

**MONOGRAF**  
**ANALISIS KOMPARASI PREDIKSI BEBAN LISTRIK**  
**JANGKA PENDEK DENGAN PENDEKATAN**  
**BACKPROPAGATION NEURAL NETWORKS**

**Ihdalhubbi Maulida**



**pena persada**

**PENERBIT CV. PENA PERSADA**

**MONOGRAF**  
**ANALISIS KOMPARASI PREDIKSI BEBAN LISTRIK**  
**JANGKA PENDEK DENGAN PENDEKATAN**  
**BACKPROPAGATION NEURAL NETWORKS**

**Penulis:**

Ihdalhubbi Maulida

**ISBN:** 978-623-315-721-6

**Design Cover:**

Retnani Nur Brilliant

**Layout:**

Hasnah Aulia

**Penerbit CV. Pena Persada**

**Redaksi:**

Jl. Gerilya No. 292 Purwokerto Selatan, Kab. Banyumas  
Jawa Tengah

Email : [penerbit.penapersada@gmail.com](mailto:penerbit.penapersada@gmail.com)

Website: [penapersada.com](http://penapersada.com) Phone: (0281) 7771388

**Anggota IKAPI**

All right reserved  
Cetakan pertama: 2021

Hak Cipta dilindungi oleh undang-undang. Dilarang memperbanyak karya tulis ini dalam bentuk apapun tanpa izin penerbit

## KATA PENGANTAR

Segala puji dan syukur dipanjatkan kehadirat Tuhan Yang Maha Esa atas segala rahmat, karunia dan kesempatan yang telah diberikan sehingga penulis dapat menyusun buku ini. Buku yang berjudul Analisis Komparasi Prediksi Beban Listrik Jangka Pendek Dengan Pendekatan Backpropagation Neural Networks dapat dipergunakan untuk menambah wawasan dan pengetahuan tentang penggunaan algoritma Backpropagation Neural Networks kepada pembaca sehingga menambah semangat dan motivasi kepada penulis untuk dapat terus meningkatkan cara menulis dan membagikan ilmu kepada pembaca.

Penulis mengucapkan terima kasih kepada rekan-rekan, keluarga dan berbagai pihak yang telah mendukung atas kelancaran pembuatan buku ini. Kritik dan saran untuk penulis diharapkan menjadikan bahan masukan yang dapat meningkatkan cara menulis dan semangat serta motivasi untuk membagikan ilmu pengetahuan kepada pembaca. Semoga buku ini dapat memberikan manfaat yang sebesar-besarnya kepada para pembaca.

## ABSTRAK

Artificial Neural Network kebanyakan dipergunakan untuk memprediksi. Untuk pilihan prediksi yang dipakai dalam penelitian ini adalah Backpropagation dengan membandingkan empat metode yaitu *Levenberg-Marquard* (trainlm), *Gradient Momentum* (traindm), *Gradient Descent* (traingda) dan *Scala Conjugugate Gradient* (trainscg) yang diukur nilai *Mean Square Error* (MSE).

Data beban listrik dari kota Berabai sebagai data prediksi dengan model prediksi berdasarkan rentet waktu yang pendekatannya dengan menggunakan metode Backpropagation dengan membandingkan nilai MSE dari empat model tersebut, maka metode *Levenberg-Marquard* (trainlm) dapat memenuhi tingkat error yang lebih paling rendah yaitu  $1,6796e-12$ .

Kata Kunci: *Beban Listrik, ANN, backpropagation, gradient descent, momentum, Levenberg.*

## DAFTAR ISI

KATA PENGANTAR .....	iii
ABSTRAK.....	iv
DAFTAR ISI .....	v
BAB I PENDAHULUAN.....	1
A. Latar Belakang.....	1
BAB II LANDASAN TEORI .....	6
A. Karakteristik Beban Tenaga Listrik .....	6
1. Kebutuhan (Demand) .....	6
2. Kebutuhan Maksimum.....	7
3. Beban Terpasang .....	7
4. Faktor Beban (load factor).....	7
5. Kurva Beban dan Beban Puncak .....	8
6. Model Prediksi.....	8
7. Model Prediksi Algoritma .....	9
8. Model Prediksi.....	10
9. Analisis Prediksi.....	12
B. Data Mining.....	13
C. Time Series (Rentet Waktu) .....	16
D. Neural Network .....	17
1. Pengertian Neural Network .....	17
2. Backpropagation .....	22
E. Arsitektur Backpropagation.....	26
1. Fungsi Aktivasi Sigmoid .....	27
2. Algoritma Pelatihan.....	29
3. Optimalitas Arsitektur Backpropagation.....	34
4. Backpropagation Dalam Peramalan .....	36
F. Kerangka Pemikiran.....	36
BAB III METODE PENELITIAN.....	38
A. Metode Penelitian .....	38
B. Metode Pengumpulan data .....	39
C. Metode Pengolahan Data Awal .....	39
D. Eksperimen dan Pengujian Model/Metode.....	41
BAB IV HASIL DAN PEMBAHASAN.....	52
A. Hasil Eksperimen.....	52

BAB V KESIMPULAN .....	58
A. Kesimpulan .....	58
B. Saran.....	58
DAFTAR PUSTAKA .....	59

# BAB I

## PENDAHULUAN

### A. Latar Belakang

Tenaga listrik mempunyai peranan penting bagi negara dalam menunjang pembangunan disegala bidang dan meningkatkan kesejahteraan rakyat. Mengingat arti penting tenaga listrik tersebut, maka dalam rangka penyelenggaraan penyediaan tenaga listrik yang lebih merata, andal, dan berkelanjutan diperlukan suatu perencanaan yang komprehensif, dengan cakrawala nasional. Karena itu pembangunan infrastruktur ketenagalistrikan harus menganut asas manfaat, efisiensi berkeadilan, berkelanjutan, optimalisasi ekonomi dalam pemanfaatan sumber daya energi, mengandalkan pada kemampuan sendiri, kaidah usaha yang sehat, keamanan dan keselamatan, kelestarian fungsi lingkungan, dan otonomi daerah. [1]

Sesuai dengan misi sector ketenagalistrikan yaitu menyelenggarakan pembangunan sarana penyediaan dan penyaluran tenaga listrik untuk memenuhi kebutuhan tenaga listrik nasional [1]. Maka negara perlu mengantisipasi kekurangan pasokan tenaga listrik, sehingga perlu adanya peramalan beban dan kebutuhan energi listrik.

Peramalan pada dasarnya merupakan suatu dugaan atau prakiraan mengenai terjadinya suatu kejadian atau peristiwa dimasa yang akan datang. Dalam kegiatan perencanaan peramalan merupakan kegiatan mula dari proses tersebut. [2]

Ramalan di bidang tenaga elektrik pada dasarnya merupakan ramalan kebutuhan energi elektrik (watt jam) dan ramalan beban tenaga elektrik (watt). Keduanya sering disebut

dengan istilah *Demand and Load Forecasting*. Hasil peramalan ini dipergunakan untuk membuat rencana pemenuhan kebutuhan maupun pengembangan penyediaan tenaga elektrik setiap saat secara cukup dan baik serta terus menerus. [2]

Prakiraan pada dasarnya ialah menduga lebih awal mengenai peristiwa atau keadaan diwaktu yang akan datang. Dalam kegiatan perencanaan, prakiraan merupakan kegiatan mula dari suatu proses perencanaan tersebut [3]

Dalam hal ini perlu disadari bahwa semakin jauh jangka waktu ke depan maka semakin sulit dan semakin ketidakpastian, tidak tentunya kejadian atau peristiwa terutama bila hal tersebut akan dinyatakan dalam angka-angka. Karena itu cara apapun (metode) yang digunakan hanya akan dapat memberikan suatu nilai perkiraan [3].

Kebutuhan tenaga listrik di Provinsi Kalimantan Selatan dipasok oleh satu sistem interkoneksi melalui jaringan transmisi 150 kV yaitu Sistem Barito dan beberapa sistem terisolasi, yaitu Sistem Batulicin/Pagatan, Sungai Kupang, dan Kotabaru. Beban puncak tahun 2011 mencapai 275,97 MW. [1]

Total kapasitas terpasang pembangkit tenaga listrik yang ada di Provinsi Kalimantan Selatan sampai dengan tahun 2011 adalah sebesar 439.82 MW. Adapun rincian pembangkit tenaga listrik tersebut adalah: PLTU Asam-Asam 130 MW, Indocement 5 MW, Tanjung Alam 3 MW dan Wijaya 6 MW. PLTG Trisakti 21 MW. PLTD tersebar 244,82 MW. PLTA Riam Kanan 30 MW. [1]

Sampai dengan tahun 2011, penjualan tenaga listrik untuk sistem kelistrikan Provinsi Kalimantan Selatan mencapai 1.467 GWh dengan komposisi penjualan per sektor pelanggan untuk rumah tangga adalah 940,1 GWh (64,1%), bisnis 259,8 GWh (17,7%), industri 137,9 GWh (9,4%), dan publik 129,3 GWh (8,81%). Rasio elektrifikasi untuk tahun 2011 adalah

75,65%. Adapun rasio desa berlistrik tahun 2011 adalah 99,90%. [1]

Diasumsikan bahwa pertumbuhan penduduk rata-rata tahun 2012 – 2031 sebesar 2% per tahun, pertumbuhan ekonomi sebesar 6,4% – 7,5% pada periode 2012 – 2014 dan sekitar 8,0% – 9,0% pada periode 2015 – 2031 yang dibarengi oleh penurunan inflasi dari sebesar 6,5% pada periode 2012 – 2014 menjadi 3,0% pada tahun 2025, rasio elektrifikasi ditargetkan menjadi sekitar 99,9% pada tahun 2020 atau penambahan konsumen rumah tangga diproyeksikan rata-rata sekitar 48.341 sambungan per tahun, dan tarif listrik disesuaikan berdasarkan inflasi. [1]

Berdasarkan asumsi dan target tersebut, maka diproyeksikan permintaan energi listrik untuk periode 2012-2031 akan tumbuh rata-rata sebesar 10,5% per tahun sehingga pada tahun 2031 kebutuhan tenaga listrik diperkirakan mencapai 10,6 TWh. Sebagai upaya untuk memenuhi pertumbuhan beban puncak rata-rata sekitar 10,2% per tahun hingga tahun 2031, maka dibutuhkan tambahan daya rata-rata sekitar 135 MW per tahun [1]

Data beban yang tercatat oleh PLN menunjukkan adanya peningkatan terhadap beban puncak. PLN perlu mengetahui kondisi beban puncak agar data-data historis mengenai beban penyulang termasuk data rentet-waktu dapat dianalisa dan diprediksi dengan pendekatan statistik dan atau *softcomputing*.

Memprediksi nilai suatu besaran pada kondisinya dimasa yang akan datang dengan tepat adalah suatu pekerjaan yang sulit, apalagi jika besaran tersebut dipengaruhi oleh banyak faktor. Tetapi dilain pihak seringkali hasil prediksi tersebut menjadi salah satu masukan yang penting dalam menyusun rencana masa depan. Berdasarkan hal tersebut maka dapat disimpulkan bahwa kebutuhan PLN adalah bagaimana memprediksi beban listrik jangka pendek dengan lebih akurat. Untuk itu dipertimbangkan pendekatan dengan *softcomputing*.

*Artificial Neural Network* (ANN) sejak diperkenalkan pada sekitar tahun 1940 telah banyak diimplementasikan pada berbagai bidang keilmuan, ANN banyak digunakan untuk melakukan prediksi atau peramalan [4]. Dalam hal tersebut untuk dapat memprediksi kebutuhan beban listrik jangka pendek dapat diimplementasikan kedalam ANN menggunakan algoritma pembelajaran *Back-propagation* sehingga hasil akurasi yang diperoleh lebih akurat.

Dari hasil penelitian yang dilakukan oleh Indra Pranata yang menunjukkan bahwa nilai *Mean Square Error* (MSE) dengan penerapan *Back-Propagation Neural Network* (BPNN) menggunakan metode *levenberg-marquardt* (lm) dan *scaled conjugate gradient* (scg) tidak menghasilkan MSE yang terendah yaitu:

**Tabel 1 Hasil Uji MSE Struktur BPNN-In dengan BPNN-scg [5]**

Arsitektur FFN			BPNN (MSE)	
Input	Hidden-1	Out	lm	Scg
9	2	1	48,82	54,78
9	4	1	44,68	63,19
9	9	1	53,49	51,23
<b>9</b>	<b>18</b>	<b>1</b>	40,52	61,52
18	2	1	47,48	50,44
<b>18</b>	<b>6</b>	<b>1</b>	43,30	49,43
18	9	1	47,22	44,79
18	18	1	43,77	61,38
18	24	1	44,10	45,79
18	32	1	45,86	53,30

Sehingga berdasarkan dari data tersebut diperlukan pengujian terhadap dataset yang berbeda dengan membandingkan beberapa teknik untuk menghasilkan MSE yang terendah dengan menerapkan BPNN.

Untuk dapat mengetahui prediksi beban penyulang listrik jangka pendek sesuai dengan berjalannya waktu dengan menerapkan *Back-Propagation Neural Network* maka penelitian ini akan mencari hasil akurasi perbandingan metode *Levenberg-Marquard* (trainlm), *Gradient Momentum* (traindm), *Gradient Descent* (traingda) dan *Scala Conjuguate Gradient* (trainscg) untuk menentukan prediksi beban penyulang listrik jangka pendek sesuai dengan berjalannya waktu”.

Tujuan dari penelitian ini adalah menerapkan *Back-Propagation Neural Network* dalam prediksi beban penyulang listrik jangka pendek sesuai dengan berjalannya waktu agar PLN tidak mengalami kesulitan dalam menyediakan pasokan listrik bagi daerah yang mengalami kekurangan tenaga listrik.

Dari hasil penelitian ini dapat dimanfaatkan untuk penerapan metode prediksi agar operator pembangkit dapat memprediksi beban lebih akurat. Selain itu dapat memberikan gambaran predeksi dalam studi kasus data rentet waktu beban listrik dalam penerapan *Back-Propagation Neural Network*. Dan hasil dari penelitian diharapkan mampu menjadi alat prediksi beban listrik bagi PLN jaringan distribusi Kalselteng.

## **BAB II**

### **LANDASAN TEORI**

#### **A. Karakteristik Beban Tenaga Listrik**

Secara umum beban yang dilayani oleh sistem distribusi ini dibagi dalam beberapa sektor yaitu sektor perumahan, sektor industri, sektor komersial dan sektor usaha. Masing-masing sektor beban tersebut mempunyai karakteristik-karakteristik yang berbeda, sebab hal ini berkaitan dengan pola konsumsi energi pada masing-masing konsumen di sektor tersebut. Karakteristik beban yang banyak disebut dengan pola pembebanan pada sektor perumahan ditunjukkan oleh adanya fluktuasi konsumsi energi elektrik yang cukup besar. Hal ini disebabkan konsumsi energi elektrik tersebut dominan pada malam hari. Sedang pada sektor industri fluktuasi konsumsi energi sepanjang hari akan hampir sama, sehingga perbandingan beban puncak terhadap beban rata-rata hampir mendekati satu. Beban pada sektor komersial dan usaha mempunyai karakteristik yang hampir sama, hanya pada sektor komersial akan mempunyai beban puncak yang lebih tinggi pada malam hari. [2]

Tujuan utama dari sistem distribusi tenaga listrik ialah mendistribusikan tenaga listrik dari gardu induk atau sumber ke sejumlah pelanggan atau beban. Suatu faktor utama yang paling penting, dalam perencanaan sistem distribusi adalah karakteristik dari berbagai beban [2].

#### **1. Kebutuhan (Demand)**

Kebutuhan sistem listrik adalah beban pada terminal terima secara rata-rata dalam suatu selang (interval) waktu

tertentu. Beban tersebut bisa dalam satuan Ampere, kiloAmpere, kiloWatts dan kiloVoltAmpere [2].

Kebutuhan beban listrik pada suatu daerah tergantung dari keadaan penduduk, pertumbuhan ekonomi, rencana pengembangannya dalam waktu dekat dan waktu yang akan datang. Sehingga kebutuhan mendatang sangat bergantung pada faktor-faktor yang dapat diketahui tersebut.

## **2. Kebutuhan Maksimum**

Kebutuhan maksimum dapat terjadi selama waktu satu jam, harian, mingguan, bulanan atau tahunan. Kebutuhan maksimum adalah sebagai kebutuhan yang terbesar yang dapat terjadi dalam suatu selang tertentu, biasanya terjadi dalam selang 15 menit, selang 30 menit atau dalam hal-hal tertentu 60 menit.

## **3. Beban Terpasang**

Beban terpasang dimaksudkan adalah jumlah kapasitas dari semua beban dengan kapasitas yang tertera pada papan nama (name plate) dan peralatan-peralatan listrik.

## **4. Faktor Beban (load factor)**

Faktor beban adalah perbandingan antara beban rata – rata terhadap beban puncak yang diukur dalam suatu periode tertentu. Beban rata – rata dan beban puncak dapat dinyatakan dalam kilowatt (kW), kilovolt –amper(kVA), amper (A) dan sebagainya, tetapi satuan dari keduanya harus sama. Faktor beban dapat dihitung untuk periode tertentu biasanya dipakai harian, bulanan atau tahunan.

Beban puncak yang dimaksud disini adalah beban puncak sesaat atau beban puncak rata-rata dalam interval tertentu (demand maksimum), pada umumnya dipakai demand maksimum 15 menit atau 30 menit.

## 5. Kurva Beban dan Beban Puncak

Kepadatan beban selalu dipakai sebagai ukuran dalam menentukan kebutuhan listrik. Sesuatu daerah kepadatan beban satuannya dapat berupa MVA/km<sup>2</sup> atau KVA/m<sup>2</sup> umumnya satuan yang dipakai adalah MVA/km<sup>2</sup>. Beban puncak (kebutuhan maksimum) didefinisikan sebagai beban (kebutuhan) terbesar/tertinggi yang terjadi selama periode tertentu. Periode tertentu dapat berupa sehari, sebulan maupun dalam setahun. Periode harian, yaitu variasi pembebanan trafo distribusi selama sehari. Selanjutnya beban puncak harus diartikan beban rata-rata selama selang waktu tertentu, dimana kemungkinan terjadinya beban tersebut.

## 6. Model Prediksi

Untuk melakukan perencanaan dalam bidang apapun, tentu harus ada metode yang baku yang digunakan. Ada berbagai model pendekatan untuk menyusun prakiraan kebutuhan tenaga listrik yang tersedia antara lain pendekatan ekonometrik, pendekatan proses, pendekatan time series, pendekatan *end-use*, pendekatan trend maupun gabungan dari berbagai model pendekatan perencanaan.

Dari beberapa metode tersebut, yang sering digunakan sebagai pendekatan untuk proyeksi kebutuhan energi listrik adalah metode pendekatan ekonometri dan pendekatan *end-use*. Perbedaan utama dari kedua metode tersebut adalah pada jenis data yang dimasukkan (data input). Pada model ekonometri, data yang digunakan

sebagai data masukan seperti pendapatan daerah, pendapatan per kapita dan data lain yang bersifat ekonomi, kemudian dihubungkan dengan kebutuhan energi.

Model yang digunakan dalam Prediksi harus dapat menggambarkan kaitan antara penjualan energi elektrik dengan variable lain yang ada dalam masyarakat seperti variable pendapatan dan tingkat konsumsi masyarakat.

Hubungan antara penjumlahan energi elektrik dengan variable-variable tersebut biasanya sangat rumit dan saling berkaitan. Oleh sebab itulah biasanya digunakan model pendekatan untuk memudahkan pembuatan ramalan. Ada tiga macam pendekatan, yaitu :

- a. **Model Mikro**, Model ini adalah model yang meninjau secara terperinci setiap komponen atau variable yang mempengaruhi penjualan energi elektrik.
- b. **Model Makro**, Model ini adalah model yang meninjau secara umum dengan menyederhanakan variable yang mempengaruhi penjualan energi elektrik.

Dalam penyusunan ramalan ini, dilakukan pembagian kebutuhan energi elektrik secara sektoral yang meliputi :

- a. Sektor Rumah Tangga
- b. Sektor Komersil
- c. Sektor Publik (pelayanan umum)
- d. Sektor Industri.

## 7. Model Prediksi Algoritma

Model ini adalah model yang dibuat untuk generalisasi dengan menyertakan pengaruh variabel secara detail atau sebaliknya, menyederhanakan variable yang mempengaruhi penjualan energi elektrik

## 8. Model Prediksi

Prediksi beban biasanya dibagi dalam dua bagian yakni :

### a. Prediksi Kebutuhan Energi Elektrik

Dengan langkah-langkah sebagai berikut:

- 1) Mengumpulkan data-data historis dari variable-variable yang mempengaruhi kebutuhana energi elektrik seperti data historis kependudukan, Produk Domestik Regoinal Bruto (PDRB), tingkat konsumsi energi masyarakat dan data produksi energi PLN untuk beberapa tahun terakhir, faktor beban, serta beberapa data lain.
- 2) Melakukan analisa awal terhadap masing-masing data yang diperoleh, kemudian menyusun secara sistematis menurut tiap sektor yang dipengaruhi.
- 3) Melakukan perhitungan matematis peramalan untuk masingmasing sektor berdasarkan data historis yang dijadikan variable.
- 4) Melibatkan beberapa kebijaksanaan yang telah ditetapkan sebagai suatu pembatasan yang harus diikuti.
- 5) Menyusun hasil ramalan kebutuhan masing-masing sektor dan ramalan secara keseluruhan untuk beberapa tahun ke depan dan menentukan tingkat pertumbuhan kebutuhan energi total pertahun.

### b. Peramalan Beban

Prediksi beban dilakukan setelah prediksi kebutuhan energi dilakukan. Pada dasarnya laju pertumbuhan kebutuhan energi sama dengan laju pertumbuhan beban rata-rata sedangkan untuk menghitung pertumbuhan beban puncak harus dicari

hubungan antara laju pertumbuhan beban puncak dengan langkah-langkah sebagai berikut:

- 1) Menentukan persamaan pertumbuhan kebutuhan energi elektrik total pertahun.
- 2) Mementukan persamaan pertumbuhan faktor beban pertahun dengan persamaan Gompertz.
- 3) Menentukan persamaan pertumbuhan beban pertahun dengan menghubungkan a dan b melalui persamaan pertama.
- 4) Menyusun hasil prediksi untuk beberapa tahun ke depan.

Terdapat tiga kelompok prediksi beban, yaitu :

- 1) Prediksi beban jangka panjang.

Prediksi beban jangka panjang adalah untuk jangka waktu diatas satu tahun. Dalam prediksi beban, masalah-masalah makro ekonomi yang merupakan masalah *ekstern* perusahaan listrik, merupakan faktor utama yang menentukan arah prediksi beban.

- 2) Prediksi beban jangka menengah.

Prediksi beban jangka menengah adalah untuk jangka waktu dari satu bulan sampai dengan satu tahun. Dalam prediksi beban jangka menengah, masalah-masalah manajerial perusahaan merupakan faktor utama yang menentukan.

- 3) Prediksi beban jangka pendek

Prediksi beban jangka pendek adalah untuk jangka waktu beberapa jam sampai satu minggu (168

jam). Besarnya beban untuk setiap jam ditentukan dengan memperhatikan *trend* beban di waktu lalu dengan memperhatikan berbagai informasi yang dapat mempengaruhi besarnya beban sistem.

## 9. Analisis Prediksi

Memprediksi nilai suatu besaran pada kondisinya dimasa yang akan datang dengan tepat adalah suatu pekerjaan yang sulit, apalagi jika besaran tersebut dipengaruhi oleh banyak faktor. Tetapi di lain pihak seringkali hasil prediksi tersebut menjadi salah satu masukan yang penting dalam menyusun rencana masa depan.

Suatu metode prediksi yang dapat digunakan untuk mengamati tingkat hubungan antar faktor yang mempengaruhi suatu besaran tertentu adalah Metode Kausal atau dapat juga disebut prediksi rentet waktu (*time-series prediction*). Salah satu model dari metoda ini adalah model regresi.

Model regresi adalah suatu model matematik yang memanfaatkan data masa lalu untuk menganalisa bentuk formulasi suatu variable terhadap variable yang lain, yang dapat digunakan dalam memprediksi pola kejadian pada masa yang akan datang. Model regresi dapat berupa regresi linier maupun regresi berganda. Pertanyaan yang sering muncul setelah ini adalah apakah penaksiran parameter dan hasil Prediksi dengan menggunakan model ini dapat dipercaya untuk digunakan lebih lanjut.

Model Koefisien beban. Metode ini dipakai untuk meramalkan beban harian suatu sistem tenaga listrik. Beban untuk setiap jam diberi koefisien yang menggambarkan besarnya beban pada jam tersebut dalam perbandingannya terhadap beban puncak. Koefisien-koefisien ini berbeda

untuk hari Senin sampai dengan Minggu dan untuk hari libur bukan minggu. Setelah didapat perkiraan kurva beban harian dengan metode koefisien, masih perlu dilakukan koreksi-koreksi berdasarkan informasi-informasi terakhir mengenai peramalan suhu dan kegiatan masyarakat.

Model yang lain adalah dengan memanfaatkan pendekatan *softcomputing*. Salah satunya adalah neural network.

## **B. Data Mining**

Istilah data mining memiliki beberapa padanan, seperti *knowledge discovery* ataupun *pattern recognition*. Istilah *knowledge discovery* atau penemuan pengetahuan tepat digunakan karena tujuan utama dari data mining memang untuk mendapatkan pengetahuan yang masih tersembunyi didalam bongkahan data. Istilah *pattern recognition* atau pengenalan pola pun tepat untuk digunakan karena pengetahuan yang hendak digali memang berbentuk pola-pola yang mungkin juga masih perlu digali dari dalam bongkahan data [9].

Secara sederhana, data mining merujuk pada ekstraksi atau "pertambangan" pengetahuan dari sejumlah data yang besar. Banyak istilah lainnya yang berarti data mining, seperti pertambangan pengetahuan dari data, ekstraksi pengetahuan, data/analisis pola, arkeologi data, dan pengerukan data. Data mining adalah kegiatan menemukan pola yang menarik dari data dalam jumlah besar, data dapat disimpan dalam database, data warehouse, atau penyimpanan informasi lainnya. Data mining berkaitan dengan bidang ilmu - ilmu lain, seperti database system, data warehousing, statistik, machine learning, information retrieval, dan komputasi tingkat tinggi. Selain itu, data mining didukung oleh ilmu lain seperti neural network, pengenalan pola, spatial data analysis, image database, signal processing [10].

Data mining didefinisikan sebagai proses menemukan pola-pola dalam data. Proses ini otomatis atau seringnya semiotomatis. Pola yang ditemukan harus penuh arti dan pola tersebut memberikan keuntungan, biasanya keuntungan secara ekonomi. Data yang dibutuhkan dalam jumlah besar [11]. Selain itu Data mining merupakan suatu proses menemukan korelasi baru yang bermakna, pola dan tren dengan memilah-milah sejumlah besar data yang disimpan dalam repositori, menggunakan teknologi pengenalan pola serta statistik dan teknik matematika. Data mining sebagai strategi pemecahan masalah secara umum dibagi menjadi beberapa kelompok fungsi berdasarkan tugas yang dapat dilakukan, yaitu [9]:

1. Fungsi deskripsi (description)

Menggambarkan sekumpulan data secara ringkas.

Data yang digambarkan berupa:

- Deskripsi grafis : diagram titik, histogram
- deskripsi lokasi : mean(rata-rata), median(nilai tengah), modus, kuartil, persentil
- Deskripsi keberagaman : range(rentang), varians dan standar deviasi

2. Fungsi estimasi (estimation)

Memperkirakan suatu hal dari sejumlah sample yang kita miliki(yg tidak kita ketahui) Estimasi hampir sama dengan klasifikasi, kecuali variable target. Estimasi lebih kearah numeric dari pada kearah kategori.

3. Fungsi prediksi (prediction)

Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada dimasa datang(memperkirakan hal yang belum terjadi). Kita bisa menunggu hingga hal itu terjadi untuk membuktikan seberapa tepat prediksi kita

4. Fungsi klasifikasi (classification)

Kegiatan menggolongkan, dengan menggunakan data historis(sebagai data yang digunakan untuk latihan dan

sebagai pengalaman). Dalam klasifikasi terdapat variabel prediktor dan target variable,

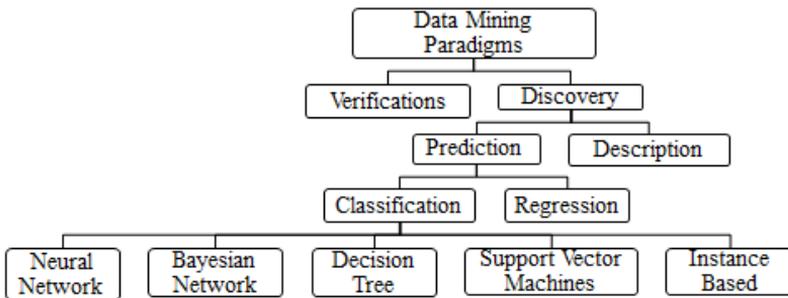
5. Fungsi pengelompokan (cluster)

Pengkulusteran merupakan pengelompokan record, pengamatan atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan. Kluster adalah kumpulan record yang memiliki kemiripan satu dengan yang lain dan memiliki ketidak miripan dengan record-record dalam cluster.

6. Fungsi assosiasi (association)

Tugas asosiasi dalam data mining adalah menemukan atribut yang muncul dalam satu waktu. Dalam dunia bisnis lebih umum disebut analisis keranjang biasa.

Ada banyak metode Data Mining yang digunakan untuk tujuan yang berbeda. Taksonomi yang disebut untuk membantu dalam memahami berbagai metode, interaksinya dan pengelompokan. Hal ini berguna untuk membedakan antara dua jenis utama Data Mining [10]:



**Gambar 1 Data Mining Taxonomy**

Berdasarkan pengelompokan data mining tersebut, masing masing memiliki kelompok fungsi antara lain:

- a. Fungsi Minor (tambahan) : deskripsi, estimasi, prediksi
- b. Fungsi Mayor (utama) : klasifikasi, pengelompokan, estimasi

Knowledge Discovery In Database (Penemuan Pengetahuan dalam Database) Data mining digambarkan sebagai proses pencarian pengetahuan yang menarik dalam database seperti pola , asosiasi, aturan, perubahan, keganjilan dan struktur penting dari sejumlah besar data yang disimpan pada bank data dan tempat penyimpanan informasi lainnya. Berikut merupakan proses KDD

- a. pemilihan data (data selection), pemilihan data relevan yang didapat dari basis data;
- b. pembersihkan data (data cleaning), proses menghilangkan noise dan data yang tidak konsisten atau data tidak relevan;
- c. pengintegrasian data (data integration), penggabungan data dari berbagai basisdata ke dalam satu basisdata baru;
- d. transformasi data, data diubah atau digabung ke dalam format yang sesuai untuk diproses dalam data mining;
- e. data mining, suatu proses di mana metoda diterapkan untuk menemukan pengetahuan berharga dan tersembunyi dari data;
- f. evaluasi pola (pattern evaluation), untuk mengidentifikasi pola-pola menarik untuk di representasikan kedalam knowledge based;
- g. representasi pengetahuan (knowledge presentation), visualisasi dan penyajian pengetahuan mengenai teknik yang digunakan untuk memperoleh pengetahuan yang diperoleh pengguna.

### C. *Time Series* (Rentet Waktu)

*Time series data* merupakan sekuens data yang nilainya berubah setiap interval waktu tertentu. *Time series* data dapat dipresentasikan dalam bentuk grafik atau kurva yang menunjukkan fungsi sebuah variabel data terhadap satuan waktu. Grafik yang dibangun dapat dimanfaatkan untuk menganalisis *trend* atau pola pada *time-series data*.

Prediksi *time series*, mengambil seri yang ada data dan perkiraan nilai data. Tujuannya adalah untuk mengamati atau model seri data yang ada untuk memungkinkan nilai-nilai data masa depan yang tidak diketahui dapat diperkirakan secara akurat. Contoh seri data meliputi serangkaian data keuangan (saham, indeks, harga, dan lain-lain), Data seri yang diamati dalam bentuk fisik (bintik matahari, cuaca, dll), dan seri data matematika (urutan Fibonacci, integral persamaan diferensial, dll). Ungkapan "time series" umum mengacu pada setiap seri data, apakah data tergantung pada interval waktu tertentu [12].

Ada dua tujuan utama analisis trend yaitu untuk memodelkan *time series data* dan untuk memprediksi *time series data*. Yang dimaksud dengan memodelkan *time series data* adalah menemukan mekanisme atau faktor-faktor yang menyebabkan terbentuknya *time series*. Yang dimaksud dengan memprediksi *time series data* adalah memprediksi nilai variabel *time series* pada suatu waktu yang akan datang.

#### **D. Neural Network**

##### **1. Pengertian Neural Network**

Jaringan syaraf tiruan (*artificial neural network*) adalah sistem komputasi yang arsitektur dan operasinya diilhami dari pengetahuan tentang sel syaraf biologis di dalam otak. Jaringan syaraf tiruan merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba menstimulasi proses pembelajaran pada otak manusia tersebut. Jaringan syaraf tiruan dapat digambarkan sebagai model matematis dan komputasi untuk fungsi aproksimasi non-linear, klasifikasi data cluster dan regresi non-parametrik atau sebuah simulasi dari koleksi model jaringan syaraf biologi. Model jaringan syaraf ditunjukkan dengan kemampuannya dalam emulasi, analisis, prediksi dan asosiasi. Kemampuan yang dimiliki jaringan syaraf tiruan dapat digunakan untuk belajar dan menghasilkan aturan

atau operasi dari beberapa contoh atau input yang dimasukkan dan membuat prediksi tentang kemungkinan output yang akan muncul atau menyimpan karakteristik input yang diberikan kepada jaringan syaraf tiruan. Salah satu organisasi yang sering digunakan dalam paradigma jaringan syaraf tiruan adalah perambatan galat mundur atau *backpropagation* [13].

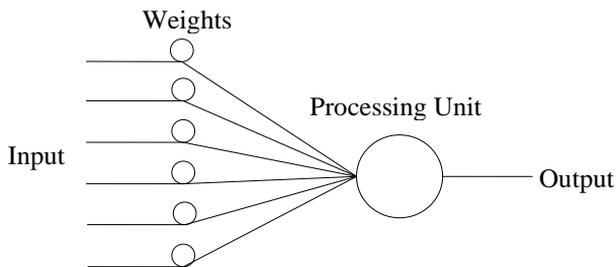
Jaringan syaraf tiruan terdiri dari beberapa neuron dan ada hubungan antar neuron- neuron seperti pada otak manusia. Neuron/selsaraf adalah sebuah unit pemroses informasi yang merupakan dasar operasi jaringan syaraf tiruan. Jaringan syaraf tiruan terdiri atas beberapa elemen penghitung tak linier yang masing-masing dihubungkan melalui suatu pembobot dan tersusun secara paralel. Pembobot inilah yang nantinya akan berubah (beradaptasi) selama proses pelatihan. Pelatihan perlu dilakukan pada suatu jaringan syaraf tiruan sebelum digunakan untuk menyelesaikan masalah. Hasil pelatihan jaringan syaraf tiruan dapat diperoleh tanggapa yang benar (yang diinginkan) terhadap masukan yang diberikan. Jaringan syaraf tiruan dapat memberikan tanggapan yang benar walaupun masukan yang diberikan terkena derau atau berubah oleh suatu keadaan [13].

Setiap neuron mempunyai keadaan internal yang disebut level aktivasi atau level aktivitas yang merupakan fungsi input yang diterima. Secara tipikal suatu neuron mengirimkan aktivitasnya kebeberapa neuron lain sebagai sinyal. Yang perlu diperhatikan adalah bahwa neuron hanya dapat mengirimkan satu sinyal sesaat, walaupun sinyal tersebut dapat dipancarkan ke beberapa neuron yang lain.

NN merupakan suatu sistem pemrosesan informasi yang memiliki karakteristik - karakteristik yang menyerupai jaringan syaraf biologi (Favset. 1994). Jaringan saraf tiruan memiliki 3 karakteristik, yaitu :

- a. Arsitektur Jaringan
- b. Algoritma Jaringan
- c. Fungsi Aktivasi.

Secara mendasar, sistem pembelajaran merupakan proses penambahan pengetahuan pada NN yang sifatnya kontinuitas sehingga pada saat digunakan pengetahuan tersebut akan dieksploitasikan secara maksimal dalam mengenali suatu objek. Neuron adalah bagian dasar dari pemrosesan suatu neural network. Dibawah ini merupakan bentuk dasar dari suatu neuron [14].



**Gambar 2 Bentuk dasar neuron**

Input merupakan masukan yang digunakan baik saat pembelajaran maupun dalam mengenali suatu objek. Seperti halnya otak manusia NN juga terdiri dari beberapa neuron dan ada hubungan antar neuron tersebut, hal ini disebut dengan bobot (weight).

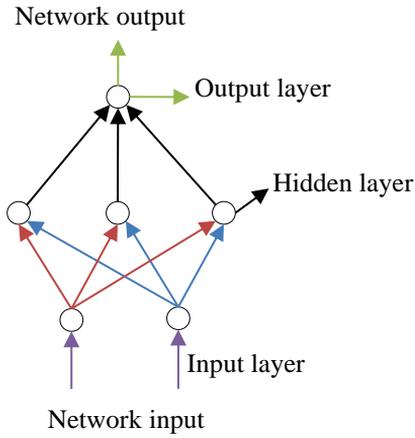
- a. Weight, beban yang selalu berubah setiap kali diberikan input sebagai proses pembelajaran.

- b. Processing Unit merupakan tempat berlangsungnya proses pengenalan suatu objek berdasarkan pembebanan yang diberikan.
- c. Output, keluaran dari hasil pengenalan suatu objek.

Keuntungan penggunaan Neural Network :

- a. Perangkat yang mampu untuk mengenali suatu objek secara non-linier.
- b. Mempermudah pemetaan input menjadi suatu hasil tanpa mengetahui proses sebenarnya.
- c. Mampu melakukan pengadaptasian terhadap pengenalan suatu objek
- d. Perangkat yang memiliki toleransi terhadap suatu kesalahan dalam pengenalan suatu objek.
- e. Neural Network mampu diimplementasikan pada suatu Hardware atau perangkat keras.
- f. Perangkat yang mampu diimplementasikan secara parallel.

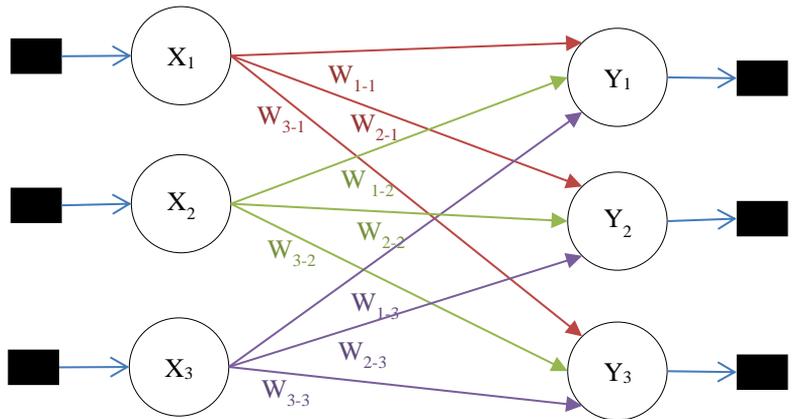
Arsitektur NN merupakan pola keterhubungan neuron, keterhubungan neuron - neuron inilah yang membentuk suatu jaringan :



**Gambar 3** Arsitektur NN

Arsitektur NN :

1. Single Layer Net



**Gambar 4.** Single Layer Net

Keterangan:

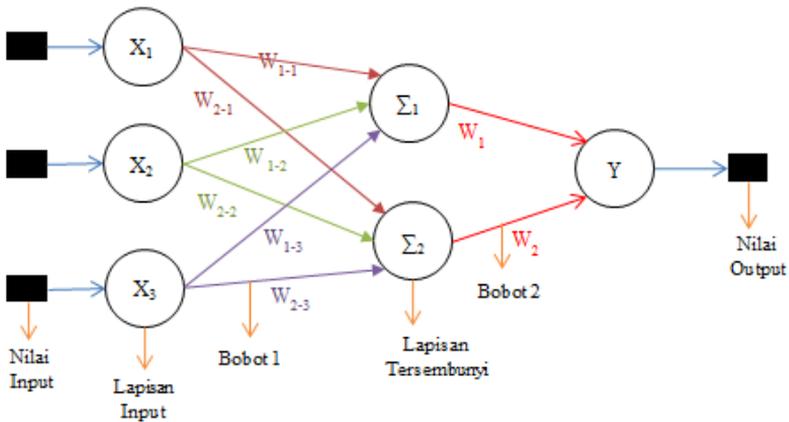
- a. n neuron sebagai input (sebanyak  $X_1, X_2, \dots, X_n$ )

- b. n neuron sebagai output (sebanyak  $Y_1, Y_2, \dots, Y_n$ )
- c. bobot masing-masing neuron (sebanyak  $W$ )

Jaringan dengan lapis tunggal hanya memiliki 1 lapis dengan bobot - bobot terhubung, jaringan ini hanya menerima input kemudian secara langsung akan mengelolanya menjadi output tanpa harus melalui lapisan tersembunyi.

## 2. Multi Layer Net

Jaringan dengan banyak lapisan memiliki suatu atau lebih lapisan yang terletak diantara lapisan input dan output (memiliki 1 atau lebih lapisan tersembunyi).



Gambar 5 Multi Layer Net

## 2. Backpropagation

*Backpropagation* merupakan algoritma pembelajaran yang terwarisi dan biasanya digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya. Algoritma *backpropagation*

menggunakan *error output* untuk mengubah nilai-nilai bobotnya dalam arah mundur (*backward*). Tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu untuk mendapatkan nilai error tersebut.

Algoritma *backpropagation* [15]:

a. Inisialisasi input, bias, epoch, learning rate, error, target, bobot awal dan bias.

b. Tahap Feedforward :

Hitung nilai masukan ( $z_{in}$ ) pada tiap pasangan elemen input pada hidden layer dengan formula:

$$z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

dimana  $v_{j0}$  = bias input,  $x$  = nilai,  $v$  = bobot

c. Jika kita menggunakan aktivasi sigmoid, hitung output ( $z$ ) dengan:

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}}$$

d. Hitung sinyal keluaran ( $y_{in}$ ) dari hidden layer untuk mendapatkan keluaran output layer dengan menggunakan persamaan:

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j w_{kj}$$

dimana  $w_{k0}$  = bias hidden

e. Jika kita menggunakan fungsi aktivasi sigmoid, hitung output ( $y$ ) dengan:

$$y_1 = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}}$$

f. Kemudian keluaran jaringan ( $y$ ) dibandingkan dengan target ( $t$ ), hitung error  $E$  dengan:

$$E = t - y_k$$

g. Tahap Backpropagation:

Pada tiap unit output, gunakan formula ini untuk memperbaiki nilai bobot dan bias ( $\delta$ ):

$$\delta_k = E f'(y_{net_k})$$

h. Perbaiki nilai bobot ( $\Delta w$ ) dan bias ( $\Delta b$ ) dengan menggunakan formula:

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

Dimana  $\alpha$  = learning rate

- i. Hitung sinyal balik ( $\delta_{in}$ ) dari output layer pada tiap unit di hidden layer

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj}$$

dimana p = jumlah node, w = bobot

- j. Pada tiap unit hidden layer, hitung delta\_1 untuk memperbaiki nilai bobot dan bias

$$\delta_j = \delta_{net_j} f'(z_{net_j})$$

- k. Perbaiki nilai bobot ( $\Delta v$ ) dan bias ( $\Delta b$ ) dengan menggunakan formula:

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

- l. Untuk semua layer, perbaiki bobot dan bias:

Pada output layer:

$$w_{jk} = w_{jk} + \Delta w_{jk}$$

$$b_{2k} = b_{2k} + \Delta b_{2k}$$

Pada hidden layer:

$$v_{ij} = v_{ij} + \Delta v_{ij}$$

$$b_{1k} = b_{1k} + \Delta b_{1k}$$

- m. Hitung MSE (Mean Square Error) dengan menggunakan formula:

$$MSE = \frac{\sum E^2}{N}$$

- n. Jika (Epoch < maximum Epoch) atau (MSE < Target Error), ulangi langkah pelatihan (mulai feedforward).

Terdapat beberapa cara dalam modifikasi bobot-bobot jaringan saraf tiruan. Pertama, modifikasi dilakukan pada setiap akhir perhitungan setiap contoh kasus yang biasa disebut *case (online) updating*. Atau cara kedua, modifikasi bobot-bobot jaringan saraf tiruan dilakukan setelah semua contoh kasus dianalisis. Cara ini disebut *epoch (batch) updating*.

Untuk perhitungan error dalam output layer dilakukan dengan persamaan berikut :

$$Err_1 = O_1(1 - O_1)(T_1 - O_1)$$

Dalam hal ini :

$O_1$  = Keluaran dari output *node* unit  $i$

$T_1$  = Nilai sesungguhnya dari output *node* dalam *data training*

Perhitungan error pada hidden layer dilakukan dengan persamaan berikut :

$$Err_1 = O_1(1 - O_1) \sum_j Err_j W_{ij}$$

Dalam hal ini :

$O_1$  = Keluaran dari hidden node unit  $i$  yang memiliki keluaran  $j$  dalam layer

$Err_j$  = Nilai error dalam node unit  $j$

$W_{ij}$  = Bobot antara kedua node (neuron)

Setelah nilai error pada setiap node dihitung, lakukan modifikasi terhadap bobot jaringan dengan menggunakan persamaan berikut :

$$W_{ij} = W_{ij} + l \cdot Err_j \cdot O_i$$

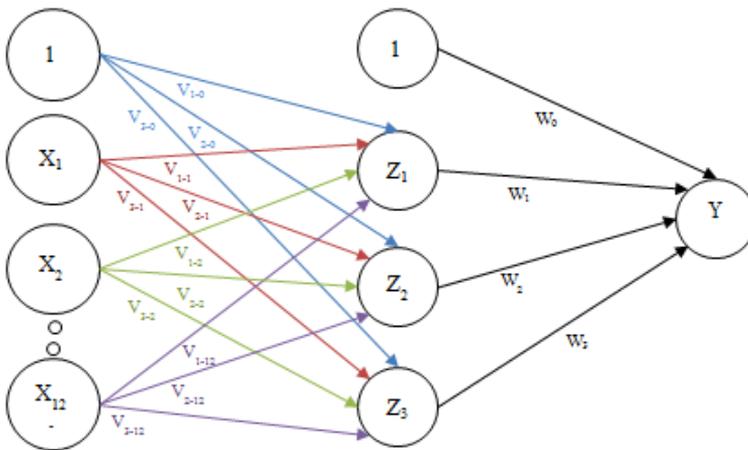
Dalam hal ini :

$l$  = *Learning rate* dengan nilai antara 0-1

Jika nilai  $l$  kecil, maka perubahan bobot akan sedikit dalam setiap interaksi.

### E. Arsitektur Backpropagation

Backpropagation memiliki beberapa unit yang ada dalam satu atau lebih layer tersembunyi. Arsitektur jaringan syaraf *backpropagation* seperti terlihat pada gambar dibawah ini [16]:

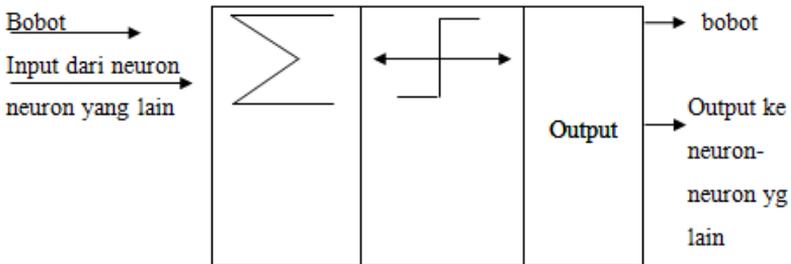


Gambar 6 Arsitektur *backpropagation*

## 1. Fungsi Aktivasi Sigmoid

Setiap neuron mempunyai keadaan internal yang disebut level aktivasi atau level aktivitas yang merupakan fungsi input yang diterima. Secara tipikal suatu neuron mengirimkan aktivitasnya kebeberapa neuron lain sebagai sinyal. Yang perlu diperhatikan adalah bahwa neuron hanya dapat mengirimkan satu sinyal sesaat, walaupun sinyal tersebut dapat dipancarkan ke beberapa neuron yang lain [17].

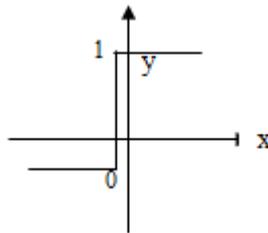
Fungsi aktivasi dalam NN yang dapat digunakan, antara lain:



Gambar 7. Fungsi aktivasi

### a. Fungsi sigmoid biner (Hard Limit)

$$y = f(x) = \frac{1}{1+e^{-\sigma x}}$$
$$Y = \begin{cases} 0, & \text{jika } x < 0 \\ 1, & \text{jika } x \geq 0 \end{cases}$$

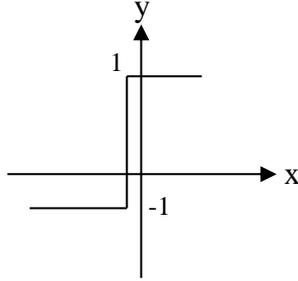


Gambar 8. Fungsi sigmoid biner

b. Fungsi bipolar (Symetik Hard Limit)

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

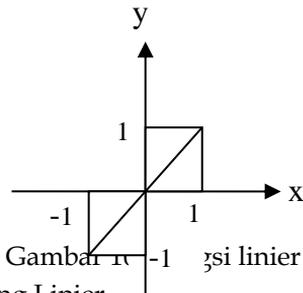
$$Y = \begin{cases} 1, & \text{jika } x \geq 0 \\ -1, & \text{jika } x < 0 \end{cases}$$



Gambar 9. Fungsi bipolar

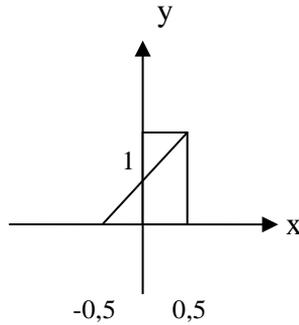
c. Fungsi linier (Identitas)

$$Y = x$$



d. Fungsi Saturating Linier

$$Y = \begin{cases} 1, & \text{jika } x \geq 0,5 \\ x + 0,5, & \text{jika } -0,5 \leq x \leq 0,5 \\ 0, & \text{jika } x \leq -0,5 \end{cases}$$



Gambar 11. Fungsi saturating linier

Dalam Backpropagation, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat yaitu : kontinu, terdeferensial dengan mudah dan merupakan fungsi yang tidak turun. Salah satu fungsi yang memenuhi ketiga syarat tersebut sehingga sering dipakai adalah fungsi sigmoid biner yang memiliki range (0,1). [16]

## 2. Algoritma Pelatihan

### a. Levenberg-Marquardt

Algoritma Levenberg-Marquardt dirancang untuk mendekati orde kedua kecepatan pelatihan tanpa harus menghitung matriks Hessian. Bila fungsi kinerja memiliki bentuk jumlah kuadrat (seperti yang khas dalam pelatihan jaringan feedforward), maka matriks Hessian dapat diperkirakan sebagai:

$$\mathbf{H} = \mathbf{J}^T \mathbf{J}$$

dan gradien dapat dihitung sebagai

$$\mathbf{g} = \mathbf{J}^T \mathbf{e}$$

dimana  $J$  adalah matriks Jacobian yang berisi derivatif pertama dari kesalahan jaringan terhadap bobot dan bias, dan  $e$  adalah vektor error jaringan. Matriks Jacobian dapat dihitung melalui teknik backpropagation standar yang jauh lebih kompleks daripada menghitung matriks Hessian.

## **b. Gradient Conjugate**

Algoritma standard yang digunakan dalam pelatihan BackPropagation Neural Network, yaitu gradient conjugate dan gradient conjugate with momentum seringkali terlalu lambat untuk keperluan praktis. Algoritma ini merupakan penggabungan dari Algoritma Gradient Conjugate with Adaptive Learning (traingda) dan Gradient Conjugate With Momentum (traingdm).

Pada standard backpropagation, perubahan bobot didasarkan atas gradien yang terjadi untuk pola yang dimasukkan saat itu. Modifikasi yang dapat dilakukan adalah melakukan perubahan bobot yang didasarkan atas arah gradien pola terakhir dan pola sebelumnya (disebut momentum) yang dimasukkan, jadi tidak hanya pola masukan terakhir saja yang diperhitungkan [18].

Penambahan momentum dimaksudkan untuk menghindari perubahan bobot yang mencolok akibat adanya data yang sangat berbeda dengan yang lain (outlier). Apabila beberapa data terakhir yang diberikan ke jaringan memiliki pola yang serupa (berarti arah gradien sudah benar), maka perubahan bobot dilakukan secara cepat. Namun, apabila data terakhir yang dimasukkan memiliki pola yang berbeda dengan pola sebelumnya, maka perubahan dilakukan secara lambat [19].

Dengan Penambahan momentum, bobot baru pada waktu ke (t+1) didasarkan atas bobot pada waktu t dan (t-1). Disini harus ditambahkan dua variabel baru yang mencatat besarnya momentum untuk dua iterasi terakhir. Jika  $\mu$  adalah konstanta ( $0 \leq \mu \leq 1$ ) yang menyatakan parameter momentum maka bobot baru dihitung berdasarkan persamaan :

$$w_{kj}(t+1) = w_{kj}(t) + \alpha \delta_k z_j + \mu(w_{kj}(t) - w_{kj}(t-1))$$

dan

$$v_{ji}(t+1) = v_{ji}(t) + \alpha \delta_j x_i + \mu(v_{ji}(t) - v_{ji}(t-1))$$

Algoritma gradient descent dengan adaptive learning rate, dasarnya sama dengan algoritma gradient descent standard dengan beberapa perubahan. Pertama-tama dihitung terlebih dahulu nilai output jaringan dan error pelatihan. Pada setiap epoch, bobot-bobot baru dihitung dengan menggunakan learning rate yang ada. Kemudian dihitung kembali output jaringan dan error pelatihan. Jika perbandingan antara error pelatihan yang baru dengan error pelatihan lama melebihi maksimum kenaikan kinerja (`max_perf_inc`), maka bobot-bobot baru tersebut akan diabaikan, sekaligus nilai learning rate akan dikurangi dengan cara mengalikannya dengan `lr_dec`. Sebaliknya, apabila perbandingan antara error pelatihan baru dengan error pelatihan lama kurang dari maksimum kenaikan kinerja, maka nilai bobot-bobot akan dipertahankan, sekaligus nilai learning rate akan dinaikkan dengan cara mengalikannya dengan `lr_inc`.

Dengan cara ini, apabila learning rate terlalu tinggi dan mengarah ke ketidak stabilan, maka learning rate akan diturunkan. Sebaliknya jika learning rate terlalu kecil untuk menuju konvergensi, maka learning rate akan dinaikkan. Dengan demikian, maka algoritma pembelajaran akan tetap terjaga pada kondisi stabil.

Algoritma gradient descent with momentum and adaptive learning(`traingdx`) merupakan penggabungan antara algoritma gradient descent with adaptive learning(`traingda`) dan algoritma gradient descent with momentum(`traingdm`). Algoritma ini merupakan algoritma default yang digunakan oleh matlab karena memiliki performa kecepatan pelatihan yang tinggi.

### c. Algoritma Scaled Conjugate Gradient

Algoritma backpropagation dasar menyesuaikan bobot dalam arah keturunan curam (negatif dari gradien), arah di mana fungsi kinerja menurun paling cepat. Ternyata, meskipun fungsi menurun paling cepat sepanjang negatif dari gradien, ini tidak selalu menghasilkan konvergensi tercepat. Dalam algoritma konjugasi gradien, pencarian dilakukan sepanjang arah konjugasi, yang umumnya lebih cepat konvergen daripada arah keturunan curam [20].

Algoritma conjugate gradient memulai dengan cara mencari di arah turunan curam (negatif dari gradien) pada iterasi pertama.

$$P_0 = -g_0$$

Sebuah pencarian garis selanjutnya dilakukan untuk menentukan jarak optimal untuk bergerak, disepanjang arah pencarian ini:

$$x_{k+1} = x_k + \alpha_k p_k$$

Kemudian arah pencarian berikutnya ditentukan sehingga berkonjugasi ke arah pencarian sebelumnya. Prosedur umum untuk menentukan arah pencarian baru adalah menggabungkan arah turunan baru dengan arah pencarian sebelumnya:

$$p_k = -g_k + \beta_k p_{k-1}$$

Setiap algoritma gradien konjugasi dibahas sejauh ini memerlukan pencarian line di setiap iterasi. Pencarian ini line adalah proses komputasi yang boros, karena memerlukan bahwa respon jaringan untuk semua input pelatihan, dan dihitung beberapa kali untuk setiap pencarian.

Modul scg (Matlab) dapat saja memerlukan iterasi yang lebih ba

Metode ini merupakan modifikasi dari *gradient descent* yang pengaruhnya terlihat pada saat beberapa data pelatihan sangat berbeda dari sebagian besar data. *Momentum* memberikan jaringan untuk membuat pengaturan bobot yang besar dan sesuai selama faktor koreksi *error* dalam arah yang sama untuk beberapa pola, sementara itu menggunakan *learning rate* yang lebih kecil untuk mencegah permukaan *error* yang besar dari setiap pola pelatihan [20].

Pada penelitian ini, pola data pelatihan menggunakan data yang berbeda cukup signifikan, dimana sebagian besar data yaitu data beban listrik yang mencapai ribuan (seperti 3042) dan karena itu aturan

penyesuaian bobot untuk mengatasi masalah tersebut perlu digunakan.

### 3. Optimalitas Arsitektur Backpropagation

Masalah utama yang dihadapi dalam *Backpropagation* adalah lamanya iterasi yang harus dilakukan. *Backpropagation* tidak dapat memberikan kepastian tentang berapa *epoch* yang harus dilalui untuk mencapai kondisi yang diinginkan. Oleh karena itu orang berusaha meneliti bagaimana parameter-parameter jaringan dibuat sehingga menghasilkan jumlah iterasi yang relatif lebih sedikit.

a. Inisialisasi bobot awal secara *random*

Pemilihan bobot awal sangat mempengaruhi jaringan syaraf dalam mencapai minimum global (atau mungkin hanya lokal saja) terhadap nilai *error*, serta cepat tidaknya proses pelatihan menuju kekonvergenan. Apabila nilai bobot awal terlalu besar, maka *input* ke setiap lapisan tersembunyi atau lapisan *output* akan jatuh pada daerah dimana turunan fungsi sigmoidnya akan sangat kecil. Sebaiknya, apabila nilai bobot awal terlalu kecil, maka *input* ke setiap lapisan tersembunyi atau lapisan *output* akan sangat kecil, yang akan menyebabkan proses pelatihan akan berjalan sangat lambat. Biasanya bobot awal diinisialisasi secara *random* dengan nilai antara -0.5 sampai 0.5 (atau -1 sampai 1, atau interval yang lainnya).

b. Jumlah unit tersembunyi

Hasil teoritis yang didapat menunjukkan bahwa jaringan dengan sebuah layer tersembunyi sudah cukup bagi *Backpropagation* untuk mengenali sembarang pola antara masukan dan target dengan tingkat ketelitian yang ditentukan. Akan tetapi penambahan jumlah layer tersembunyi kadangkala membuat pelatihan lebih mudah.

Dalam propagasi maju, keluaran harus dihitung untuk tiap layar, dimulai dari layar tersembunyi paling bawah (terdekat dengan masukan). Sebaliknya, dalam propagasi mundur, faktor  $\delta$  perlu dihitung untuk tiap layar tersembunyi, dimulai dari layar keluaran.

c. Jumlah pola pelatihan

Tidak ada kepastian tentang berapa banyak pola yang diperlukan agar jaringan dapat dilatih dengan sempurna. Jumlah pola yang dibutuhkan dipengaruhi oleh banyaknya bobot dalam jaringan serta tingkat akurasi yang diharapkan. Aturan kasarnya dapat ditentukan berdasarkan rumusan :

$$\text{Jumlah pola} = \text{Jumlah bobot} / \text{tingkat akurasi}$$

Untuk jaringan dengan 80 bobot dan tingkat akurasi 0.1, maka 800 pola masukan diharapkan akan mampu mengenali dengan benar 90 % pola diantaranya.

d. Lama iterasi

Tujuan utama penggunaan *Backpropagation* adalah mendapatkan keseimbangan antara pengenalan pola pelatihan secara benar dan respon yang baik untuk pola lain yang sejenis (disebut data pengujian). Jaringan dapat dilatih terus menerus hingga semua pola pelatihan dikenali dengan benar. Akan tetapi hal itu tidak menjamin jaringan akan mampu mengenali pola pengujian dengan tepat. Jadi tidaklah bermanfaat untuk meneruskan iterasi hingga semua kesalahan pola pelatihan = 0.

Umumnya data dibagi menjadi dua bagian, yaitu pola data yang dipakai sebagai pelatihan dan data yang dipakai untuk pengujian. Perubahan bobot dilakukan berdasarkan pola pelatihan. Akan tetapi selama pelatihan (misal setiap 10 *epoch*), kesalahan yang terjadi dihitung berdasarkan semua data (pelatihan dan pengujian). Selama kesalahan ini menurun, pelatihan terus dijalankan. Akan tetapi jika kesalahannya sudah meningkat, pelatihan tidak ada gunanya untuk

diteruskan lagi. Jaringan sudah mulai mengambil sifat yang hanya dimiliki secara spesifik oleh data pelatihan (tapi tidak dimiliki oleh data pengujian) dan sudah mulai kehilangan kemampuan melakukan generalisasi [21].

#### **4. Backpropagation Dalam Peramalan**

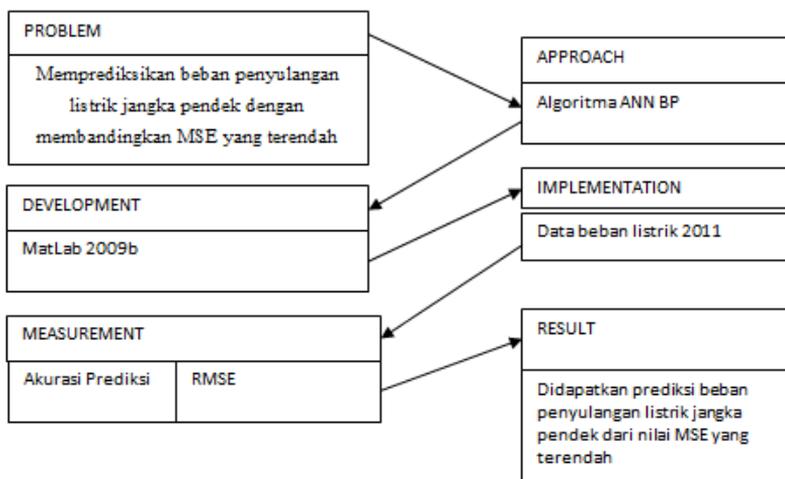
Backpropagation dapat diaplikasikan dengan baik adalah bidang peramalan (forecasting). Secara umum, masalah peramalan dapat dinyatakan dengan sejumlah data runtun waktu (time series) yang bertujuan untuk memperkirakan berapa harga  $x_{n+1}$  berdasarkan data  $x_1, x_2, \dots, x_n$ . Langkah-langkah membangun struktur jaringan untuk peramalan sebagai berikut [16].

- a. Transformasi Data. Sebab-sebab utama data ditransformasi adalah agar kestabilan taburan data dicapai. Selain itu berguna untuk menyesuaikan nilai data dengan range fungsi aktivasi yang digunakan dalam jaringan. Ada beberapa transformasi yang digunakan, yaitu transformasi polinomial, transformasi normal dan transformasi linear.
- b. Pembagian Data. Data dibagi menjadi data pelatihan, pengujian dan validasi.
- c. Perancangan Struktur Jaringan Yang Optimum. Struktur jaringan terdiri dari simpul input, bilangan hidden layer, bilangan simpul hidden layer, dan bilangan simpul output.

#### **F. Kerangka Pemikiran**

Penelitian ini dilakukan untuk mengamati dan memprediksi beban listrik agar dalam penerapannya dapat menekan tingginya beban listrik yang dipergunakan.

Berikut adalah kerangka pemikiran dalam bentuk gambar.



**Gambar 12 Kerangka Pemikiran**

## BAB III METODE PENELITIAN

### A. Metode Penelitian

Metode penelitian yang dilakukan adalah metode penelitian eksperimen, dengan tahapan penelitian seperti berikut:

1. Pengumpulan Data (*Data Gathering*)  
Penelitian ini memakai data rentet waktu
2. Pengolahan Awal Data (*Data Pre-processing*)  
Data perolehan di-transformasi untuk mendapatkan atribut yang relevan dan sesuai dengan format input algoritma soft computing.
3. Model/Metode Yang Diusulkan (*Proposed Model/Method*)  
Metode yang diusulkan adalah metode evaluasi tingkat akurasi dari model algoritma yang dapat digunakan untuk memprediksi data rentet waktu.
4. Eksperimen dan Pengujian Metode (*Method Test and Experiment*)  
Sebagian data digunakan sebagai data training dan sebagian lagi sebagai data checking. Perhitungan akan diulang beberapa kali untuk mendapatkan besaran parameter terbaik.
5. Evaluasi dan Validasi Hasil (*Result Evaluation and Validation*)  
Evaluasi dilakukan dengan mengamati hasil prediksi menggunakan algoritma softcomputing. Validasi dilakukan dengan mengukur hasil prediksi dibandingkan dengan data asal, sehingga dapat diketahui algoritma yang lebih akurat.

## B. Metode Pengumpulan data

Penelitian ini memakai data beban listrik yang didapatkan dari PLN Kalselteng. Data yang dibutuhkan dalam penelitian ini adalah:

### 1. Data Sekunder

Penelitian ini memakai data beban, 1 April 2010 - 30 Maret 2011 untuk beban pkl. 6.00, 10.00, 14.00, 18.00, 19.00, 20.00, 21.00, 22.00, dan 23.00 WITA

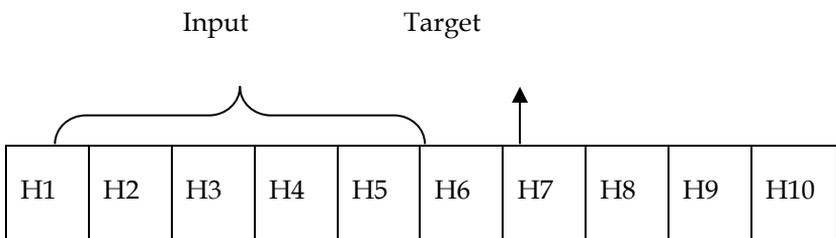
### 2. Data Primer

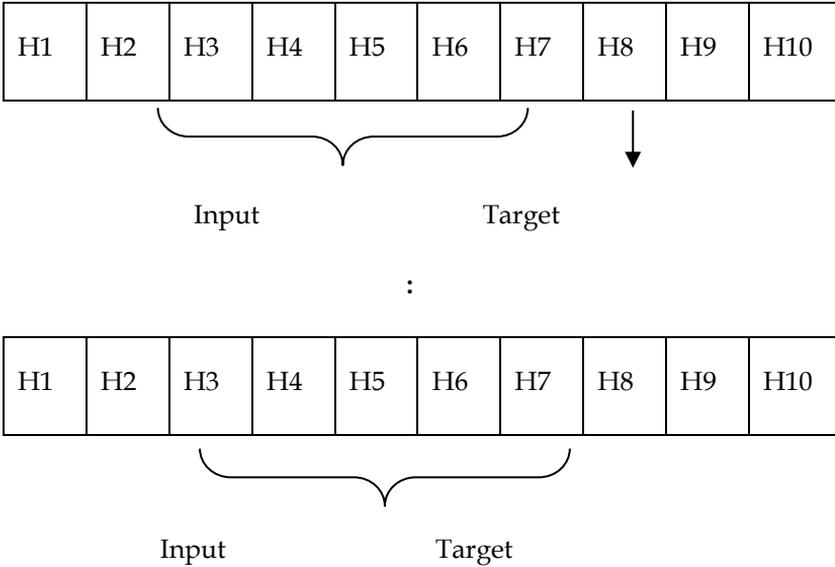
Data primer yang digunakan adalah data hasil komputasi algoritma prediksi.

## C. Metode Pengolahan Data Awal

Data yang didapatkan dari instansi terkait masih berupa data yang terdiri dari berbagai parameter, sehingga harus direkapitulasi terlebih dahulu. Rekapitulasi tersebut dilakukan dengan memperhatikan kebutuhan. Data hasil proses ini adalah data dengan atribut: Beban MW. Tiap baris data adalah data dari 9 jam pilihan. Data dikumpulkan selama satu tahun, sehingga tersedia 3042 baris data.

Ilustrasi prediksi jaringan seperti gambar di bawah ini:





**Gambar 13 Step a head**

Jaringan akan dilatih untuk peramalan satu langkah ke depan (ukuran langkah maju satu dari data input). Ketika prediksi dibuat, dapat ditentukan target perkiraan, yang merupakan jumlah titik data yang diperkirakan di depan. Dalam hal ini, perkiraan horizontal satu di depan disebut juga one step ahead. Maka tabel data input akan *tampak seperti di bawah ini*:

**Tabel 2 Tabel data input pada matlab**

5,15	6,65	9,049	...	9,86	9,59	8,09
5,21	5,15	6,65	...	9,73	9,86	9,59
5,64	5,21	5,15	...	6,56	9,73	9,86

Dan target dari prediksi adalah satu langkah dari 3 data input, seperti tampak di bawah ini:

**Tabel 3 Tabel data input pada matlab**

5,15	6,65	9,049	...	9,86	9,59	8,09
------	------	-------	-----	------	------	------

Lebih jelasnya dapat dilihat pada tabel penelitian di bawah ini:

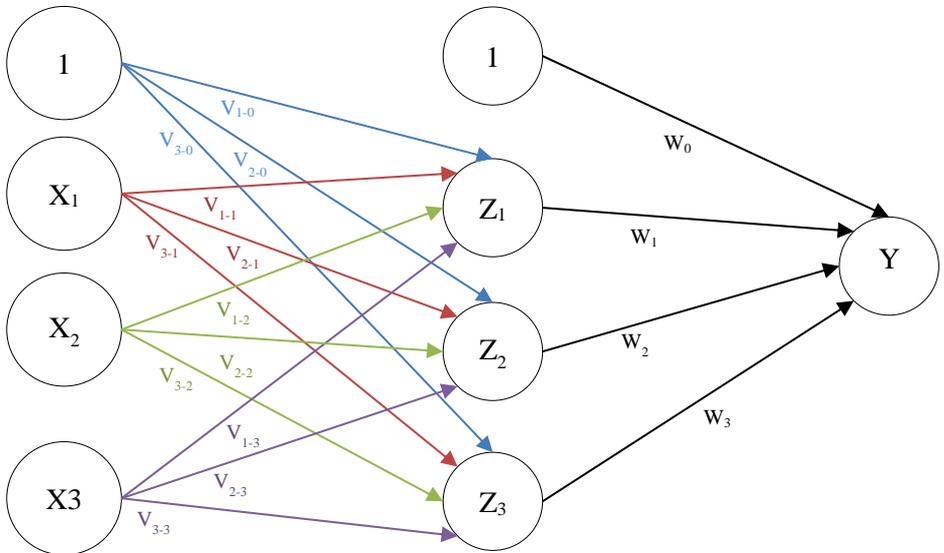
**Tabel 4 Tabel data pelatihan 3 input 1 target**

Pola	$X_1$	$X_2$	$X_3$	Target
1	5,15	6,65	9,049	6,65
2	5,21	5,15	6,65	9,049
3	5,64	5,21	5,15	8,512
...	...	...	...	...
1000	9,59	8,09	7,15	9,59

#### **D. Eksperimen dan Pengujian Model/Metode**

Untuk memprediksi beban listrik berdasarkan rentet waktu digunakan metode Backpropogatian Neural Network. Data akan melewati tahap pelatihan selanjutnya adalah tahap

pengujian. Pengujian jaringan memprediksi beban listrik untuk hari berikutnya, dimana terdapat nilai error yang merupakan selisih antara nilai beban listrik hari ini dengan nilai beban listrik hari berikutnya, lalu diadakan evaluasi. Algoritma dari metode yang digunakan akan di implementasikan dengan Matlab. Pada tabel pengujian di atas, akan dilatih dengan metode Backpropagation dengan 3 tahap yaitu tahap feedforward, backforward dan modifikasi bobot. Sebelum pelatihan feedforward, perlu menentukan besarnya  $\alpha = 0,3$ , error = 0,00005, epoch=100, fungsi aktivasi = sigmoid biner dan menentukan data pelatihan.



**Gambar 14. Arsitektur jaringan 3 input 3 hidden layer dan 1 output**

Dari gambar diatas, data masukan dinyatakan dengan matrik:

$$X = [ x_1, x_2, x_3 ]$$

Bobot-bobot link neuron masukan adalah  $a_{ij}$ , sehingga dalam bentuk matrik menjadi:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

Keluaran dari tiap-tiap neuron pada lapisan masukan adalah:

$$Z = [z_1 \ z_2 \ z_3]$$

Bobot-bobot link neuron pada lapisan keluaran yaitu:

$$W = [w_0 \ w_1 \ w_2 \ w_3]$$

Pelatihan Backpropagation merupakan suatu penyesuaian bobot dengan memperhatikan kesalahan keluaran yang nyata dengan hasil prediksi. Pada prinsipnya bobot awal diambil secara acak kemudian bobot tersebut berubah kearah bawah yang disebut '*gradient descent*'. Hal tersebut dilakukan secara berulang hingga harga kesalahan akan sama dengan nol [22].

Fungsi aktivasi menggunakan sigmoid biner (range 0 sampai dengan 1), maka data harus ditransformasikan [0,1]. Tapi akan lebih baik di transformasikan pada range [0,1 0,9] karena fungsi sigmoid nilainya tidak pernah 0 ataupun 1 [23].

Untuk mentransformasikan dengan rumus:

$$X' = \frac{(rangemax - rangemin)(x - a)}{b - a} + rangemin$$

Dimana:

a = nilai minimum dari data

b = nilai maximum dari data

$$X_1' = \frac{(0,9-0,1)(5,15-0)}{11,3-0} + 0,1$$

$$X_1' = 0,464602$$

**Tabel 5 Hasil transformasi**

Pola	X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	Target
1	0,464602	0,570796	0,740637	0,570796
2	0,46885	0,464602	0,570796	0,740637
3	0,499292	0,46885	0,464602	0,702619
...	...	...	...	...
1000	0,778938	0,672743	0,606195	0,778938

Pelatihan algoritmanya adalah sebagai berikut:

Inisialisasi bobot awal secara acak.

**Tabel 6 Nilai bobot-bobot dari layer input ke layer tersembunyi**

	$V_1$	$V_2$	$V_3$
1	0.1	0.3	-0.1
$X_1$	0.5	0.1	-0.5
$X_2$	-0.2	0.1	0.2
$X_3$	0.1	-0.2	-0.3

**Tabel 7 Nilai bobot-bobot dari layer tersembunyi ke layer output**

	$W$
1	-0.1
$Z_1$	0.2
$Z_2$	0.1
$Z_3$	-0.4

Lakukan langkah 3 sampai dengan 8 untuk melatih data

**Tabel 8 Data pelatihan**

$X_1$	$X_2$	$X_3$	Target
0,464602	0,570796	0,740637	0,570796

Tahap I: Propagasi Maju

Langkah 3

Unit yang masuk akan menerima sinyal dan akan menuju ke unit yang tersembunyi

Langkah 4

Keluarnya yang berada pada unit tersembunyi ( $Z_j$ ) akan dihitung sebagai berikut :

$$z_{netj} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

$$z_{net1} = v_{10} + \sum_{i=1}^2 x_i v_{ji}$$

$$z_{net1} = v_{10} + (x_1 \cdot v_{11}) + (x_2 \cdot v_{12}) + (x_3 \cdot v_{13})$$

$$= 0.1 + (0,464602 \cdot 0.5) + (0,570796 \cdot (-0.2)) + (0,740637 \cdot 0.1)$$

$$= 0,2922053$$

Dengan cara yang sama hasil  $Z_{net2}$  dan  $Z_{net3}$  adalah sebagai berikut:

$$Z_{net2} = 0,25541$$

$$Z_{net3} = -0,44033$$

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}}$$

$$z_1 = f(z_{net_1}) = \frac{1}{1 + e^{-z_{net_1}}} = \frac{1}{1 + e^{-0,2922053}} = 0,5725$$

$$z_2 = f(z_{net_2}) = \frac{1}{1 + e^{-z_{net_2}}} = \frac{1}{1 + e^{-0,25541}} = 0,5635$$

$$z_3 = f(z_{net_3}) = \frac{1}{1 + e^{-z_{net_3}}} = \frac{1}{1 + e^{+0,44033}} = 0,39166$$

Langkah 5

Perhitungan semua jaringan di unit keluaran (yk)

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j w_{kj}$$

$$y_{net_1} = w_{10} + \sum_{j=1}^p z_j w_{1j} = w_0 + z_1 w_1 + z_2 w_2 + z_3 w_3$$

$$= -0,1 + 0,5725 \cdot 0,2 + 0,5635 \cdot 0,1 + 0,39166 \cdot (-0,4) = -0,8581$$

$$y_1 = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}} = \frac{1}{1 + e^{0,8581}} = 0,47856$$

Fase II : Propagasi Maju

Langkah 6

$$\delta_k = (t_k - y_k) f'(y_{\text{net}_k}) = (t_k - y_k) (y_k (1 - y_k))$$

$$\delta_1 = (t_1 - y_1) f'(y_{\text{net}_1}) = (t_1 - y_1) (y_1 (1 - y_1))$$

$$= (0,1328 - 0,47856) ((0,47856)(1 - 0,47856)) = -0,08626$$

Misal dengan menggunakan  $\alpha = 0,2$

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

$$\Delta w_{10} = \alpha \delta_1 z_0 = 0,2 \cdot (-0,08626) \cdot (1) = -0,01725$$

Dengan cara yang sama maka :

$$\Delta w_{11} \quad - \quad 0,00972$$

$$\Delta w_{12} \quad - \quad 0,00676$$

$$\Delta w_{13} \quad - \quad 0,02861$$

Langkah 7

Menghitung factor  $\delta$  unit yang tersembunyi berdasarkan kesalahan di setiap unit tersembunyi  $z_j$

$$(j=1,2,3,\dots,p)$$

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj}$$

$$\delta_{net_1} = \delta_1 \cdot w_{11} = (-0,08626) \cdot (-0,00988) = 0,00085$$

$$\delta_{net_2} = \delta_1 \cdot w_{12} = (-0,08626) \cdot (-0,00972) = 0,00139$$

$$\delta_{net_3} = \delta_1 \cdot w_{13} = (-0,08626) \cdot (-0,0676) = 0,00091$$

Faktor kesalahan  $\delta$  unit tersembunyi

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1-z_j)$$

$$\delta_1 = \delta_{net_1} z_1 (1-z_1) = (0,00085) \cdot 0,5725 \cdot (1-0,5725) = 0,00021$$

Langkah yang sama untuk perhitungan faktor kesalahan  $\delta$  unit tersembunyi

$$\delta_2 = \delta_{net_2} z_2 (1-z_2) = 0,00034$$

$$\delta_3 = \delta_{net_3} z_3 (1-z_3) = 0,00022$$

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

$$\Delta v_{11} = \alpha \delta_1 x_1 = 0,2 \cdot (0,00021) \cdot 0,4646 = 0,00002$$

Dengan langkah yang sama maka dapat nilai sebagai berikut:

$$\Delta v_{12} = 0,00002$$

$$\Delta v_{13} = 0,00002$$

$$\Delta v_{21} = 0,00003$$

$$\Delta v_{22} = 0,00004$$

$$\Delta v_{23} = 0,00003$$

$$\Delta v_{31} = 0,00002$$

$$\Delta v_{32} = 0,00002$$

$$\Delta v_{33} = 0,00002$$

Fase III : Perubahan Bobot

Langkah 8

Perubahan bobot garis yang menuju unit keluaran

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

$$w_{10} \text{ (baru)} = w_{10} \text{ (lama)} + \Delta w_{10}$$

$$= -0,1 - 0,1725$$

$$= -0,1173$$

$$w_{11} \text{ (baru)} = w_{11} \text{ (lama)} + \Delta w_{11}$$

$$= 0,2 - 0,00988$$

$$= 0,1901$$

$$w_{12} \text{ (baru)} = w_{12} \text{ (lama)} + \Delta w_{12}$$

$$= 0,1 - 0,00972$$

$$= 0,0903$$

$$\begin{aligned}
w_{13}(\text{baru}) &= w_{13}(\text{lama}) + \Delta w_{13} \\
&= -0,4 - 0,00676 \\
&= -0,40676 \\
v_{kj}(\text{baru}) &= v_{kj}(\text{lama}) + \Delta v_{kj} \\
v_{11}(\text{baru}) &= 0,50002 \\
v_{12}(\text{baru}) &= -0,199998 \\
v_{13}(\text{baru}) &= -0,100031 \\
v_{21}(\text{baru}) &= 0,10003 \\
v_{22}(\text{baru}) &= 0,10004 \\
v_{23}(\text{baru}) &= -0,19994 \\
v_{31}(\text{baru}) &= -0,05 \\
v_{32}(\text{baru}) &= 0,20002 \\
v_{33}(\text{baru}) &= -0,29997
\end{aligned}$$

Pengukuran kinerja dilakukan dengan menghitung rata-rata error yang terjadi melalui besaran Mean Square Error (MSE). Semakin kecil nilai MSE menyatakan semakin dekat nilai prediksi dengan nilai sebenarnya.

# BAB IV HASIL DAN PEMBAHASAN

## A. Hasil Eksperimen

### 1. BPNN

Data yang dipergunakan:

**Tabel 4. 1 Sebagian dari 1343 data (4 Input)**

X2	0.04	2.32	2.72	0.02	0.048	0.213	0.343	0.07	3.28	7.04	2.58	2.98	0.062	0.328	0.42	0.38	3.92	0.2	0.7	0.43	0.08	3.04	0.04	0.08	0.02	0.32	2.03	2.04	2.34	0.12	0.008
X8	2.12	0.02	0.06	0.212	0.243	0.31	3.28	2.83	2.04	2.92	0.062	0.328	0.42	0.38	3.92	0.2	0.7	0.43	0.08	3.04	0.04	0.08	0.02	0.32	2.03	2.04	2.34	0.12	0.008	0.312	
X3	2.12	0.02	0.06	0.212	0.243	0.31	3.28	2.83	2.04	2.92	0.062	0.328	0.42	0.38	3.92	0.2	0.7	0.43	0.08	3.04	0.04	0.08	0.02	0.32	2.03	2.04	2.34	0.12	0.008	0.312	
X5	0.02	0.048	0.213	0.343	0.07	3.28	2.83	2.04	2.92	0.062	0.328	0.42	0.38	3.92	0.2	0.7	0.43	0.08	3.04	0.04	0.08	0.02	0.32	2.03	2.04	2.34	0.12	0.008	0.312	0.002	3.02
X1	0.048	0.213	0.343	0.07	3.28	2.83	2.04	2.92	0.062	0.328	0.42	0.38	3.92	0.2	0.7	0.43	0.08	3.04	0.04	0.08	0.02	0.32	2.03	2.04	2.34	0.12	0.008	0.312	0.002	3.02	
BPNN	1	5	3	4	2	6	3	8	0	10	11	13	13	14	12	16	13	18	18	10	10	17	15	18	18	18	18	18	18	18	18

Dengan menggunakan Matlab, data pengujian yang sudah diinputkan diproses dengan menggunakan metode BPNN dengan membandingkan hasil uji dari metode trainlm, traindm, traingda dan transcg .

Adapun cara untuk menguji hasil BPNN dengan beberapa metode dengan menggunakan matlab sebagai berikut :

a. BPNN dengan metode Trainlm:

```
tes = newff(inputptest,targettest,7,{'},'trainlm');
```

```
tes.trainParam.epochs = 1000;
```

```
tes.trainParam.Ir = 0.02;
```

```
BPNN = train(tes,inputptest,targettest);
```

```
y= sim(BPNN,inputptest);
```

b. BPNN dengan metode Traindm:

```
tes = newff(inputtest,targettest,7,{'traindm'});  
tes.trainParam.epochs = 1000;  
tes.trainParam.Ir = 0.02;  
BPNN = train(tes,inputtest,targettest);  
y= sim(BPNN,inputtest);
```

c. BPNN dengan metode Traingda:

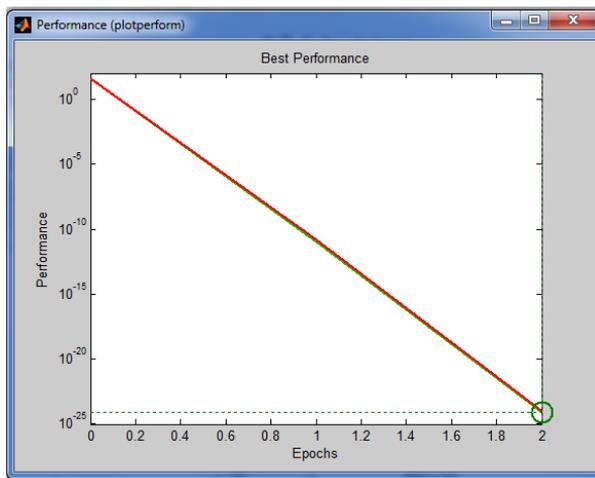
```
tes = newff(inputtest,targettest,7,{'traingda'});  
tes.trainParam.epochs = 1000;  
tes.trainParam.Ir = 0.02;  
BPNN = train(tes,inputtest,targettest);  
y= sim(BPNN,inputtest);
```

d. BPNN dengan metode Trainscg:

```
tes = newff(inputtest,targettest,7,{'trainscg'});  
tes.trainParam.epochs = 1000;  
tes.trainParam.Ir = 0.02;  
BPNN = train(tes,inputtest,targettest);  
y= sim(BPNN,inputtest);
```

Variabel tes akan mengidentifikasi data input dan data target dengan melihat berapa hidden yang akan dipergunakan untuk melakukan training. Metode yang dipergunakan akan menentukan besar kecilnya error yang dihasilkan begitu pula dengan hidden dan epochs yang diberikan.

Dengan menggunakan coding tersebut diatas maka akan menghasilkan sebuah grafik dimana akan dapat menampilkan hasil performance yang baik untuk mendapatkan hasil prediksi yang lebih akurat. Berikut ini adalah hasil dari penginputan data dengan struktur 4 input-7 hidden-1output dengan menggunakan lm menghasilkan MSE  $1,6796e-12$  (gambar 15). Pengujian akan dilakukan sesuai dengan metode yang lain seperti metode *Levenberg-Marquard* (*trainlm*), *Gradient Momentum* (*traindm*), *Gradient Descent* (*traingda*) dan *Scala Conjugugate Gradient* (*trainscg*) dan mengubah nilai hidden.



**Gambar 15 Hasil struktur 4-7-1 dengan menggunakan metode *trainlm***

Arsitektur jaringan, dalam hal ini jumlah *hidden layer* serta jumlah *node* di dalamnya, akan mempengaruhi kinerja jaringan tersebut. Arsitektur yang terlalu sederhana tidak akan mampu menemukan pola data yang kompleks, mendapatkan *error* pelatihan dan *error* generalisasi yang tinggi pada saat *underfitting*. Dan pada arsitektur yang terlalu kompleks dapat menyebabkan jaringan mempunyai banyak keragaman nilai dan terpaku pada nilai-nilai data *training* yang diberikan, bukan pola umum dari data-data tersebut. Masalah ini dikenal dengan istilah *overfitting*.

Untuk dapat menghindari *overfitting*, penentuan jumlah *hidden layer* serta *hidden nodes* di dalamnya harus dilakukan tidak hanya berdasarkan jumlah *epoch* dalam pelatihan, tetapi juga akurasi dalam pengujian. Karena itu penentuan arsitektur jaringan tidak dapat terlepas dari proses pengujian dan dilakukan secara bersamaan.

Dari pengujian tersebut didapat hasil sebagai berikut:

**Tabel 4. 2 Hasil uji struktur BPNN**

Arsitektur FFN			BPNN (MSE)			
Input	Hidden-1	Out	Da	Dm	lm	Scg
4	3	1	2,2632e-10	0,9917	9,3511e-12	2,0576e-09
4	3	1	2,6348e-10	13,3917	5,4555e-12	1,4912e-07
4	3	1	4,0318e-10	10,4493	4,3646e-12	6,2371e-10
4	5	1	3,0372	18,1547	6,2652e-12	2,3535e-09
4	5	1	4.8003	6,0145	5,5082e-12	2,1557e-07
4	5	1	3,9163	10,0532	4,0056e-12	5,7319e-11
4	7	1	5,1165	8,4120	5,2952e-12	1,9607e-06
4	7	1	3,3713	5,1616	2,4161e-12	1,4593e-09
4	7	1	5,486	7,4230	1,6796e-12	1,4050e-06

Menurut hasil uji, semakin banyak hidden layer maka akan akurat data tersebut dan semakin kecil error yang dihasilkan.

### 1) Evaluasi dan Validasi Hasil

Terlihat bahwa nilai MSE yang dihasilkan oleh algoritma BP-Levenberg-Marquardt lebih baik nilainya dari metode algoritma yang lain. Dengan demikian, maka dapat disimpulkan bahwa algoritma BP-Levenberg-Marquardt memprediksi beban listrik lebih akurat.

## 2) Implikasi Penelitian

Data hasil perbandingan tersebut disimpulkan algoritma BP-Levenberg-Marquardt memprediksi lebih akurat dari pada BPNN-dm, BPNN-scg ataupun BPNN-da, dalam memprediksi beban listrik jangka pendek.

Berdasarkan hasil penelitian dan pengukuran, penerapan algoritma BPNN dengan teknik Levenberg-Marquardt ini memiliki nilai lebih dalam proses prediksi beban listrik yaitu menjadikan perkiraan beban listrik memiliki tingkat akurasi yang lebih baik.

Dengan demikian, adanya penerapan algoritma BPNN dengan teknik Levenberg-Marquardt mampu memberikan solusi bagi petugas maupun perencana, serta mampu menjadi metode prediksi beban listrik yang dapat digunakan PLN guna optimasi jaringan distribusi Kalselteng

## **BAB V**

### **KESIMPULAN**

#### **A. Kesimpulan**

Dari hasil penelitian yang dilakukan dari tahap awal hingga pengujian, dan pengukuran, dapat disimpulkan bahwa penerapan algoritma BPNN dengan teknik Levenberg-Marquardt ini memiliki prediksi beban listrik dengan tingkat akurasi yang lebih baik, dengan tingkat error lebih rendah untuk pola beban di wilayah Berabai (BR).

Dengan demikian penerapan algoritma BPNN dengan teknik Levenberg-Marquardt mampu membantu operator perencana dengan metode prediksi yang kinerjanya lebih akurat dalam memprediksi beban listrik. Di samping itu mampu dijadikan metode prediksi beban listrik bagi PLN jaringan distribusi Kalselteng.

#### **B. Saran**

Berdasarkan hasil penelitian dan pengukuran, penerapan algoritma BPNN dengan teknik Levenberg-Marquardt memiliki tingkat akurasi yang lebih baik dalam prediksi beban listrik jangka pendek. Namun berapa hal perlu disampaikan untuk penerapan algoritma BPNN dengan teknik Levenberg-Marquardt yang lebih baik:

1. Data sebagai sumber masukan bagi sistem dapat lebih rinci (perjam) dan dengan jumlah lebih banyak lagi.
2. Mempertimbangkan variabel-variabel lain seperti suhu/cuaca. Karena berdampak kepemakaian peralatan elektronik seperti lampu (di siang hari ketika cuaca agak gelap karena mendung), kipan angin dan AC.

## DAFTAR PUSTAKA

- [1] Jero Wacik, "RENCANA UMUM KETENAGALISTRIKAN NASIONAL 2012-2031," Keputusan Menteri, Oktober 2012.
- [2] Daman Suswanto, "Analisis Peramalan Beban dan Kebutuhan Energi Listrik," November 2010.
- [3] Tan Chong and Christian N.S. Pedersen, Financial Time Series Forecasting Using Improved Wavelet Neural Network, 2009.
- [4] Janett Williams and Yan Li, "A Case Study Using Neural Network Algorithms: Horse Racing Predictions in Jamaica," International Conf. on Artificial Intelligence (ICAI'08), 2008.
- [5] Indra Pranata, "Predeksi WNN untuk Prediksi Beban Penyulangan Listrik Jangka Pendek," Thesis, 2012.
- [6] Yong Wang, Dawu Gu, Jianping Xu, and Jing Li, "Back Propagation Neural Network for Short-term Electricity Load Forecasting with Weather Features," in International Conference on Computational Intelligence and Natural Computing, 2009, pp. 58-61.
- [7] Fang Liu, Raymond D. Findlay, and Qiang Son, "A Neural Network Based Short Term Electric Load Forecasting In Ontario Canada," in International Conference on Computational Intelligence for Modelling Control and Automation, and International Conference on Intelligent

Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), 2006.

- [8] Jui Fang Chang, Chi Ming Kuan, and Yu Wen Lin, "Forecasting Exchange Rates by Genetic Algorithms Based Back Propagation Network Model," IEEE, 2009.
- [9] Sani Susanto and Dedy Suryadi, Pengantar Data Mining : Menggali Pengetahuan Dari Bongkahan Data. Yogyakarta: Andi, 2010.
- [10] I H Witten and Eibe Frank, Data Mining : Practical Machine Learning Tools and Techniques Second Edition. San Francisco: Morgan Kauffman, 2005.
- [11] Daniel T Larose, Discovering knowledge in data an introduction to data mining. Canada: John Wiley & Sons, Inc., Hoboken, New Jersey, 2005.
- [12] Eric A Plummer, "TIME-SERIES FORECASTING WITH FEED-FORWARD NEURAL NETWORKS," A Thesis from Department of Computer Science, 2000.
- [13] A Hermawan, Jaringan Syaraf Tiruan Teori dan Aplikasi. Yogyakarta: Andi, 2006.
- [14] Chris Brenton, Mastering Network Security. Sybex, San Fransisco, 1999.
- [15] S Kusumadewi, Artificial Intellegence : Teknik dan Aplikasinya. Yogyakarta: Graha Ilmu, 2003.

- [16] Jong Jek Siang, Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab. Yogyakarta: Andi, 2009.
- [17] D Puspaningrum, Pengantar Jaringan Syaraf Tiruan. Yogyakarta: Andi Offset, 2006.
- [18] Matlab 7.9.0 (R2009b), , M.T Hagan, Demuth H.B, and Beale M.H, Eds.: Boston, MA: PWS Publishing., 1996.
- [19] Matlab 7.9.0 (R2009P), , R Fletcher and Reeves C.M, Eds.: Computer Journal, Vol. 7, 1964, pp. 149-154.
- [20] The MathWorks, Neural Network Toolbox: User's Guide ver.7, Mark Hudson Beale, Martin T. Hagan, and Howard B Demuth, Eds., 2010.
- [21] Jong Jek Siang, Jaringan Syaraf Tiruan & pemrogramnya Menggunakan MATLAB.: ANDI, 2004.
- [22] Bogdan M Wilamowski, Neural Network Learning.: Auburn University, 2010.
- [23] Xiang Ligu, Zhang Yun, and Yong He, "Study of Application Model on BP Neural Network Optimized by Fuzzy Clustering," 2005.
- [24] kusrini and Emha Taufiq Luthfi, Algoritma Data Mining. Yogyakarta: Andi, 2009.

- [25] Nicolae Morariu, "A NEURAL NETWORK MODEL FOR TIME-SERIES FORECASTING," 2005.
- [26] Howard B. et,al Demuth, Neural Network Toolbox™ 7.: Mathwork, 2010.
- [27] Hudson Beale Mark, T Martin, and B D Howard, The MathWorks, 7th ed., 2010.
- [28] Kusumadewi Sri and Hartati Sri, Neuro-Fuzzy: Integrasi Sistem Fuzzy Dan Jaringan Syaraf, 2nd ed.: Graha Ilmu, 2010.
- [29] Yadi Mulyadi, Ade Gafar Abdullah, and Dadang Lukman Hakim, Aplikasi Logika Fuzzy Dan jaringan Syaraf Tiruan Sebagai Metode Alternatif Prediksi Beban Listrik Jangka Pendek, 2006.
- [30] LIU Chengshui and YI Hongmei, "Research on Power System Load Forecasting Model Based on Data Mining Technology," in 2010 International Conference on Intelligent System Design and Engineering Application, 2010, pp. 2040-243.
- [31] Gao Guorong and Liu Yanping, "Traffic Flow Forecasting based on PCA and Wavelet Neural Network," in 2010 International Conference of Information Science and Management Engineering, 2010, pp. 158-161.

- [32] QianZhang and Tongna Liu, "A Fuzzy Rules and Wavelet Neural Network Method for Mid-long-term Electric Load Forecasting," in Second International Conference on Computer and Network Technology, 2010, pp. 442-446.
- [33] Hai-shuang Guan, Wen-ge Ma, and Qiu-ping Wang, "Real-time Optimal Control of Traffic Flow Based on Fuzzy Wavelet Neural Networks," in Fourth International Conference on Natural Computation, 2008, pp. 509-511.
- [34] Reza Ezuan Samin, Azme Khamis, Ruhaila Md. Kasmani, and Syahirbanun Isa, "Forecasting Sunspot Numbers with Recurrent Neural Networks (RNN) using 'Sunspot Neural Forecaster' System," in Second International Conference on Advances in Computing, Control, and Telecommunication Technologies, 2010, pp. 10-14.
- [35] Bank Indonesia, "Kajian Ekonomi Regional : Kalimantan Selatan," Triwulan II - 2008, Agustus 2008.
- [36] Departemen Energi dan Sumber Daya Mineral, Keputusan Menteri ESDM Nomor: 0954 K / 30 / MEM / 2004 Tentang Rencana Umum Ketenagalistrikan Nasional, 2004.
- [37] Suyanto, *Evolutionary Computing: Komputasi Berbasis 'Evolusi' dan 'Genetika'*. Bandung: Informatika, 2008.
- [38] Li Chungui, Xu Shu'an, and Wen Xin, "Traffic Flow forecasting Algorithm Using Simulated Annealing Genetic BP Network," in 2010 International Conference on

Measuring Technology and Mechatronics Automation, 2010, pp. 1043-1046.

- [39] Ganji Huang and Lingzhi Wang, "Hybrid Neural Network Models for Hydrologic Time Series Forecasting Based on Genetic Algorithm," in 2011 Fourth International Joint Conference on Computational Sciences and Optimization, 2011, pp. 1347-1350.
- [40] Siang Jong Jek, Jaringan Syaraf Tiruan dan Pemrogramannya, 2nd ed. Yogyakarta: Penerbit Andi, 2009.
- [41] Yuhui Wang, Yunzhong Jiang, Xiaohui Lei, and Wang Hao, "Rainfall-Runoff Simulation Using Simulated Annealing Wavelet BP Neural Network," in 2010 International Conference on Intelligent Computation Technology and Automation, 2010, pp. 963-967.
- [42] Qin Ju et al., "Streamflow simulation with an integrated approach of wavelet analysis and artificial neural networks," in Fourth International Conference on Natural Computation, 2008, pp. 564-569.
- [43] The MathWorks, Wavelet Toolbox: User's Guide ver.1, Misiti Michel et al., Eds.: The MathWorks, Inc., 1996.
- [44] Yafei Huang, "Short-term Traffic Flow Forecasting Based on Wavelet Network Model Combined with PSO," in 2008 International Conference on Intelligent Computation Technology and Automation, 2008, pp. 249-253.

- [45] David Veitch, "Transformasi wavelet and their application in the study of dynamical systems," Department of Mathematics - University of York - UK, Dissertation submitted for the MSc in Data Analysis, Networks and Nonlinear Dynamics 2005.
- [46] J.S.R. Jang, C.T. Sun, and E. Mizutani., Neuro-Fuzzy and Soft Computing. New Jersey: Prentice-Hall, 1997.
- [47] Phang Ming-bao and Zhao Xin-ping, "Traffic Flow Prediction of Chaos Time Series by Using Subtractive Clustering for Fuzzy Neural Network Modeling," in Second International Symposium on Intelligent Information Technology Application, Beijing, 2008, pp. 23-27.
- [48] Suyanto, Artificial Intelligenc: Searching, Reasoning, Planning, and Learning. Bandung: Penerbit Informatika, 2007.
- [49] Krose Ben and Smagt Patrick van der, An Introduction To Neural Networks, English ed., 1996.
- [50] Jerico Wacik, Keputusan Menteri Energi dan Sumber Daya Mineral., 2012.