

Peramalan Beban Listrik Kabupaten Cilacap

Evan Dhia Aruna¹, Bagus Fatkhurrozi², Andriyatna Agung Kurniawan³

^{1,2,3}Program Studi Teknik Elektro Jurusan Teknik Elektro Fakultas Teknik Universitas Tidar

E-mail: evan.dhia.aruna23@students.untidar.ac.id¹, bagusf@untidar.ac.id², andriyana@untidar.ac.id³

Article History:

Received: 03 Januari 2024

Revised: 09 Januari 2024

Accepted: 11 Januari 2024

Keywords:

Ann,
Backpropagation, Electrical
Load, Load Forecasting

Abstract: *This study examines electricity load, emphasizing the need for accurate prediction and optimal distribution. Utilizing artificial neural networks and the backpropagation algorithm, the research leverages data from BPS Kabupaten Cilacap and PT. PLN (Persero) UP3 Kabupaten Cilacap. Various configurations for hidden layer neurons, epochs, and learning rates are explored to determine the optimal network architecture for forecasting. The selected model, with specific criteria, demonstrates high accuracy during training (MSE: 0.00099999, MAPE: 5.44%, Regression: 0.98226) and testing (MSE: 0.0009493, MAPE: 3.99%, Regression: 0.90709) phases. The conclusion affirms the effectiveness of the Backpropagation ANN method in predicting electricity load in Kabupaten Cilacap for the period 2023-2030, meeting PLN's tolerance of $\leq 10\%$ based on the MAPE criteria.*

PENDAHULUAN

Beban listrik, vital sebagai kebutuhan daya yang bervariasi, harus optimal disalurkan oleh pembangkit listrik sebagai bagian integral dari infrastruktur distribusi tenaga listrik. Dengan daya listrik yang tidak dapat disimpan dalam jumlah besar, prediksi yang akurat menjadi krusial untuk memenuhi kebutuhan saat diperlukan. Maka, pengembangan sistem tenaga listrik superior dalam kuantitas dan kualitas menjadi kebutuhan mendesak. Salah satu pendekatan untuk itu adalah peramalan beban listrik. Sejalan dengan Peraturan Gubernur Jawa Tengah Nomor 19 Tahun 2019, yang menegaskan pentingnya proyeksi kebutuhan tenaga listrik daerah, terutama dalam meningkatkan kualitas sektor kelistrikan. Perencanaan yang baik diperlukan untuk menghindari potensi kerugian bagi konsumen dan perusahaan penyedia listrik.

Kenaikan beban listrik dipengaruhi oleh beberapa faktor yang sulit diprediksi, maka sukar mendapatkan yang pasti di masa depan. pemicu meningkatnya konsumsi daya listrik suatu daerah disebabkan oleh faktor pertumbuhan penduduk dan faktor ekonomi pada daerah itu sendiri (M. H. M. R. Shyamali Dilhani & Kumara, 2021). Namun demikian, hasil peramalan yang optimal adalah yang mendekati target dan didistribusikan pada waktu yang tepat. Oleh karena itu, peramalan beban listrik sangat dianjurkan untuk memproyeksikan kebutuhan beban listrik di masa mendatang dalam jangka waktu tertentu.

Perkembangan teknologi komputasi saat ini menuju pada teknologi kecerdasan buatan sehingga menghasilkan metode alternatif untuk peramalan beban listrik jangka panjang. Penggunaan teknologi komputasi selain lebih efektif, hasil perhitungannya juga lebih akurat. Para ahli berusaha membuat sistem cerdas (*artificial intelligence*) yang bisa meramalkan beban listrik

di masa mendatang. Salah satu sistem cerdas yang baik digunakan untuk peramalan yaitu jaringan saraf tiruan (JST) (Fadilah, dkk., 2021).

Jaringan saraf tiruan merupakan algoritma penyelesaian masalah komputasi dengan sistem operasi yang mirip dengan jaringan saraf manusia. Salah satu jenis algoritma ANN-nya adalah *backpropagation*. Metode ini dipilih karena memungkinkan proses pembelajaran dilakukan berdasarkan contoh-contoh yang diberikan dan baik dalam menemukan hubungan antara karakteristik beban *nonlinier* dengan data lainnya. Karnoto, dkk., (2018), karakteristik *nonlinier* ini mencakup faktor ekonomi dan faktor lainnya. Apalagi jika keluarannya tidak sesuai dengan tujuan yang diinginkan, maka keluaran tersebut diteruskan kembali ke lapisan tersembunyi dan diteruskan ke unit lapisan masukan untuk mendapatkan umpan balik guna memverifikasi hasil keluaran ANN.

Berdasarkan kajian diatas, dalam penelitian ini telah dilakukan penggunaan metode jaringan saraf tiruan (JST) dengan algoritma *backpropagation* sebagai peramalan beban listrik Kabupaten Cilacap periode 2023-2030 menggunakan beberapa faktor yang mempengaruhinya. Variabel yang digunakan berupa data jumlah penduduk, PDRB, jumlah pelanggan, daya tersambung, dan konsumsi daya listrik. Data tersebut merupakan data historis yang diperoleh dari *website* resmi Badan Pusat Stastistika Kabupaten Cilacap dan PT. PLN (Persero) UP3 dari tahun 2001 sampai dengan 2022. Data ini digunakan untuk meramalkan periode tahun 2023-2030.

LANDASAN TEORI

Peramalan beban listrik merupakan proses estimasi atau prediksi besarnya permintaan atau kebutuhan daya listrik pada suatu waktu tertentu di masa depan. Tujuan utama dari peramalan beban listrik adalah untuk membantu penyedia listrik, pembangkit listrik, dan otoritas terkait dalam merencanakan dan mengelola produksi, distribusi, dan konsumsi energi listrik secara efisien (Afinda & Budiono, 2020).

Peramalan Beban

Peramalan beban sistem tenaga listrik merupakan proses penting dan sebagai dasar untuk perencanaan sistem tenaga. Estimasi permintaan beban untuk sistem tenaga harus setepat mungkin. Berdasarkan jangka waktunya, peramalan beban listrik dapat dikelompokkan menjadi 3 (Djohar & Musarudin, 2017):

1. Peramalan jangka pendek

Peramalan beban jangka pendek adalah proyeksi untuk rentang waktu beberapa jam hingga satu minggu (168 jam). Besar beban untuk setiap jam ditetapkan dengan mempertimbangkan pola beban sebelumnya, sehingga dapat menghindari kelebihan atau kekurangan pasokan energi. Dalam konteks kelistrikan, peramalan jangka pendek bermanfaat untuk memprediksi konsumsi daya listrik dalam interval jam, hari, minggu, atau bulan dari suatu sistem manajemen energi, termasuk menjaga keandalan listrik setiap harinya.

2. Peramalan jangka menengah

Peramalan beban jangka menengah merujuk pada estimasi untuk periode antara satu bulan hingga satu tahun. Tindakan yang dilakukan dalam peramalan jangka menengah melibatkan pengembangan jaringan distribusi dan penyelesaian proyek transmisi saluran. Selain itu, peramalan ini juga berperan dalam menentukan jadwal perawatan peralatan listrik. Permintaan daya listrik dari konsumen selama periode peramalan jangka menengah dipengaruhi oleh faktor musiman.

3. Peramalan jangka panjang

Peramalan jangka panjang merujuk pada estimasi untuk periode lima hingga sepuluh tahun.

Tujuan dari peramalan jangka panjang adalah untuk menilai potensi perluasan jaringan listrik dan proyek pembangunan pembangkit listrik yang memiliki jangka waktu yang lebih panjang.

Peramalan beban selalu dipengaruhi oleh beberapa faktor ketidakpastian yang meningkat semakin jauh. Peramalan beban dimaksudkan untuk diproyeksikan ke masa yang akan datang. Metode peramalan beban dikelompokkan menjadi 5 (Moshinsky, 2019):

1. Peramalan beban menggunakan faktor peningkatan beban;
 2. Peramalan beban berdasarkan data karakteristik ekonomi;
 3. Peramalan beban dengan nilai perkiraan;
 4. Peramalan beban berdasarkan nilai beban spesifik dan derajat kelistrikan;
- peramalan beban dengan kurva beban standar.

Jaringan Syaraf Tiruan

Jaringan saraf tiruan (JST) adalah suatu sistem pengolahan informasi yang memiliki ciri-ciri kinerja khusus yang menyerupai jaringan saraf otak manusia, yang melibatkan fase pembelajaran melalui perubahan bobot sinaptik. Jaringan saraf tiruan memiliki kemampuan untuk mengenali pola data berdasarkan informasi historis. Dengan memeriksa data historis, jaringan saraf tiruan memungkinkan pengambilan keputusan berdasarkan informasi yang sebelumnya belum pernah diperiksa (Kumara, dkk., 2021).

Jaringan saraf tiruan (JST) adalah jenis arsitektur yang menghubungkan secara paralel dengan beberapa *node* (*neuron hidden layer*) dan koneksi antar *node*. Setiap titik koneksi dari satu *node* ke *node* lainnya mempunyai nilai yang diasumsikan dengan bobot. Setiap *node* mempunyai nilai, dan nilai tersebut digunakan sebagai nilai aktivasi *node*. Ciri-ciri jaringan saraf tiruan ditentukan oleh (Siang, 2005):

1. Pola koneksi *neuron* satu dengan yang lainnya disebut arsitektur jaringan;
2. Metode penentuan bobot yang saling terhubung disebut pelatihan atau proses pembelajaran jaringan;
3. Fungsi aktivasi.

Parameter Penyusun Jaringan Syaraf Tiruan

Parameter penyusun (*training parameters*) pada algoritma jaringan syaraf tiruan berupa fungsi aktivasi, *epoch*, dan laju pembelajaran (*learning rate*) (Septiarani, 2022):

1. Fungsi aktivasi;

Fungsi aktivasi adalah fungsi yang bertujuan sebagai penentu nilai *output* berdasarkan jumlah keseluruhan nilai *input* pada *neuron*. Fungsi aktivasi suatu algoritma jaringan mungkin berbeda dengan fungsi aktivasi untuk algoritma jaringan lainnya. Beberapa fungsi aktivasi yang digunakan dalam pembelajaran jaringan *backpropagation* adalah:

a. *sigmoid bipolar* (*tansig*). $f(net) = \frac{2}{1+e^{-net}} - 1$. Fungsi ini digunakan secara *default*. Rentang nilai yang dimiliki fungsi *sigmoid bipolar* yaitu [-1,1].

b. *sigmoid biner* (*logsig*) $f(net) = \frac{1}{1+e^{-net}}$. Fungsi *sigmoid biner* mempunyai bentuk yang hampir sama dengan *sigmoid bipolar* hanya saja rentangnya adalah [0,1]

c. fungsi identitas (*purelin*) $f(net) = net$.

2. *Epoch*;

Epoch merupakan banyaknya iterasi yang dilakukan oleh jaringan hingga tercapai kondisi yang diinginkan. Ketika kriteria terpenuhi proses training akan berhenti. Apabila nilai *epoch* sebesar 20 artinya jaringan telah melakukan 20 kali *loop* proses *training*. Jika jumlah maksimum

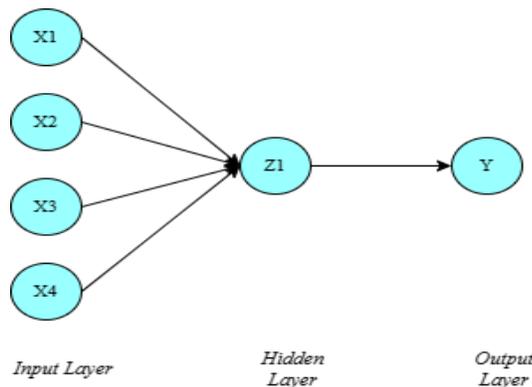
epoch terlalu kecil, maka model tidak terlatih dengan baik, dan hasil peramalan tidak akurat. Sedangkan, jika jumlah maksimum *epoch* terlalu besar, maka model akan terlatih terlalu banyak pada data pelatihan dan menyebabkan *overfitting*. Selain itu, semakin banyak *epoch*, maka semakin lama waktu yang dibutuhkan jaringan untuk mencapai fungsi tujuan.

3. Laju pembelajaran.

Laju pembelajaran (*learning rate*) disimbolkan *alpha*, digunakan untuk mengendalikan tahap penyesuaian bobot. Laju pembelajaran mempengaruhi proses pelatihan (*training*). Nilai *alpha* terlalu kecil mengakibatkan konvergensi jaringan lambat, dan jika nilai *alpha* terlalu besar dapat mengakibatkan ketidakseimbangan jaringan yang dapat menyebabkan *overfitting*. *Overfitting* yaitu kondisi dimana model atau algoritma pembelajaran terlalu “memahami” atau “memorize” data pelatihan yang digunakan untuk melatihnya, hingga akhirnya kinerjanya menurun ketika diterapkan pada data yang belum pernah dilihat sebelumnya atau data uji. Laju pembelajaran juga mempengaruhi nilai *error*.

Backpropagation

Algoritma *backpropagation* merupakan jenis jaringan saraf tiruan yang umumnya digunakan untuk menyelesaikan masalah prediksi. Keberhasilannya terletak pada fungsinya sebagai metode pelatihan JST yang terawasi (Yuliawanti, 2022). Proses pelatihan ini melibatkan sepasang pola, terdiri dari pola *input* dan pola target. Saat suatu pola melewati jaringan, bobot diubah agar perbedaan antara pola *output* dan pola target diminimalkan. Iterasi pelatihan dilakukan secara berulang hingga semua pola yang dihasilkan oleh jaringan sesuai dengan pola target. Struktur jaringan ini terdiri dari lapisan masukan (*input layer*), lapisan tersembunyi (*hidden layer*), dan lapisan keluaran (*output layer*). Arsitektur *backpropagation* disajikan dalam bentuk gambar pada Gambar 1



Gambar 1. Arsitektur Jaringan *Backpropagation* dengan 1 *Hidden Layer*

Aturan pelatihan *backpropagation* dikembangkan dari aturan delta dengan menambahkan lapisan tersembunyi. Pelatihan jaringan dengan menggunakan metode *backpropagation* menggunakan tiga fase (Siang, 2005) sebagai berikut:

Langkah 0 :

1. Menginisialisasi bobot awal dengan nilai *random* kecil mulai dari 0 hingga 1
2. Menentukan *max. epoch*, *target error*, *learning rate*, dan momentum
3. Selagi *epoch* masih < *max. epoch* dan *error* > *target error*, maka lakukan langkah-langkah berikut:

Langkah 1 :

Selama kondisi penghentian tidak terpenuhi (keadaan penghentian $error < target\ error$), kerjakan langkah 2-9.

Langkah 2 :

Setiap pasang data pelatihan kerjakan langkah 3-8.

Fase 1 : *feedforward*

Langkah 3 :

Setiap *neuron* masukan (X_i) mendapatkan sinyal dan melanjutkannya ke lapisan tersembunyi. Lapisan tersembunyi yang digunakan untuk penelitian yaitu 1, menggunakan beberapa jumlah *neuron* lapisan tersembunyi (*node*) yang berbeda pada tiap arsitektur jaringan.

Langkah 4 :

Setiap *hidden unit* z_j ($j = 1, 2, \dots, p$) akan menambahkan sinyal *input* berbobot dengan bias menggunakan persamaan (1). Selanjutnya melakukan perhitungan pada *output* lapisan tersembunyi (Z_j) dengan persamaan (2) dan (3) :

$$z_net_j = V_{j0} + \sum W_{ij}X_i \quad (1)$$

$$z_j = f(z_net_j) \quad (2)$$

$$z_j = f(z_net_j) = \frac{1}{1+e^{-z_in_j}} \quad (3)$$

Keterangan :

- z_net_j = sinyal masukan lapisan tersembunyi ke- j
 W_{ij} = bobot masukan ke-i dan lapisan tersembunyi ke-j
 X_i = nilai masukan ke- i
 V_{j0} = bias pada lapisan tersembunyi ke- j
 Z_i = aktivasi pada lapisan tersembunyi ke- j

Langkah 5 :

Melakukan perhitungan keluaran pada lapisan keluaran (y_k) ($k = 1, 2, \dots, m$), dan tanpa menggunakan fungsi aktivasi karena pada penelitian ini menggunakan *purelin*. Untuk menghitung y_k menggunakan persamaan (4), (5), dan (6)

$$Y_net_k = V_{k0} + \sum Z_jW_{jk} \quad (4)$$

$$y_k = f(Y_net_k) \quad (5)$$

$$f(Y_net_k) = Y_net_k \quad (6)$$

Keterangan :

- Y_net_k = sinyal *input output layer* ke- k
 W_{jk} = bobot antara *hidden layer* ke- j dan *output layer* ke- k
 Z_j = aktivasi *hidden layer* ke- j
 V_{k0} = bias pada *hidden layer* ke- k
 Y_k = aktivasi pada *output layer* ke- k

Fase 2 : *backward*

Langkah 6 :

Selanjutnya pada proses *backward* melakukan evaluasi dengan cara koreksi *error* pada setiap unit lapisan keluaran (δ_k) dan koreksi *error* pada setiap unit *hidden layer* (δ_j). Untuk menghitung (δ_k) dan (δ_j) menggunakan persamaan (7) dan (8)

$$\delta_k = (t_k - y_k)f'(Y_net_k) = (t_k - y_k) \cdot 1 \quad (7)$$

$$\delta_j = \delta_k W_{jk} f'(z_net_j) = \delta_k W_{jk} Z_j(1 - Z_j) \quad (8)$$

Keterangan :

- δ_k = faktor koreksi *error* bobot pada *output layer*
 t_k = target *output*

- α = learning rate
 $f'(z_{net_j})$ = fungsi derivatif dari $f(z_{net_j})$
 $f'(Y_{net_k})$ = fungsi derivatif dari $f(Y_{net_k})$

Langkah 7 :

Kemudian melakukan evaluasi bobot dan bias pada lapisan tersembunyi dan lapisan keluaran menggunakan persamaan (9), (10), (11), dan (12)

$$\Delta W_{ij} = (-\alpha \cdot \delta_j \cdot x_i + \beta \cdot W_{ij}) \quad (9)$$

$$\Delta V_{j0} = (-\alpha \cdot \delta_j + \beta \cdot V_{j0}) \quad (10)$$

$$\Delta W_{jk} = (-\alpha \cdot \delta_k \cdot f(z_{net_j}) + \beta \cdot W_{jk}) \quad (11)$$

$$\Delta V_{jk} = (-\alpha \cdot \delta_k + \beta \cdot V_{jk}) \quad (12)$$

Keterangan :

ΔW_{ij} = koreksi *error* pada bobot W_{ij}

ΔV_{j0} = koreksi *error* pada bias V_{j0}

ΔW_{jk} = koreksi *error* pada bobot W_{jk}

ΔV_{jk} = koreksi *error* pada bias V_{jk}

β = momentum

Fase 3 : Perubahan Bobot

Langkah 8 :

Kemudian proses mengubah bobot lama menjadi bobot baru, yang pertama dilakukan mengubah bobot dari lapisan tersembunyi menuju unit lapisan keluaran menggunakan persamaan (13)

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta W_{jk} \quad (13)$$

Selanjutnya mengubah bobot masukan menuju lapisan tersembunyi menggunakan persamaan (14)

$$W_{ij}(\text{baru}) = W_{ij}(\text{lama}) + \Delta W_{ij} \quad (14)$$

Keterangan :

$W_{jk}(\text{baru})$ = nilai bobot W_{jk} yang baru

$W_{jk}(\text{lama})$ = nilai bobot W_{jk} pada perhitungan sebelumnya

$W_{ij}(\text{baru})$ = nilai bobot W_{ij} yang baru

$W_{ij}(\text{lama})$ = nilai bobot W_{ij} pada perhitungan sebelumnya

Langkah 9 :

Menguji apakah kondisi penghentian terpenuhi. Kondisi penghentian ini tercapai apabila nilai $error < target\ error$.

Persentase Kesalahan

Hasil dari peramalan berupa nilai perkiraan beserta persentase kesalahan (*error*) sebagai perbandingan antara prediksi (hasil peramalan) dengan nilai aktualnya. Apabila nilai *error* belum memenuhi/mendekati target, jaringan mengulang modifikasi bobot. Semakin kecil nilai *error* maka peramalan semakin akurat. Namun, perlu diingat bahwa jika mengoptimalkan nilai *error* yang kecil pada tahap pelatihan tidak selalu menjamin kualitas prediksi yang baik pada data baru. Oleh karena itu, selain nilai *error* yang kecil, validasi (menguji jaringan) menggunakan data baru yang tidak dipakai pada saat proses pelatihan juga penting untuk menilai kinerja jaringan secara menyeluruh. Terdapat dua jenis pengukuran kesalahan (*error*) yang biasa digunakan pada prediksi, yaitu MSE (*Mean Square Error*) dan MAPE (*Mean Absolute Percentage Error*). Penggunaan pengukuran ini

mewakili sudut yang berbeda untuk mengevaluasi model peramalan.

1. Mean Square Error (MSE)

Mean Square Error (MSE) merupakan metrik yang umum digunakan dalam statistika dan pembelajaran mesin untuk mengukur seberapa baik model regresi memodelkan data. MSE mengukur rata-rata dari kuadrat selisih antara nilai yang diprediksi oleh model dan nilai sebenarnya dari data. Dihitung dengan mengkuadratkan selisih antara nilai aktual (target) dan nilai prediksi (*output*) dan membaginya dengan jumlah deret dalam prediksi (Amalia, 2020). Untuk menghitung MSE menggunakan persamaan (15) sebagai berikut:

$$MSE = \frac{1}{n} \sum_{t=1}^n (T_i(x) - Y_i(x))^2 \quad (15)$$

Keterangan:

T_i = nilai target

Y_i = *output* JST

N = jumlah data penelitian

2. Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) metrik evaluasi yang digunakan untuk mengukur tingkat akurasi model peramalan atau prediksi dalam konteks peramalan deret waktu atau regresi. MAPE mengukur seberapa besar persentase rata-rata dari kesalahan absolut relatif terhadap nilai sebenarnya. (Amalia, 2020). Semakin kecil nilai MAPE maka semakin akurat hasil peramalannya (Fadlilah dan Novita, 2021). Nilai MAPE dapat dihitung menggunakan persamaan (16) sebagai berikut:

$$MAPE = \frac{1}{n} \times \frac{T_i - Y_i}{Y_i} \times 100\% \quad (16)$$

Keterangan:

T_i = nilai target

Y_i = *output* JST

N = jumlah data penelitian

Menurut (Gustriansyah, 2017), nilai MAPE dari hasil persamaan (16) dikelompokkan menjadi beberapa kriteria seperti yang tertera pada Tabel 1.

Tabel 1. Kriteria Nilai MAPE

| Nilai MAPE | Akurasi Prediksi |
|-------------------------|------------------|
| $MAPE \leq 10\%$ | Sangat baik |
| $10\% < MAPE \leq 20\%$ | Baik |
| $20\% < MAPE \leq 50\%$ | Cukup baik |
| $MAPE > 50\%$ | Tidak akurat |

Normalisasi dan Denormalisasi Data

Normalisasi data atau transformasi data merupakan proses mengubah nilai-nilai data ke dalam rentang yang lebih kecil atau standar sebelum digunakan sebagai masukan untuk melatih model JST. Normalisasi data berguna untuk meningkatkan konvergensi pembelajaran dan meminimalkan masalah numerik selama proses pelatihan (Rohman, 2022). Pada umumnya normalisasi data dilakukan setelah data dikelompokkan menjadi: data latih, data uji, dan data peramalan. Denormalisasi data merupakan proses mengembalikan nilai hasil peramalan ke dalam skala semula (N) setelah proses pelatihan, pengujian, dan peramalan selesai. Dalam hal ini, data akan

dinormalisasikan dalam rentang 0 hingga 1. Persamaan untuk normalisasi data dan denormalisasi data ditunjukkan pada persamaan (17) dan (18)(Afinda dan Budiono, 2020):

Normalisasi :

$$N' = \frac{0.8(N-N_{min})}{N_{max}-N_{min}}+0.1 \quad (17)$$

Keterangan :

N' = data hasil normalisasi

N = data asli

N_{max} = data terbesar

N_{min} = data terkecil

Denormalisasi :

$$N = \frac{(N'-0.1) \cdot N_{max}-N_{min}}{0.8}+N_{min} \quad (18)$$

Keterangan :

N' = data hasil normalisasi

N = data asli

N_{max} = data terbesar

N_{min} = data terkecil

Korelasi antar Variabel

Salah satu teknik statistik yang sering digunakan untuk mencari hubungan antara dua variabel atau lebih adalah teknik korelasi(Darto, dkk., 2005). Pada umumnya, besar kecilnya hubungan dinyatakan dengan bilangan. Bilangan yang menyatakan besar kecilnya hubungan tersebut disebut koefisien korelasi. Koefisien korelasi tersebut berkisar antara 0.00 sampai +1.00 (korelasi positif) dan diantara 0.00 sampai -1.00 (korelasi negatif), tergantung pada arah hubungan positif ataukah negatif. Koefisien yang bernilai positif menunjukkan bahwa arah korelasi tersebut positif apabila kedua variabel berubah dalam arah yang sama. Sedangkan untuk korelasi bernilai negatif yaitu apabila salah satu variabel bertambah maka variabel lain berkurang atau dalam hal ini adalah berbanding terbalik. Dalam bahasa statistika, tetapan korelasi dinyatakan dengan simbol (r). Berikut merupakan konstruksi penafsiran atas koefisien korelasi antar variabel pada Tabel 2(Darto, 2005):

Tabel 2. Penafsiran Koefisien Korelasi antar Variabel

| No. | Nilai Korelasi (r) | Penafsiran |
|-----|------------------------|---|
| 1. | 0,00 - 0,19 | Hubungan yang sangat kecil dan bisa diabaikan |
| 2. | 0,20 - 0,39 | Hubungan yang kecil (tidak erat) |
| 3. | 0,40 - 0,69 | Hubungan yang moderat |
| 4. | 0,70 - 0,89 | Hubungan yang erat |
| 5. | 0,90 - 1,00 | Hubungan yang sangat erat |

Dalam menganalisis hubungan antara dua variabel yang memiliki skala pengukuran interval, digunakan metode korelasi *product moment* yang dikembangkan oleh Karl Pearson. Metode ini memberikan ukuran sejauh mana dua variabel bergerak bersamaan atau berlawanan dalam suatu hubungan linier. Koefisien korelasi Pearson dihitung menggunakan persamaan yang membandingkan jumlah deviasi dari masing-masing nilai variabel terhadap *mean* (rerata) variabel

tersebut. Persamaan korelasi Pearson yang umumnya digunakan dapat dirumuskan sebagai berikut (Dewi, 2015):

$$r_{xy} = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{\{n\sum x^2 - (\sum x)^2\}\{n\sum y^2 - (\sum y)^2\}}} \quad (19)$$

Keterangan :

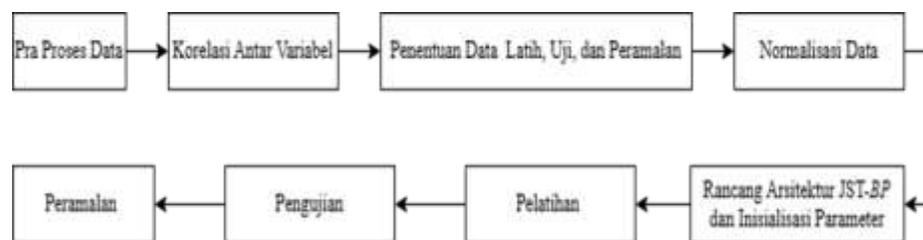
- rx_y = koefisien korelasi
- x = variabel terikat
- y = variabel bebas
- n = banyaknya subjek yang diteliti

Persamaan ini memberikan nilai antara -1 hingga 1, di mana nilai positif menunjukkan hubungan linier positif, nilai negatif menunjukkan hubungan linier negatif, dan nilai 0 menunjukkan tidak adanya hubungan linier. Sebagai alat statistik yang sangat umum digunakan, korelasi Pearson membantu dalam mengidentifikasi dan mengukur derajat ketergantungan antara dua variabel yang diukur pada skala interval.

METODE PENELITIAN

Penelitian ini dilakukan di Universitas Tidar. Pada penelitian menggunakan Matlab R2017a sebagai alat bantu untuk proses pembuatan model peramalan beban listrik. Sedangkan bahan yang digunakan sebagai penunjang kelancaran proses penelitian merupakan data historis yang diperoleh dari PT. PLN (Persero) UP3 Cilacap dan data dari Badan Pusat Statistika (BPS) Kabupaten Cilacap.

Data akan dibagi menjadi tiga yaitu data latih, data uji, dan data peramalan. Kemudian data akan diproses menggunakan metode jaringan syaraf tiruan (JST) *backpropagation*. Variabel data yang digunakan berupa data yang berkaitan dengan pertumbuhan beban listrik. Beberapa data yang digunakan yaitu data jumlah penduduk, produk domestik regional bruto (PDRB), jumlah pelanggan, dan daya tersambung sebagai *input* data. Sedangkan data konsumsi daya listrik sebagai target dan *output* data. Mekanisme dalam melakukan peramalan dapat dilihat pada Gambar 2.



Gambar 2. Diagram Alir Tahap Penelitian

Merancang Arsitektur Jaringan

Tahap ini merupakan proses penting sebelum melakukan pelatihan. Pada tahap ini menggunakan 3 layer yaitu *input layer*, *hidden layer*, dan *output layer*. Pada *input layer* terdapat 4 variabel *input* yang berbeda. Untuk *hidden layer* menggunakan 1 unit *hidden layer* begitu juga dengan *output layer* menggunakan 1 unit *output*. Pada tahap ini arsitektur jaringan yang berbeda yang terletak pada jumlah *neuron hidden layer*. Jumlah *neuron hidden layer* yang digunakan sebesar 25, 30, dan 37 unit *neuron hidden layer*. Karena bertujuan untuk mendapatkan arsitektur jaringan terbaik untuk melakukan peramalan. Arsitektur terbaik dapat dilihat pada nilai *error* pada proses pelatihan dan pengujian. Karena semakin rendah nilai *error*nya maka tingkat akurasi peramalan semakin baik.

Inisialisasi Parameter

Dalam konteks peramalan, inisialisasi parameter merujuk pada proses menetapkan nilai awal dalam suatu jaringan sebelum memulai proses pelatihan. Proses inisialisasi parameter ini memiliki peran penting dalam konvergensi pembelajaran dan kinerja keseluruhan model. Dalam penelitian ini telah menetapkan parameter untuk melakukan proses pelatihan yang dapat dilihat pada Tabel 3

Tabel 3. Parameter Pelatihan JST *Backpropagation*

| Parameter | Digunakan |
|----------------------------|---------------------------|
| <i>Epoch</i> | 10000 |
| <i>Fail</i> | 100 |
| <i>Learning Rate</i> | 0.01 |
| <i>MSE Goals</i> | 0.001 |
| <i>Max. Gradient</i> | 0.00001 |
| Jumlah <i>Hidden Layer</i> | 1 |
| Fungsi Pelatihan | <i>TrainGDX</i> |
| Fungsi Aktivasi | <i>Logsig dan Purelin</i> |
| Fungsi Performa | MSE |
| Fungsi Pembelajaran | <i>LearnGD</i> |

HASIL DAN PEMBAHASAN

Setelah melalui serangkaian pengujian menggunakan tiga model Jaringan Saraf Tiruan (JST) *backpropagation* dengan variasi pada jumlah *neuron* pada *hidden layer*, penulis berhasil menentukan satu model terbaik dengan arsitektur jaringan 4-37-1 yang digunakan pada tahap selanjutnya yaitu peramalan. Pada tahap pelatihan model jaringan ini mendapatkan MSE sebesar 0.00099999 dan MAPE sebesar 5,44%. Sedangkan pada tahap pengujian mendapatkan MSE sebesar 0.0009493 dan MAPE sebesar 3,99%. Model ini dipilih berdasarkan hasil evaluasi kinerja dan akurasi yang diperoleh selama tahap pengujian. Selanjutnya, hasil peramalan dapat dilihat pada Tabel 4 dan divisualisasikan dalam bentuk grafik pada Gambar 3

Tabel 4. Hasil Peramalan

| Tahun | Konsumsi Daya Listrik (KWH) |
|-------|-----------------------------|
| 2023 | 1,657,180,675 |
| 2024 | 1,903,459,281 |
| 2025 | 2,005,081,611 |
| 2026 | 2,046,560,113 |
| 2027 | 2,113,444,197 |
| 2028 | 2,146,454,172 |
| 2029 | 2,234,250,334 |
| 2030 | 2,398,263,244 |



Gambar 3. Hasil Peramalan

Dari analisis hasil peramalan pada Gambar 3, dapat diperhatikan bahwa terjadi peningkatan signifikan dalam konsumsi daya listrik setiap tahunnya. Kenaikan ini secara langsung terkait dengan pertumbuhan jumlah penduduk, perkembangan ekonomi, peningkatan jumlah pelanggan, dan peningkatan penggunaan daya listrik secara umum. Hal ini dapat dibuktikan dengan melakukan koefisien korelasi pada data peramalan dengan variabel *input* X1 (jumlah penduduk); X2 (PDRB); X3 (jumlah pelanggan); dan X4 (daya tersambung); dengan variabel *output/target* T yang merupakan Konsumsi daya listrik. Hasil dari koefisien korelasi dapat dilihat pada Tabel 5

Tabel 5. Koefisien Korelasi antar Variabel

| Variabel <i>Input</i> | Korelasi Terhadap Y |
|-----------------------|---------------------|
| X1 | 0.92578 |
| X2 | 0.97105 |
| X3 | 0.9828 |
| X4 | 0.96321 |

KESIMPULAN

Dari hasil analisis dan pembahasan, dapat disimpulkan bahwa penggunaan metode jaringan saraf tiruan (JST) *backpropagation* dengan arsitektur dan parameter yang telah ditentukan, seperti jumlah *epoch*, *learning rate*, dan jumlah *neuron hidden layer* berpengaruh besar terhadap konvergensi data, pola regresi, dan waktu pemrosesan pada saat proses pelatihan dan pengujian.

Pada pelatihan dan pengujian yang telah dilakukan ketiga arsitektur dengan parameter yang berbeda didapatkan model arsitektur jaringan terbaik untuk melakukan peramalan yaitu dengan menggunakan arsitektur jaringan 4 input – 37 unit neuron hidden layer- 1 *output* dengan *maximum epoch* 10000; *learning rate* 0.01 menghasilkan MSE sebesar 0,0009999 dan MAPE sebesar 5,44%. Sedangkan pada proses pengujian menghasilkan MSE sebesar 0,0009493 dan MAPE sebesar 3,99%. Kemudian jika dilihat dari kriteria nilai MAPE, model JST *Backpropagation* dengan 4 input – 37 unit *neuron hidden layer*- 1 *output* dengan *epoch* 10000; *learning rate* 0.01 memiliki akurasi sangat baik untuk melakukan peramalan.

DAFTAR REFERENSI

- Afinda, Y. E., & Budiono, G. (2020). Peramalan Jangka Panjang Beban Listrik Sektor Rumah Tangga di Jawa Timur Menggunakan Metode Trend Proyeksi dan Regresi Linier. *El Sains : Jurnal Elektro*, 2(1). <https://doi.org/10.30996/elsains.v2i1.4012>
- Amalia, A. M. (2020). Modifikasi Jaringan Backpropagation Dengan Particle Swarm Optimization Untuk Peramalan Curah Hujan. In *Repository.Unej.Ac.Id*. Retrieved from <https://repository.unej.ac.id/handle/123456789/102589>
- Darto, et all. (2005). *Buku Referensi Statistika untuk Ekonomi dan Bisnis*. 1–22.
- Dewi, I. P. (2015). STATISTIKA. In *A psicanalise dos contos de fadas. Tradução Arlene Caetano*. Retrieved from <https://batukota.bps.go.id/publication/download>
- Djohar, A., & Musarudin, M. (2017). Analisis Kebutuhan dan Penyediaan Energi Listrik di Kabupaten Konawe Kepulauan Tahun 2017-2036 dengan Menggunakan Perangkat Lunak Leap. *Fortei 2017*, 293–298.
- Fadilah, M. N., Yusuf, A., & Huda, N. (2021). Prediksi Beban Listrik Di Kota Banjarbaru Menggunakan Jaringan Syaraf Tiruan Backpropagation. *Jurnal Matematika Murni Dan Terapan Epsilon*, 14(2), 81. <https://doi.org/10.20527/epsilon.v14i2.2961>
- Fadlilah, N., Harjanto, I., & Novita, M. (2021). Prediksi Beban Listrik Jangka Panjang Di Wilayah Jawa Tengah Menggunakan Algoritma Jaringan Syaraf Tiruan Backpropagation. *Science and Engineering ...*, 6(Sens 5), 598–604.
- Gustriansyah, R. (2017). Analisis Metode Single Exponential Smoothing Dengan Brown Exponential Smoothing Pada Studi Kasus Memprediksi Kuantiti Penjualan Produk Farmasidi Apotek. *Seminar Nasional Teknologi Informasi Dan Multimedia*, 7–12.
- Kristianto, A., Handoko, S., & Karnoto. (2018). *Aplikasi Jaringan Syaraf Tiruan Untuk Proyeksi Kebutuhan Energi Listrik Provinsi D.I.Yogyakarta Tahun 2016-2025*.
- M. H. M. R. Shyamali Dilhani, N. M. W., & Kumara, and K. J. C. (2021). Electricity Load Forecasting Using Optimized Artificial Neural Network. *US Patent 6,601,053*. Retrieved from <https://patents.google.com/patent/US6601053B1/en>
- Moshinsky, M. (2019). Power System Engineering Planning, Design, and Operation of Power System and Equipment. In *Nucl. Phys.* (Vol. 13).
- Rohman, F. (2022). Prediksi Beban Listrik Dengan Menggunakan Jaringan Syaraf Tiruan Metode Backpropagation. *Jurnal Surya Energy*, 5(2), 55–60. <https://doi.org/10.32502/jse.v5i2.3092>
- Septiarani, C. I. (2022). Peramalan Harga Ethereum Menggunakan Metode PSO-Backpropagation Neural Network. *Sains Dan Teknologi*, (11150331000034), 1–147.
- Siang, J. (2005). Jaringan Syaraf Tiruan dan Pemograman Menggunakan Matlab. *Pemograman Backpropagation Dengan Matlab*, pp. 247–275.
- Yuliawanti, F. D. (2022). *Optimasi jaringan syaraf tiruan menggunakan particle swarm optimization untuk prediksi kasus covid-19 di indonesia*. Retrieved from <http://digilib.uinsby.ac.id/id/eprint/53013>