**BAB 4  
IMPLEMENTASI DAN PENGUJIAN**

Bab ini akan membahas tentang hasil yang diperoleh dari implementasi metode yang diajukan, yaitu *extreme learning machine* (ELM), untuk melakukan proses prediksi kualitas air di Danau Toba. Bab ini akan menjabarkan hasil perancangan antarmuka yang digunakan dalam proses prediksi, prosedur operasional dari aplikasi yang dirancang, tahapan pengujian yang dilakukan, serta hasil yang didapat dari setiap pengujian.

# Implementasi Sistem

Dalam penelitian ini, tahap preprocessing akan diimplementasikan ke dalam sistem menggunakan bahasa pemrograman Python, sedangkan tahap training dan testing serta visualisasi hasil pengujian akan diimplementasikan ke dalam sistem menggunakan bahasa pemrograman MATLAB.

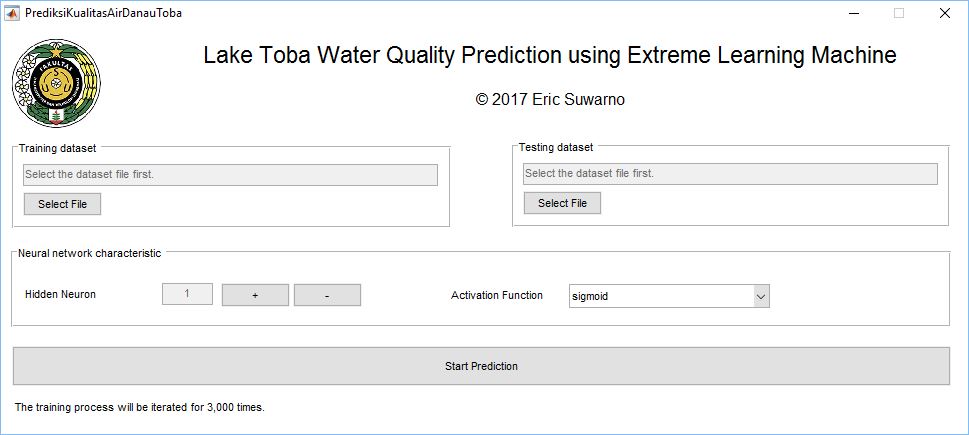
## Spesifikasi perangkat keras dan perangkat lunak yang digunakan

Spesifikasi perangkat keras dan perangkat lunak yang digunakan untuk perancangan sistem pada penelitian ini adalah sebagai berikut:

1. Prosesor Intel(R) Core(TM) i5-3317U CPU @ 1.70 GHz
2. Kapasitas memori RAM sebesar 4 GB
3. Sistem operasi Windows 10 Pro 64-bit
4. *Hard drive* yang memiliki kapasitas sebesar 500 GB
5. *Software* yang digunakan adalah Python versi 2.7 dan MATLAB versi R2015a (8.5.0.197613)
6. *Library* yang digunakan adalah *library* *elm\_train* dan *elm\_predict* yang diterbitkan oleh Zhu *et al.* (2004)

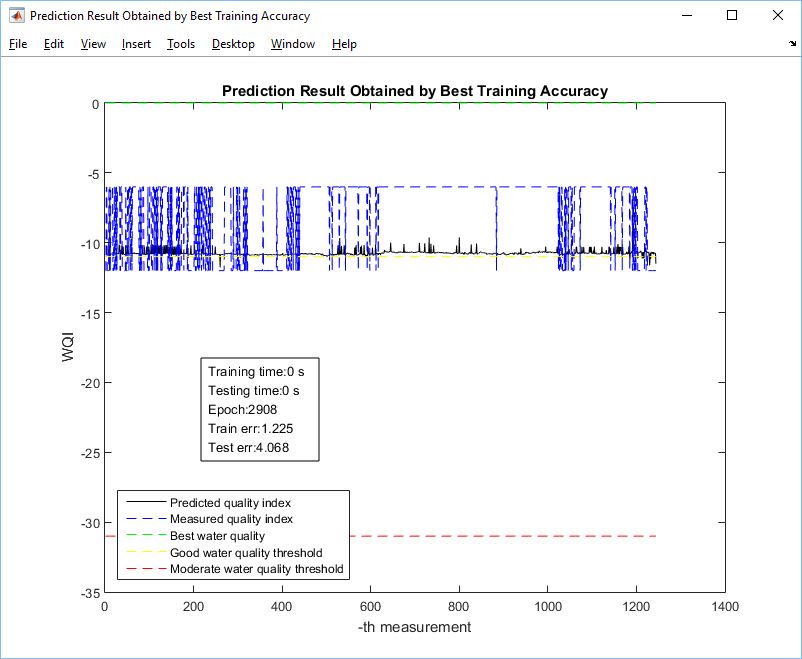
## Implementasi perancangan antarmuka

Perancangan antarmuka sistem dibuat berdasarkan rancangan yang telah dipaparkan pada bab 3. Antarmuka aplikasi yang telah dirancang pada penelitian ini ditunjukkan oleh Gambar 4.1. Antarmuka yang dirancang terdiri dari halaman utama, di mana dalam halaman utama terdapat menu-menu yang digunakan untuk menjalankan proses prediksi kualitas air menggunakan *extreme learning machine*.



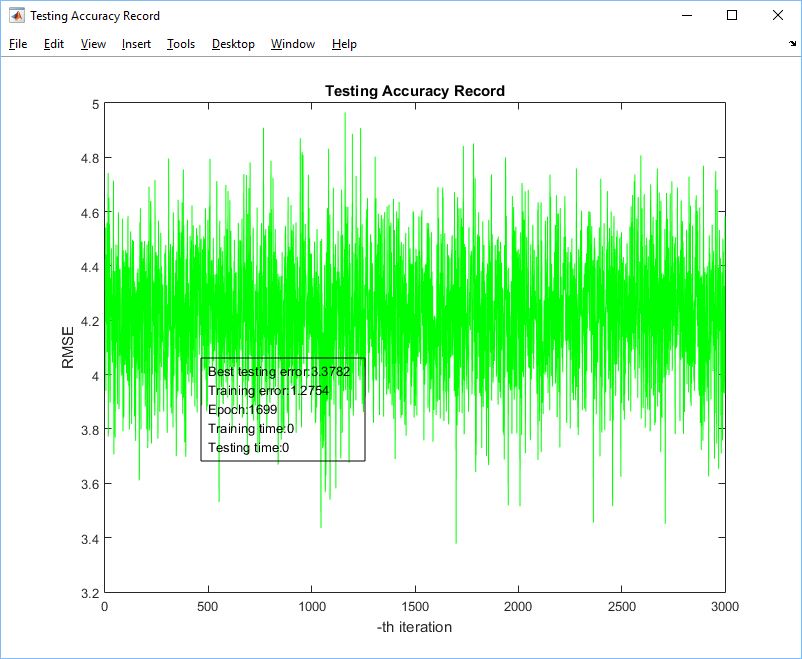
**Gambar 4.1.** Hasil rancangan antarmuka aplikasi

Setelah data latih, data uji, jumlah *hidden neuron*, dan fungsi aktivasi untuk proses prediksi telah diberikan kepada aplikasi, tahap *training* dan *testing* dalam proses prediksi akan dimulai, dan akan menghasilkan grafik hasil prediksi, yang menunjukkan pergerakan indeks kualitas air yang diperkirakan, dengan indeks kualitas air terukur sebagai perbandingan hasil prediksi. Ilustrasi dari grafik hasil prediksi yang akan ditampilkan setelah proses prediksi selesai ditunjukkan oleh Gambar 4.2. Grafik hasil prediksi akan ditampilkan pada dua jendela, berdasarkan *training error* dan *testing error* terendah. Grafik hasil prediksi juga menampilkan batas-batas tipe kualitas air, yaitu kualitas air tipe A (sangat baik), dengan nilai 0 skala Skoret, kualitas air tipe B (baik), dengan batas bawah pada -11 skala Skoret, dan kualitas air tipe C (sedang), dengan batas bawah pada -31 skala Skoret.



**Gambar 4.2.** Ilustrasi tampilan grafik hasil prediksi

Selain itu, juga akan ditampilkan dua buah grafik, seperti yang ditunjukkan pada Gambar 4.3, yang menunjukkan nilai training error dan testing error yang diperoleh pada setiap perulangan. Grafik ini akan menunjukkan nilai training error dan testing error yang diperoleh dalam tiap perulangan dalam sebuah proses prediksi.



**Gambar 4.3.** Ilustrasi grafik *testing error* dalam setiap perulangan

## Implementasi data

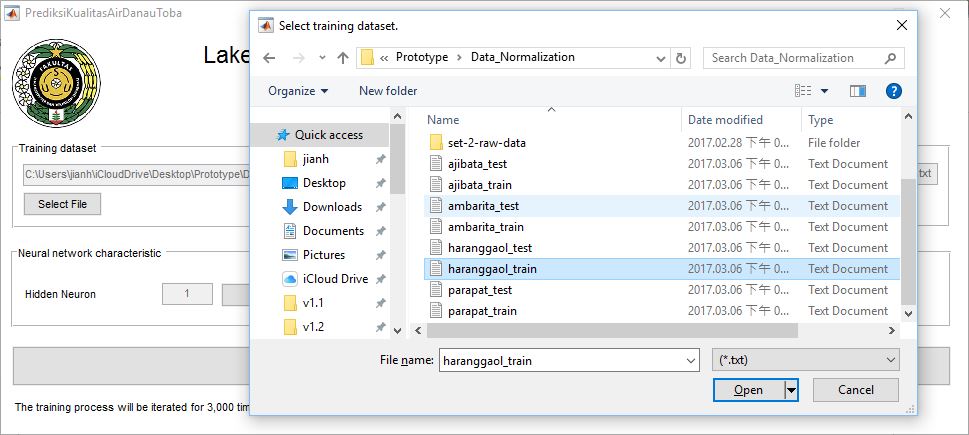
Data yang digunakan dalam penelitian ini bersumber dari penelitian yang dilakukan oleh Rahmat *et al.* (2016), di mana parameter-parameter kualitas air diukur dalam kurun waktu yang sedemikian rupa, dan disimpan dalam format dokumen teks, dengan hasil pengukuran setiap parameter dipisahkan oleh tanda titik koma. Pengukuran dilakukan pada beberapa lokasi, dan dikumpulkan dalam beberapa *file*. *File* tersebut akan diolah terlebih dahulu dalam tahap *preprocessing* sehingga dihasilkan *training dataset* dan *testing dataset* yang dapat digunakan oleh *extreme learning machine*. Perincian dari *file* yang digunakan dalam penelitian ini ditunjukkan oleh Tabel 4.1. Pemberian nama “*resume*” pada nama *file* awal menunjukkan bahwa proses pengukuran dilakukan pada waktu yang berbeda.

**Tabel 4.1.** Perincian data yang digunakan dalam penelitian

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Nama *file* awal** | **Lokasi** | **Jumlah baris data** | | | |
| **Awal** | **Setelah penyaringan** | **Latih** | **Uji** |
| 1 | DATA ajibata.txt | Ajibata | 2203 | 2112 | 1268 | 844 |
| 2 | DATA Haranggaol.txt | Haranggaol | 6374 | 3532 | 2120 | 1412 |
| 3 | DATA parapat.txt | Parapat | 2446 | 1452 | 872 | 580 |
| 4 | DATA parapat resume.txt |
| 5 | DATA samosir.txt | Ambarita | 6129 | 3113 | 1869 | 1244 |
| 6 | DATA samosir resume.txt |

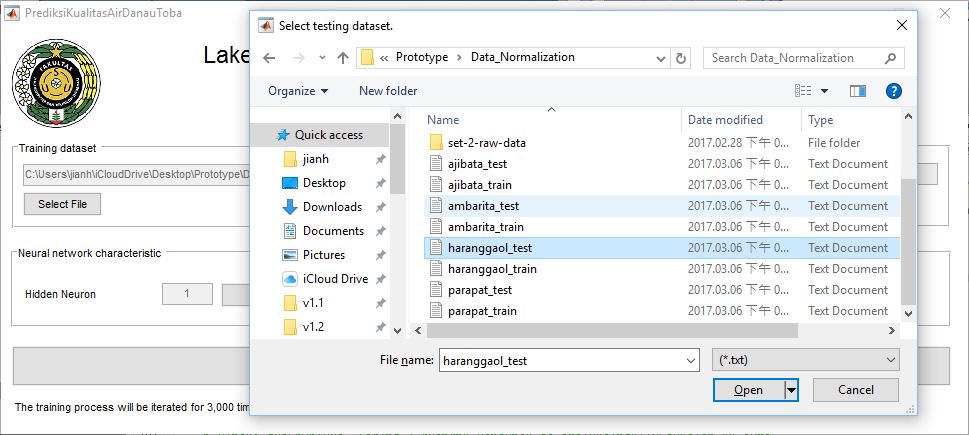
# Prosedur Operasional

Bagian ini akan memaparkan prosedur yang harus dilakukan dalam menggunakan aplikasi yang dikembangkan dalam penelitian ini. Setelah antarmuka tampil di layar, pengguna dapat memilih *training dataset* (himpunan data latih) dengan memilih tombol “*Select File*” pada bagian “*Training dataset*”. Tombol “*Select File*” akan mengarahkan pengguna pada sebuah dialog pemilihan file, di mana file yang dapat dipilih di dalam dialog ini adalah dokumen teks dengan ekstensi “.txt”. Ilustrasi dari tahap pemilihan *training dataset* ditunjukkan oleh Gambar 4.4.



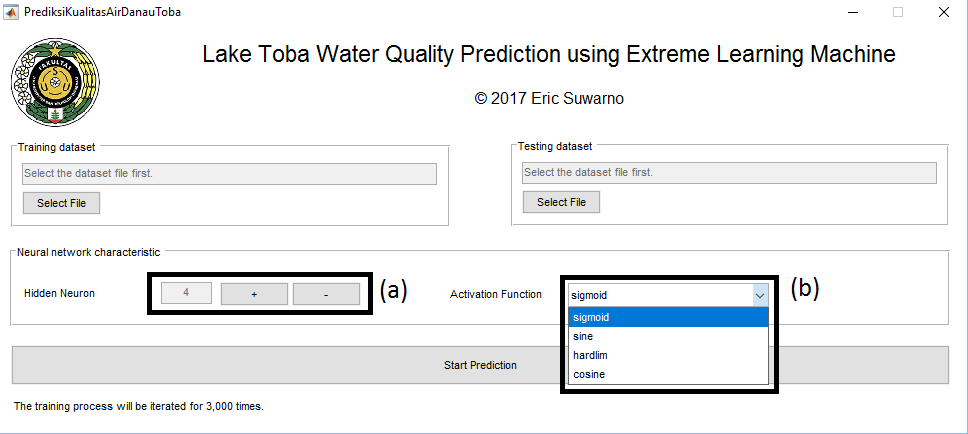
**Gambar 4.4.** Ilustrasi pemilihan *training dataset*

Setelah itu, pengguna juga dapat memilih *testing dataset* (himpunan data uji) dengan memilih tombol “*Select File*” pada bagian “*Testing dataset*”. Ketika dipilih, tombol ini akan memunculkan sebuah dialog pemilihan file, di mana pengguna dapat memilih *file* berupa dokumen teks dengan ekstensi “.txt” sebagai *testing dataset*. Ilustrasi dari tahap pemilihan testing dataset ditunjukkan oleh Gambar 4.5.



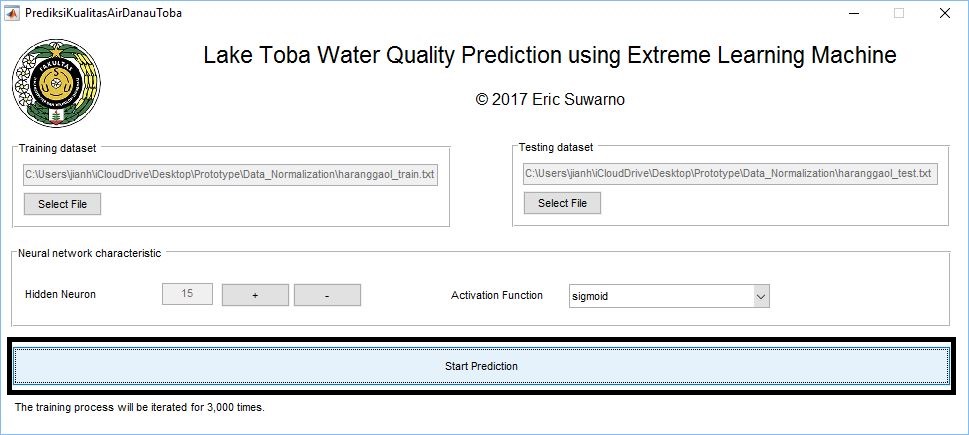
**Gambar 4.5.** Ilustrasi pemilihan *testing dataset*

Setelah itu, pengguna dapat menentukan karakteristik dari *artificial neural network* yang akan digunakan dalam proses *training*, yaitu jumlah *hidden neuron* dan fungsi aktivasi neuron. Pengguna dapat mengubah jumlah *hidden neuron* melalui tombol *plus* dan *minus* yang terdapat pada bagian “*Hidden Neuron*”, sedangkan fungsi aktivasi yang digunakan oleh setiap neuron dapat dipilih dengan memilih salah satu *item* yang terdapat dalam *combo box*. *Combo box* ini berada pada bagian “*Activation Function*”. Ilustrasi dari pemilihan hidden neuron dan fungsi aktivasi neuron ditunjukkan oleh Gambar 4.6.



**Gambar 4.6.** Ilustrasi pengaturan jumlah *hidden neuron* (a) dan fungsi aktivasi (b)

Setelah *training dataset*, *testing dataset*, dan karakteristik *artificial neural network* diberikan kepada sistem, pengguna dapat memulai proses prediksi dengan menekan tombol “*Start Prediction*”, seperti yang ditunjukkan pada Gambar 4.7. Sistem akan memulai proses prediksi, serta menampilkan grafik hasil prediksi berdasarkan *training error* dan *testing error* terendah.



**Gambar 4.7.** Ilustrasi tombol “*Start Prediction*” pada aplikasi

# Hasil Pengujian

Bagian ini akan memaparkan hasil yang didapatkan dari implementasi *extreme learning machine* (ELM) dalam melakukan proses prediksi kualitas air di Danau Toba. Proses pengujian akan dilakukan pada setiap dataset dengan menggunakan parameter yang dijelaskan pada Tabel 4.2.

**Tabel 4.2.** Parameter *artificial neural network* pada pengujian setiap *dataset*

|  |  |  |
| --- | --- | --- |
| **Pengujian ke-** | **Fungsi aktivasi** | **Jumlah neuron pada *hidden layer*** |
| 1 | *Sigmoid* | 15 |
| 2 | *Sigmoid* | 30 |
| 3 | *Sine* | 15 |
| 4 | *Sine* | 30 |
| 5 | *Hardlim* | 20 |
| 6 | *Hardlim* | 40 |
| 7 | *Hardlim* | 60 |
| 8 | *Cosine* | 15 |
| 9 | *Cosine* | 30 |

Pada setiap pengujian terhadap *dataset*, pengujian akan dilakukan sebanyak 3.000 kali perulangan. Tingkat akurasi yang diperoleh dari proses training dan testing dalam penelitian ini akan direpresentasikan melalui nilai *root mean square error* (RMSE). Nilai dari *root mean square error* (RMSE) dapat dihitung melalui persamaan 4.1:

|  |  |  |
| --- | --- | --- |
|  |  | (4.1) |

di mana adalah nilai hasil prediksi pada sampel ke-, adalah nilai yang diharapkan pada sampel ke-, , dan adalah jumlah sampel pengujian.

## Hasil pengujian terhadap dataset Ajibata

Pengujian dilakukan terhadap *dataset* hasil pengukuran di Ajibata dengan menggunakan parameter *artificial neural network* yang ditentukan pada Tabel 4.2. Hasil pengujian akan dipaparkan berdasarkan *training error* terbaik dan *testing error* terbaik yang diperoleh dalam setiap pengujian.

*Training error* terbaik yang diperoleh dalam setiap pengujian menggunakan *dataset* Ajibata dijelaskan pada Tabel 4.3. Menurut hasil pengujian pada tabel, diketahui bahwa pertambahan jumlah neuron pada *hidden layer* mengakibatkan peningkatan terhadap tingkat akurasi terbaik yang dapat dihasilkan pada tahap *training*. Hal ini dapat dibuktikan dari perbandingan hasil pengujian yang dilakukan.

Salah satu contoh dari hubungan antara jumlah neuron pada *hidden layer* dan *training error* terbaik dapat ditunjukkan oleh hasil pengujian pertama dan kedua pada Tabel 4.3. Parameter *artificial neural network* yang digunakan pada pengujian pertama adalah 15 neuron pada *hidden layer* dengan fungsi *sigmoid* sebagai fungsi aktivasi, sedangkan parameter *artificial neural network* yang digunakan pada pengujian kedua adalah 30 neuron pada *hidden layer* dengan fungsi *sigmoid* sebagai fungsi aktivasi. Pada pengujian pertama, *training error* terendah yang didapat adalah 0,12161 yang didapat pada perulangan ke-1353, dengan durasi *training* sebesar 0,046875 detik. Sedangkan pada pengujian kedua, *training error* terendah yang didapat adalah 0,053148 yang didapat pada perulangan ke-310, dengan durasi *training* sebesar 0,046875 detik.

**Tabel 4.3.** *Training error* terbaik dalam pengujian *dataset* Ajibata

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Pengujian ke-** | ***Training error* terbaik** | | | | |
| ***Training error* (RMSE)** | ***Testing error* (RMSE)** | ***Epoch* ke-** | **Durasi *training* (s)** | **Durasi *testing* (s)** |
| 1 | 0,12161 | 1,0968 | 1353 | 0,046875 | 0,03125 |
| 2 | 0,053148 | 5,1126 | 310 | 0,046875 | 0,03125 |
| 3 | 0,11341 | 1,9972 | 123 | 0,0625 | 0,03125 |
| 4 | 0,05369 | 5,9416 | 1208 | 0,09375 | 0,046875 |
| 5 | 2,2261 x 10-14 | 4,2301 x 10-14 | 1582 | 0,046875 | 0,046875 |
| 6 | 2,0764 x 10-14 | 0,77258 | 2235 | 0,046875 | 0,046875 |
| 7 | 1,9697 x 10-14 | 1,3994 | 897 | 0,046875 | 0,03125 |
| 8 | 0,12742 | 0,99394 | 1394 | 0,078125 | 0,046875 |
| 9 | 0,055086 | 4,3224 | 125 | 0,0625 | 0,0625 |

Hasil pengujian yang dijelaskan pada Tabel 4.3 juga menunjukkan bahwa bertambahnya jumlah neuron pada *hidden layer* akan memperburuk kemampuan *artificial neural network* dalam melakukan prediksi. Hal ini dapat dibuktikan melalui bertambahnya *testing error* seiring dengan bertambahnya jumlah neuron pada *hidden layer*, yang menunjukkan penurunan tingkat akurasi pada proses prediksi.

Salah satu sampel dari hasil pengujian yang membuktikan pernyataan tersebut adalah pengujian kelima, keenam, dan ketujuh. Pengujian kelima, keenam, dan ketujuh menggunakan fungsi *hardlim* sebagai fungsi aktivasi dengan jumlah *hidden neuron* yang berbeda-beda. Jumlah *hidden neuron* yang digunakan pada pengujian kelima, keenam, dan ketujuh, masing-masing adalah 20, 40, dan 60 neuron. *Testing error* yang didapat pada saat mencapai *training error* terbaik dalam pengujian kelima adalah 4,2301 x 10-14 dengan durasi *testing* sebesar 0,046875 detik. *Testing error* yang didapat pada saat mencapai *training error* terbaik dalam pengujian keenam adalah 0,77258 dengan durasi *testing* sebesar 0,046875 detik. Sedangkan pada pengujian ketujuh, *testing error* yang didapat saat mencapai *training error* terbaik adalah 1,3994 dengan durasi *testing* sebesar 0,03125 detik.

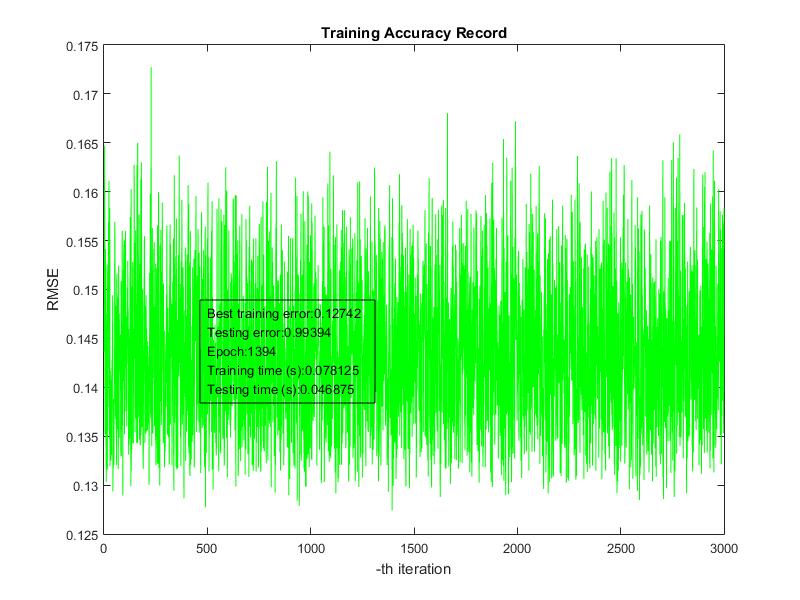
Hasil *testing error* terbaik yang didapat dari setiap pengujian menggunakan *dataset* Ajibata dijelaskan pada Tabel 4.4. Seperti yang ditunjukkan pada Tabel 4.3, hasil pengujian pada tabel menunjukkan bahwa jumlah *hidden neuron* yang digunakan pada *artificial neural network* mempengaruhi tingkat akurasi terbaik yang mampu diperoleh dari proses prediksi pada setiap pengujian. Hasil pengujian pada tabel juga menunjukkan bahwa *testing error* yang didapat pada setiap perulangan tidak selalu berkaitan dengan *training error* yang didapat. Sebagai perbandingan, pada pengujian ketiga, *training error* terbaik yang didapat adalah 0,11341 dengan *testing error* sebesar 1,9972 yang didapat pada perulangan ke-123. Sedangkan, *testing error* terbaik yang didapat pada pengujian ketiga adalah 0,076934 dengan *training error* sebesar 0,15285 yang didapat pada pengulangan ke-975.

**Tabel 4.4.** *Testing error* terbaik dalam pengujian *dataset* Ajibata

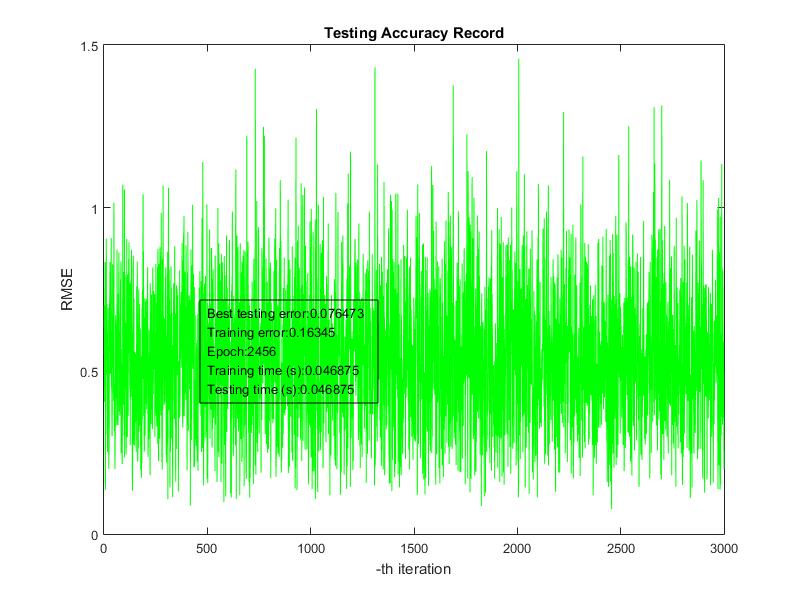
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Pengujian ke-** | ***Testing error* terbaik** | | | | |
| ***Training error* (RMSE)** | ***Testing error* (RMSE)** | ***Epoch* ke-** | **Durasi *training* (s)** | **Durasi *testing* (s)** |
| 1 | 0,16083 | 0,10341 | 382 | 0,046875 | 0,046875 |
| 2 | 0,12143 | 0,40694 | 1503 | 0,046875 | 0,046875 |
| 3 | 0,15285 | 0,076934 | 975 | 0,0625 | 0,03125 |
| 4 | 0,11242 | 0,36725 | 2822 | 0,046875 | 0,03125 |
| 5 | 2,2261 x 10-14 | 4,2301 x 10-14 | 1582 | 0,046875 | 0,046875 |
| 6 | 2,039 x 10-13 | 1,9932 x 10-13 | 1318 | 0,0625 | 0,03125 |
| 7 | 1,0297 x 10-13 | 1,1631 x 10-13 | 2366 | 0,0625 | 0,0625 |
| 8 | 0,16345 | 0,076473 | 2456 | 0,046875 | 0,046875 |
| 9 | 0,11972 | 0,32694 | 2899 | 0,0625 | 0,03125 |

Hasil pengujian juga menunjukkan bahwa pada extreme learning machine, *training error* dan *testing error* yang diperoleh pada setiap pengulangan dipengaruhi oleh input weight dan bias yang dihasilkan dari proses training. Hal ini dikarenakan *input weight* dan *bias* pada *extreme learning machine* diperoleh melalui proses pengacakan.

Seperti yang ditunjukkan pada Gambar 4.8 dan Gambar 4.9 pada pengujian kedelapan, *training error* dan *testing error* cenderung berubah-ubah untuk setiap perulangan. Hal ini menyebabkan hasil pengujian terbaik dapat diperoleh dengan waktu yang lebih singkat.



**Gambar 4.8.** Grafik *training error* dalam setiap perulangan yang diperoleh  
pada pengujian kedelapan menggunakan *dataset* Ajibata



**Gambar 4.9.** Grafik *testing error* dalam setiap perulangan yang diperoleh  
pada pengujian kedelapan menggunakan *dataset* Ajibata

## Hasil pengujian terhadap dataset Ambarita

Pengujian dilakukan terhadap *dataset* hasil pengukuran di Ambarita dengan menggunakan parameter *artificial neural network* yang ditentukan pada Tabel 4.2. Hasil pengujian akan dipaparkan berdasarkan *training error* terbaik dan *testing error* terbaik yang diperoleh dalam setiap pengujian.

Tabel 4.5 menunjukkan *training error* terbaik yang berhasil diperoleh dalam setiap pengujian menggunakan *dataset* Ambarita. Seperti hasil yang diperoleh pada pengujian menggunakan *dataset* Ajibata, hasil pengujian yang ditunjukkan pada tabel menunjukkan bahwa pertambahan jumlah *hidden neuron* mempengaruhi tingkat akurasi terbaik yang diperoleh pada saat *training*.

**Tabel 4.5.** *Training error* terbaik dalam pengujian *dataset* Ambarita

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Pengujian ke-** | ***Training error* terbaik** | | | | |
| ***Training error* (RMSE)** | ***Testing error* (RMSE)** | ***Epoch* ke-** | **Durasi *training* (s)** | **Durasi *testing* (s)** |
| 1 | 1,20750 | 3,6577 | 1802 | 0,078125 | 0,046875 |
| 2 | 1,14940 | 3,5106 | 2507 | 0,062500 | 0,046875 |
| 3 | 1,20940 | 3,7979 | 918 | 0,062500 | 0,062500 |
| 4 | 1,15340 | 3,5509 | 2348 | 0,078125 | 0,062500 |
| 5 | 0,86710 | 2,1500 | 1347 | 0,062500 | 0,062500 |
| 6 | 0,85058 | 2,9877 | 1552 | 0,093750 | 0,093750 |
| 7 | 0,79288 | 4,7899 | 1836 | 0,093750 | 0,031250 |
| 8 | 1,21270 | 3,6513 | 514 | 0,078125 | 0,109380 |
| 9 | 1,15280 | 3,6166 | 1384 | 0,078125 | 0,062500 |

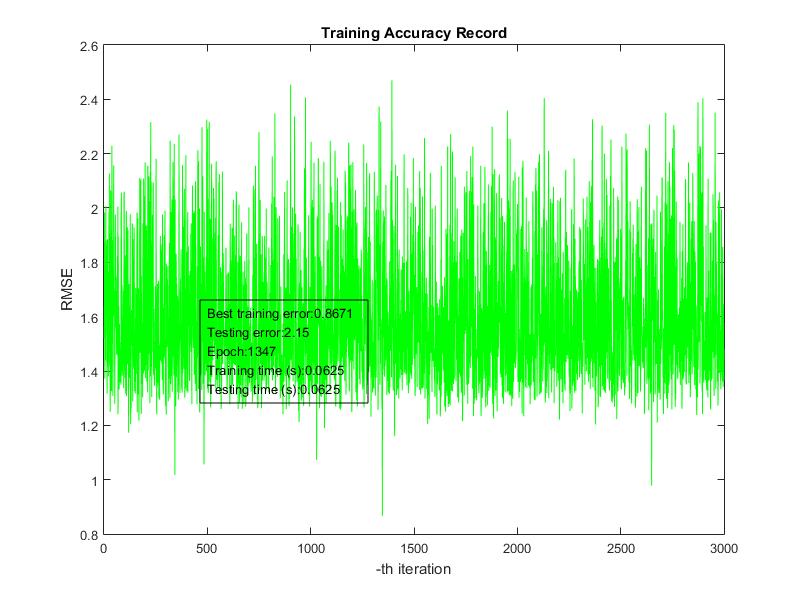
Dari hasil pengujian pada Tabel 4.5, diketahui juga bahwa dari empat fungsi aktivasi yang digunakan dalam pengujian, fungsi *hardlim* mampu memberikan hasil prediksi dengan tingkat *error* terendah. Hal ini dapat dibuktikan dari *training error* terendah yang diperoleh dari pengujian kelima, keenam, dan ketujuh menggunakan dataset Ambarita. *Training error* terbaik yang dihasilkan dari pengujian kelima, keenam, dan ketujuh lebih rendah dibandingkan dengan *training error* terbaik yang diperoleh pada pengujian lainnya.

*Testing error* terendah yang diperoleh dari pengujian menggunakan *dataset* hasil pengukuran di Ambarita ditunjukkan pada Tabel 4.6. Seperti pada *testing error* terendah yang diperoleh menggunakan dataset Ajibata, diketahui juga bahwa dari hasil pengujian dataset, testing error yang diperoleh tidak berhubungan erat dengan training error yang dihasilkan untuk setiap perulangan. Namun, training error yang lebih rendah akan meningkatkan kemungkinan untuk mendapatkan hasil prediksi dengan testing error yang lebih rendah, yang akan meningkatkan tingkat akurasi dari proses prediksi.

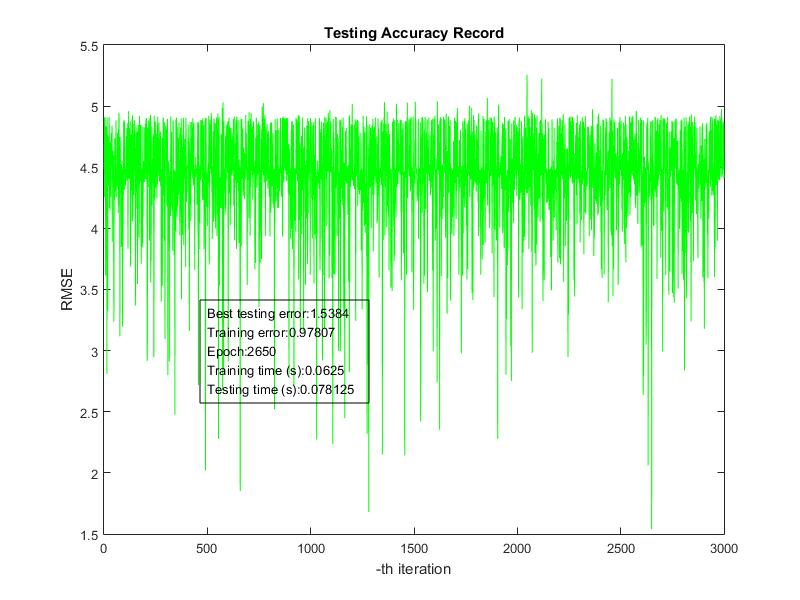
**Tabel 4.6.** *Testing error* terbaik dalam pengujian *dataset* Ambarita

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Pengujian ke-** | ***Testing error* terbaik** | | | | |
| ***Training error* (RMSE)** | ***Testing error* (RMSE)** | ***Epoch* ke-** | **Durasi *training* (s)** | **Durasi *testing* (s)** |
| 1 | 1,27030 | 3,4397 | 1554 | 0,046875 | 0,031250 |
| 2 | 1,20190 | 3,3278 | 2654 | 0,062500 | 0,046875 |
| 3 | 1,23950 | 3,4364 | 2827 | 0,046875 | 0,046875 |
| 4 | 1,18190 | 3,3549 | 2056 | 0,062500 | 0,093750 |
| 5 | 0,97807 | 1,5384 | 2650 | 0,062500 | 0,078125 |
| 6 | 1,16880 | 1,3560 | 1315 | 0,093750 | 0,078125 |
| 7 | 1,12140 | 1,4468 | 274 | 0,125000 | 0,046875 |
| 8 | 1,24690 | 3,4402 | 2419 | 0,046875 | 0,031250 |
| 9 | 1,18810 | 3,3000 | 452 | 0,078125 | 0,031250 |

Seperti hasil pengujian terhadap dataset hasil pengukuran di Ajibata, nilai training error dan *testing error* yang diperoleh pada setiap perulangan dalam pengujian sangat dipengaruhi oleh nilai *input weight* dan *bias* yang diperoleh dari proses pengacakan. Sehingga, hasil dengan *testing error* terendah dapat diperoleh tanpa memerlukan waktu komputasi yang lebih lama, seperti yang ditunjukkan pada Gambar 4.10 dan 4.11, yang menunjukkan training error dan testing error yang diperoleh pada pengujian kelima. Dari Gambar 4.10 dan 4.11, dapat dibuktikan bahwa dalam pengujian kelima, hasil dengan training error terendah dapat diperoleh pada perulangan ke-1.347 dalam 3.000 perulangan. Begitu juga dengan hasil dengan testing error terendah, yang dapat diperoleh pada perulangan ke-2.650 dalam 3.000 perulangan.



**Gambar 4.10.** Grafik *training error* yang diperoleh pada setiap perulangan  
dalam pengujian menggunakan *dataset* Ambarita



**Gambar 4.10.** Grafik *testing error* yang diperoleh pada setiap perulangan  
dalam pengujian menggunakan *dataset* Ambarita

## Hasil pengujian terhadap dataset Haranggaol

Hasil pengujian terhadap dataset Haranggaol dengan menggunakan parameter *artificial neural network* sesuai dengan Tabel 4.2 ditunjukkan oleh Tabel 4.\*.

## Hasil pengujian terhadap dataset Parapat

Hasil pengujian terhadap dataset Parapat dengan menggunakan parameter *artificial neural network* sesuai dengan Tabel 4.2 ditunjukkan oleh Tabel 4.\*.

## Perbandingan hasil antara fungsi aktivasi

## Perbandingan dengan algoritma backpropagation