Digital Transformation Technology (Digitech) | e-ISSN : 9999-9999
Volume 1, **Number** 1, September 2021
https://doi.org/10.47709/digitech.v1i2.1111

Penerapan Jaringan Syaraf Tiruan Untuk Memprediksi Penjualan Mobil Dengan Menggunakan Metode *Backpropagation* (Studi Kasus : Toyota Auto 2000 Medan)

Nurhayati[1*], Juliana Naftali Sitompul[2], Bagus Alwi Setiawan[3]
[1,2,3] STMIK Kaputama, Indonesia
[1]nurhayatiazura059@gmail.com, [2]joellyanna07@gmail.com, [3]bagusalwisetiawan10@gmail.com

open access
Histori Artikel:
Diajukan: 24 June 2021
Disetujui: 25 June 2021
Dipublikasi: 30 September 2021

Kata Kunci:
Backpropagation; Data; Jaringan Syaraf Tiruan; Kenderaan; Penjualan Mobil.

Abstrak
Penjualan kendaraan merek Toyota ditangani oleh Divisi Kendaraan yang berkedudukan di kantor pusat Jakarta dan untuk seluruh cabang ditangani oleh Departemen Penjualan. Data produksi yang digunakan adalah data tahun 2016, 2017 dan 2018 berupa data bulanan. Dengan *epoch* maksimum antara 0-10000, *learning rate* 0,1 dan *target error* 0,01-0,5 untuk mendapatkan hasil yang konvergen. Data penjualan mobil dapat dikenali oleh sistem jaringan syaraf tiruan dengan metode *backpropagation*, hasil pengujian mengalami kenaikan dan penurunan. Prediksi penjualan New Agya meningkat rata-rata 5.99/bulan, Calya meningkat rata-rata 5.99/bulan, All New Rush meningkat rata-rata 12.06/bulan, New Avanza meningkat rata-rata 7.72/bulan, New Vios menurun rata-rata 0.33/bulan, New Corolla meningkat rata-rata 0.13/bulan, New Camry menurun rata-rata 0.48/bulan, Etios menurun rata-rata 0.60/bulan, Yaris menurun rata-rata 0.57/bulan, New Yaris menurun rata-rata 3.38/bulan, Rush menurun rata-rata 3.12/bulan, New Kijang Innova menurun rata-rata 2.23/bulan, New Fortuner menurun rata-rata 2.23/bulan, All New Hilux menurun rata-rata 0,18/bulan, Hilux menurun rata-rata 0,14/bulan, dan New Hilux meningkat rata-rata 0,08/bulan.

PENDAHULUAN

Penjualan merupakan salah satu indikator terpenting dalam sebuah perusahaan. Jika tingkat penjualan tinggi maka laba yang dihasilkan juga tinggi sehingga perusahaan dapat bertahan dalam persaingan bisnis dan mengembangkan perusahaan. Prediksi penjualan merupakan cara yang efektif untuk meningkatkan keuntungan perusahaan. Data dan informasi penjualan sangat penting bagi perusahaan untuk merencanakan penjualan di masa yang akan datang. Misalnya data pelanggan, jumlah kendaraan, harga mobil, suku cadang dan jenis kendaraan. Peneliti menggunakan sistem kecerdasan buatan yang salah satunya adalah jaringan syaraf tiruan yang diimplementasikan ke dalam teknik prediksi penjualan. Hipotesis yang diajukan adalah merancang pemodelan jaringan syaraf tiruan untuk meningkatkan penjualan mobil menggunakan metode *Backprogation* menggunakan bahasa pemrograman *Matlab*. Bertujuan untuk mengetahui jumlah penjualan mobil pertahun agar persediaan mobil dapat diketahui dengan *software* sehingga diketahui bertambah atau berkurang.

STUDI LITERATUR

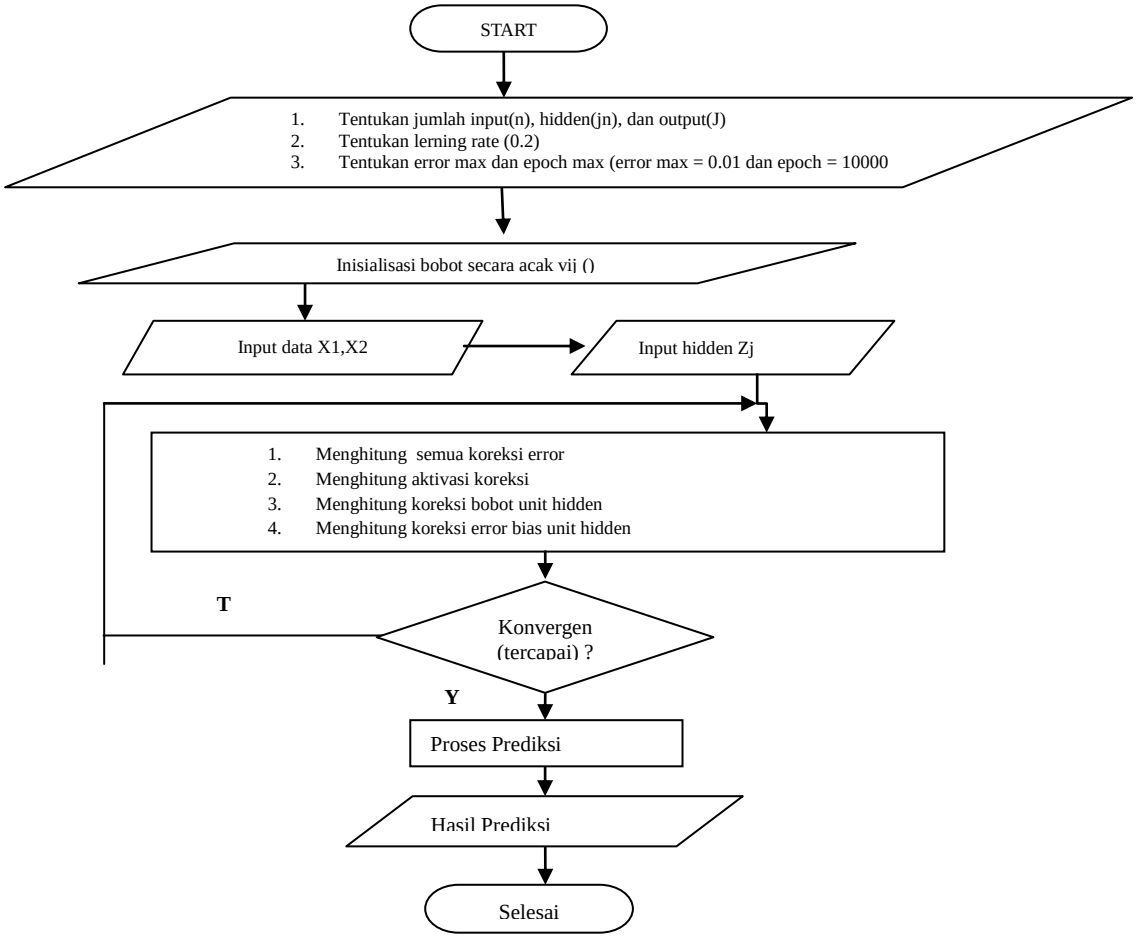
Jong Jek Siang dalam Suci Andriyani dan Norenta Sihotang (2018), Jaringan Syaraf Tiruan adalah sistem pemrosesan informasi yang memiliki karakteristik mirip dengan Jaringan Syaraf. Jaringan Syaraf Tiruan dibentuk sebagai generalisasi dari model matematis jaringan syaraf biologis, dengan asumsi bahwa: pemrosesan informasi terjadi pada banyak elemen sederhana (*neuron*), sinyal dikirim antar *neuron* melalui konektor, koneksi antar *neuron* memiliki bobot yang akan memperkuat atau melemahkan sinyal, untuk menentukan keluaran; setiap *neuron* menggunakan fungsi aktivasi (biasanya bukan fungsi linier) yang ditetapkan ke jumlah input yang diterima dan besarnya output ini kemudian dibandingkan dengan ambang batas. Jaringan saraf tiruan dicirikan oleh tiga hal berikut: pola koneksi antar *neuron* (disebut arsitektur jaringan), Metode untuk menentukan bobot tautan (disebut metode pelatihan/pembelajaran/algoritma) dan fungsi aktivasi.

Jurnal Teknik ITS (Volume: 6, Nomor: 1, 2017, oleh Razak dan Riksakomara Peramalan Jumlah Produksi Ikan Menggunakan Jaringan Saraf Tiruan *Backpropagation* (Studi Kasus: UPTD Pelabuhan Perikanan Banjarmasin) menyatakan jumlah produksi tangkapan ikan tidak pasti setiap bulan yang berdampak pada aktivitas penyediaan es pendingin dan penyediaan air bersih untuk proses produksi ikan. Jumlah produksi ikan akan diprediksi untuk beberapa periode yang akan datang dengan menggunakan data historis yang ada dengan menggunakan metode *Backpropagation*.

METODE

Menurut Prastyo 2014 *backpropagation* merupakan jaringan syaraf terawasi. Dalam pembelajaran teknik *feedforward learning* pada *backpropagation* adalah untuk mendapatkan nilai *error* keluaran pada jaringan syaraf tiruan *backpropagation*. Setelah mendapatkan nilai *error*, lakukan *backward learning* dengan mengubah atau memperbaharui nilai bobot. Maksud dari kedua penelitian tersebut adalah agar nilai *error* yang dihasilkan konvergen dengan nilai 0. Menurut F. Suhandi 2009, *backpropagation* adalah jenis pelatihan terkontrol yang menggunakan pola penyesuaian bobot untuk mencapai nilai error minimum antara output yang diprediksi dan keluaran sebenarnya.

Algoritma *backpropagation* memiliki 3 *layer* yaitu input *layer*, *hidden layer*, *output layer*, setiap *layer* berisi *neuron*/*node*. Pada lapisan input tujuan dari *node*/*neuron* untuk menerima input setelah banyak variabel telah diinput. *Neuron* lapisan tersembunyi berfungsi untuk menerima nilai dari lapisan masukan yang telah dikalikan dengan nilai bobot. Pada lapisan tersembunyi, nilai input diubah menjadi nilai lapisan tersembunyi dengan menggunakan fungsi, setelah itu nilai diteruskan ke lapisan output. Lapisan keluaran menerima nilai lapisan tersembunyi yang telah dihitung dengan nilai bobot. Lapisan keluaran digunakan untuk mendapatkan nilai keluaran. Setelah mendapatkan nilai output, dicari nilai *error*nya dengan membandingkannya dengan nilai target. Jika nilai output sama dengan nilai target, maka pembelajaran selesai, jika tidak mendekati target, pembelajaran dilanjutkan dengan melakukan teknik *backward*, yaitu pembelajaran dari output kembali ke input dengan memperbarui berat. Setelah pemutakhiran bobot selesai, proses pembelajaran dilanjutkan hingga konvergen dengan nilai target.



Gambar 1 *Flowchart* Perhitungan Sistem Metode *Backpropagation*

Gambar diatas merupakan proses *flowchart* perhitungan manual yang dilakukan dengan metode *backpropagation* untuk memprediksi penjualan mobil mulai dari proses *start*. Kemudian tentukan jumlah input, *hidden*, dan input pertama, tentukan *lerning rate* dan tentukan *max error* dan *epoch*. Setelah itu, proses penentuan inisialisasi bobot dilakukan secara acak dari bobot yang telah ditentukan secara acak. Selanjutnya input data X1, X2 karena disini penulis menggunakan 2 input. Kemudian melakukan proses tersembunyi yang dimulai dengan menghitung koreksi kesalahan, kemudian menghitung aktivasi koreksi, kemudian menghitung

koreksi bobot unit tersembunyi dan menghitung koreksi kesalahan bias unit tersembunyi. Jika proses kalkulasi konvergen atau mendekati capaian prediksi maka proses terus selesai, jika tidak maka proses kalkulasi lapisan tersembunyi akan kembali.

Langkah – langkah perhitungan dalam pelatihan backpropagation dijabarkan sebagai berikut:

1. Inisialisasi bobot
   Pertama kali yang dilakukan adalah penentuan bobot / *weight* secara acak pada setiap *node* yang berada pada input layer, hidden layer, dan output layer.
2. Akitivasi *hidden layer*
   Menghitung nilai total masukan pada masing *neuron* / *node* pada *hidden layer*

$$n_h(p) = \sum_{i=1}^{r} x_i(p) \times w_{ih}(p) \quad (1)$$

   $n_n(p)$: nilai total bobot yang masuk ke *hidden neuron*
   x_1 : nilai input
   $w_{in}(p)$: nilai *weight* yang menuju ke *hidden layer*
   r : jumlah masukan ke *hidden* neuron
   Setelah mendapatkan nilai total masukan pada setiap *hidden layer*, maka dilakukan perhitungan fungsi aktivasi pada setiap *neuron* di *hidden layer*.

$$y_h(p) = \frac{1}{1 + e^{-n_h(p)}} \quad (2)$$

   $y_h(p)$: nilai aktivasi *hidden neuron*.
3. Aktivasi *output layer*
   Menghitung total nilai yang masuk ke output *neuron*

$$n_o(p) = \sum_{h=1}^{m} y_h(p) \times w_{ho}(p) \quad (3)$$

   $n_o(p)$: nilai total bobot yang masuk ke *output* neuron
   y_h : nilai aktivasi dari neuron *hidden layer*
   $w_{ho}(p)$: nilai *weight* yang menuju ke *output layer*
   m : Jumlah *neuron* yang ada pada *hidden layer*
   Setelah mendapatkan nilai total masukan ke *neuron output layer*, maka dilakukan perhitungan fungsi aktivasi pada setiap *neuron* di *output layer*

$$y_o(p) = \frac{1}{1 + e^{-n_o(p)}} \quad (4)$$

   $y_o(p)$: nilai aktivasi *output neuron*
4. Gradien *error layer output*
   Menghitung nilai *error* pada setiap *neuron output layer*
   $eo(p) = ydo(p) - yo(p)$
   $e_o(p)$: nilai *error* dari *neuron output layer*
   $y_{do}(p)$: nilai target *neuron* pada *output layer*
   $y_o(p)$: nilai aktivasi *neuron* pada *output layer*
   Setelah mendapatkan nilai *error neuron* dari *output layer*, lalu dicari nilai *gradien error* dari *neuron* tersebut.
   $\delta o\ (p) = yo(p) \times [1 - yo(p)] \times eo(p)$
   $\delta_o\ (p)$: nilai *gradien error*
   $y_o(p)$: nilai aktivasi *output neuron*
   $e_o(p)$: nilai *error* dari *neuron output layer*
5. Perhitungan *update weight hidden layer* ke *output layer*
   Menghitung delta bobot
   $\Delta w_{ho}(p) = \eta \times y_h(p) \times \delta_o\ (p)$
   $\Delta w_{ho}(p)$: nilai delta bobot
   η : *learning rate*
   $y_h(p)$: nilai aktivasi *neuron hidden layer*
   $\delta_o\ (p)$: nilai *gradien error* Setelah didapat delta bobot kemudian melakukan *update* bobot *weight* dari *hidden layer* menuju ke output layer.
   $w_{ho}(p+1) = w_{ho}(p) + \Delta w_{ho}(p)$
   $w_{ho}(p+1)$: *weight* baru dari *neuron hidden layer* ke *neuron output* layer $w_{ho}(p)$: *weigh*t lama dari *neuron hidden layer* ke *neuron output* layer $\Delta w_{ho}(p)$: nilai delta bobot

6. Perhitungan *update weight input layer ke hidden layer*
   Menghitung *gradien* error pada *hidden layer*

$$\delta_h(p) = y_h(p) \times [1 - y_h(p)] \times \sum_{o=1}^{l} \delta_o(p).w_{ho}(p) \quad (5)$$

   $\delta_h(p)$: nilai *gradien error* dari *neuron hidden layer*
   $y_h(p)$: nilai aktivasi pada *hidden layer*
   l :banyak *neuron output layer* yang terhubung ke *neuron hidden layer*
   $\delta o\ (p)$: nilai *gradien error neuron output layer*
   $who(p)$:nilai *weight* lama *neuron hidden layer* ke *neuron output layer* Menghitung delta bobot dari input *layer* ke *hidden layer*

$$\Delta w_{ih}(p) = \eta \times x_i(p) \times \delta_h(p)$$

   $\Delta w_{ih}(p)$: nilai delta bobot untuk *hidden layer* ke input *layer*
   η : *learning rate*
   $x_i(p)$: nilai input
   $\delta_h\ (p)$: nilai *gradien error neuron* pada *hidden layer*
   Mengupdate nilai *weight* input *layer* ke *hidden layer*

$$wih(p+1) = wih(p) + \Delta wih(p)$$

   $w_{ih}(p+1)$: weight baru dari *input layer* ke *hidden layer*
   $w_{ih}(p)$: nilai *weight* lama
   $\Delta w_{ih}(p)$: delta bobot dari *hidden layer* ke *input layer*
7. Nilai iterasi (p) dinaikan sebanyak 1 dan diulangi prosesnya dari langkah 2 sampai nilai *error* tercapai.

HASIL

Data tersebut akan dilatih dengan pengaruh model algoritma jaringan syaraf tiruan. Jumlah data masukan, jumlah lapisan tersembunyi dan keluaran jumlah Penjualan Mobil digunakan untuk menghasilkan iterasi tercepat dengan nilai lapisan tersembunyi berubah. Hasil pelatihan dilakukan dalam dua tahap yaitu tahap pelatihan data yang dilatih, dan tahap pengujian data baru yang belum pernah dilatih yang terdiri dari 24 data yaitu jumlah penjualan mobil selama 2 tahun, 2016 dan 2017. Setelah itu jaringan akan diuji dengan 12 data baru yaitu jumlah Penjualan Mobil tahun 2018. Hal ini berfungsi untuk menguji seberapa besar jaringan syaraf tiruan dapat mengenali data baru.

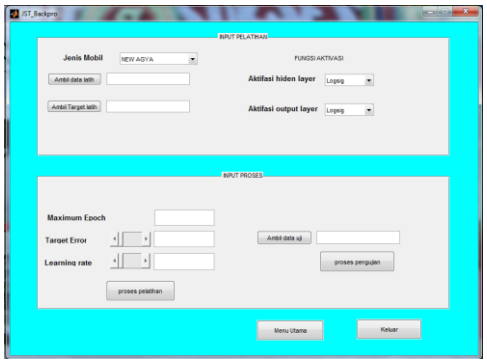
Data input diambil dari data Penjualan Mobil selama 3 tahun, 2016, 2017 dan 2018 yang akan dilatih pada jaringan 12 data pelatihan, 12 data target pelatihan dan 12 data target. Data tersebut akan digunakan untuk menguji keakuratan sistem dalam mengenali data masukan lainnya. Semua data variabel akan dipisahkan menjadi dua bagian, yaitu data input dan data output. Data masukan berfungsi sebagai proses pelatihan dan pengujian. Sedangkan data keluaran sebagai data target pencapaian proses

PEMBAHASAN

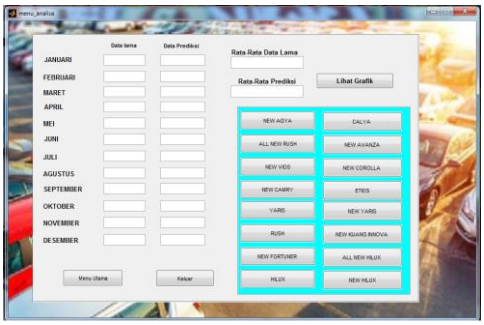
Menu *ANN_Backpro* berisi proses input data yang akan dilatih dan diuji menggunakan metode *backpropagation*. Dengan menginput data *training* dan *target training*, kemudian memilih fungsi aktivasi dan melanjutkan ke proses input *epoch* maksimum, target *error* dan *learning rate* kemudian klik tombol proses *training*. Dan langkah selanjutnya adalah proses input data pengujian, setelah mengklik tombol proses pengujian, Anda akan mendapatkan hasil prediksi data. Tampilan menu *ANN_Backpro* untuk memprediksi jumlah penjualan mobil menggunakan metode *backpropagation* seperti terlihat pada Gambar 2.

Menu analisis data ini dirancang untuk menampilkan hasil perbandingan antara data lama dan data prediksi. Pilih tombol sesuai dengan data yang telah dilatih pada menu JST_*Backpropagation* untuk menampilkan menu hasil. Tampilan menu analisis data memprediksi jumlah penjualan mobil menggunakan metode *backpropagation*, seperti terlihat pada Gambar 3.

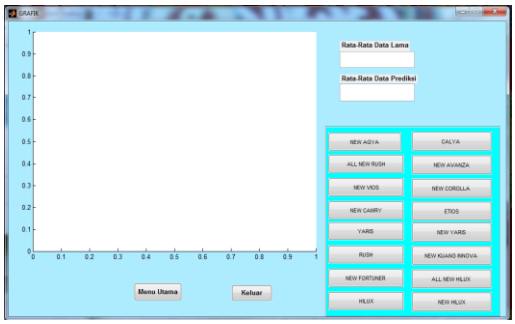
Menu grafik data ini berisi data rata-rata, baik data rata-rata penjualan lama maupun data prediksi rata-rata. Pilih tombol sesuai dengan data yang telah dilatih pada menu JST_*Backpropagation* untuk menampilkan hasilnya. Tampilan menu grafik data untuk memprediksi jumlah penjualan mobil dengan metode *backpropagation* seperti pada Gambar 4.



Gambar 2 Menu *ANN_Backpro* untuk Prediksi Jumlah Penjualan Mobil Menggunakan Metode *Backpropagation*



Gambar 3 Analisis Menu Data Prediksi Jumlah Penjualan Mobil Menggunakan Metode *Backpropagation*



Gambar 4 Data Grafik Menu Prediksi Jumlah Penjualan Mobil Menggunakan Metode *Backpropagation*

KESIMPULAN

Kesimpulannya data penjualan mobil dapat dikenali oleh sistem jaringan syaraf tiruan dengan metode *backpropagation*, hasil pengujian mengalami kenaikan dan penurunan. Prediksi penjualan New Agya meningkat rata-rata 5.99/bulan, Calya meningkat rata-rata 5.99/bulan, All New Rush meningkat rata-rata 12.06/bulan, New Avanza meningkat rata-rata 7.72/bulan, New Vios menurun rata-rata 0.33/bulan, New Corolla meningkat rata-rata 0.13/bulan, New Camry menurun rata-rata 0.48/bulan, Etios menurun rata-rata 0.60/bulan, Yaris menurun rata-rata 0.57/bulan, New Yaris menurun rata-rata 3.38/bulan, Rush menurun rata-rata 3.12/bulan, New Kijang Innova menurun rata-rata 2.23/bulan, New Fortuner menurun rata-rata 2.23/bulan, All New Hilux menurun rata-rata 0,18/bulan, Hilux menurun rata-rata 0,14/bulan, dan New Hilux meningkat rata-rata 0,08/bulan.

REFERENSI

Hedriyanto, 2013. Analisa Retrun Saham Sektor Di Bursa Efek Indonesia Menggunakan Model Arbitrage Princing Theory.

Ibrahim M.PD.I. 2014. *Pengantar Jaringan Saraf Tiruan*. CV. Andi Offset, Yogyakarta.

Jek Jong Siang. 2009. *Jaringan Syaraf Tiruan dan Pemogramannya Menggunakan Matlab*. CV. Andi Offset, Yogyakarta.

Sugiarti. 2013. Kecerdasan Buatan. CV.Andi Offset, Yogyakarta.

Irawan dkk. 2017. Analisis Penambahan Nilai Momentum Pada Prediksi Produktivitas Kelapa Sawit Menggunakan Backpropagation. Jurnal Informatika. Vol. 1, No. 2.

Rinta Berutu. 2017. Iplementasi Jaringan Syaraf Tiruan Untuk Memprediksi Penjualan Kosmetik Decorative Dengan Metode Backpropagation. Jurnal Informatika. Vol. 12, No. 1.

Razak dan Riksakomara. 2017. Peramalan Jumlah Produksi Ikan dengan Menggunakan Backpropagation Neural Network (Studi Kasus: UPTD Pelabuhan Perikanan Banjarmasin. Jurnal Informatika. Vol 6, No. 1.

Suci Andriyani, dkk. 2018. Implementasi Motode Backpropagation Untuk Prediksi Harga Jual Kelapa Sawit Berdasarkan Kualitas Buah.Jurnal Informatika. Vol. 4, No. 2.

Yatini Indra. 2010. *Flowchart, Algoritma, dan Pemrograman Menggunakan Bahasa C++ Builder.* Penerbit Graha Ilmu, Yogyakarta.