

**PREDIKSI KEBUTUHAN ENERGI LISTRIK SULAWESI UTARA
MENGUNAKAN *ARTIFICIAL NEURAL NETWORK* DAN METODE
*EXPONENTIAL SMOOTHING***

Febry Hontong¹⁾, Tritiya Arungpadang²⁾, Johan Neyland³⁾

Jurusan Teknik Mesin Universitas Sam Ratulangi Manado

Jln. Kampus UNSRAT, Manado

ABSTRAK

Untuk memprediksi seberapa besar kebutuhan energi listrik Sulawesi Utara untuk satu tahun ke depan diperlukan metode yang benar. Metode terpercaya yang digunakan untuk tugas prediksi dalam penelitian ini adalah *Artificial Neural Network* dan Metode *Exponential Smoothing*.

Hasil prediksi menggunakan *Artificial Neural Network* adalah 110.38, 112.62, 111.56, 108.05, 107.95, 110.32, 109.90, 110.58, 113.26, 107.11, 115.60, 105.40 GWh. Hasil prediksi menggunakan *Exponential Smoothing* adalah 112.32, 112.70, 113.07, 113.45, 113.82, 114.19, 114.57, 114.94, 115.32, 115.69, 116.07, 116.44 GWh.

Kata Kunci : *Artificial Neural Network, Exponential Smoothing, Prediksi, Kebutuhan Energi Listrik.*

ABSTRACT

To predict the electrical energy need of North Sulawesi for one year ahead requires correct methods. The reliable methods used for the prediction task in this research are Artificial Neural Network and Exponential Smoothing.

The prediction results using Artificial Neural Network are 110.38, 112.62, 111.56, 108.05, 107.95, 110.32, 109.90, 110.58, 113.26, 107.11, 115.60, 105.40 GWh. The prediction results using Exponential Smoothing are 112.32, 112.70, 113.07, 113.45, 113.82, 114.19, 114.57, 114.94, 115.32, 115.69, 116.07, 116.44 GWh.

Key words: *Artificial Neural Network, Exponential Smoothing, Prediction, Electrical Energy Need.*

I. Pendahuluan

1.1 Latar Belakang

PT. PLN (Persero) Wilayah Suluttenggo sebagai perusahaan penyedia energi listrik untuk Sulawesi Utara harus tahu kebutuhan energi listrik yang harus

disiapkan pada periode waktu tertentu.

Penyediaan energi listrik yang sesuai dengan kebutuhan energi yang digunakan menjadi masalah yang cukup serius. Oleh karena itu prediksi kebutuhan energi listrik dalam jangka waktu tertentu perlu

dilakukan, sehingga kebutuhan energi listrik bisa terpenuhi secara tepat.

Artificial neural network (jaringan syaraf tiruan) digunakan dalam penelitian ini untuk memprediksi kebutuhan energi listrik. Keunggulan *artificial neural network* adalah dapat mengolah data yang tidak memiliki hubungan linear. Selain itu tingkat keakuratan *artificial neural network* juga sangat tinggi, dengan syarat data yang dimasukkan memenuhi syarat jumlah dan validitasnya tinggi dan juga digunakan metode *Exponential Smoothing* untuk melihat hasil mana yang terbaik jika dibandingkan dengan *Artificial Neural Network*.

Skripsi ini dilatarbelakangi oleh adanya keinginan untuk mengetahui seberapa besarkah energi listrik yang diperlukan Sulawesi Utara selama satu tahun kedepan.

1.2 Rumusan Masalah

Rumusan masalah penelitian adalah:

1. Bagaimana memprediksi kebutuhan energi listrik untuk Sulawesi Utara.
2. Bagaimana mengembangkan model *artificial neural network* metode *backpropagation* untuk memprediksi kebutuhan listrik Sulawesi Utara.
3. Bagaimana membandingkan hasil simulasi *artificial neural network* dan

hasil perhitungan menggunakan metode *exponential smoothing*.

1.3 Batasan Penelitian

Dalam penelitian ini, penulis memberikan batasan pada penelitian sebagai berikut :

1. Prediksi menggunakan *artificial neural network*, metode *backpropagation*.
2. Sebagai pembandingan digunakan metode *exponential smoothing*.
3. Data yang digunakan adalah data energi terjual (MWh).

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah memprediksi kebutuhan listrik Sulawesi Utara dengan menerapkan penggunaan *artificial neural network* dan metode *exponential smoothing*.

1.5 Manfaat Penelitian

1. Penelitian ini dilakukan untuk menerapkan ilmu yang dipelajari selama menempuh perkuliahan.
2. Berguna dalam memenuhi syarat untuk mendapat gelar Sarjana Teknik.
3. Memberi manfaat bagi PT. PLN (Persero) Wilayah Suluttenggo untuk menggunakan hasil penelitian ini sebagai tolok ukur dalam mempersiapkan kapasitas energi listrik untuk Sulawesi Utara.

II. Landasan Teori

2.1. Energi Terjual

Energi listrik yang terjual kepada pelanggan, adalah energi (kWh) yang terjual kepada pelanggan-pelanggan TT (tegangan tinggi), TM (tegangan menengah) dan TR (tegangan rendah) sesuai dengan jumlah kWh yang dibuat rekening. (Data Statistik PLN, 2015)

2.2. Metode Pemulusan Eksponensial Ganda (*Double Exponential Smoothing*): Linear Satu-Parameter Dari Brown

Persamaan yang dipakai dalam penggunaan Metode Pemulusan Eksponensial Ganda Linear Satu Parameter dari Brown adalah sebagai berikut:

$$S'_t = \alpha_p X_t + (1 - \alpha_p) S'_{t-1} \quad (2.1)$$

$$S''_t = \alpha_p X_t + (1 - \alpha_p) S''_{t-1} \quad (2.2)$$

$$a_t = S'_t + (S'_t - S''_t) = 2S'_t - S''_t \quad (2.3)$$

$$b_t = \frac{\alpha_p}{1 - \alpha_p} (S'_t - S''_t) \quad (2.4)$$

$$F_{t+m} = a_t + b_t m \quad (2.5)$$

2.3. Artificial Neural Network

(Jaringan Syaraf Tiruan)

2.3.1 Pemrograman Backpropagation Dengan Matlab

Matriks adalah elemen dasar pada Matlab yang digunakan untuk memasukan elemen-elemen matriks di antara dua kurung siku. Sebagai pembatas elemen-elemen matriks dalam satu baris digunakan spasi dan pembatas matrik digunakan semicolon (;).

Dalam *incremental*, bobot diubah setiap kali masukan diberikan ke jaringan. Sebaliknya dalam metode kelompok bobot diubah setelah semua pola masukan diberikan ke jaringan, *error* (dan suku perubahan bobot) yang terjadi dalam setiap pola masukan dijumlahkan untuk menghasilkan bobot baru. Matlab menggunakan metode pelatihan kelompok dalam iterasinya, perubahan bobot dilakukan per *epoch*.

1. Membentuk Jaringan

- Inisialisasi Jaringan

Perintah yang dipakai untuk membentuk jaringan adalah *newff* yang formatnya adalah sebagai berikut:

$$net = newff(PR, [S_1, S_2, \dots, S_N], \{TF_1, TF_2, \dots, TF_N\}, BTF, BLF, PF)$$

- Inisialisasi Bobot

Setiap kali membentuk jaringan *backpropagation*, matlab akan memberi nilai bobot dan bias awal dengan bilangan agak kecil. Bobot dan bias ini akan berubah setiap kali jaringan dibentuk. Akan tetapi jika diinginkan memberi bobot tertentu, bisa dilakukan dengan memberi nilai pada *net.IW*, *net.LW* dan *net.b*.

2) Pelatihan *Backpropagation*

Untuk melatih jaringan digunakan perintah *train*. Pelatihan dilakukan untuk meminimumkan kuadrat kesalahan rata-rata (*mse = mean square error*). Pencarian

titik minimum dengan metode gradient dilakukan dengan memberikan parameter “*trainlm*” setelah fungsi aktivasi pada perintah “*newff*”.

Agar lebih efisien, untuk nilai vektor masukan cukup dengan menggunakan perintah *minmax* (p).

Perhatikan besarnya penurunan *error* sebelum dan sesudah pelatihan. Tentu saja kita tidak harus mencari bobot awal maupun *error*. Setelah jaringan terbentuk melalui perintah *newff*, kita bisa langsung menggunakan perintah *train* untuk melatih dan melihat bobot akhirnya. Dengan memberi nilai yang diinginkan pada parameter-parameter tersebut dapat diperoleh hasil yang lebih optimal.

- *net.trainParam.show*
- *net.trainParam.epoch*
- *net.trainParam.goal*

Untuk mengetahui hasil keluaran, dibutuhkan pola masukan dan target. Untuk mengetahui besarnya *error* parameter masukan harus ditambah dengan variable untuk menyimpan *error* (*e*). Perintah yang digunakan adalah

$$[y] = \text{sim}(\text{net}, p)$$

$$[e] = t - y$$

- Dengan parameter masukan
net : nama jaringan dalam perintah *newff*
p : vektor masukan jaringan

t : vektor target jaringan.

- Dan parameter hasil
y : keluaran jaringan
e : *error* jaringan = $t - y$

3) Simulasi Jaringan

Proses simulasi dalam *Backpropagation* adalah untuk mengetahui hasil keluaran jaringan berdasarkan arsitektur jaringan, pola masukan dan fungsi aktivasi yang dipakai. Sebelum menghitung keluaran jaringan, tentukan pola masukan. Pada proses simulasi tidak menggunakan data target, untuk mengetahui hasil keluaran jaringan hanya dibutuhkan pola masukan. Semua pola data *x'* digunakan sebagai pola masukan dengan menggunakan nilai bobot dan bias awal, untuk memprediksi kebutuhan energi listrik berdasarkan hasil simulasi jaringan. Untuk mengetahui hasil keluaran pada simulasi jaringan, digunakan perintah $[y] = \text{sim}(\text{net}, p)$.

III. Metodologi Penelitian

3.1 Tempat dan Waktu Penelitian

3.1.1 Waktu :

Penelitian ini dilakukan terhitung tanggal 4 Mei 2016 sampai 30 Juli 2016.

3.1.2 Tempat :

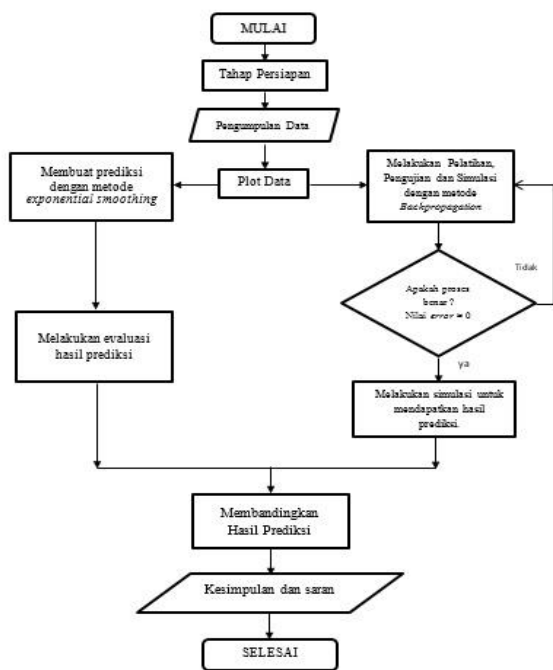
Data diambil dari Bidang Niaga Kantor Induk PT. PLN (Persero) Wilayah Suluttenggo.

3.2 Bahan dan Peralatan

Bahan yang digunakan dalam penelitian ini adalah data energi terjual (MWh) di Sulawesi Utara. Peralatan yang digunakan adalah laptop, *software* MATLAB 2014a dan *software* Microsoft Excel 2013.

3.3 Prosedur Penelitian

Metode yang digunakan dalam penelitian ini adalah *artificial neural network* dan *exponential smoothing*. Prosedur penelitian seperti diperlihatkan pada Gambar 3.1.



Gambar 3.1 Diagram Alir Penelitian

IV. HASIL DAN PEMBAHASAN

4.1. Artificial Neural Network

(Jaringan Syaraf Tiruan)

4.1.1. Pengolahan Data

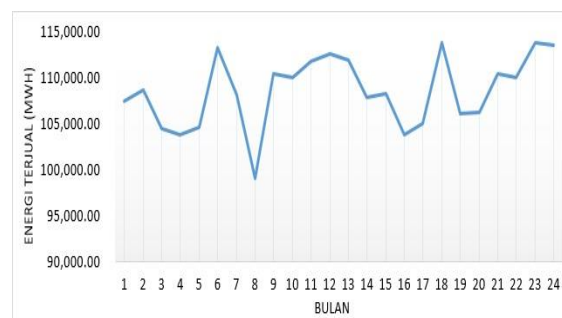
Tabel 4.1. Data Energi Terjual Sulawesi Utara Periode Juli 2013 – Juni 2016

Bulan	Periode Waktu	Energi Terjual (MWh)	Konversi Data
Juli 2014	1	107,501.05	0.5602
Agustus 2014	2	108,632.58	0.6213
September 2014	3	104,484.53	0.3972
Oktober 2014	4	103,820.79	0.3613
November 2014	5	104,576.96	0.4022
Desember 2014	6	113,197.71	0.8679
Januari 2015	7	108,162.06	0.5959
Februari 2015	8	98,983.08	0.1000
Maret 2015	9	110,456.55	0.7198
April 2015	10	110,036.42	0.6971
Mei 2015	11	111,792.43	0.7920
Juni 2015	12	112,600.97	0.8357
Juli 2015	13	111,937.52	0.7998
Agustus 2015	14	107,827.52	0.5778
September 2015	15	108,283.13	0.6024
Oktober 2015	16	103,797.93	0.3601
November 2015	17	104,957.17	0.4227
Desember 2015	18	113,752.88	0.8979
Januari 2016	19	106,091.96	0.4840
Februari 2016	20	106,220.22	0.4910
Maret 2016	21	110,388.58	0.7162
April 2016	22	110,004.62	0.6954
Mei 2016	23	113,791.48	0.9000
Juni 2016	24	113,530.31	0.8859

Sumber: PT. PLN (Persero) Wilayah Suluttenggo – Bidang Niaga

- Grafik (x') Energi Terjual Sulawesi Utara

Pembuatan grafik (x') dilakukan dengan memplot kembali volume ekspor perikanan ke dalam bentuk grafik. Pada grafik dapat dilihat hasil pengamatan selama 24 bulan.



Gambar 4.1. Grafik (x') Energi Terjual

- Pengelompokan Data Masukan dan Target

Data yang digunakan merupakan data selama 24 bulan dari nilai x' selanjutnya data dikelompokkan menjadi 12 bulan dan

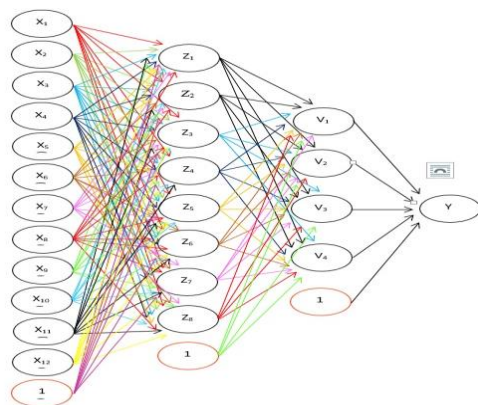
target adalah data pada bulan ke-13. Seperti terlihat pada tabel di bawah ini.

Tabel 4.2. Pengelompokan Pola Masukan dan Target

Pola	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	Target
1	0.5602	0.6213	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998
2	0.6213	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778
3	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024
4	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601
5	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227
6	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979
7	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840
8	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910
9	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162
10	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954
11	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954	0.9000
12	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954	0.9000	0.8859

- Arsitektur Jaringan 12-8-4-1

Gambar di bawah ini menunjukkan arsitektur jaringan yang terdiri dari 12 unit masukan (ditambah sebuah bias), 2 buah layer tersembunyi dan 1 unit keluaran (y). Pada layer tersembunyi-1 terdiri dari 8 unit (ditambah sebuah bias) dan pada layer tersembunyi-2 terdiri dari 4 unit (ditambah sebuah bias).



Keterangan:

- $x_1 - x_{12}$ = Pola masukan
- $z_1 - z_8$ = Layer tersembunyi 1
- $v_1 - v_4$ = Layer tersembunyi 2
- 1 = Bias

y = Keluaran

4.1.2. Hasil Pengolahan Data

- Proses Pelatihan

Untuk proses pelatihan, data yang digunakan untuk masukan dan target adalah data pola 1 sampai pola 6, seperti pada tabel 4.3.

Tabel 4.3. Data Pelatihan Kebutuhan Energi Listrik

Pola	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	target
1	0.5602	0.6213	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998
2	0.6213	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778
3	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024
4	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601
5	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227
6	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979

Berikut adalah algoritma pemrograman untuk proses pelatihan pada menu *command window* Matlab:

```
p=[];%input
t=[];%target
%%%%pembentukan jaringan 12-8-4-1
Kebutuhan Energi
net = newff(minmax(p), [8,4,1],
{'logsig','logsig','logsig'}, 'trainlm');
net.trainParam.epochs = 1000; %jumlah
epoch max pelatihan
net.trainParam.show = 50; %frekuensi
perubahan mse
net.trainParam.goal = 1e-5; %batas nilai
mse_penghentian iterasi
net = train(net,p,t); %training jaringan
```

%%%% Nilai Bobot dan Bias

```
net.IW {1,1}
net.b {1}
net.LW {2,1}
net.b {2}
```

$net.LW\{3,2\}$

$net.b\{3\}$

$[y] = sim(net,p)$

$[e] = t - y$

Hasil *training* menunjukkan bahwa iterasi diselesaikan dalam 9 *epoch*. Pada *epoch* ke 9 iterasi sudah dihentikan karena *mse* sudah lebih kecil dari batas toleransi yang ditetapkan 10^{-5} .

Tabel 4.4. Hasil Pelatihan Kebutuhan Energi Listrik

Pola	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	X ₁₁	X ₁₂	Target	HPL	Error	Ket
1	0.5602	0.6213	0.3972	0.5613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.7998	0.0002	Baik
2	0.6213	0.3972	0.5613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.5778	-0.0005	Baik
3	0.3972	0.5613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.6024	-0.0003	Baik
4	0.5613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.3601	0.0002	Baik
5	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.4227	0.0004	Baik
6	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.8979	0.0006	Baik

Tabel di atas menunjukkan hasil pelatihan dan nilai *error*. Berdasarkan hasil pelatihan, dapat dilihat bahwa jaringan mampu mengenali semua data dari 12 data yang dilatihkan. Ini ditunjukkan oleh hasil keluaran yang ditampilkan dan nilai *error* yang dicapai.

Nilai bobot dan bias dari unit masukan dan layar tersembunyi dapat dilihat pada tabel berikut:

Tabel 4.5. Bobot Dan Bias Unit Masukan

LT-1	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	X ₁₁	X ₁₂	B
Z ₁	2.7279	6.9767	0.9962	-0.3321	-2.08	-0.8265	1.7812	4.3865	11.1249	8.3215	-1.3653	-1.8731	-22.
Z ₂	4.584	4.5841	2.7319	2.2184	-4.1062	-1.1973	1.4781	-2.0409	9.8346	-5.9591	3.7557	0.907	-15.
Z ₃	-3.7983	-5.0625	1.2388	3.4105	-3.735	2.4236	-1.8668	0.2527	0.5656	10.6897	0.5404	-5.6085	0.1
Z ₄	4.9885	5.4717	3.4536	-0.6761	2.621	2.5552	1.4505	-2.1112	-8.2648	-3.5389	0.6474	-5.465	1.2
Z ₅	2.3994	6.005	0.5392	2.6817	1.2184	-2.9335	2.248	1.7964	-10.0137	-8.4159	5.6495	0.3617	1.2
Z ₆	-4.7068	-0.2823	-4.6929	-2.0426	-1.6291	0.3252	-2.7483	-2.3596	-6.5523	-6.4575	-3.2556	4.2826	17.1
Z ₇	-3.344	3.1722	2.5991	1.9762	2.6888	-1.0988	-3.4623	-0.9973	0.0448	2.4777	2.9772	5.0263	-20.
Z ₈	0.5415	-4.5626	3.6143	-3.9315	-3.9406	1.1779	-0.0429	1.884	-6.1495	-0.7095	3.3835	-3.0111	12.1

Tabel 4.6. Bobot Dan Bias Layar Tersembunyi-1

LT-2	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	Bias
V ₁	1.3447	-2.8096	-2.7861	1.2483	-2.7221	-0.3913	-3.3391	2.3605	-0.9351
V ₂	-1.2841	1.8912	0.3246	2.0814	3.044	3.9458	2.719	2.4021	-6.7481
V ₃	-3.7593	-1.2262	-1.5382	0.1069	-1.7287	-2.138	-3.1102	-2.6361	7.6774
V ₄	-1.6285	0.3416	1.286	-4.335	3.0677	-0.7876	2.1684	-1.4619	-3.9866

Tabel 4.7. Bobot Dan Bias Layar Tersembunyi-2

Keluaran	V ₁	V ₂	V ₃	V ₄	Bias
Y	-2.0129	3.1863	2.0519	3.64	-1.8413

Keterangan:

Tabel 4.7 adalah nilai bobot dari layar tersembunyi-2 (v) menuju keluaran (y).

- Proses Pengujian

Ini adalah proses untuk menguji seberapa besar pola atau data yang dikenali jaringan. Pengujian dilakukan secara bertahap yaitu dengan melakukan pengujian terhadap data yang dilatihkan dan pengujian data baru. Untuk data pengujian dapat dilihat pada tabel berikut.

Tabel 4.8. Data Pengujian Kebutuhan Energi Listrik

Pola	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	Target
7	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840
8	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910
9	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162
10	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954
11	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954	0.9000
12	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954	0.9000	0.8859

Selanjutnya dalam proses pengujian, bobot yang diperoleh dari hasil pelatihan dimasukkan dalam pengujian kemudian diuji dengan data baru untuk menentukan seberapa besar jaringan mengenali pola.

Berikut adalah algoritma pemrograman untuk proses pengujian pada menu *command window* Matlab:

$p=[]; \%input$

```

t=[ ];%target
%%%%%%%%pembentukan jaringan 12-8-4-1
Kebutuhan Energi
net = newff(minmax(p), [8,4,1],
{'logsig','logsig','logsig'}, 'trainlm');
%%%%%%%% Nilai Bobot dan Bias
net.IW {1,1}=[];
net.b {1}=[];
net.LW {2,1}=[];
net.b {2}=[];
net.LW {3,2}=[];
net.b {3}=[];
[y] = sim(net,p)
[e] = t - y
net.trainParam.epochs = 1000; %jumlah
epoch max pelatihan
net.trainParam.show = 50; %frekuensi
perubahan mse
net.trainParam.goal = 1e-5; %batas nilai
mse_penghentian iterasi
net = train(net,p,t); %training jaringan
net.IW {1,1}
net.b {1}
net.LW {2,1}
net.b {2}
net.LW {3,2}
net.b {3}
[y] = sim(net,p)
[e] = t - y

```

Pada pengujian menunjukkan bahwa iterasi diselesaikan dalam 5 *epoch*. Pada *epoch* ke 5 iterasi sudah dihentikan karena *mse* sudah lebih kecil dari batas toleransi yang ditetapkan 10^{-5} .

Untuk hasil keluaran jaringan dari pengujian dapat dilihat pada tabel berikut:

Tabel 4.9. Hasil Keluaran Pengujian Kebutuhan Energi Listrik

Pola	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	Target	HPU	Error	Kat
7	0.9939	0.1000	0.7190	0.6971	0.7920	0.8337	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4834	0.0006	Baik
8	0.1000	0.7190	0.6971	0.7920	0.8337	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4900	0.4906	0.0004	Baik
9	0.7190	0.6971	0.7920	0.8337	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4900	0.7162	0.7166	0.0002	Baik
10	0.6971	0.7920	0.8337	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4900	0.7162	0.6954	0.6954	-0.001	Baik
11	0.7920	0.8337	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4900	0.7162	0.6954	0.6900	0.8985	0.0007	Baik
12	0.8337	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4900	0.7162	0.6954	0.6900	0.8859	0.8852	0.0007	Baik

Dari 6 data baru yang telah diujikan, tampak bahwa 6 data dikenali jaringan.

- Simulasi Jaringan

Berikut adalah algoritma pemrograman untuk simulasi jaringan pada menu *command window* Matlab:

```

p=[];
%%%%%%%%pembentukan jaringan 12-8-4-1
Kebutuhan Energi
net = newff(minmax(p), [8,4,1],
{'logsig','logsig','logsig'}, 'trainlm');
%%%%%%%% Nilai Bobot dan Bias
net.IW {1,1}=[];
net.b {1}=[];
net.LW {2,1}=[];
net.b {2}=[];
net.LW {3,2}=[];
net.b {3}=[];
[y] = sim(net,p)

```

Tabel 4.10. Pola Masukan Dan Hasil Simulasi

Pola	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	HSM
1	0.5602	0.6213	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998
2	0.6213	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778
3	0.3972	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024
4	0.3613	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601
5	0.4022	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227
6	0.8679	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979
7	0.5959	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840
8	0.1000	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910
9	0.7198	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162
10	0.6971	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954
11	0.7920	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954	0.9000
12	0.8357	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954	0.9000	0.8859
13	0.7998	0.5778	0.6024	0.3601	0.4227	0.8979	0.4840	0.4910	0.7162	0.6954	0.9000	0.8859	0.7155

Dari hasil simulasi yang dilakukan, dapat dilihat bahwa jaringan mengenali semua data yang disimulasikan. Berdasarkan hasil simulasi inilah yang akan digunakan untuk memprediksi kebutuhan energi listrik pada bulan Juli 2016. Demikian juga untuk mencari hasil untuk bulan selanjutnya, digunakan cara yang sama sampai mendapat hasil untuk 12 bulan berikutnya.

Berikut adalah hasil simulasi untuk 12 bulan selanjutnya, yaitu bulan Juli 2016 sampai bulan Juni 2017.

Tabel 4.11. Hasil Simulasi Bulan Juli 2016 – Juli 2017

No.	Bulan	Kebutuhan Energi Listrik (MWh)
1	Juli 2016	110,376.29
2	Agustus 2016	112,619.77
3	September 2016	111,560.96
4	Oktober 2016	108,053.23
5	November 2016	107,947.72
6	Desember 2016	110,318.91
7	Januari 2017	109,904.28
8	Februari 2017	110,576.21
9	Maret 2017	113,262.08
10	April 2017	107,109.19
11	Mei 2017	115,599.96
12	Juni 2017	105,402.52

4.1.3. Pembahasan

- Analisa Hasil Pelatihan Kebutuhan Energi Listrik Dengan Artificial

Neural Network pada Tool Box Matlab

Hasil keluaran dari pelatihan menunjukkan bahwa jaringan dapat mengenali pola yaitu pada:

- Pola ke-1 hasil keluaran 0,7998 nilai *error* 0.0002 artinya pola dikenali
- Pola ke-2 hasil keluaran 0,5779 nilai *error* -0.0005 artinya pola dikenali
- Pola ke-3 hasil keluaran 0,6024 nilai *error* -0.0003 artinya pola dikenali
- Pola ke-4 hasil keluaran 0,3601 nilai *error* 0.0002 artinya pola dikenali
- Pola ke-5 hasil keluaran 0,4227 nilai *error* 0.0004 artinya pola dikenali
- Pola ke-6 hasil keluaran 0,8978 nilai *error* 0.0006 artinya pola dikenali

Semua pola yang dilatihkan dapat dikenali oleh jaringan hal ini berdasarkan nilai *error* yang dicapai. Dalam *artificial neural network* metode *backpropagation*, jaringan dikatakan terlatih atau dikenali jika nilai *mse* = 0 atau nilainya mendekati 0. Hal ini juga dapat dilihat pada grafik penurunan *mse*.

- Analisa Hasil Pengujian Kebutuhan Energi Listrik dengan Artificial Neural Network pada Tool Box Matlab

Dari hasil pengujian menunjukkan bahwa:

- Pola 1 hasil keluaran 0,4834 nilai *error* = 0.0006 (baik)

- b. Pola 2 hasil keluaran 0,4906 nilai *error* = 0.0004 (baik)
- c. Pola 3 hasil keluaran 0,7160 nilai *error* = 0.0002 (baik)
- d. Pola 4 hasil keluaran 0,6964 nilai *error* = -0.001 (baik)
- e. Pola 5 hasil keluaran 0,8983 nilai *error* = 0.0017 (baik)
- f. Pola 6 hasil keluaran 0.8852 nilai *error* = 0.0017 (baik)

Dari hasil pengujian menunjukkan bahwa jaringan mengenali semua pola yang di uji.

- Kriteria penilaian

Jika nilai *error* ≥ 0 , maka bernilai buruk

Jika nilai *error* ≈ 0 , maka bernilai baik

- Analisa Hasil Simulasi Kebutuhan Energi Listrik dengan Artificial Neural Network pada Tool Box Matlab

Dalam simulasi adalah untuk mengetahui hasil keluaran jaringan dengan memasukan semua pola data, pada simulasi tidak menggunakan data target hanya pola masukan dan nilai bobot awal.

- a. No 1 menunjukkan hasil keluaran pada bulan ke-13
- b. No 2 menunjukkan hasil keluaran pada bulan ke-14
- c. No 3 menunjukkan hasil keluaran pada bulan ke-15
- d. No 4 menunjukkan hasil keluaran pada bulan ke-16

- e. No 5 menunjukkan hasil keluaran pada bulan ke-17
- f. No 6 menunjukkan hasil keluaran pada bulan ke-18
- g. No 7 menunjukkan hasil keluaran pada bulan ke-19
- h. No 8 menunjukkan hasil keluaran pada bulan ke-20
- i. No 9 menunjukkan hasil keluaran pada bulan ke-21
- j. No 10 menunjukkan hasil keluaran pada bulan ke-22
- k. No 11 menunjukkan hasil keluaran pada bulan ke-23
- l. No 12 menunjukkan hasil keluaran pada bulan ke-24
- m. No 13 menunjukkan hasil keluaran pada bulan ke-25

4.2. Pemulusan Eksponensial Ganda (Double Exponential Smoothing): Metode Linear Satu-Parameter Dari Brown

4.2.1. Hasil Pengolahan Data

Berikut ini akan dilakukan peramalan dengan nilai $\alpha=0,2$, dimana nilai parameter α besarnya antara $0 < \alpha < 1$, sesuai dengan langkah dalam pemecahan Metode Linear Satu Parameter dari Brown.

Bulan ke-2 (Agustus 2014), $X_2 = 108,632.58$

- a. Perhitungan Eksponensial Tunggal

$$S'_t = \alpha X_t + (1 - \alpha)S'_{t-1}$$

$$S'_2 = 0.2 (108.632,58) + 0.8 (107.501,36) = 107.727,36$$

b. Perhitungan Eksponensial Ganda

$$S''_t = \alpha S'_t + (1 - \alpha)S''_{t-1}$$

$$S''_t = 0.2 (107.727,36) + 0.8 (107.501,36) = 107.908,40$$

c. Perhitungan nilai a

$$a_t = 2S'_t - S''_t$$

$$a_2 = 2 (107.727,36) - 107.546,31 = 107.908,40$$

d. Perhitungan nilai b

$$b_t = \frac{\alpha}{1 - \alpha} (S'_t - S''_t)$$

$$b_t = \frac{0.2}{0.8} (107.727,36 - 107.546,31) = 45,26$$

e. Peramalan untuk bulan ke-3 (September 2014), $m=1$

$$F_{t+m} = a_t + b_t m$$

$$F_{2+1} = 107,908 + 45,26 (1)$$

$$F_{Sept-14} = 107.953,66$$

Peramalan untuk bulan ke-26 (Agustus 2016), $m=2$

$$F_{26} = a_{26} + b_{26} (2)$$

$$F_{Agst-16} = 111.946,74 + 374,58 (2) = 112.695,91$$

Demikian seterusnya untuk periode selanjutnya dan dapat dilihat pada tabel berikut:

Tabel 4.12. Prediksi Kebutuhan Energi Listrik dengan $\alpha=0.2$

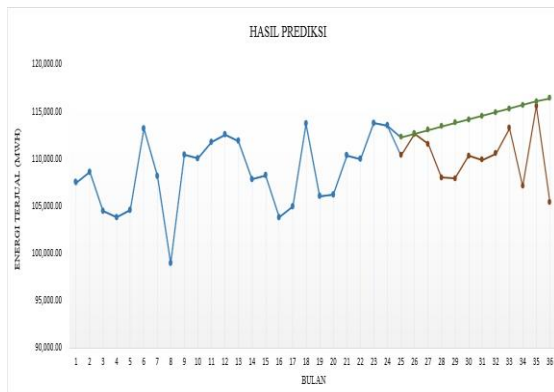
Bulan	Periode Waktu	Rasio Topikal (Data Aktual)	Eksponensial Tanggal	Eksponensial Ganda	Nilai a	Nilai b	Hasil Prediksi
Juli 2014	1	107,501.05	107,501.05	107,501.05			
Agustus 2014	2	108,632.58	107,727.36	107,546.31	107,908.40	45.26	
September 2014	3	104,484.55	107,078.79	107,452.81	106,704.77	(96.50)	107,953.66
Oktober 2014	4	109,820.79	106,427.19	107,267.48	108,606.70	(258.12)	108,611.27
November 2014	5	104,574.94	106,057.14	107,009.58	109,104.71	(219.11)	108,401.57
Desember 2014	6	111,197.71	107,683.26	107,104.71	107,869.80	95.14	104,846.61
Januari 2015	7	108,142.04	107,620.62	107,207.89	108,032.34	103.18	107,940.94
Februari 2015	8	98,981.08	105,899.11	106,944.94	104,841.28	(242.94)	108,124.52
Maret 2015	9	110,456.58	106,802.80	106,917.11	108,694.49	(27.83)	104,378.22
April 2015	10	110,056.42	107,451.92	107,424.07	107,879.77	108.94	104,646.64
Mai 2015	11	111,792.42	108,220.02	107,283.26	109,236.79	239.19	107,994.74
Juni 2015	12	112,699.97	109,176.21	107,861.88	110,699.57	378.59	109,615.98
Juli 2015	13	111,997.52	109,728.67	108,075.18	111,280.77	419.21	111,049.14
Agustus 2015	14	107,827.52	109,548.28	108,229.80	110,546.77	294.42	111,798.10
September 2015	15	108,282.12	109,125.25	108,490.89	109,779.42	140.09	110,421.29
Oktober 2015	16	106,797.90	108,067.79	108,406.27	107,729.21	(84.62)	109,940.71
November 2015	17	104,987.17	107,462.68	108,214.15	106,877.18	(192.12)	107,644.69
Desember 2015	18	112,742.88	108,707.11	108,212.74	109,100.48	91.59	104,482.04
Januari 2016	19	106,091.94	108,184.08	108,287.01	108,081.15	(25.73)	109,200.07
Februari 2016	20	106,232.22	107,791.21	108,187.87	107,394.78	(99.14)	108,059.42
Maret 2016	21	110,248.58	108,510.76	108,212.45	108,409.08	24.58	107,288.61
April 2016	22	110,004.62	108,649.59	108,299.96	108,999.20	87.42	108,422.64
Mai 2016	23	112,791.48	109,677.92	108,575.48	110,780.57	279.41	109,066.42
Juni 2016	24	112,590.21	110,648.60	108,950.06	111,944.74	274.28	111,029.98
Juli 2016	25						112,321.32 $m=1$
Agustus 2016	26						112,695.91 $m=2$
September 2016	27						113,070.49 $m=3$
Oktober 2016	28						113,445.08 $m=4$
November 2016	29						113,819.66 $m=5$
Desember 2016	30						114,194.25 $m=6$
Januari 2017	31						114,568.83 $m=7$
Februari 2017	32						114,943.42 $m=8$
Maret 2017	33						115,318.00 $m=9$
April 2017	34						115,692.59 $m=10$
Mai 2017	35						116,067.17 $m=11$
Juni 2017	36						116,441.76 $m=12$

4.3. Analisa Perbandingan Hasil Prediksi Artificial Neural Network Dan Metode Exponential Smoothing

Berdasarkan hasil yang telah dilakukan untuk setiap model peramalan, dapat dilihat perbandingan hasil tiap model seperti pada tabel berikut:

Tabel 4.13. Perbandingan Hasil

Hasil Prediksi Artificial Neural Network	Hasil Prediksi Exponential Smoothing
110,376.29	112,321.32
112,619.77	112,695.91
111,560.96	113,070.49
108,053.23	113,445.08
107,947.72	113,819.66
110,318.91	114,194.25
109,904.28	114,568.83
110,576.21	114,943.42
113,262.08	115,318.00
107,109.19	115,692.59
115,599.96	116,067.17
105,402.52	116,441.76



Gambar 4.5. Perbandingan Hasil Prediksi

Dari hasil yang didapatkan dari kedua metode, bisa dilihat bahwa *Artificial Neural Network* memiliki hasil yang lebih baik untuk memprediksi kebutuhan energi listrik di waktu yang akan datang karena kemampuan mengolah data yang akurat dan dapat membaca fluktuasi data, sesuai dengan pola data aktual. Jika dibandingkan dengan metode *Exponential Smoothing* yang hanya menggunakan satu data terakhir untuk memprediksi bulan selanjutnya.

V. PENUTUP

5.1 Kesimpulan

Hasil prediksi untuk 1 tahun kedepan adalah :

Bulan	Hasil <i>Artificial Neural Network</i>	Hasil <i>Exponential Smoothing</i>
Juli 2016	110,376.29	112,321.32
Agustus 2016	112,619.77	112,695.91
September 2016	111,560.96	113,070.49
Oktober 2016	108,053.23	113,445.08
November 2016	107,947.72	113,819.66
Desember 2016	110,318.91	114,194.25
Januari 2017	109,904.28	114,568.83
Februari 2017	110,576.21	114,943.42
Maret 2017	113,262.08	115,318.00
April 2017	107,109.19	115,692.59
Mei 2017	115,599.96	116,067.17
Juni 2017	105,402.52	116,441.76

5.2. Saran

- Proses Pelatihan pada *Artificial Neural Network* harus dilakukan

berulang-ulang sampai mendapat nilai *error* yang paling mendekati nol.

- Coba gunakan metode peramalan yang lain sebagai pembanding.

DAFTAR PUSTAKA

- Data Statistik PLN. 2015. www.pln.co.id
- Jong Jek, S. 2004. *Jaringan Saraf Tiruan Dan Pemogramannya Menggunakan MATLAB*. Andi Yogyakarta.
- Makridakis, Spyros., Wheelwright, S.C. and McGee, V.E. 1993. *Metode Dan Aplikasi Peramalan*. Jilid 1. Tangerang: Binarupa Aksara.
- Tritiya, A. 2007. *Algoritma Backpropagation*. Teknik Mesin Universitas Sam Ratulangi. Manado.