangguan Perkembangan Pada Anak**. Media Informatika, Vol. 6, No. 1, Juni 2008, 1-23
- [5] Dhaneswara; V,S. Moertini. 2004. **Jaringan Saraf Tiruan Propagasi Balik Untuk Klasifikasi Data**. FMIPA Unpar, v1ol.9 no.3, Nov. 2004
- [6] Suhari, 2010. **Jaringan Syaraf Tiruan : Aplikasi Pemilihan Merek**, Jurnal Teknologi Informasi DINAMIK Volume XV No.2, Juli 2010 : 90-95
- [7] Suprianto, 2004. **Penerapan Jaringan Syaraf Tiruan Untuk Memprediksi Harga Saham**, Skripsi-FTIK Universitas Komputer Indonesia Bandung. 2004
- [8] [http://id.m.wikipedia.org/wiki/Jaringan_Syaraf_Tiruan_\(Artificial_Neural_Networks\)](http://id.m.wikipedia.org/wiki/Jaringan_Syaraf_Tiruan_(Artificial_Neural_Networks)), 2 – 02 -12, 21.43.