

# PREDIKSI HARGA SAHAM MENGUNAKAN JARINGAN SYARAF TIRUAN *BACKPROPAGATION*

Siti Amiroch

Universitas Islam Darul 'Ulum Lamongan, amirast\_117@yahoo.com

**Abstract.** *In the stock market, stock price prediction is an important issue for the perpetrators of capital transactions to help making the right decision. Most traders have their own application software to predict the stock price so that it can be decided would buy the shares or sell them. By using a neural networks, prediction of stock prices can be done by using the backpropagation algorithm. Artificial neural networks can be used either to predict the level or price of the stock index, stock movement (trend), and the return earned on stocks. This study discusses the use of techniques Backpropagation Neural Network to predict the stock price closing (Close) in AKR Tbk (AKRA Corporindo) engaged in the petroleum, chemical, logistics, manufacturing and coal are simulated in Matlab. Of some testing done, the prediction results obtained are very close to the price actually with very small MSE value.*

**Keywords:** *stock price, neural network, backpropagation.*

**Abstrak.** Dalam bursa saham, prediksi harga saham merupakan isu penting bagi pelaku transaksi modal untuk membantu pengambilan keputusan yang tepat. Kebanyakan para trader mempunyai software aplikasi sendiri untuk memprediksi harga saham sehingga bisa diputuskan akan membeli saham tersebut atau bahkan menjualnya. Dengan menggunakan *neural network* atau jaringan syaraf tiruan, prediksi harga saham bisa dilakukan dengan menggunakan algoritma *backpropagation*. Jaringan syaraf tiruandapat digunakan baik untuk memprediksi level atau harga indeks saham, pergerakan saham (*trend*), maupun *return* yang diperoleh atas saham. Penelitian ini membahas penggunaan teknik Jaringan Syaraf Tiruan Backpropagation untuk memprediksi harga sahampenutupan (*Close*) pada PT AKR Corporindo Tbk (AKRA Corporindo) yang bergerak di bidang petroleum, chemical, logistik, pabrikan dan batubara yangdisimulasikan dalam matlab. Dari beberapa testing yang dilakukan, diperoleh hasilprediksi sangat mendekati harga sebenarnya dengan nilai MSE yang sangat kecil.

**Kata kunci :** *harga saham, jaringan syaraf tiruan, backpropagation*

## 1 Pendahuluan

Fluktuasi harga saham dipengaruhi oleh banyak faktor seperti kondisi dan kinerja perusahaan, resiko, deviden, tingkat suku bunga, kondisi perekonomian, kebijaksanaan pemerintah, laju inflasi, penawaran dan permintaan serta masih banyak lagi. Karena dimungkinkan adanya perubahan faktor-faktor di atas, harga saham dapat naik atau turun. Sebelum melakukan pembelian atas saham, trader

akan melakukan analisa dari informasi fluktuasi harga tersebut, tidak cukup dengan mengandalkan informasi harga saham pada saat ini saja tetapi juga informasi harga saham dari waktu yang lampau (*past*) juga perlu diketahui. Dari informasi tersebut dapat dibuat sebuah model (*trend*) yang menggambarkan informasi harga saham di waktu lampau hingga saat ini (*present*). Dengan trend ini informasi harga saham dapat diprediksi/diramalkan sehingga investor dapat melihat bagaimana prospek investasi saham dari sebuah perusahaan di masa datang. Prediksi harga saham juga dapat digunakan untuk mengantisipasi naik turunnya harga saham sehingga para trader sangat terbantu dalam pengambilan keputusan.

Kebanyakan para trader membeli *software* aplikasi sendiri, sedangkan harga dari *software* dan pelatihan dalam menggunakannya relatif mahal. Sehingga diperlukan solusi untuk mengatasi hal tersebut. Adanya *Neural Network* bertujuan menggantikan *software* tersebut tanpa harus membeli *software*, tentunya diharapkan dengan hasil yang mendekati harga yang sebenarnya.

*Neural network* merupakan model *learning* yang menyerupai sistem neuron pada makhluk hidup. *Neural network* terdiri atas sekumpulan unit input dan output yang terhubung satu dengan lainnya dan masing-masing hubungan antar unit mempunyai bobot. Setiap unit input dan output pada *network* merupakan bagian dari sebuah lapisan/*layer* dalam *network*. Sebuah *neural network* dapat mempunyai tiga atau lebih lapisan, yaitu satu lapisan input, satu atau lebih lapisan tersembunyi dan satu lapisan output. Tahap *learning* pada *neural network* mencakup evaluasi dan penyesuaian kembali bobot setiap hubungan antar unit di dalam *network* sehingga *tuple-tuple* data yang masuk ke dalam *network* dapat diberi kelas atau label yang tepat. *Neural network* dapat digunakan baik untuk memprediksi level atau harga indeks saham, pergerakan saham (*trend*), maupun *return* yang diperoleh dari saham.

*Backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama proses training serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola inputan yang serupa (tapi tak sama) dengan pola yang dipakai selama *training*. Algoritma *backpropagation* menggunakan *error output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan error ini, tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu. Pada saat perambatan maju, neuron-neuron diaktifkan dengan menggunakan fungsi aktivasi yang dapat dideferensialkan, seperti fungsi sigmoid biner dengan range (0,1).

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

$$f'(x) = f(x)[1 - f_1(x)] \quad (2)$$

Dari paparan diatas, sebagai alternatif bagi pelaku pasar untuk mengambil keputusan dalam berinvestasi di dunia saham, penelitian ini bertujuan untuk memprediksi harga saham yang dilakukan dengan *neural network* menggunakan algoritma *backpropagation*. Harga saham yang diprediksi adalah saham penutupan (*close*) dari saham AKR *CorporindoTbk* (AKRA).

## 2 Metode Penelitian

Pada penelitian ini data yang digunakan adalah data historikal AKR *Corporindo Tbk* (AKRA) dari 7 Januari 2013 sd 24 Juni 2013 yang diambil dari <http://finance.yahoo.com>. Terdapat 120 data ditampilkan, namun hanya 24 data yang *ditraining* dan 5 data sebagai *testing*. Prediksi dilakukan untuk mengetahui harga saham penutupan pada tanggal 11, 12, 13, 14, dan 15 Pebruari 2013. Adapun alasan dipakainya data AKRA pada tahun 2013 karena dalam dunia saham, pada saat itu AKRA merupakan perusahaan dengan harga kenaikan saham paling tinggi sehingga memberikan keuntungan yang optimal untuk para *trader*.

Pada penelitian ini, simulasi menggunakan *software* Matlab 2012 dengan spesifikasi komputer yang digunakan yaitu : *Intel(R) Core(TM) 2 Duo CPU T6500 @ 2.10GHz*, RAM 2048 Mb.

Sebelum dilakukan prediksi terhadap harga saham, beberapa hal yang perlu dirancang adalah sebagai berikut :

### a. Perancangan Data *Input Output*

Untuk data *input*, data yang digunakan adalah data harga saham penutupan (*close*) yang diperlakukan secara '*time series*'. Data harian selama periode tertentu, misal  $X_t$  ( $t=1,2,3,\dots$ ) dipakai untuk menentukan harga saham pada periode berikutnya. Dengan kata lain, data ke-1 sampai data ke-24 dijadikan data input untuk memprediksi data yang ke-25. Dilanjutkan dengan data ke-2 sampai data ke-25 untuk memprediksi data yang ke-26 dan seterusnya.

Untuk data *output* sebanyak satu, yaitu harga saham keesokan harinya. Namun, karena proses *testing* dilakukan 5 kali untuk 24 data secara berulang, maka *output* yang ditampilkan dalam laporan ini adalah 5 data. Untuk memperoleh *output* diperlukan dua rangkaian proses yaitu: *training* dan *testing*. Pada penelitian ini, data *training* sebanyak 24 data, sedangkan untuk *testing* digunakan satu data saja.

### b. Perancangan Proses Training

Sebelum proses *training*, setiap variabel *input* harus dinormalkan menjadi data yang bernilai antara 0 dan 1, tujuannya agar proses *training* pada Jaringan Syaraf Tiruan lebih efisien dan efektif apabila data-data yang masuk berada pada suatu *range* tertentu. Oleh karena itu, data *input* harus melalui proses normalisasi terlebih dahulu sehingga berada pada *range* yang sama dengan fungsi aktivasinya, yaitu antara 0 sampai 1. Tapi akan lebih baik jika ditransformasikan ke interval yang lebih kecil, misal pada interval [0.1, 0.9]. Hal ini mengingat bahwa fungsi *sigmoid* adalah fungsi asimtotik yang nilainya tidak pernah mencapai 0 ataupun 1. [2] Pada penelitian ini normalisasi data menggunakan rumus :

$$x' = \frac{0.8(x-a)}{b-a} + 0.1 \quad (3)$$

dengan :

$x'$  : data yang akan dinormalisasi

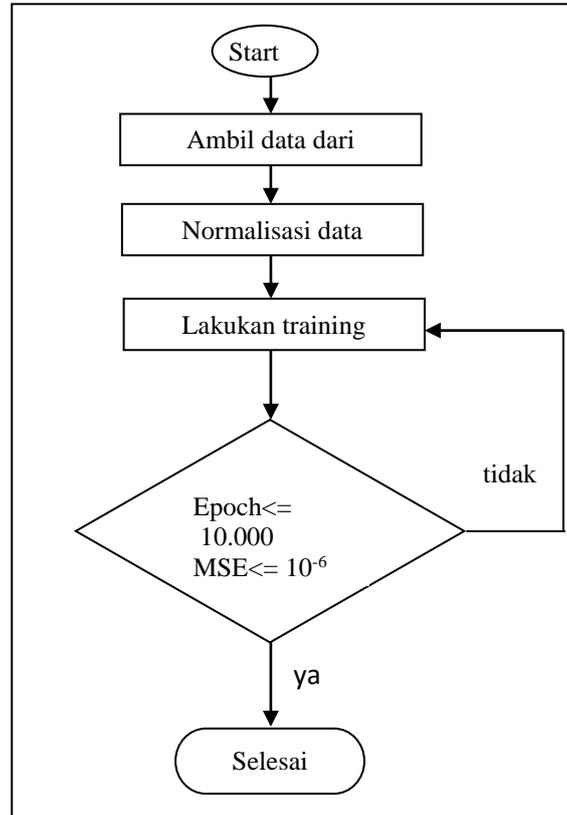
$x$  : data yang dimaksud

$a$  : data terkecil

$b$  : data terbesar

Akibatnya,, pada akhir proses *training*, diperoleh nilai-nilai bobot dari data *training* terakhir yang selanjutnya disimpan untuk proses *testing*.

Berikut adalah diagram alir perancangan proses *training*.



Gambar 1. Diagram alir proses *training*

### c. Perancangan Proses *Testing*

Pada proses *testing*, yang menjadi variabel *input* ( $x_i$ ) sama seperti pada proses *training*. Data *input* juga harus dinormalkan terlebih dahulu. *Output* dari proses *testing* adalah ‘harga saham prediksi jaringan syaraf tiruan’. Selain itu juga ditampilkan *error testing*. Dari *error* tersebut akan dihitung rata-rata *error testing* guna mengukur sejauh mana sistem jaringan syaraf tiruan ini bekerja untuk memprediksi harga saham. Rumus untuk menghitung rata-rata *error* jaringan pada saat *testing* adalah:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \bar{y}_i)^2}{N}} \quad (4)$$

dengan :

$y_i$  = nilai aktual data

$\bar{y}_i$  = nilai hasil prediksi

$N$  = jumlah data testing

Pada proses *testing*, *output* jaringan syaraf tiruan yang bernilai antara 0 sampai dengan 1 perlu dikonversikan kembali supaya menjadi harga saham normal. Proses tersebut dinamakan denormalisasi data. Adapun rumus yang digunakan adalah:

$$X_i = Y(X_{max} - X_{min}) + X_{min} \quad (5)$$

dengan :

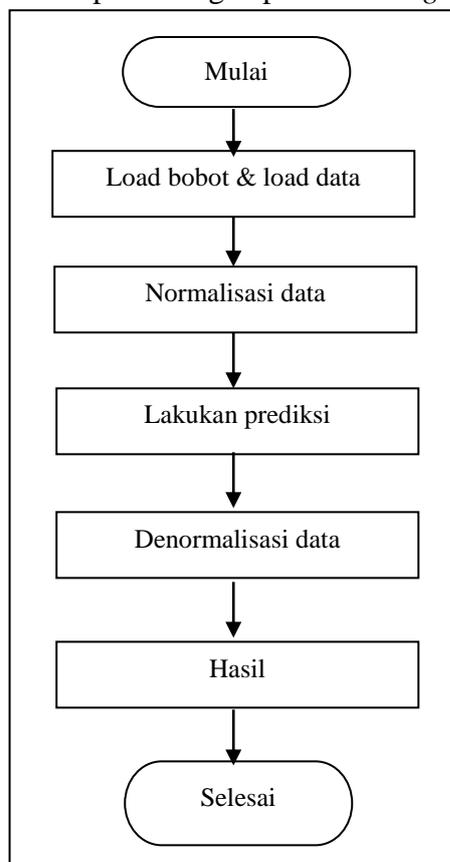
$X_i$  = harga saham normal

$X_{min}$  = data dengan nilai minimum

$Y$  = hasil *output* jaringan

$X_{max}$  = data dengan nilai maksimum

Berikut adalah diagram alir perancangan proses *testing*.



Gambar 2. Diagram alir proses testing

#### d. Algoritma

**Step 0 :** Inisialisasi semua bobot dengan bilangan acak kecil

Bobot bias *input* ( $v0j$ ) = bilangan acak dari  $-\beta$  dan  $\beta$ , dengan

Bobot *input* ( $vij$ ) = bilangan acak dari -0.5 dan 0.5

Bobot bias *hidden* ( $w0k$ ) dan bobot *hidden* ( $wjk$ ) = bilangan acak dari -1 dan 1.

**Step 1 :** Jika kondisi penghentian belum terpenuhi, lakukan Step 2-9

**Step 2 :** Untuk setiap pasang data pelatihan, lakukan langkah 3-8

**Step 3 :** Tiap unit input ( $X_i, i=1, \dots, n$ ) menerima sinyal dan meneruskannya ke unit *hidden*

**Step 4 :** Hitung semua *output* di unit *hidden* ( $Z_j, j=1, \dots, p$ )

$$z\_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

$$z_j = f(z\_in_j)$$

**Step 5 :** Hitung semua *output* di unit *output* ( $Y_k, k=1, \dots, m$ )

$$y\_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk}$$

$$y_k = f(y\_in_k)$$

**Step 6 :** Hitung  $\delta$  unit *output* berdasarkan *error* di setiap unit *output*  $Y_k$

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

Hitung suku perubahan bobot *hidden* dan bobot bias *hidden* dengan *learning rate*  $\alpha$

$$\Delta w_{jk} = \alpha \delta_k z_j$$

$$\Delta w_{0k} = \alpha \delta_k$$

**Step 7 :** Hitung  $\delta$  unit *hidden* berdasarkan *error* di setiap unit *hidden*  $Z_j$

$$\delta\_in_j = \sum_{k=1}^m \delta_k w_{jk}$$

$$\delta_j = \delta\_in_j f'(z\_in_j)$$

Hitung suku perubahan bobot *input* dan bobot bias *input* dengan *learning rate*  $\alpha$

$$\Delta v_{ij} = \alpha \delta_j x_i$$

$$\Delta v_{0j} = \alpha \delta_j$$

**Step 8 :** Hitung semua perubahan bobot

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

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

**Step 9 :** *Test* kondisi penghentian

### 3 Hasil dan Pembahasan

Dari 120 data yang ada, data yang ditraining sebanyak 24 sedangkan data testngnya 5. Prediksi dilakukan pada data ke-25, 26, 27, 28, dan 29. Prediksi data

ke-25 yaitu pada tanggal 11 Pebruari 2013 adalah merupakan hasil *training* dari 24 data sebelumnya, demikian seterusnya.

Sebelum proses *training* dilakukan, ditentukan dulu parameter yang dibutuhkan yaitu: maksimum *epoch* ditentukan sebesar 10.000, *target error*  $1 \times 10^{-6}$  dan *learning rate* ( $\alpha$ ) sama dengan 1. Dengan parameter tersebut, diperoleh hasil simulasi program untuk data ke-25 yaitu:

*Epoch* = 10000  
 MSE = 4.5275e-05  
 RMSE = 0.0067.

Diperoleh *V* sebagai hasil dari pelatihan bobotnya sebagai berikut :

0.6504 0.5769 0.5032 0.0999.

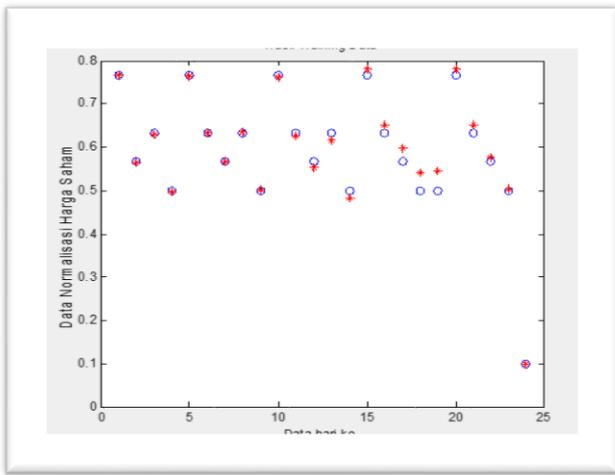
Diperoleh juga *output* di unit *output* (prediksi)

$y = 0.5416$ .

Setelah didenormalisasi, diperoleh hasil prediksi adalah :

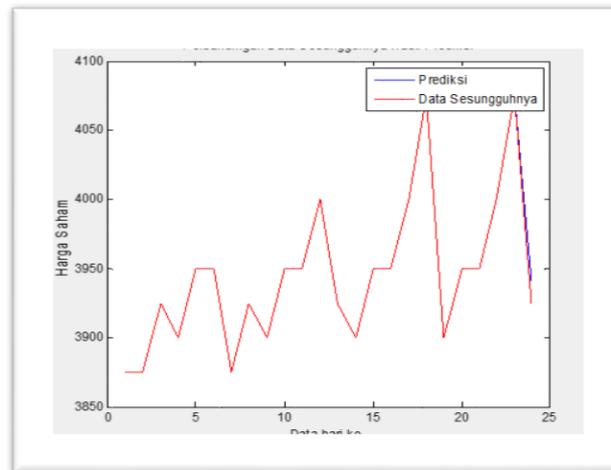
Prediksi = 3941  
 selisih = 16.

Nilai selisih diperoleh dari selisih nilai prediksi dikurangi nilai sebenarnya. Berikut adalah gambar hasil training data ke-25 :



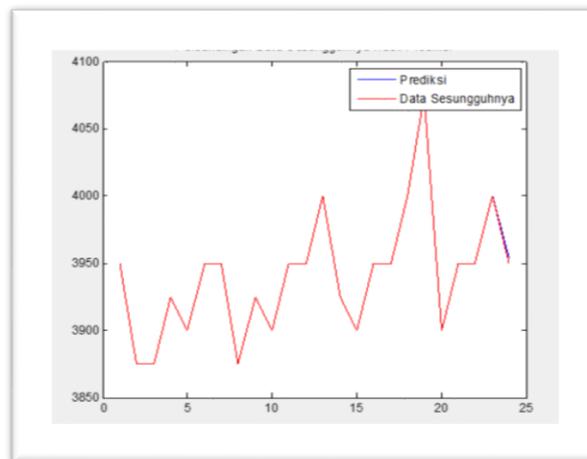
**Gambar 3.** Bobot dan target

Pada Gambar 3 diatas, output digambarkan sebagai \*, dan o adalah targetnya. Target mewakili nilai sebenarnya. Tampak bobot pelatihan (prediksi) sangat mendekati target terutama pada data ke-1,2,3,4,5,6,7,8,9,23 dan 24..Pada data ke-24 tampak nilai bobot pelatihan (\*) hampir persis dengan target (o). Dan apabila dilihat grafik dari perbandingan data sebenarnya dengan hasil prediksi Jaringan Syaraf Tiruan *backpropagation* pada data ke-25, gambar prediksinya sangat mendekati gambar data sebenarnya.



**Gambar 4.** Perbandingan data sesungguhnya dengan hasil prediksi data ke-25

Pada Gambar 4 diatas, prediksi yang dihasilkan yaitu Rp 3941,- sedangkan data sesungguhnya adalah Rp 3925,-. Ada selisih Rp 16,- antara data yang sebenarnya dan target. Dengan MSE  $4.5275e-05$  dan RMSE 0.0067 menunjukkan ketelitian yang sangat tinggi. Begitu juga untuk prediksi data ke-27, dengan MSE sebesar  $1.3137e-04$  dan RMSE 0.0115 diperoleh hasil prediksi Rp 3954,- sedangkan data sesungguhnya Rp. 3950,-. Dari data tersebut tampak selisih Rp 4,- merupakan nilai yang sangat kecil dan menunjukkan hasil prediksi yang bagus untuk sebuah prediksi saham.



**Gambar 5.** Perbandingan data sesungguhnya dengan hasil prediksi data ke-27

Pada Gambar 5 terlihat grafik prediksi dan data sesungguhnya sangat berimpit karena selisih antara prediksi dan data sesungguhnya sangat kecil. Untuk hasil prediksi secara lengkap ditunjukkan pada tabel berikut:

**Tabel 1.** Hasil prediksi secara lengkap

Data ke-	Tanggal	Data sesungguhnya	Hasil prediksi	Selisih	MSE	RMSE
25	11-02-2013	3925	3941	16	4.5275e-05	0.0067
26	12-02-2013	4025	3957	68	1.4509e-04	0.0120
27	13-02-2013	3950	3954	4	1.3137e-04	0.0115
28	14-02-2013	3975	3947	28	1.0768e-04	0.0104
29	15-02-2013	3925	3947	22	1.2296e-04	0.0111

#### 4 Kesimpulan

Dari hasil pelatihan data saham AKRA menggunakan algoritma *backpropagation*, diperoleh hasil prediksi saham mendekati harga sesungguhnya dengan ditunjukkannya selisih prediksi dengan data sebenarnya yang sangat kecil. Selain itu, nilai MSE yang sangat kecil juga menunjukkan hasil *training* yang bagus. Hal ini bisa diartikan bahwa algoritma *backpropagation* bisa diperhitungkan untuk memprediksi harga saham sekaligus sebagai pengganti *software* alternatif yang lebih ekonomis dibandingkan dengan *software* prediktor yang harganya relatif mahal.

#### Daftar Pustaka

- [1] Fausett, L. 1994. *Fundamentals of Neural Network (Architectures, Algorithms, and Applications)*. Prentice Hall. New Jersey.
- [2] Jek Siang, J. 2005. *Jaringan Syaraf Tiruan dan Pemrogramannya menggunakan Matlab*. Andi Offset. Yogyakarta.
- [3] Kusumadewi, S. 2004. *Membangun Jaringan Syaraf Tiruan menggunakan Matlab dan Excel Link*. Graha Ilmu. Yogyakarta.