BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Tahap implementasi merupakan tahap pembuatan perangkat lunak, tahap lanjut dari tahap perancangan. Tahap yang dilakukan untuk menerjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dimengerti oleh komputer serta penerapan perangkat lunak pada keadaan yang sebenarnya.

4.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras (*hardware*) yang digunakan untuk membangun sistem pengidentifikasi penyakit mata menular ini dapat dilihat pada tabel 4.1.

Tabel 4.1 Spesifikasi Implementasi Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Processor	Intel Core i3 2.00 GHz
2	Monitor	14.1”
3	Memori	2 Gb

4.1.2 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak (*software)* yang digunakan untuk membangun sistem pengidentifikasi penyakit mata menular ini dapat dilihat pada tabel 4.2.

Tabel 4.2 Spesifikasi Implementasi Perangkat Lunak

No	Perangkat Lunak
1	Sistem operasi Windows 7
2	Visual Studio 2010
3	MySQL ver 5
4	MySQL connector ver 6

4.1.3 Implementasi Database

Pembuatan basis data (*database*), dilakukan dengan menggunakan MySQL. Implementasi basis data (*database*) dalam bahasa SQL adalah seperti dapat dilihat pada tabel 4.3.

Tabel 4.3 Implementasi Database

No	Proses	SQL
1	Pembuatan Database	CREATE DATABASE db_mata_b;
2	Tabel “penderita”	CREATE TABLE `penderita` (`kode_penderita` int(11) NOT NULL AUTO_INCREMENT, `x1` set('0','1') NOT NULL DEFAULT '0', `x2` set('0','1') NOT NULL DEFAULT '0', `x3` set('0','1') NOT NULL DEFAULT '0', `x4` set('0','1') NOT NULL DEFAULT '0', `x5` set('0','1') NOT NULL DEFAULT '0', `x6` set('0','1') NOT NULL DEFAULT '0', `x7` set('0','1') NOT NULL DEFAULT '0', `x8` set('0','1') NOT NULL DEFAULT '0', `x9` set('0','1') NOT NULL DEFAULT '0', `x10` set('0','1') NOT NULL DEFAULT '0', `x11` set('0','1') NOT NULL DEFAULT '0', `x12` set('0','1') NOT NULL DEFAULT '0', `x13` set('0','1') NOT NULL DEFAULT '0', `x14` set('0','1') NOT NULL DEFAULT '0', `x15` set('0','1') NOT NULL DEFAULT '0', `x16` set('0','1') NOT NULL DEFAULT '0', `x17` set('0','1') NOT NULL DEFAULT '0', `x18` set('0','1') NOT NULL DEFAULT '0', `x19` set('0','1') NOT NULL DEFAULT '0', `identifikasi` set('Konjungtivitis','Keratokonjungtivitas

		vernalis','Endoftalmitis','Selulitis Orbitalis','Trakoma','Blefaritis','Ulkus Kornea') NOT NULL, PRIMARY KEY (`kode_penderita`)) ENGINE=InnoDB;
3	Tabel "penyakit"	CREATE TABLE `penyakit` (`kode_penyakit` varchar(3) NOT NULL, `nama_penyakit` varchar(30) NOT NULL, PRIMARY KEY (`kode_penyakit`)) ENGINE=InnoDB;

4.1.4 Implementasi Class

Implementasi *class* merupakan implementasi dari analisis kelas yang ada pada *class diagram.* Deskripsi implementasi *class* pada sistem yang dibangun dapat dilihat pada tabel 4.4.

Tabel 4.4 Implementasi Class

No	Nama Class	Nama File
1	cl_penderita	cl_penderita.cs
2	cl_pelatihan	cl_pelatihan.cs

4.1.5 Implementasi Antarmuka

Implementasi antarmuka dilakukan untuk setiap tampilan program yang dibangun dalam bentuk *file code*. Berikut ini adalah implementasi antarmuka untuk pengguna. Deskripsi implementasi antarmuka pada sistem pengidentifikasi penyakit mata menular ini dapat dilihat pada tabel 4.5.

Tabel 4.5 Implementasi Antarmuka

No	Nama Antarmuka	Proses	Nama File	Keterangan
1	Form Utama	Tampilan awal	FormUtama.cs	Tampilan awal sistem
2	Form Penderita	Proses	FormPenderita.cs	Tampilan untuk

		pengolahan data penderita		menampilkan data penderita.
3	Form Identifikasi	Proses metode *backpropagation* untuk mengidentifikasi pola baru	FormIdentifikasi.cs	Tampilan untuk mengidentifikasi penyakit dari pola baru.
4	Form Simulasi	Proses metode *backpropagation* untuk simulasi	FormSimulasi.cs	Tampilan untuk melakukan simulasi metode *backpropagation*

4.2 Pengujian Metode

Pengujian metode bertujuan untuk mengetahui performansi dari metode *backpropagation* yang diimplementasikan pada sistem pengidentifikasi penyakit mata menular. Pengujian metode ini menggunakan *cross validation* dengan 10 *fold*, data keseluruhan dibagi menjadi 10 *subset* dimana setiap *subset* memiliki data yang berbeda. Data yang digunakan dalam pengujian ini merupakan data dari 7 macam penyakit mata menular sebanyak 20 data setiap penyakitnya. Jumlah keseluruhan data yang digunakan adalah 140 data, data ini dibagi menjadi data 10 *subset* yang terdiri dari 14 data untuk setiap *subset*. Setiap *subset* akan diuji dengan menggunakan *subset* lainnya sebagai data pelatihan, sehingga setiap pengujian dilakukan sebanyak 10 kali. Pengujian ini menghasilkan besaran akurasi dari metode *backpropagation* berdasarkan jumlah data yang teridentifikasi sesuai dengan target dan jumlah data yang teridentifikasi tidak sesuai dengan target keluaran. Hasil dari pengujian ini akan ditampilkan dalam bentuk *confusion matrix.* Pengujian ini dilakukan menggunakan sistem pengidentifikasi penyakit mata menular yang sudah dibuat.

Pada metode *backpropagation* terdapat empat buah variabel yang dapat dirubah yaitu jumlah *hidden layer*, *learning rate*, batas epoch(iterasi), dan batas

error. Perubahan nilai batas epoch(iterasi) dan batas *error* dilakukan untuk membatasi proses pelatihan metode *backpropagation,* sedangkan perubahan dari jumlah *hidden layer* dan *learning rate* mempengaruhi laju pembelajaran dan akurasi dari metode *backpropagation*.

a. Laju Pembelajaran

Laju pembelajaran dari metode ini dapat dilihat dari perubahan *error* pada tahap pelatihan metode *backpropagation*. Jika nilai *error* mecapai nilai batas *error* dengan waktu singkat maka laju pembelajaran dikatakan cepat, sedangkan jika nilai *error* mecapai nilai batas *error* dengan waktu lama maka laju pembelajaran dikatakan lambat.

b. Akurasi

Akurasi dari metode ini dapat dilihat dari banyaknya hasil prediksi sistem yang sesuai dengan hasil prediksi aslinya dan hasil prediksi sistem yang tidak sesuai. Jumlah hasil prediksi yang akurat dan jumlah hasil prediksi yang tidak akurat dirubah menjadi nilai persen untuk memudahkan dalam mengetahui nilai akurasi dari sistem.

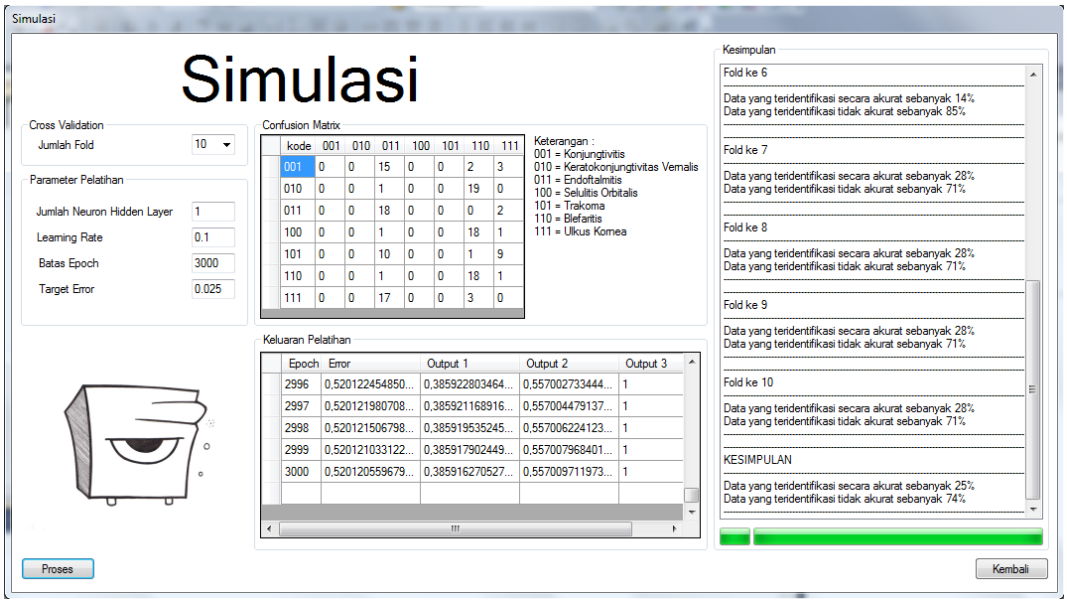
4.2.1 Skenario Pengujian 1

Pada pengujian skenario 1, pengujian metode *backpropagation* dilakukan dengan mengubah nilai jumlah *hidden layer* dari nilai 1 hingga 10, sedangkan ketiga variabel lainnya diisi dengan nilai yang sama. Berikut ini adalah nilai variabel yang digunakan pada skenario pengujian 1.

Jumlah *neuron* pada *hidden layer*	=	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
Learning Rate (α)	=	0.1
Batas Epoch	=	3000
Target Error	=	0.025

1. *Hidden Layer* dengan 1 *Neuron*

Pengujian ini dimulai dengan *neuron* pada *hidden layer* yang berjumlah 1, hasil dari pengujian ini dapat dilihat pada gambar 4.1.



Gambar 4.1 Pengujian Backpropagation dengan 1 neuron pada hidden layer

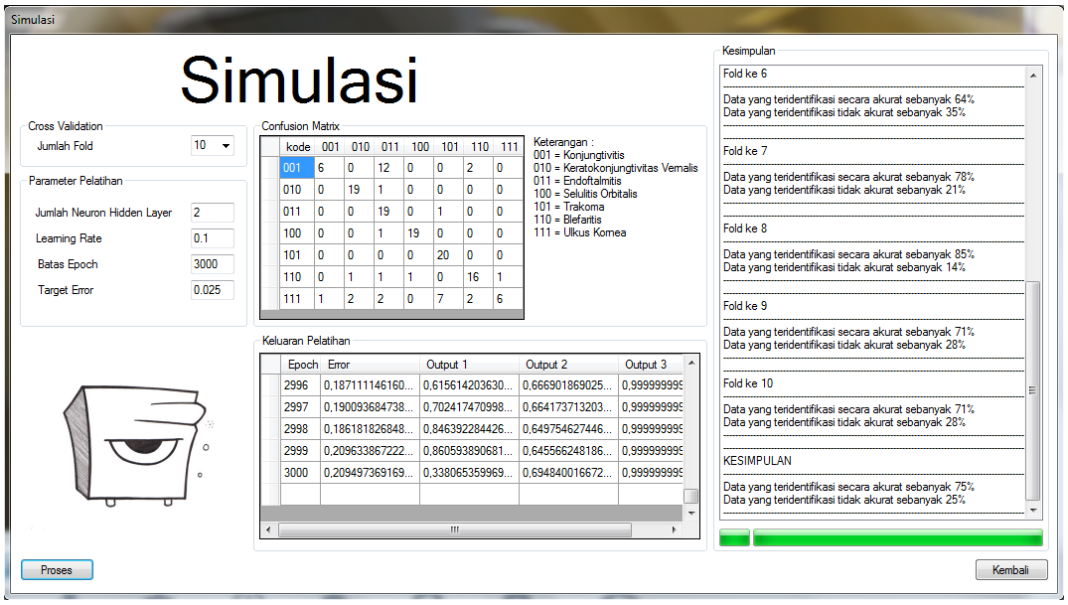
Pada pengujian dengan menggunakan 1 *neuron* pada *hidden layer* menjadikan laju pembelajaran tergolong lambat, karena pada epoch ke 3000 nilai *error* masih belum mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 25.714%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.6.

Tabel 4.6 Hasil pengujian hidden layer dengan 1 neuron

Fold	**Akurasi**
1	28%
2	28%
3	14%
4	28%
5	28%
6	14%
7	28%
8	28%
9	28%
10	28%
Rata-rata akurasi	**25.714%**

2. *Hidden Layer* dengan 2 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 2, hasil dari pengujian ini dapat dilihat pada gambar 4.2.



Gambar 4.2 Pengujian Backpropagation dengan 2 neuron pada hidden layer

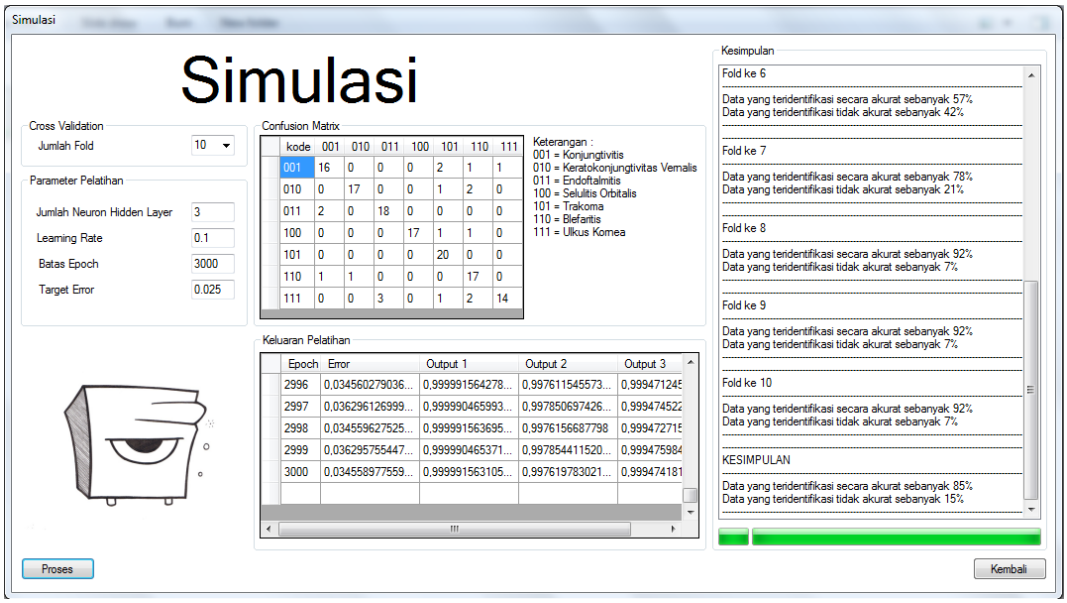
Pada pengujian dengan menggunakan 2 *neuron* pada *hidden layer* menjadikan laju pembelajaran masih tergolong lambat, karena pada epoch ke 3000 nilai *error* masih belum mencapai target *error*, namun terdapat peningkatan rata-rata akurasi. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 75%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.7.

Tabel 4.7 Hasil pengujian hidden layer dengan 2 neuron

Fold	**Akurasi**
1	78%
2	78%
3	71%
4	78%
5	71%
6	64%
7	78%
8	85%
9	71%
10	71%
Rata-rata akurasi	**75%**

3. *Hidden Layer* dengan 3 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 3, hasil dari pengujian ini dapat dilihat pada gambar 4.3.



Gambar 4.3 Pengujian Backpropagation dengan 3 neuron pada hidden layer

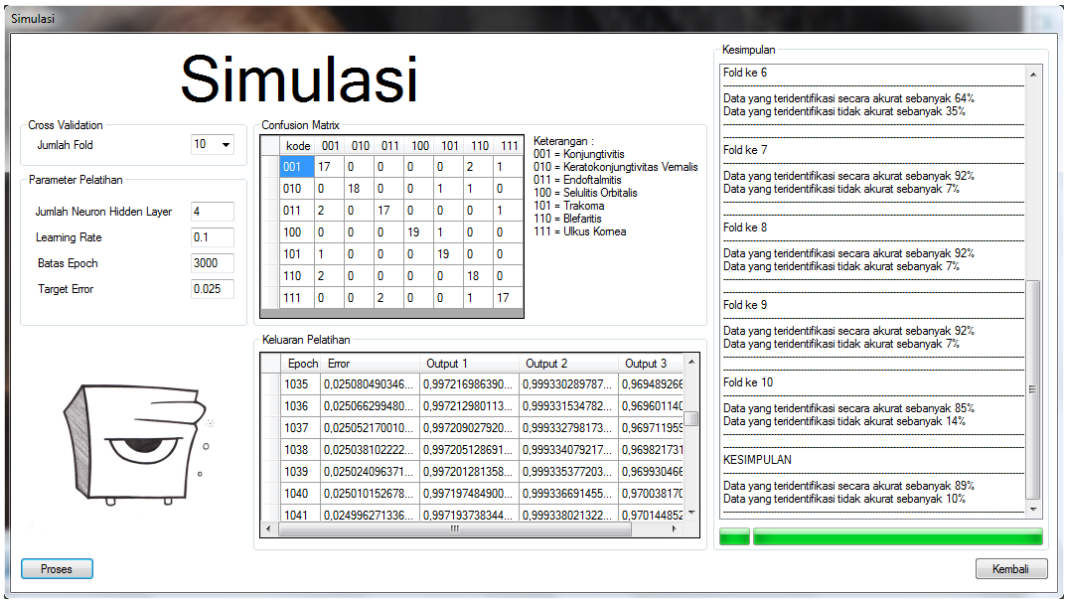
Pada pengujian dengan menggunakan 3 *neuron* pada *hidden layer* terjadi peningkatan laju pembelajaran, namun pembelajaran masih tergolong lambat karena pada epoch ke 3000 nilai *error* masih belum mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 85%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.8.

Tabel 4.8 Hasil pengujian hidden layer dengan 3 neuron

Fold	**Akurasi**
1	92%
2	92%
3	71%
4	92%
5	85%
6	57%
7	78%
8	92%
9	92%
10	92%
Rata-rata akurasi	**85%**

4. *Hidden Layer* dengan 4 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 4, hasil dari pengujian ini dapat dilihat pada gambar 4.4.



Gambar 4.4 Pengujian Backpropagation dengan 4 neuron pada hidden layer

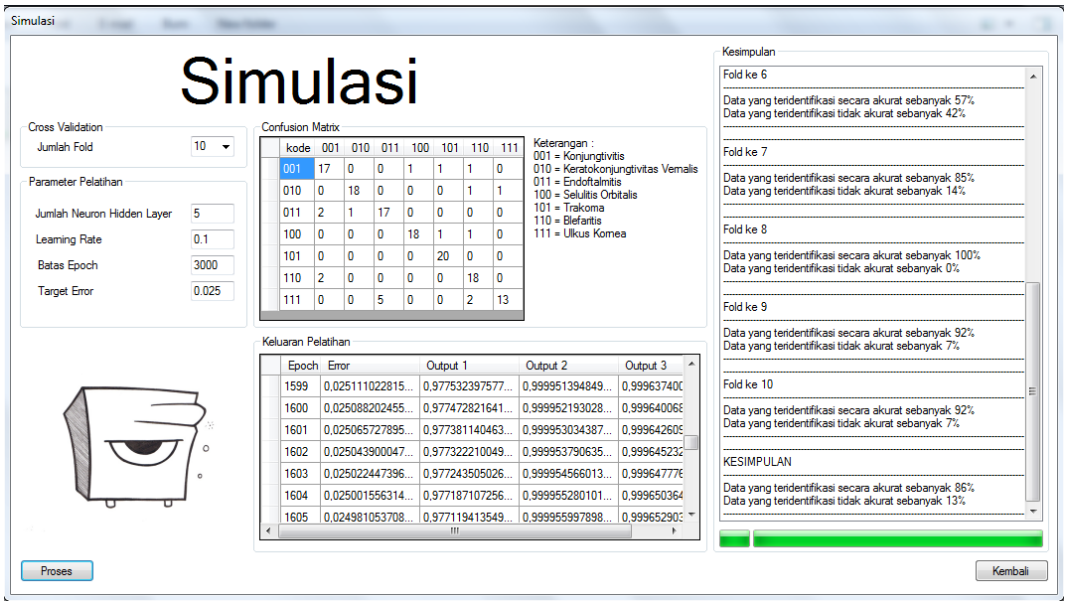
Pada pengujian dengan menggunakan 4 *neuron* pada *hidden layer* menjadikan laju pembelajaran tergolong cepat, karena pada epoch ke 1041 nilai *error* sudah mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 89.286%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.9.

Tabel 4.9 Hasil pengujian hidden layer dengan 2 neuron

Fold	**Akurasi**
1	100%
2	100%
3	92%
4	92%
5	78%
6	64%
7	92%
8	92%
9	92%
10	85%
Rata-rata akurasi	**89.286%**

5. *Hidden Layer* dengan 5 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 5, hasil dari pengujian ini dapat dilihat pada gambar 4.5.



Gambar 4.5 Pengujian Backpropagation dengan 5 neuron pada hidden layer

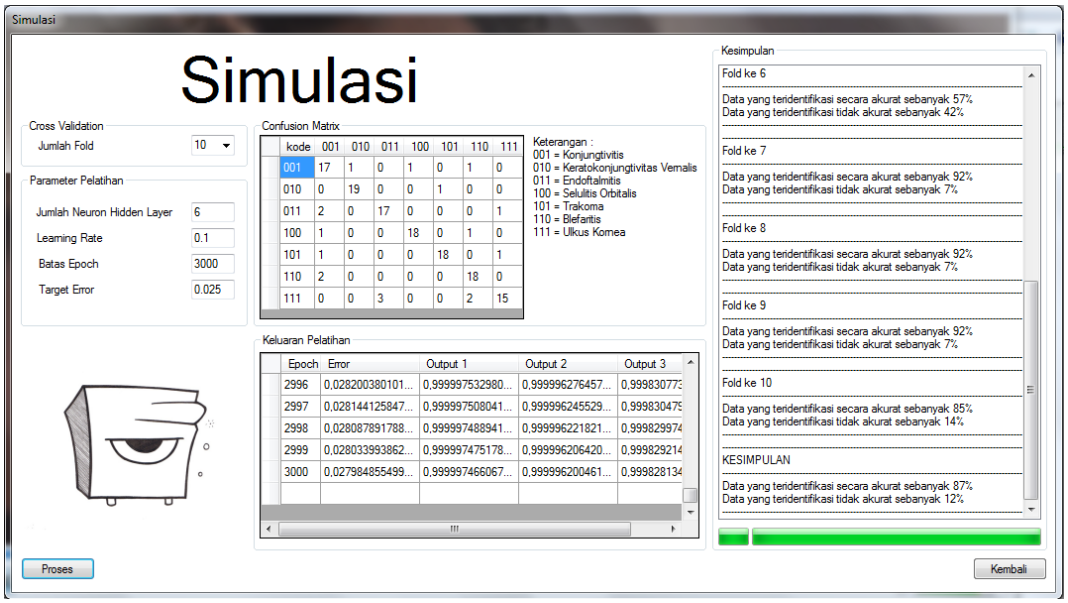
Pada pengujian dengan menggunakan 5 *neuron* pada *hidden layer* laju pembelajaran tergolong cepat, namun terjadi penurunan laju pembelajaran dari pengujian sebelumnya karena nilai *error* baru mencapai target *error* pada epoch ke 1605. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 86.428%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.10.

Tabel 4.10 Hasil pengujian hidden layer dengan 5 neuron

Fold	**Akurasi**
1	92%
2	78%
3	92%
4	85%
5	85%
6	57%
7	85%
8	100%
9	92%
10	92%
Rata-rata akurasi	**86.428%**

6. *Hidden Layer* dengan 6 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 6, hasil dari pengujian ini dapat dilihat pada gambar 4.6.



Gambar 4.6 Pengujian Backpropagation dengan 6 neuron pada hidden layer

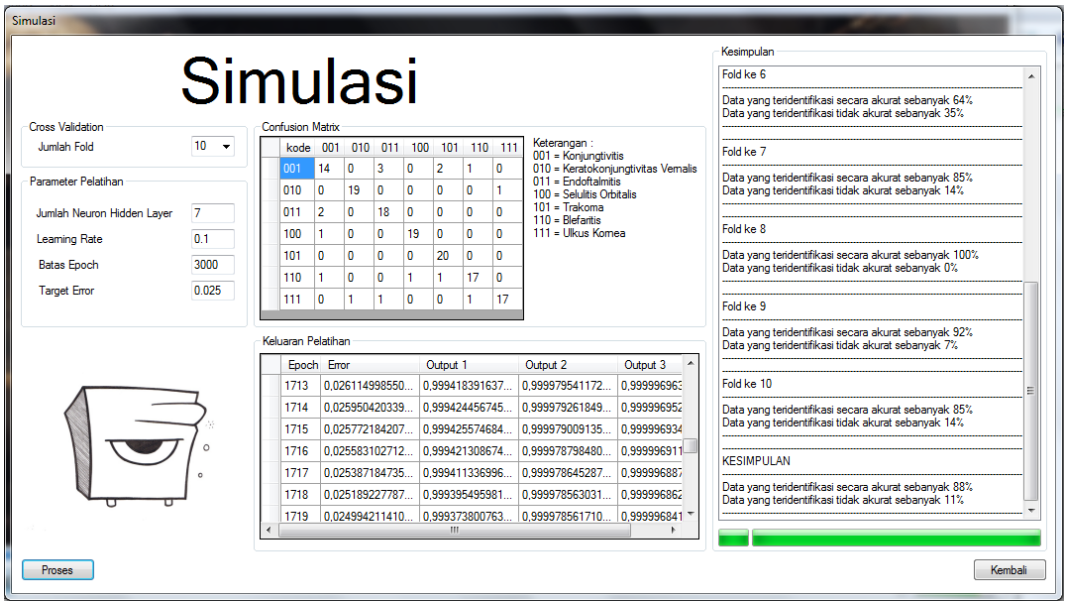
Pada pengujian dengan menggunakan 6 *neuron* pada *hidden layer* terjadi penurunan laju pembelajaran, karena pada epoch ke 3000 nilai *error* belum mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 87.143%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.11.

Tabel 4.11 Hasil pengujian hidden layer dengan 6 neuron

Fold	**Akurasi**
1	92%
2	100%
3	85%
4	92%
5	78%
6	57%
7	92%
8	92%
9	92%
10	85%
Rata-rata akurasi	**87.143%**

7. *Hidden Layer* dengan 7 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 7, hasil dari pengujian ini dapat dilihat pada gambar 4.7.



Gambar 4.7 Pengujian Backpropagation dengan 7 neuron pada hidden layer

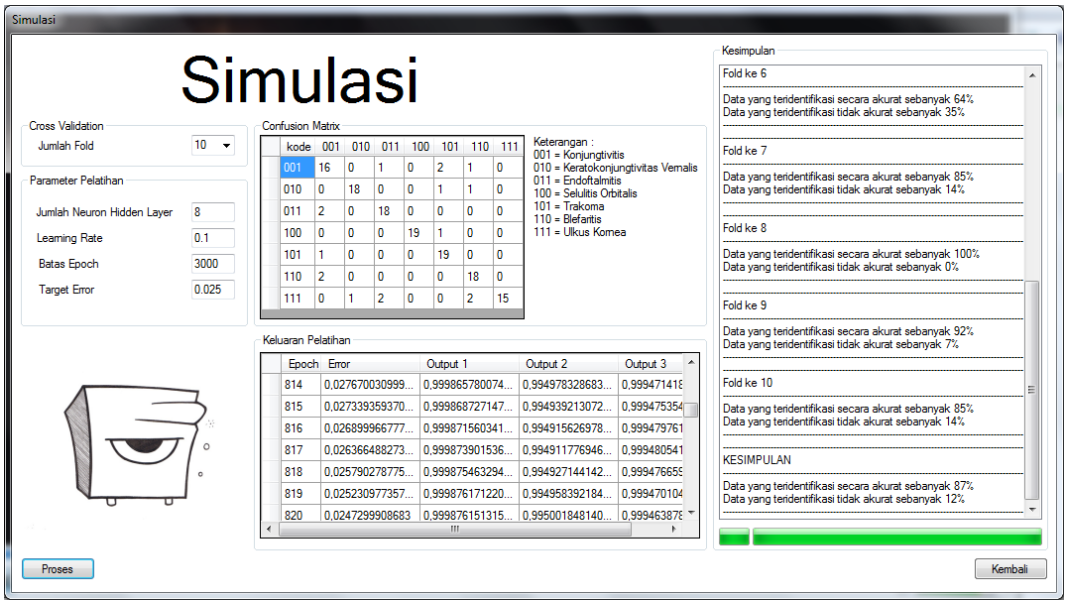
Pada pengujian dengan menggunakan 7 *neuron* pada *hidden layer* laju pembelajaran tergolong cepat, karena pada epoch ke 1719 nilai *error* sudah mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 88.571%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.12.

Tabel 4.12 Hasil pengujian hidden layer dengan 7 neuron

Fold	**Akurasi**
1	100%
2	100%
3	85%
4	85%
5	85%
6	64%
7	85%
8	100%
9	92%
10	85%
Rata-rata akurasi	**88.571%**

8. *Hidden Layer* dengan 8 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 8, hasil dari pengujian ini dapat dilihat pada gambar 4.8.



Gambar 4.8 Pengujian Backpropagation dengan 8 neuron pada hidden layer

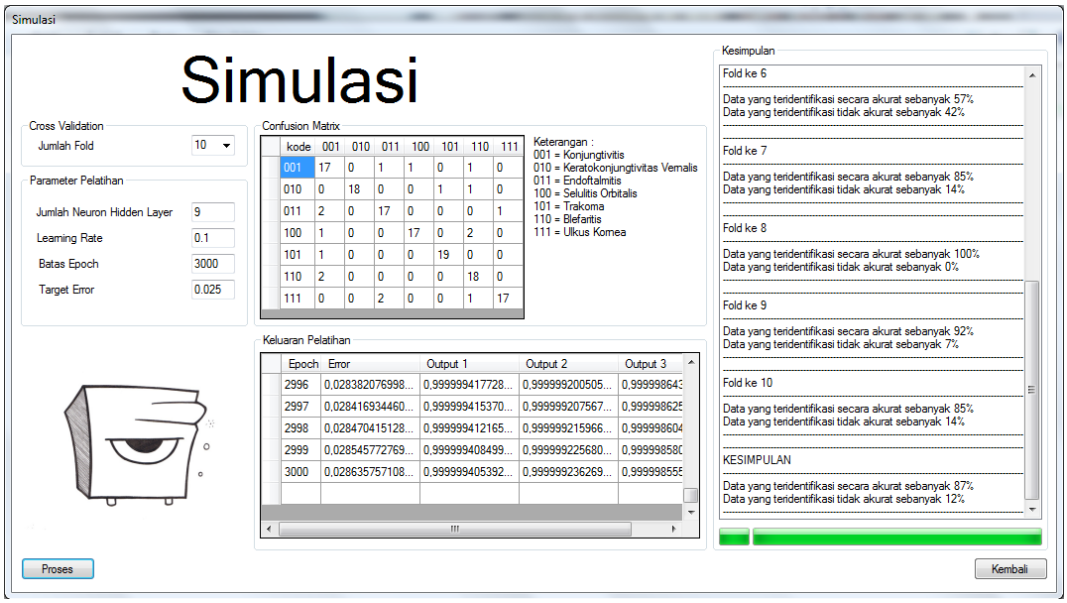
Pada pengujian dengan menggunakan 8 *neuron* pada *hidden layer* laju pembelajaran tergolong cepat, karena pada epoch ke 820 nilai *error* sudah mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 87.857%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.13.

Tabel 4.13 Hasil pengujian hidden layer dengan 8 neuron

Fold	**Akurasi**
1	100%
2	92%
3	85%
4	92%
5	78%
6	64%
7	85%
8	100%
9	92%
10	85%
Rata-rata akurasi	**87.857%**

9. *Hidden Layer* dengan 9 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 9, hasil dari pengujian ini dapat dilihat pada gambar 4.9.



Gambar 4.9 Pengujian Backpropagation dengan 9 neuron pada hidden layer

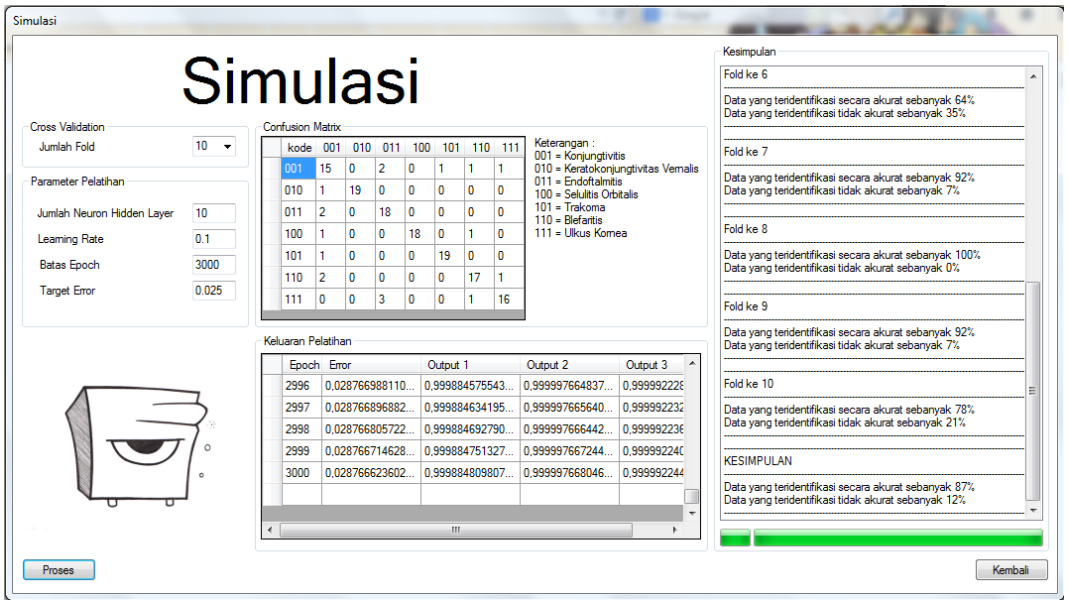
Pada pengujian dengan menggunakan 9 *neuron* pada *hidden layer* laju pembelajaran tergolong lambat, namun pada epoch ke 3000 nilai *error* hampir mendekati nilai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 87.857%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.14.

Tabel 4.14 Hasil pengujian hidden layer dengan 9 neuron

Fold	**Akurasi**
1	100%
2	100%
3	78%
4	92%
5	85%
6	57%
7	85%
8	100%
9	92%
10	85%
Rata-rata akurasi	**87.857%**

10. *Hidden Layer* dengan 10 *Neuron*

Pengujian selanjutnya jumlah *neuron* pada *hidden layer* diubah menjadi 10, hasil dari pengujian ini dapat dilihat pada gambar 4.10.



Gambar 4.10 Pengujian Backpropagation dengan 10 neuron pada hidden layer

Pada pengujian dengan menggunakan 10 *neuron* pada *hidden layer* laju pembelajaran tergolong lambat, namun pada epoch ke 3000 nilai *error* hampir mendekati nilai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 87.143%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.15.

Tabel 4.15 Hasil pengujian hidden layer dengan 10 neuron

Fold	**Akurasi**
1	85%
2	100%
3	78%
4	92%
5	85%
6	64%
7	92%
8	100%
9	92%
10	78%
Rata-rata akurasi	**87.143%**

Dari skenario pengujian 1 didapatkan hasil secara keseluruhan yang dapat dilihat pada tabel 4.16.

Tabel 4.16 Data Hasil Skenario Pengujian 1

Jumlah *neuron*	Akurasi	Epoch	Nilai Error yang dicapai
1	25.714%	3000	0.520120559679
2	75%	3000	0.209497369169
3	85%	3000	0.034558977559
4	89.286%	1041	0.024996271336
5	86.428%	1605	0.024981053708
6	87.143%	3000	0.027984855499
7	88.571%	1719	0.024994211410
8	87.857%	820	0.0247299908683
9	87.857%	3000	0.028635757108
10	87.143%	3000	0.028766623602

Dari data hasil skenario pengujian 1 ini dapat diambil kesimpulan bahwa jumlah *neuron* yang memiliki akurasi paling tinggi pada kasus ini adalah 4 *neuron* dengan akurasi sebesar 89.286%, meskipun terdapat jumlah *neuron* yang memiliki laju pembelajaran lebih cepat dari 4 *neuron*, dengan menggunakan 4 *neuron* pada *hidden layer* laju pembelajaran masih terbilang cepat karena nilai *error* telah mencapai nilai batas *error* sebelum epoch mencapai batas epoch.

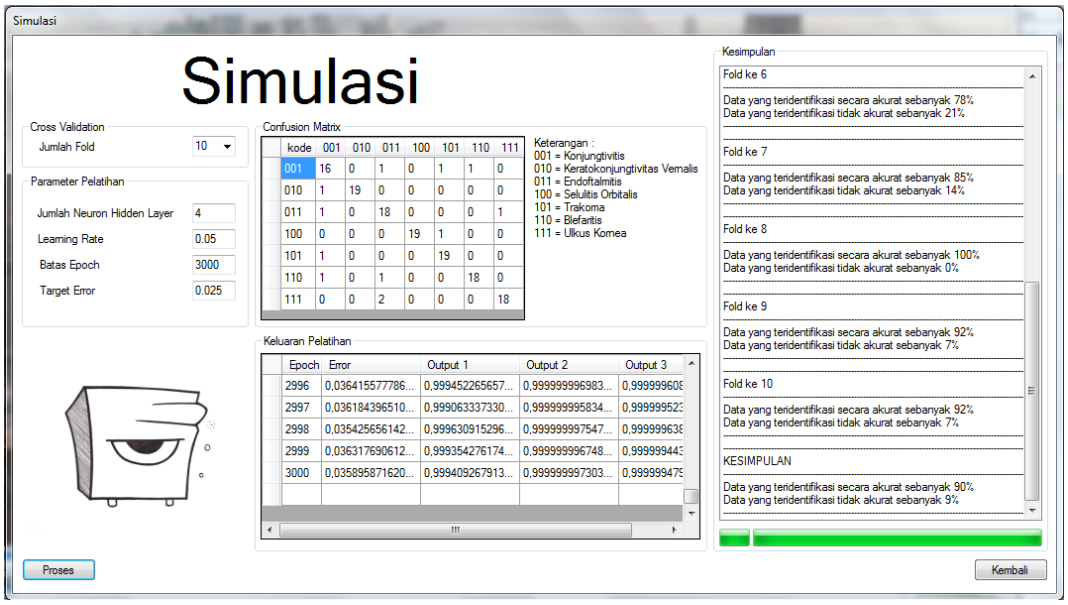
4.2.2 Skenario Pengujian 2

Pada pengujian skenario 2, pengujian metode *backpropagation* dilakukan dengan mengubah nilai jumlah *learning rate* dari nilai 0.05, 0.1, 0.15, 0.2, 0.25, dan 0.03, sedangkan ketiga variabel lainnya diisi dengan nilai yang sama. Berikut ini adalah nilai variabel yang digunakan pada skenario pengujian 2.

Jumlah *neuron* pada *hidden layer*	=	4
Learning Rate (α)	=	0.05, 0.1, 0.15, 0.2, 0.25, 0.3
Batas Epoch	=	3000
Target Error	=	0.025

1. *Learning rate* 0.05

Pengujian skenario 2 dimulai dengan *learning rate* yang bernilai 0.05, hasil dari pengujian ini dapat dilihat pada gambar 4.11.



Gambar 4.11 Pengujian Backpropagation dengan learning rate bernilai 0.05

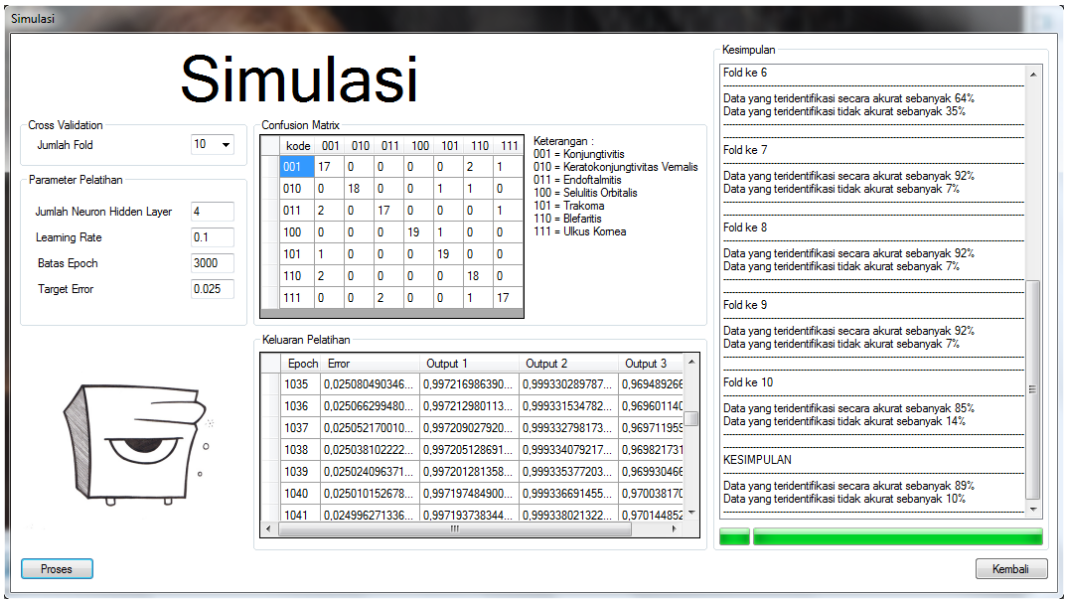
Pada pengujian dengan menggunakan *learning rate* yang bernilai 0.05 laju pembelajaran tergolong lambat, namun pada epoch ke 3000 nilai *error* hampir mendekati nilai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 90.714%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.17.

Tabel 4.17 Hasil pengujian learning rate bernilai 0.05

Fold	**Akurasi**
1	92%
2	100%
3	92%
4	92%
5	78%
6	78%
7	85%
8	100%
9	92%
10	92%
Rata-rata akurasi	**90.714%**

2. *Learning rate* 0.1

Pengujian selanjutnya nilai *learning rate* diubah menjadi 0.1, hasil dari pengujian ini dapat dilihat pada gambar 4.12.



Gambar 4.12 Pengujian Backpropagation dengan learning rate bernilai 0.1

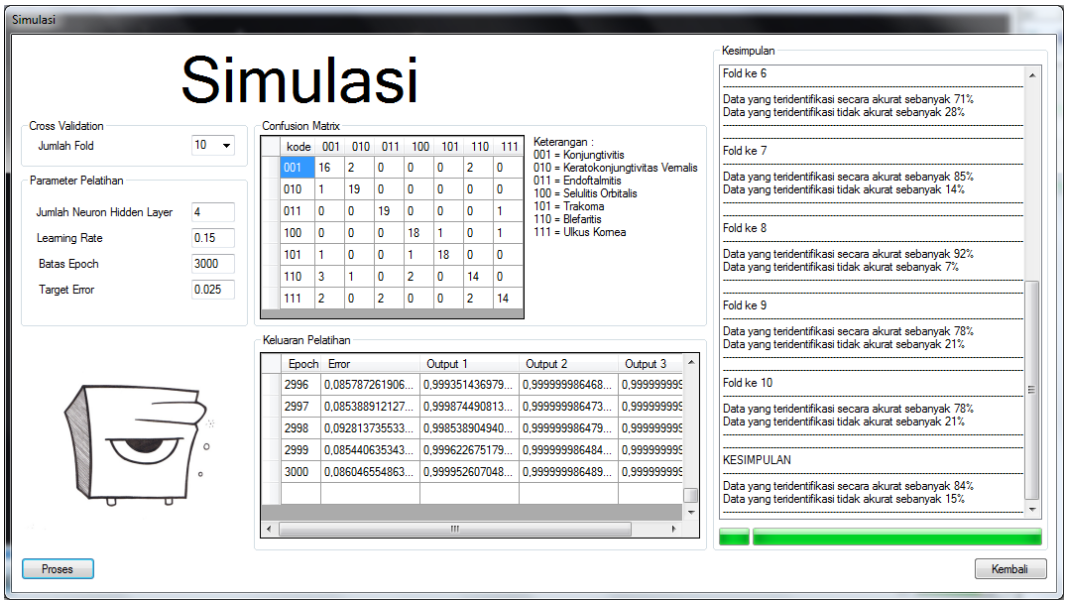
Pada pengujian dengan menggunakan *learning rate* yang bernilai 0.1 menjadikan laju pembelajaran tergolong cepat, karena pada epoch ke 1041 nilai *error* sudah mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 89.286%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.18.

Tabel 4.18 Hasil pengujian learning rate bernilai 0.1

Fold	**Akurasi**
1	100%
2	100%
3	92%
4	92%
5	78%
6	64%
7	92%
8	92%
9	92%
10	85%
Rata-rata akurasi	**89.286%**

3. *Learning rate* 0.15

Pengujian selanjutnya nilai *learning rate* diubah menjadi 0.15, hasil dari pengujian ini dapat dilihat pada gambar 4.13.



Gambar 4.13 Pengujian Backpropagation dengan learning rate bernilai 0.15

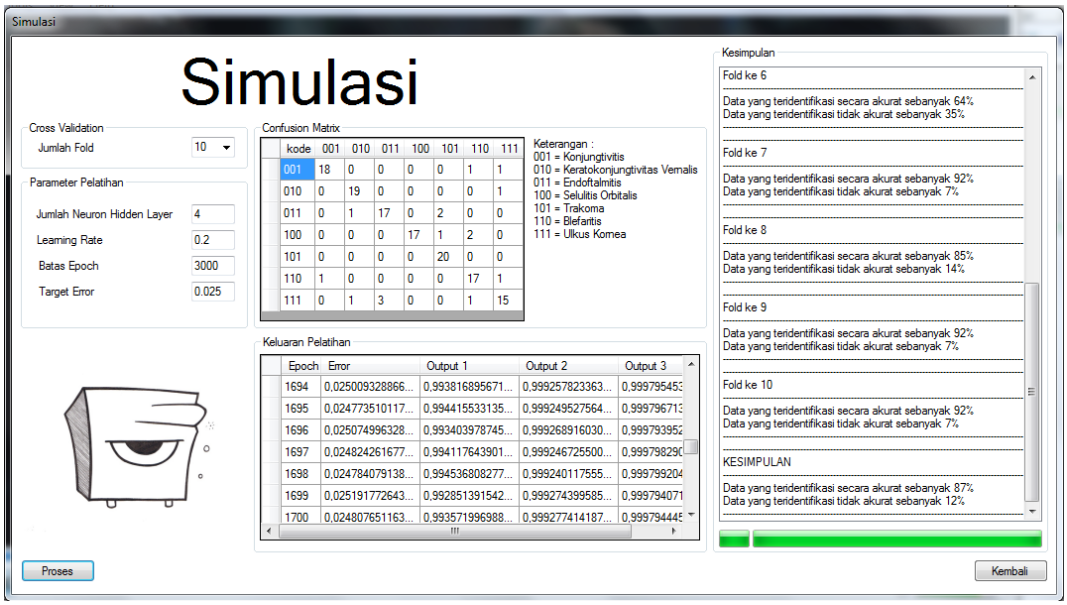
Pada pengujian dengan menggunakan *learning rate* yang bernilai 0.15 menjadikan laju pembelajaran tergolong lambat, karena pada epoch ke 3000 nilai *error* belum mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 84.286%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.19.

Tabel 4.19 Hasil pengujian learning rate bernilai 0.15

Fold	**Akurasi**
1	92%
2	92%
3	78%
4	85%
5	85%
6	71%
7	85%
8	92%
9	78%
10	78%
Rata-rata akurasi	**84.286%**

4. *Learning rate* 0.2

Pengujian selanjutnya nilai *learning rate* diubah menjadi 0.2, hasil dari pengujian ini dapat dilihat pada gambar 4.14.



Gambar 4.14 Pengujian Backpropagation dengan learning rate bernilai 0.2

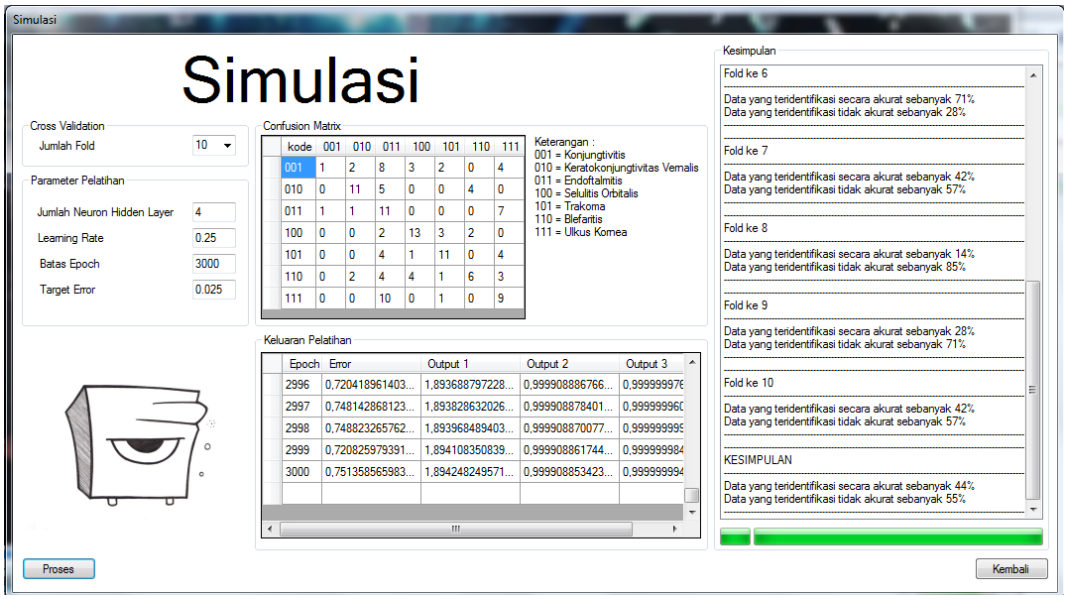
Pada pengujian dengan menggunakan *learning rate* yang bernilai 0.2 menjadikan laju pembelajaran tergolong cepat, karena pada epoch ke 1700 nilai *error* sudah mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 87.857%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.20.

Tabel 4.20 Hasil pengujian learning rate bernilai 0.2

Fold	**Akurasi**
1	100%
2	100%
3	71%
4	92%
5	85%
6	64%
7	92%
8	85%
9	92%
10	92%
Rata-rata akurasi	**87.857%**

5. *Learning rate* 0.25

Pengujian selanjutnya nilai *learning rate* diubah menjadi 0.25, hasil dari pengujian ini dapat dilihat pada gambar 4.15.



Gambar 4.15 Pengujian Backpropagation dengan learning rate bernilai 0.25

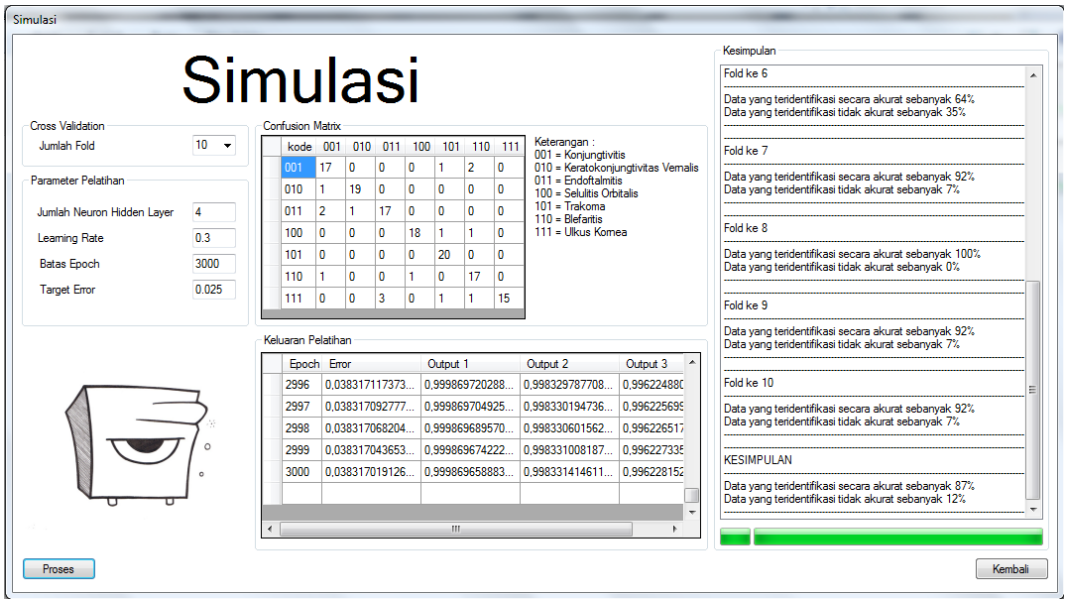
Pada pengujian dengan menggunakan *learning rate* yang bernilai 0.25 menjadikan laju pembelajaran tergolong lambat, karena pada epoch ke 3000 nilai *error* belum mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 44.286%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.21.

Tabel 4.21 Hasil pengujian learning rate bernilai 0.25

Fold	**Akurasi**
1	50%
2	21%
3	57%
4	57%
5	57%
6	71%
7	42%
8	14%
9	28%
10	42%
Rata-rata akurasi	**44.286%**

6. *Learning rate* 0.3

Pengujian selanjutnya nilai *learning rate* diubah menjadi 0.3, hasil dari pengujian ini dapat dilihat pada gambar 4.16.



Gambar 4.16 Pengujian Backpropagation dengan learning rate bernilai 0.3

Pada pengujian dengan menggunakan *learning rate* yang bernilai 0.3 menjadikan laju pembelajaran tergolong lambat, karena pada epoch ke 3000 nilai *error* belum mencapai target *error*. Rata-rata Akurasi yang didapat dari pengujian ini sebesar 87.857%. Hasil akurasi yang didapat pada setiap *fold* dapat dilihat pada tabel 4.22.

Tabel 4.22 Hasil pengujian learning rate bernilai 0.25

Fold	**Akurasi**
1	100%
2	85%
3	85%
4	92%
5	71%
6	64%
7	92%
8	100%
9	92%
10	92%
Rata-rata akurasi	**87.857%**

Dari skenario pengujian 2 didapatkan hasil secara keseluruhan yang dapat dilihat pada tabel 4.23.

Tabel 4.23 Data Hasil Skenario Pengujian 2

Learning Rate	Akurasi	Epoch	Nilai Error yang dicapai
0.05	90.714%	3000	0.035895871620
0.1	89.286%	1041	0.024996271336
0.15	84.286%	3000	0,086046554863
0.2	87.857%	1700	0.024807651163
0.25	44.286%	3000	0.751358565983
0.3	87.857%	3000	0.038317019126

Dari data hasil skenario pengujian 2 ini dapat diambil kesimpulan bahwa nilai *learning rate* yang memiliki akurasi paling tinggi pada kasus ini adalah *learning rate* yang bernilai 0.05, akan tetapi laju pembelajarannya terbilang lambat yang disebabkan kecilnya nilai dari *learning rate.*

4.3 Kesimpulan Pengujian

Berdasarkan hasil skenario pengujian 1 dan skenario pengujian 2 dapat ditarik kesimpulan bahwa metode *backpropagation* dapat mengidentifikasi penyakit mata menular dengan akurasi terbesar sebesar 90.714%. Akurasi ini didapatkan dari susunan variabel yang memiliki jumlah *neuron* pada *hidden layer* sebanyak 4 *neuron, learning rate* yang bernilai 0.05, batas epoch sebesar 3000, dan target *error* sebesar 0.025.