

PREDIKSI INFLASI INDONESIA DENGAN MODEL ARTIFICIAL NEURAL NETWORK

Diah Wahyuningsih
Idah Zuhroh
Zainuri

Fakultas Ekonomi Universitas Muhammadiyah Malang

ABSTRACT

This research examines and analyzes the use of Artificial Neural Networks (ANN) as a forecasting tool. Specifically a neural network's ability to predict future trends of inflation is tested. Accuracy is compared against a traditional forecasting method, multiple linear regression analysis. Finally, the probability of the model's forecast being correct is calculated using conditional probabilities. While only briefly discussing neural network theory, this research determines the feasibility and practicality of using neural networks as a forecasting tool for inflation in Indonesia. This study builds upon the work done by Edward Gately in his book Neural Networks for Financial Forecasting. This research validates the work of Gately and describes the development of a neural network that achieved an 86 percent probability of predicting an inflation rise, while multiple regression analysis is only to predict inflation that achieved a 16%. It was concluded that neural networks do have the capability to forecast inflation and, if properly trained, we could benefit from the use of this forecasting tool.

Keywords: *neural networks, inflation, time series analysis, forecasting, artificial intelligence*

A. LATAR BELAKANG

Tujuan dari penelitian ini adalah meramalkan inflasi di Indonesia menggunakan model *Artificial Neural Networking* (ANN) dan membandingkan-nya dengan capaian spesifikasi model linier tradisional dalam hal ini analisis regresi. *Neural network* dapat dianggap sebagai model “*black box*”, sebab kadang-kadang sulit untuk memberikan arti ekonomi atas hubungan yang diperkirakan muncul dalam model. Meskipun demikian, model ini sudah membuktikan kemanfaatannya sebagai alat peramalan di bidang ilmu pengetahuan (Kartopolous, 1996). Model *Neural Network*, dalam kaitan ketiadaan struktur mereka, dapat dipandang terbaik sebagai indikator model. Seperti halnya dalam kajian ini, hendaknya dapat dipandang sebagai suatu kontribusi dalam pengembangan pengetahuan yang relatif baru, khususnya sebagai model riset di Indonesia yang dapat, khususnya dalam kemampuannya memprediksi tingkat inflasi. *Neural Networks* secara umum menyediakan hubungan non linier antar variabel.

Sesungguhnya tingkat inflasi merupakan indikator *performance* ekonomi makro, mengingat inflasi akan memberikan dampak perekonomian secara menyeluruh, yang di negara manapun merupakan fenomena moneter (Gregory, 2000). Tingkat inflasi tidak dapat dianggap remeh dalam sistem perekonomian suatu negara dan pelaku bisnis pada umumnya. Jika inflasi dapat diramalkan dengan akurasi yang tinggi, tentunya dapat dijadikan dasar pengambilan kebijakan pemerintah dalam mengantisipasi aktivitas ekonomi di masa depan.

Kajian yang dilakukan terkait dengan pengujian model ANN, dirangkai sebagai upaya untuk menjawab beberapa pertanyaan berikut:

- Apakah model ANN secara akurat mampu meramalkan inflasi di Indonesia.
- Jika dibandingkan, apakah model ANN mampu meramalkan inflasi lebih baik dibanding dengan model analisis regresi berganda.

Neural Networks: Suatu Penjelasan Ringkas

ANN lahir dari usaha memodelkan otak manusia karena manusia dianggap sebagai sistem yang paling sempurna. Berbagai usaha memodelkan otak manusia telah dilakukan dan memunculkan tiga golongan model (Affandi, 1994). Pertama, golongan pertama meniru pola manusia dalam mengambil keputusan. Seperangkat diinputkan dalam otak mesin atau komputer, sehingga komputer dapat mengambil keputusan sesuai dengan pengetahuan yang sesuai dengan input (“pengetahuan”) yang diberikan. Golongan ini disebut sebagai sistem pakar (*expert system*). Kedua, golongan berikutnya menirukan cara kerja manusia yang tidak pernah dilakukan dalam variabel tegas (*crisp*). Semua variabel yang diolah dalam otak manusia bersifat samar (*fuzzy*). Dengan menggabungkan variabel samar dengan sistem pakar maka lahirlah *fuzzy logic*. Ketiga, golongan berikutnya lahir dari usaha memodelkan sel syaraf. Oleh karena itu disebut sebagai ANN (*artificial neural network*).

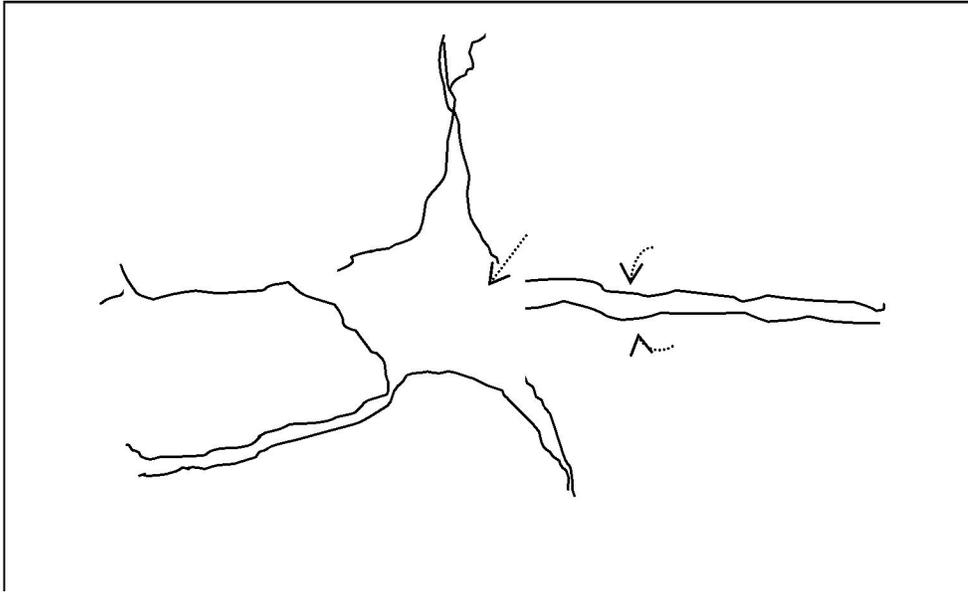
Model ANN pertama kali dikenalkan oleh Mc. Culloh dan Pitts sebagai komputasi aktivitas syaraf. Hasil karyanya kemudian menjadi arah bagi penelitian di bidang ini pada masa berikutnya. Hebb mempostulatkan bahwa neuron berhubungan satu sama lain pada pola terorganisasi secara mandiri. Hubungan secara kontinyu berubah ketika sebuah organisasi mempelajari suatu tugas baru, yaitu penjelasan mengenai model syaraf biologis.

Pada tahun 1958, Rosenblat, Widrow, dan Hoff menemukan aturan pembelajaran untuk pertama kalinya pada perceptron. Minski dan Papert (1969), mengemukakan bahwa *perceptron* sangat terbatas penggunaannya, yakni hanya sebagai metode perhitungan pada kehidupan nyata. Bernard Widrow menemukan unsur neural sederhana yang hampir sama dengan perceptron yang dinamakan ADALINE (*adaptive linear neuron*), dan jaringan multi layernya dikenal sebagai MADALINE (*multiple adalines*). Berikutnya Widrow juga mengembangkan prosedur pembelajaran terawasi yang dikenal sebagai *Least Mean Square* (LMS) atau *Widrow Hoff Learning Method*. Pada era selajutnya, ANN berkembang sedemikian hingga ditemukan berbagai macam metode dan aturan pembelajaran.

Untuk kepentingan aplikasi, manusia memilih fungsi-fungsi dan hubungan antar neuron tidak terlalu terikat oleh jaringan biologis yang nyata. Hal ini mempercepat penemuan-penemuan teknik komputasi berbasis ANN yang handal. Sebuah ANN yang berorientasi pada aplikasi memiliki tiga karakteristik. Pertama, bersifat adaptif. Artinya, ANN mampu mengubah parameter dan struktur dirinya berdasarkan masukan yang diberikan, serta menanganikan masukan yang sebelumnya belum pernah dikenal sebelumnya. Kedua, merupakan pemrosesan non linear, fungsi aktivasi merupakan unit non linear dari ANN. Ketiga, merupakan pemrosesan paralel, seperti halnya sistem syaraf real jutaan neuron yang dimilikinya bekerja secara paralel, sehingga masing-masing melakukan proses secara bersamaan atau simultan.

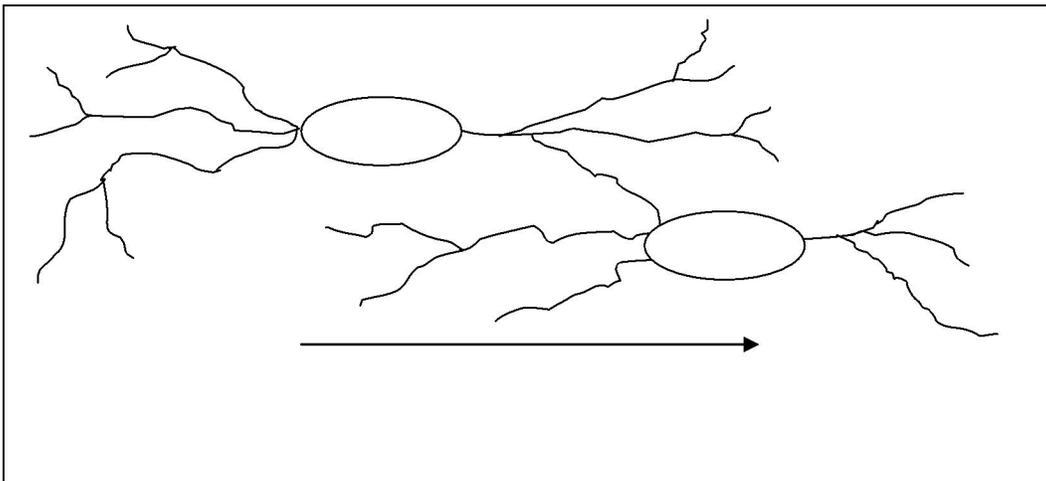
Neuron Biologis

ANN merupakan jaringan yang dibuat dengan meniru jaringan syaraf manusia dengan diilhami oleh struktur dan cara kerja otak dan sel syaraf manusia. Dalam otak manusia mengandung kurang lebih 10^{11} neuron dan sekitar 10^4 hubungan perneuron. Sebuah neuron mengandung *soma* (badan sel), *axon* (pengirim sinyal), dan *dendrite* (penerima sinyal). Yang menghubungkan antara satu neuron dan neuron yang lain adalah *synapse* (sinapsis). *Dendrite* berfungsi sebagai penerima sinyal dari neuron dari neuron lain melalui sinapsis. Sedangkan *axon* berfungsi meneruskan sinyal ke ujung serat yang berhubungan dengan sinapsis. Gambar neuron biologis dapat dilihat pada Gambar 1.



Gambar 1. Neuron Biologis

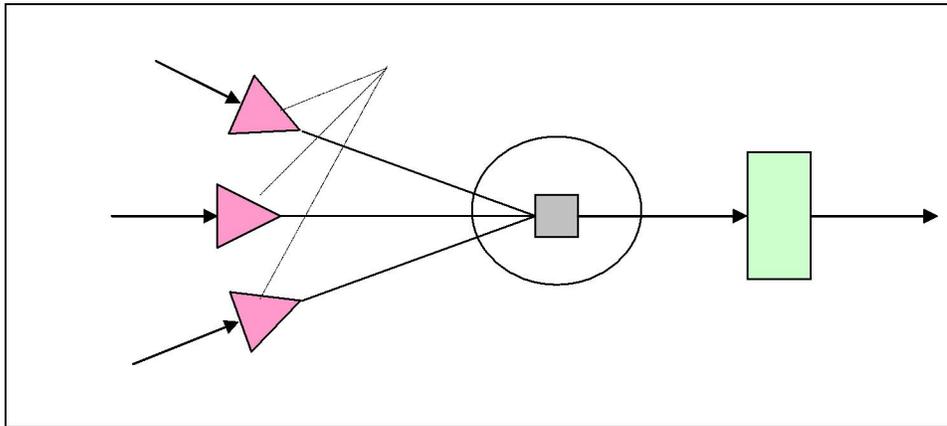
Sebuah neuron akan bereaksi apabila potensial listrik mencapai satu batasan tertentu (threshold). Cara kerja dari neuron adalah dengan menjumlahkan sinyal yang masuk melalui dendrite yang berasal dari axon neuron lain yang dikalikan dengan pembobot snapsis. Proses pembelajaran terjadi dengan perubahan pada sinapsis. Ada dua jenis sinapsis yang bersifat membangkitkan (*exite*) dan sinapsis yang bersifat menghambat (*inhibit*). Sinyal yang masuk dijumlahkan oleh badan sel dan dikonversi oleh fungsi aktivasi tertentu, sehingga menghasilkan sinyal pemicu yang dialirkan ke neuron melalui akson. Skema fisiologis neuron dapat dilihat pada Gambar 2.



Gambar 2. Skema Fisiologis Neuron

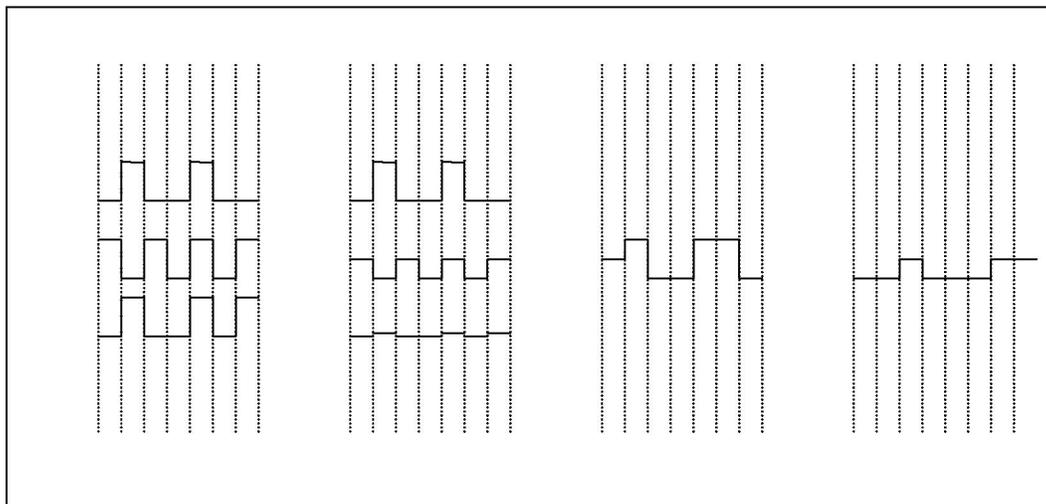
B. REPRESENTASI MATEMATIS DARI NEURON

Model matematik orde pertama dari neuron dapat dilihat pada Gambar 3.



Gambar 3. Representasi Matematis dari Neuron

Hubungan (*connection*) yang masuk pada neuron digambarkan dengan garis input dengan bobot tertentu. Neuron hanya melakukan akumulasi dan memberikan nilai pembatas (*threshold*) untuk pulsa yang datang dari input. Jika sebuah pulsa datang dari suatu hubungan, maka pulsa tersebut akan dilipatkan nilainya dengan suatu nilai yang disebut dengan bobot dari hubungan yang menentukan kepentingan dari hubungan tersebut (identik dengan besar kecilnya ukuran dendrit biologis). Nilai dari hubungan-hubungan diakumulasikan menjadi nilai overall unit aktivasi, melewati nilai batas tertentu, dan akan mengeluarkan pulsa jika nilai overall mencapai nilai batas tersebut. Keluaran dari tahap threshold akan menjadi masukan dan neuron yang lain, dan terbentuklah suatu jaringan syaraf yang lengkap.



Gambar 4. Operasi dari Model Matematis

Gambar 4 menunjukkan bobot dari, masing-masing input adalah 1; 0,5; dan 0,1; serta nilai threshold sebesar 2 Berdasarkan penjelasan ini dapat digambarkan operasi dan model matematika di atas. Fungsi operasi di atas diaplikasikan pada neuron buatan dan dikenal sebagai Threshold Logic Unit (TLU) dikemukakan oleh McCulloch and Pitts (1943). Berikut ini adalah gambaran dari konsep yang dikemukakan oleh McCulloch and Pitts dan disempurnakan oleh Rosenblatt (1958).

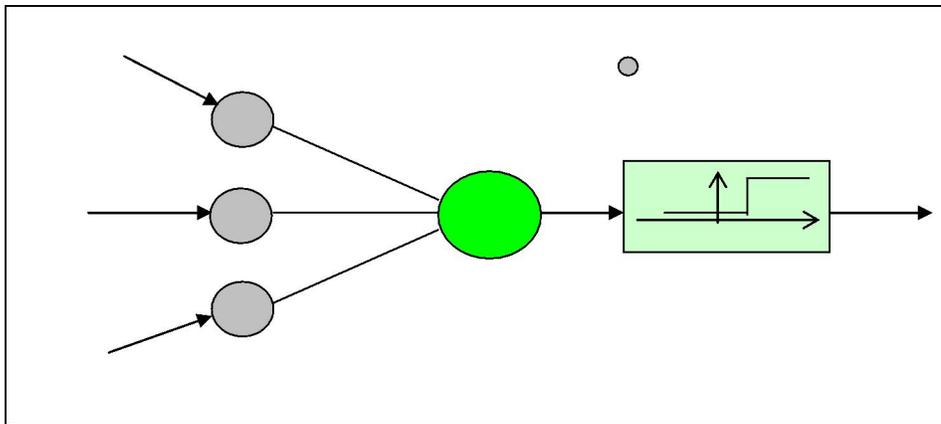
Elemen dasar yang membangun sebuah ANN adalah sebuah simpul (unit). Simpul berfungsi untuk mengubah sinyal masukan menjadi sebuah keluaran. Model ini mempunyai masukan berupa x_1, x_2, \dots, x_n , bobot adalah w_1, w_2, \dots, w_n . Sinyal berharga 0 dan 1 yang merupakan nilai boolean (hal ini sesuai dengan contoh gerbang logika dalam rangkaian elektronika).

Nilai aktivasi (α) adalah sebagai berikut ini:

$$\alpha = w_1x_1 + w_2x_2 + \dots + w_nx_n \quad \text{atau} \quad \alpha = \sum_{i=1}^n w_ix_i \quad (1)$$

Keluaran dari y diberikan dengan membandingkan nilai aktivasi dengan nilai batas (threshold) sebagai berikut ini:

$$Y = \begin{cases} 1 & \text{if } \alpha \geq \theta \\ 0 & \text{if } \alpha < \theta \end{cases} \quad (2)$$

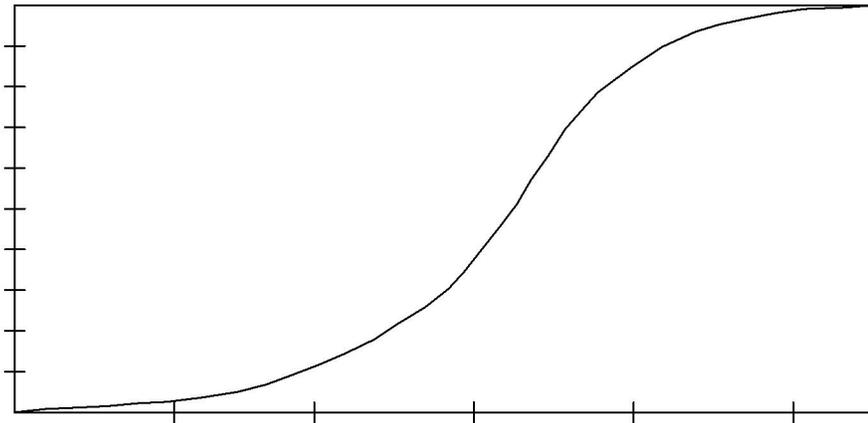


Gambar 5. Gambar dari TLU

Threshold (θ) seringkali bernilai 0, fungsi threshold sering disebut sebagai fungsi step atau pembatas keras. Cara kerja dari TLU adalah menjumlahkan seluruh masukan setelah diberi pembobot dan menyesuaikan hasil penjumlahan ini ke dalam sebuah fungsi aktivasi yang berfungsi untuk mengubah suatu nilai tidak terbatas (*unbounded*) menjadi nilai yang terbatas (*bounded*) atau dikenal dengan fungsi pemampat. Pada model asli Rosenblatt fungsi yang digunakan adalah fungsi pembatas keras seperti pada Gambar 4.

Jika dianalogikan dengan neuron biologis, munculnya aksi potensial disimbolkan dengan nilai biner 1 dan jika tidak disimbolkan dengan nilai biner 0. Sesuai dengan yang dikemukakan oleh Minski dan Papert (1969), bahwa perceptron terbatas penggunaannya sebagai metode perhitungan pada aplikasi nyata, masalah akan timbul jika masukan yang diumpangkan adalah bilangan real, dan bilangan real pada keluaran. Proses analisis matematis untuk jaringan semacam ini cukup kompleks, dan dapat dilakukan dengan salah satu varian dari ANN, yaitu multilayer perceptron.

Penggunaan bilangan real untuk masukan dan keluaran mensyaratkan penggunaan threshold dengan fungsi matematis antara lain fungsi sigmoid, arctangen, arcsin, dan lain-lain. Fungsi yang digunakan haruslah mulus dan terus menerus (tidak diperkenankan menggunakan potongan fungsi linear atau fungsi step) dan mempunyai nilai batas atas dan batas bawah absolut. Fungsi sigmoid merupakan fungsi yang umum digunakan dalam ANN.

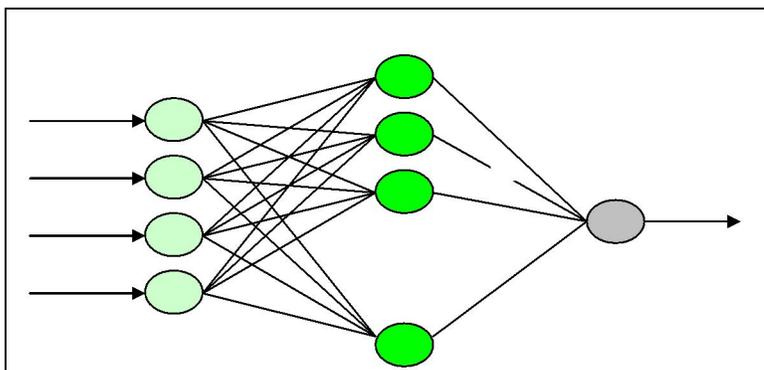


Gambar 6. Fungsi Sigmoid

Arsitektur dari Multilayer Perceptron

Kemampuan dari sebuah simpul teratas pada pengenalan pola-pola yang linear dan fungsi-fungsi logika sederhana. Kemampuan lebih tinggi dapat diperoleh dengan menggabungkan beberapa simpul membentuk ANN. Gambar 7 menunjukkan skema dari ANN lapisan jamak (multilayer). Hal yang terpenting dari ANN bukan hanya bagaimana neuron diimplementasikan tetapi juga bagaimana hubungan antar neuron dibangun (seringkali disebut dengan arsitektur). Lebih lanjut, Arsitektur ANN dibagi menjadi empat golongan, yakni (1) ANN lapisan Tunggal (single layer neural network); (2) ANN lapisan Majemuk (multi layer neural network); (3) ANN arah depan (feedforward neural network); dan (4) ANN recurrent (recurrent neural network).

Setiap lingkaran yang terdapat pada gambar 6 merupakan simpul yang melakukan perhitungan kecuaii pada lapisan masukan. Lapisan tersembunyi dan lapisan keluaran melakukan perhitungan dengan fungsi tertentu yang disebut dengan fungsi aktivasi. Fungsi aktivasi yang sering disebut pula dengan fungsi sigmoid merupakan suatu fungsi kontinyu, monoton tidak turun, keluaran terbatas, dan mudah untuk dideferensialkan. Lapisan tersembunyi pada ANN bisa lebih dari satu lapisan. Menurut Wang et al. (1999) pemakaian lebih dari satu lapisan tersembunyi tidak akan meningkatkan kinerja dari jaringan.



Gambar 7. Jaringan Syaraf Tiruan Multilayer¹

¹ Notasi yang digunakan: x_i : masukan; v_{ij} : nilai pembobot antara lapisan i dan lapisan j (lapisan masukan dan lapisan tersembunyi); z_j : keluaran pada simpul j (pada lapisan tersembunyi); w_{jk} : nilai pembobot antara lapisan j dan lapisan k (lapisan tersembunyi dan lapisan keluaran); dan y_k : keluaran dari simpul k (lapisan keluaran)

Aturan pembelajaran dari ANN bertujuan untuk mengubah-ubah faktor pembobot dan merupakan serangkaian algoritma yang dapat beradaptasi sehingga diperoleh pembobot yang diinginkan. Metode pembelajaran dalam ANN terdiri dari tiga macam yaitu pelatihan terawasi (supervised learning), pelatihan tak terawasi (*unsupervised learning*), dan pelatihan perkuatan (reinforced learning). Pelatihan terawasi merupakan proses pembelajaran menggunakan pembandingan pada keluaran ANN, sehingga diperoleh sinyal kesalahan (*error*). Besarnya kesalahan ini digunakan untuk menata faktor pembobot pada jaringan, sehingga diperoleh keluaran jaringan yang mendekati keluaran yang diinginkan. Prinsip kerja jaringan berdasarkan kesalahan yang digunakan untuk mengoreksi faktor pembobot jaringan ini disebut dengan error correction learning. Algoritma dari error correction learning ini adalah metode pembelajaran penurunan gradien, yaitu kuadrat rata-rata (least mean square) dan penjaralan balik kesalahan (*error backpropagation*) [Demuth dan Bealle, 1994].

Pelatihan yang tak terawasi tidak memerlukan pola sasaran, sehingga tidak ada proses pembandingan keluaran terhadap respon yang diharapkan. ANN dengan pelatihan tidak terawasi dapat belajar dengan cara memasukkan data ke dalam jaringan, dan jaringan-jaringan membentuk kelas-kelas tertentu dan mengklasifikasikan data masukan dalam kelas tertentu. Algoritma metode pelatihan ini dapat ditemukan pada jaringan Hamming, Linsker, dan Kohonen (Peterson, 1996).

Pelatihan perkuatan berhubungan dengan pembaruan pembobot jaringan dengan mengevaluasi sinyal. Hal yang membedakan dengan pelatihan terawasi adalah pada pelatihan terawasi sinyal pembandingan dianggap sebagai sinyal yang benar. Algoritma yang menggunakan metode pelatihan ini adalah learning automata.

Backpropagation

Kekuatan utama dari backpropagation adalah klasifikasi patern, yaitu mengklasifikasikan pasangan input dan output. Selain itu, jaringan syaraf backpropagation juga dapat digunakan untuk memprediksi output suatu sistem, ataupun pengolahan sinyal digital. Jika suatu data diaplikasikan pada ANN dan keluarannya tidak sesuai dengan yang diharapkan, maka yang dilakukan adalah memodifikasi bobot hubungan. Karena inisialisasi (kondisi awal) bobot adalah random, maka kemungkinan mendapatkan output tidak sesuai dengan yang diharapkan adalah tinggi. Untuk meningkatkan kinerja jaringan tidak dapat dilakukan dengan semata-mata memodifikasi bobot hubungan, karena ketidaktahuan akan kontribusi masing-masing bobot hubungan terhadap kesalahan output. Oleh karena itu, digunakan algoritma yang secara efisien dapat memodifikasi hubungan atau faktor bobot untuk meminimisasi kesalahan (error) dari output.

Algoritma yang digunakan adalah pada kondisi di atas adalah error corection learning yang dapat ditemukan pada backpropagation. Proses pembelajaran (*learning*) dari backpropagation terdiri atas tiga tahap. Pertama, pemilihan dan persiapan data training. Pemilihan dan persiapan data training merupakan faktor yang penting dalam algoritma backpropagation. Kedua, inialisasi pembobot. Artinya, pembobot mula-mula dipilih secara acak. Ketiga, modifikasi bobot hubungan neuron. Untuk menjelaskan tahap ini dapat dilihat Gambar 6, masukan x_i diberikan pada lapisan masukan, simpul pada lapisan masukan akan mendistribusikan sinyal tersebut ke simpul pada lapisan tersembunyi. Masukan pada lapisan tersembunyi disebut dengan nilai z_{inj} . Persamaan untuk mendapatkan z_{inj} adalah sebagai berikut ini;

$$Z_{inj} = x_i v_{ij} \quad (3)$$

Setelah memasuki simpul pada lapisan tersembunyi maka akan terjadi perhitungan masukan z_{inj} dengan fungsi aktivasi. Fungsi aktivasi yang digunakan adalah fungsi sigmoid.

$$f(x) = 1/[1 + \exp(-\sigma x)] \quad (4)$$

di mana: x : nilai masukan

σ : konstanta persamaan sigmoid

Nilai keluaran pada lapisan tersembunyi z_j merupakan hasil masukan z_{inj} ke fungsi aktivasi

$$z_j = f(z_{inj}) \quad (5)$$

z_j kemudian menjadi masukan lapisan keluaran

$$y_{ink} = z_j w_{jk} \quad (6)$$

Masukan pada lapisan keluaran (y_{ink}) akan mengalami perhitungan pada simpul keluaran dengan fungsi aktivasi

$$y_k = f(y_{ink}) \quad (7)$$

Aturan belajar diturunkan dengan mengoptimasi suatu fungsi harga (cost function). Harga yang digunakan adalah jumlah kuadrat galat (sum squared error) sebagai berikut ini:

$$E = \frac{1}{2} (t_k - y_k)^2 \quad (8)$$

Di mana; t_k : keluaran sebenarnya (target)

y_k : keluaran ANN

Diferensial dari persamaan (2.9) terhadap w_{jk} maka;

$$\begin{aligned} \frac{1}{2}[t_k - y_k]^2 &= \frac{1}{2}[t_k - f(y_{ink})]^2 \\ &= -[t_k - y_k] f'(y_{ink}) \\ &= -[t_k - y_k] f'(y_{ink}) f'(y_{ink}) z_j \\ &= -[t_k - y_k] f''(y_{ink}) z_j \\ &= -\delta_k z_j \end{aligned} \quad (9)$$

Dari persamaan di atas, maka dapat ditentukan perubahan pembobot antara lapisan keluaran dan lapisan tersembunyi sebagai berikut:

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (10)$$

Di mana : konstanta laju pembelajaran

Dari diferensial persamaan (9) terhadap v_{ij} maka didapat;

$$\begin{aligned} &= -[t_k - y_k] y_k \\ &= -[t_k - y_k] f'(y_{ink}) y_{ink} \\ &= -\delta_k (y_{ink}) \\ &= -\delta_k w_{jk} (z_j) \\ &= x_i - \delta_k w_{jk} f'(z_{inj}) \\ &= \delta_j x_i \end{aligned} \quad (11)$$

Dari persamaan (2.12) dapat ditentukan perubahan pembobot antara lapisan tersembunyi dan lapisan masukan sebagai berikut:

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (12)$$

Prosedur memperbarui pembobot adalah;

$$\Delta w_{jk} (\text{new}) = w_{jk} (\text{old}) + \Delta w_{jk} \quad (13)$$

$$\Delta v_{ij} (\text{new}) = v_{ij} (\text{old}) + \Delta v_{ij} \quad (14)$$

Sedangkan pengaruh konstanta momentum terhadap perubahan nilai pembobot adalah:

$$\Delta w_{jk} (t+1) = \alpha \delta_k z_j + \mu \Delta w_{jk} (t) \quad (15)$$

$$\Delta v_{ij} (t+1) = \alpha \delta_j x_i + \mu \Delta v_{ij} (t) \quad (16)$$

Di mana μ : konstanta laju pembelajaran

Berdasarkan hal di atas untuk memperbaiki proses pembelajaran dilakukan perbaikan dengan menambahkan nilai konstanta momentum (u) antara 0 hingga 0.9, sehingga nilai learning rate dapat ditingkatkan dan osilasi pada jaringan diminimumkan. Metode heuristik yang lain adalah dengan cara menambahkan layer pada hidden unit, menambahkan hubungan pada input unit ke output unit, dan mengkostumasi fungsi sigmoid.

Pengujian pada proses ini adalah pengujian terhadap kinerja ANN. Jika memungkinkan data yang digunakan adalah data nyata, di mana jaringan belum pernah menganalisis dan tidak pernah muncul pada input. Ada beberapa macam cara pengujian kinerja ANN, berikut ini adalah dua di antaranya:

a. RMSE (*root mean square error*)

Kinerja ANN didasarkan atas nilai RMSE (*root mean square error*) pada proses generalisasi terhadap contoh data validasi. Semakin kecil nilai RMSE semakin baik kinerja dari ANN. Persamaan RMSE adalah:

$RMSE =$

Di mana: $y_i =$ nilai prediksi ANN

$t_i =$ nilai aktual yang diberikan

$n =$ jumlah contoh pada pengujian

b. Validasi

Validasi dilakukan sebagai pengujian kinerja atau ketepatan prediksi ANN terhadap contoh yang diberikan selama proses pembelajaran. Proses ini dilakukan dengan memberikan sampel data yang lain dari proses pembelajaran, dan melihat kemampuan ANN memberikan jawaban yang benar. Validasi model dirumuskan dengan persamaan:

Validasi (%) = $x \times 100\%$

Di mana: A = Jumlah data hasil pendugaan yang sama dengan target

B = Jumlah data target

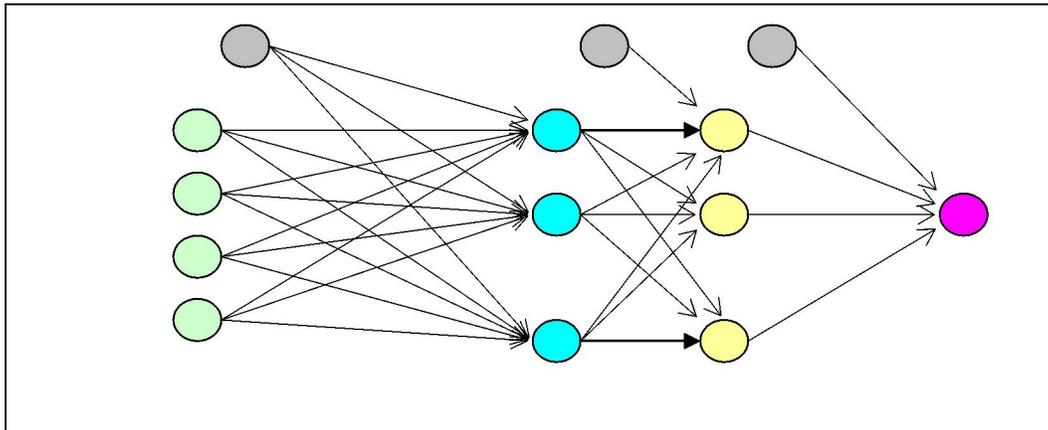
C. METODE PENELITIAN DAN ANALISIS DATA

Data dan Sumber Data

Data yang digunakan dalam penelitian ini adalah data bulanan yang di mulai pada Bulan Juli 1999 dan diakhiri pada Bulan Desember 2004. Awal data pada bulan tersebut dikarenakan pada periode tersebut sudah tidak terjadi gejolak inflasi yang tajam akibat krisis ekonomi tahun 1996. Selain data inflasi, data-data lainnya yang dibutuhkan adalah IHSG, tingkat suku bunga, dan nilai tukar rupiah terhadap dollar. Data inflasi berupa laju kenaikan Indeks Harga Konsumen untuk tujuh kota besar di Indonesia yang memasukkan harga sembilan kebutuhan pokok. IHSG merupakan Indeks Harga Saham Gabungan yang merupakan indeks komposit dari keseluruhan saham yang diperdagangkan di Bursa Efek Jakarta. Tingkat suku bunga merupakan tingkat suku bunga Sertifikat Bank Indonesia (SBI) tempo 3 bulan. Keseluruhan data diperoleh dari publikasi : monthly statistik Jakarta Stock Exchange, dan statistik Bank Indonesia dengan berbagai terbitan.

Model Artificial Neural Network

ANN yang digunakan pada sistem ini menggunakan jenis *multi layer perceptron* dengan dua *hidden layer*. Struktur syaraf tiruan pada sistem terlihat pada Gambar 8. Lapisan input yang terdiri dari 4 variabel yang akan digunakan untuk prediksi inflasi, hal ini diharapkan agar dapat diketahui variabel apa yang paling baik digunakan untuk prediksi inflasi. Variable input yang digunakan untuk prediksi tingkat inflasi pada penelitian ini, yaitu, indeks harga saham gabungan, jumlah uang beredar, nilai kurs rupiah terhadap dollar USA, dan tingkat suku bunga.



Gambar 8. Struktur Jaringan Syaraf Tiruan Sistem Prediksi Inflasi

Pemilihan faktor-faktor di atas tergantung kebutuhan yang diinginkan dalam melakukan prediksi. Dalam sistem ini tidak terdapat faktor politik yang dimasukkan jariangan. Data-data yang ada dalam faktor tersebut sebelum menjadi input pada jaringan terlebih dahulu dilakukan normalisasi agar data-data tersebut berada pada interval 0-1. Cara ini disarankan oleh LeCun (1998), yang menunjukkan bahwa konvergensi umumnya akan lebih cepat tercapai jika nilai rata-rata dari input data training mendekati nol.

Algoritma Pembelajaran

Sistem prediksi inflasi ini menggunakan algoritma pembelajaran backpropagation dengan fungsi aktivitas sigmoid:

$$g(x) = 1/(1 + e^{-x}) \tag{17}$$

Fungsi error yang digunakan pada sistem ini, yaitu:

$$E = \frac{1}{2} \sum (y - d)^2 \tag{18}$$

Derivatif fungsi error yang mengacu pada bobot dapat digunakan untuk meng update bobot sehingga dapat menurunkan nilai error. Untuk melakukan update bobot digunakan algoritma gradient descent yang menghasilkan rangkaian bobot menggunakan rumus dibawah ini:

$$\Delta w_{ij} = -\varepsilon(t) \tag{19}$$

Pilihan nilai learning rate akan sangat berpengaruh pada proses training. Jika terlalu kecil, training akan memerlukan interaksi yang banyak, sehingga lama untuk mencapai konvergen. Jika terlalu besar, bias timbul osilasi sehingga tidak akan mencapai nilai error yang diharapkan. Untuk menghindari terjadinya osilasi untuk learning rate yang besar adalah dengan membuat bobot berubah secara dependent (berpengaruh) pada perubahan bobot sebelumnya dengan menambahkan suatu momentum, yakni:

$$\Delta w_{ij} = -\varepsilon(t) + \mu \Delta w_{ij}(t-1) \tag{20}$$

Penggunaan momentum ini agar titik minimum akan dicapai dengan waktu yang lebih cepat, pada sistem ini momentum dirancang dengan nilai $0.01 \leq \mu \leq 0.99$. Learning rate yang dirancang pada sistem mempunyai nilai $0.01 \leq \eta \leq 0.99$. Penggunaan learning rate dapat mempercepat proses pembelajaran dan error pada epoch sebelumnya akan diikutkan dalam perhitungan peng-update-an bobot berikutnya.

Dari proses pembelajaran yang dilakukan oleh sistem akan diperoleh nilai bobot dari hasil training yang akan digunakan untuk testing dan prediksi data. Sistem prediksi inflasi menggunakan

metode split sample. Artinya, data yang digunakan pada saat training tidak digunakan lagi pada testing. Nilai error pada saat testing akan memberikan gambaran estimasi model yang cukup general.

Algoritma Backpropagation

Suatu jaringan backpropagation dengan Q lapisan, $q = 1, 2, \dots, n$. Q Input jaringan dilambangkan dengan ${}^q\text{net}_i$ dan output lapisan ke-q dilambangkan dengan ${}^q\text{y}_i$. Jaringan backpropagation memiliki n neuron input dan m neuron output. Sinapsis antara ${}^{q-1}\text{y}_j$ dengan ${}^q\text{y}_i$ dilambangkan dengan ${}^q\text{w}_{ij}$.

a. Inisialisasi bobot awal

Dengan menggunakan variabel random yang cukup kecil antara 0 sampai 1, kerjakan langkah-langkah berikut, selama kondisi berhenti bernilai FALSE. Untuk tiap penawaran elemen yang akan dilakukan pembelajaran, kerjakan:

b. Feed Forward

1. Tiap-tiap unit input ($X_i, i=1, 2, 3, \dots, n$) menerima sinyal x_i dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (lapisan tersembunyi).
2. Tiap-tiap unit tersembunyi ($Z_j, j = 1, 2, \dots, p$) menjumlahkan sinyal-sinyal input terbobot.

$$z_in_j = v_{0j} + x_i v_{ij} \quad (3.5)$$

Gunakan fungsi aktivitas untuk menghitung sinyal outputnya:

$$z_j = f(z_in_j) \quad (3.6)$$

Kirimkan sinyal tersebut ke semua unit dilapisan atasnya (unit-unit output)

3. Tiap-tiap output ($Y_k, k=1, 2, \dots, m$) menjumlah sinyal-sinyal input bobot

$$y_in_k = w_{ok} + z_j w_{jk} \quad (3.7)$$

Gunakan fungsi aktivitas untuk menghitung sinyal output:

$$y_k = f(y_in_k) \quad (3.8)$$

Dan kirimkan sinyal tersebut ke semua unit dilapisan atasnya (unit-unit output).

c. Backpropagation

1. Tiap-tiap output ($Y_k, k=1, 2, \dots, m$) menerima target pola yang berhubungan dengan pola input pembelajaran, hitung informasi errornya:

$$\delta_k = (t_k - y_k) f'(y - in_k)$$

Kemudian hitung koreksi bobot (yang nanti akan digunakan untuk memperbaiki nilai w_{jk}).

$$\Delta w_{jk} = -\varepsilon \delta_k z_j + \mu \Delta w_{jk-1} \quad (3.9)$$

2. Tiap-tiap tersembunyi menjumlahkan delta inputnya (dari unit-unit yang berada pada lapisan atasnya):

$$\delta_in_j = \delta_k w_{kj} \quad (3.10)$$

Kalikan nilai ini dengan turunan dari fungsi aktivitas untuk menghitung informasi error:

$$\delta_j = \delta_in_j f'(z_in_j) \quad (3.11)$$

Hitung koreksi bobot untuk memperbaiki nilai v_{ij} :

$$\Delta v_{ij} = \varepsilon \delta_j + \mu \Delta v_{ij-1} \quad (3.12)$$

3. Tiap-tiap output ($Y_k, k = 1, 2, \dots, m$) memperbaiki bobotnya ($j = 0, 1, 2, \dots, p$)

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (3.13)$$

Tiap-tiap unit tersembunyi ($Z_j, j = 1, 2, \dots, p$) memperbaiki bobotnya ($i = 1, 2, \dots, n$).

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (3.14)$$

Hasil: Perbandingan Model Regresi dengan ANN dalam Proyeksi Inflasi

Model ekonometrika selalu mengalami perkembangan, khususnya time series analysis. Model analisis time series mulai dari ARCH, GARCH, ARIMA, ARCES, sampai penggunaan jaringan syarat atau dikenal dengan istilah ANN. Penggunaan ANN saat ini mulai masuk pada fase penyempurnaan, sehingga banyak diterapkan pada berbagai bidang ilmu. Pada bidang ekonomi, terutama digunakan untuk melakukan forecasting suatu kejadian. Misalkan, prediksi inflasi, pertumbuhan ekonomi, pertumbuhan penjualan, valas, dan permasalahan ekonomi lainnya.

Pendekatan yang akan digunakan dalam permodelan ANN dapat menggunakan pendekatan kausal dan pendekatan time series, dan atau kombinasi dari keduanya. Software yang digunakan dalam analisis ini, untuk ANN menggunakan S-plus 2000, sedangkan regresi linear menggunakan Minitab Vs 14.

Analisis Regresi Linear Berganda

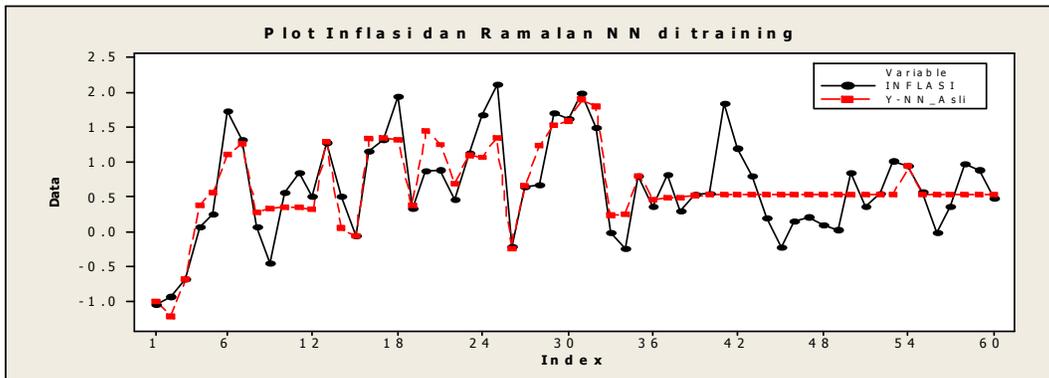
Sebelum dilakukan analisis, maka dilakukan pemetaan untuk melihat pola hubungan antara variabel dependent (variabel inflasi) dengan masing-masing variabel independen (variabel uang yang beredar, variabel kurs, variabel tingkat suku bunga, dan variabel indeks harga saham gabungan). Hasil analisis korelasi menjelaskan bahwa dari keempat variabel terpilih, hanya uang yang beredar yang memiliki korelasi sangat kecil (8,6%), sedangkan ketiga variabel lain (IHSG, KURS, dan BUNGA) memiliki korelasi yang cukup tinggi terhadap inflasi. Lebih lanjut, dari hasil potting didapat bahwa pola hubungan antara variabel inflasi dengan masing-masing variabel independen (IHSG, UANG, KURS, dan BUNGA) tidak semuanya berpola hubungan linear. Hanya ***inflasi dengan KURS yang memiliki pola hubungan yang linear***, sedangkan ketiga lainnya memiliki pola hubungan yang non-linear. Dengan melihat hasil tersebut, maka dapat disimpulkan bahwa penggunaan analisis regresi linear tidak akan menghasilkan persamaan regresi yang baik.

Hasil regresi linear berganda dengan menggunakan jumlah observasi sebanyak 60, sebagai training. Dari hasil regresi di dapat hasil, sebagai berikut:

$$INFLASI = - 2.75 + 0.00036 IHSG + 0.00112 UANG + 0.000312 KURS + 0.0164 BUNGA$$

Berdasarkan hasil regresi tersebut, hanya variabel kurs yang memiliki arti penting variabel inflasi. Artinya, variabel kurs yang nyata memengaruhi inflasi. Hal ini sejalan dengan hasil plotting, di mana kurs yang memiliki pola hubungan linear dengan inflasi. Sedangkan ketiga variabel lainnya tidak berpengaruh nyata terhadap inflasi.

Walaupun hasil tersebut cukup baik berdasarkan tuntutan model, yang dapat dilihat dari nilai analisis variannya, baik dari P_value maupun dari D.W. Namun variabel tersebut belum menunjukkan kemampuan untuk memprediksi variabel inflasi, yang ditunjukkan dari besarnya $R^2_{(adj)} = 16,2$ persen. Kecilnya koefisien korelasi tersebut bisa disebabkan oleh beberapa variabel lain yang tidak dimasukkan dalam model, tetapi juga bisa disebabkan pilihan model yang kurang tepat (misspesifikasi model). Selanjutnya, untuk melihat kemampuan prediksi dari hasil regresi di training, kemudian diuji cobakan pada data testingnya. Berdasarkan hasil testingnya didapat bahwa $SSE = 1.69427$, $MSE = 0.282378$, dan $RMSE = 0.531392$.

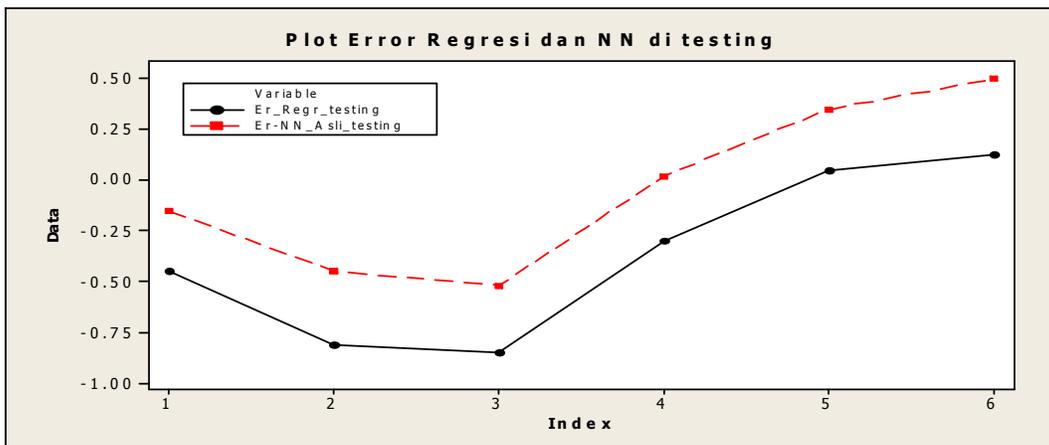


Gambar 9. Plot Inflasi dan Ramalan NN di Training

Analisis ANN

Analisis ANN dengan menggunakan dua macam data, yaitu data training dan testing. Seperti halnya yang dilakukan pada regresi linear, data training sebanyak 60 observasi, sedangkan data testing 6 observasi. Data training digunakan untuk pembelajaran komputer, dalam upaya melakukan prediksi. Data testing digunakan sebagai validasi, dari hasil pembelajaran dari data training. Langkah kedua, menentukan jumlah hidden layer untuk mendapatkan hasil yang terbaik.

Untuk melihat hasil ANN dengan riil inflasi dapat dilihat pada Gambar 9. Hasil tersebut dapat disimpulkan bahwa analisis ANN lebih baik dibandingkan dengan regresi, karena kemampuan ANN dalam mendekati dengan riil inflasi. Atau secara jelas dapat dibandingkan error antara analisis regresi dengan ANN, yang secara jelas dapat dilihat pada Gambar 10.



Gambar 10. Plot Error Regresi dan NN di Testing

Secara ringkas dapat dilihat kemampuan ANN dengan regresi linear dalam melakukan prediksi inflasi. Perbedaan tersebut dapat dilihat pada Tabel 1.

Tabel 1. Perbedaan ANN dan Regresi Linier

Keterangan	Regresi Linear		ANN	
	Training	Testing	Training	Testing
SSE	23,17	1,69	9,28	0,87
MSE	0,42	0,28	0,16	0,15
RMSE	0,65	0,53	0,39	0,38

D. KESIMPULAN DAN REKOMENDASI

Kesimpulan

1. Pola hubungan antara variabel inflasi dengan variabel uang beredar, variabel kurs, variabel tingkat suku bunga, variabel indeks harga saham gabungan tidak semua linear. Hanya inflasi dengan KURS yang memiliki pola hubungan yang linear, sedangkan ketiga lainnya memiliki pola hubungan yang non-linear.
2. Berdasarkan hasil regresi linear hanya variabel kurs yang memiliki arti penting bagi variabel inflasi, artinya variabel kurs yang nyata memengaruhi inflasi. Hal ini sejalan dengan hasil plotting dimana kurs yang memiliki pola hubungan linear dengan inflasi. Sedangkan ketiga variabel (variabel uang beredar, variabel tingkat suku bunga, variabel indeks harga saham gabungan) tidak berpengaruh nyata terhadap inflasi.
3. Hasil prediksi inflasi dengan menggunakan analisis ANN lebih baik dibandingkan dengan analisis regresi linear. Hal ini dapat dilihat dari korelasi keempat variabel terhadap inflasi, dimana pada ANN sebesar 0,83 sedangkan pada regresi linear hanya 0,16.

Rekomendasi

Dalam analisis ditemukan predictive power sebesar 83 % dan analisis regresi sebesar 16%. Hasil ini perlu diinterpretasi hati-hati mengingat dalam model analisis hanya melibatkan 66 observasi. Pada analisis regresi masih memungkinkan diuji dengan model lainnya untuk mencapai model terbaik yang selanjutnya dapat dibandingkan dengan model ANN. Hal ini diharapkan untuk lebih menguatkan kesimpulan akurasi *predictive power* ANN lebih tinggi dari model lainnya.

DAFTAR PUSTAKA

- Anonim. *Ekonomi Makro Indonesia*. Bank Indonesia
———. *SEMI*. Vol. IV No. 24. 2004. Bank Indonesia
———. *Berita Resmi Statistik*. Biro Pusat Statistik
- Apotolos, Paul Refenes. *Neural Network In The Capital Market*. Editor 1995 John Wiley dan Sons Ltd. Baffins Lane. Chichester West Sussex Po 19. England.
- Bishop, C.M. 1996. *Neural Network for Pattern Recognition*. Oxford University
- Faraway, Julian dan Chatfield, Chris. 1995. *Time Series Forecasting with Neural Network: A Case Study*. Research Report 95-06 of The Statistics Group. University of Bath
- Fahlman, S.E. 1988. *An Empirical Study of Learning Speed in Back-Propagation Network*. Technical Report CMU-CS-88-162. Carnegie Mellon University. USA

- Fausett, L. 1994. *Fundamentals of Neural Network: Architectures, Algorithms, and Applications*. Prentice-Hall Inc. USA
- Gately, Edward J. 1996. *Neural Networks for Financial Forecasting*. John Wiley & Son. New York
- Gregory, N. Mankiw. 2000. *Macroeconomics*. Worth Publishers, Inc. New York and Basingstoke. 4th edition
- Kartalopoulos, Stamatios V. 1996. *Understanding Neural Networks and Fuzzy Logic: Basic Concepts and Applications*. IEEE Press. New York
- LeCun, Y., Bottou, L., Orr, G.B., Muller, K.R. 1998. *Efficient BackProp, Neural Network: Tricks of The Trade*. Springer
- LiMin Fu. 1994. *Neural Network In Computer Intelligence*. McGraw-Hill Internasional Editions
- Marimin, Yani N. dan Teguh Wibowo. 2002. *Sistem Prediksi Harga Saham Menggunakan Logika Fuzzy dan Jaringan Syaraf Tiruan*. Prosiding Ilmu Komputer dan Teknologi Informasi III. 27 - 28 Agustus 2002. Fakultas Informasi Universitas Tarumanegara
- Mederros Marcelo C., Timo Terasvirta, dan Gianluigi R. Building Neural Network Models for Time Series: A Statistical Approach. Departamento De Economia, www.econ.puc-rio.br.
- Nakamura, Emi. 2001. *Inflation Forecasting Using A Neural Network*, www.harvard.edu/nakamura
- Nopirin. 1990. *Ekonomi Moneter*. BPFE. Yogyakarta
- Riedmiller, M. 1994. *Advanced Supervised Learning in Multi-Layer Perceptrons From Backpropagation to Adaptive Learning Algorithms*. Int. Journal of Computer Standard and Interfaces (16)
- Rumelhart, D.E., Hinton, G.E., Williams, R.J. 1986. *Learning Internal Representations by Error Propagation*. MIT Press, Cambridge, MA
- Reed, R.D. 1999. *Supervised Learning in Feedforward Artificial Neural Network*. The MIT Press
- Sarles, W.S. ed. 1997. *Neural Network FAQ*. URL: ftp://ftp.sas.com/pub/neural/FAQ.html.
- Schiffmann, W., Joost, M., Werner, R. 1993. *Comparison of Optimized Backpropagation Algorithms*. Proc. Of the European Symposium on Artificial Neural Network (ESANN). 1993. Brussels
- Sofyan, Assauri. 1984. *Teknik dan Peramalan: Penerapannya dalam Ekonomi dan Usaha*. Lembaga Penerbit FE-UI. Jakarta
- Stern, Hal, S. 1996. *Neural Network in Applied Statistics*. Departemen of Statistics Iowa State University Arnes, Americans Statistical Association and the American Society for Quality Control, Technometrics, August, Vol. 38 No. 3
- Sukirno, Sadono. 2003. *Pengantar Teori Makro Ekonomi*. Edisi kedua. Lembaga Penerbit FE-UI. Jakarta
- Winardi. 1990. *Pengantar Ekonomi Makro*. Penerbit Tarsito Bandung