

BAB 4.
IMPLEMENTASI DATA SPEKTROFOTOMETER DAN ANALISA
DENGAN *SELF ORGANIZING MAPS* DAN ALGORITMA PCA

4.1 Deteksi Dengue dengan Dengue Duo (NS1 dan IgG)

Data yang diperoleh berawal dari pengklasifikasian sampel menggunakan Dengue Duo (NS1 dan IgG) dengan hasil seperti pada Tabel 4.1.

Tabel 4.1 Jumlah Sampel yang Diperoleh

Status	Jumlah Sampel
Dengue	15
Non Dengue	15
Normal	7

4.2 Pengukuran Absorbansi

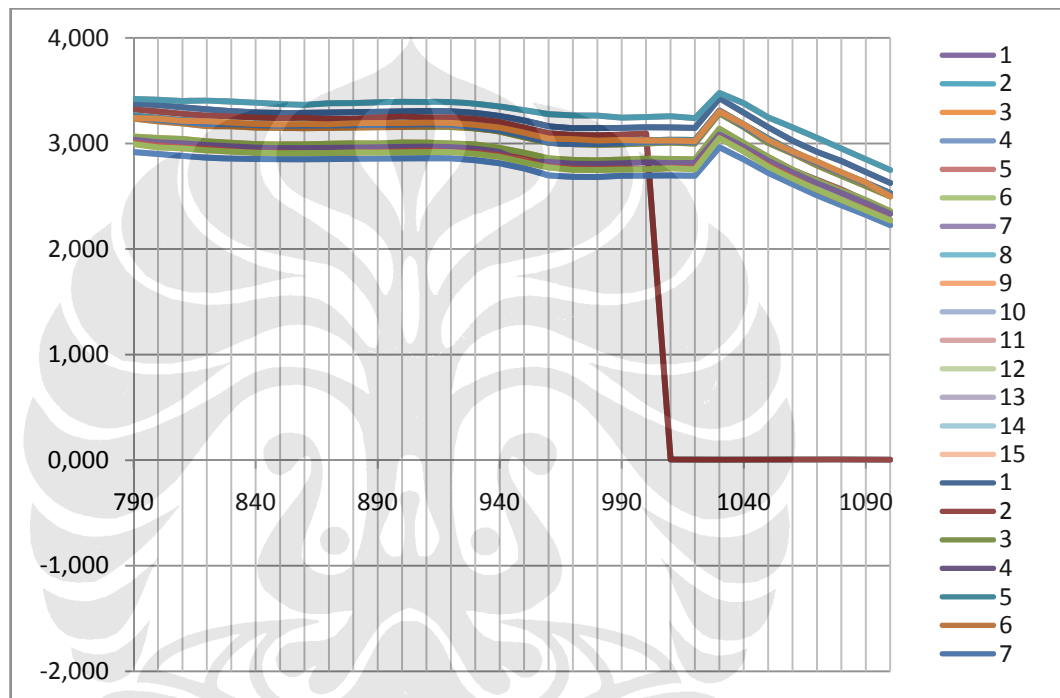
Kemudian dilakukan pengambilan data absorbansi dengan rentang pengukuran 190 s/d 1100 nm per setiap 10 nm dalam bentuk print out untuk 3 kelompok sampel, antara lain Dengue Positif, Negatif dan Normal. Hasil yang diperoleh dari pengukuran rentang 190 s/d 1100 nm di tabulasikan kedalam spreadsheet lalu di kelompokkan berdasarkan penggunaan lampu dan pergantian filter untuk mengeliminasi pergeseran absorbansi pada panjang gelombang yang diakibatkan pergantian filter dan lampu. Rentang panjang gelombang diklasifikasikan dalam Tabel 4.2

Tabel 4.2 Rentang Klasifikasi Panjang Gelombang Berdasarkan Pergantian Filter dan Lampu

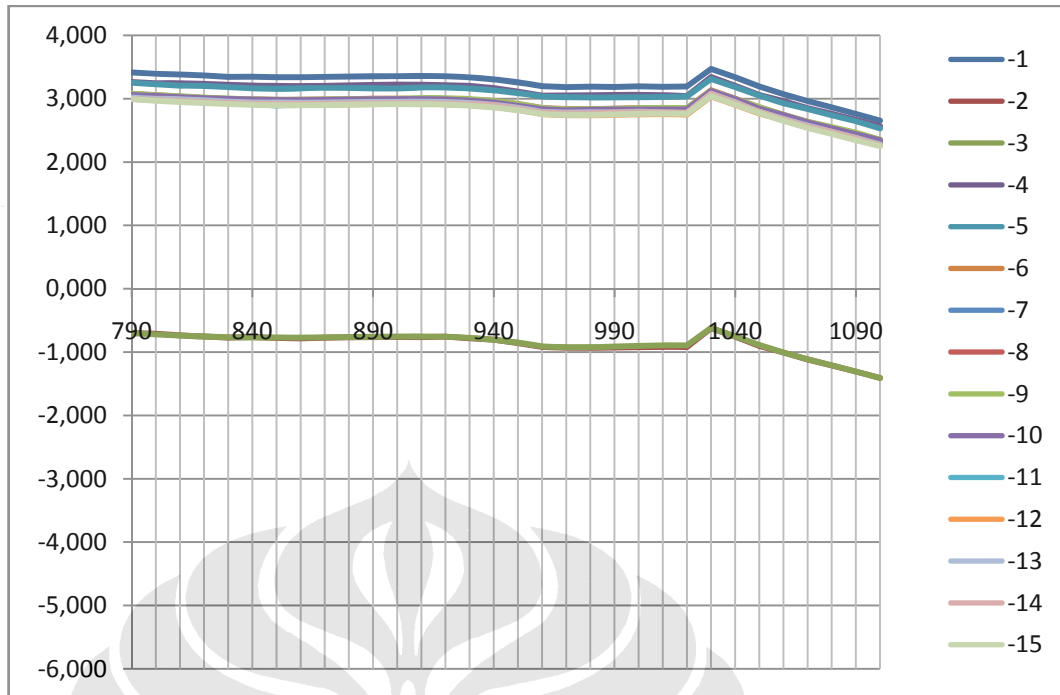
λ	Rentang Klasifikasi	Kelompok	n
780 nm	1100 s/d 790 nm	Near Infra Red	32
600 nm	780 s/d 610 nm	Visible	18
390 nm	600 s/d 400 nm	Visible	21
340 nm	390 s/d 350 nm	Near UV	5
	340 s/d 190 nm	Near UV	16

Pada Gambar 4.1 s/d Gambar 4.15, memperlihatkan spektrum berdasarkan rentang klasifikasi pada Tabel 4.2. Pada kelompok rentang klasifikasi 600 s/d 400 nm ditemukan karakteristik yang unik, dan diperkirakan merupakan data yang signifikan untuk mewakili karakteristik kelompok sampel kemudian diolah datanya seperti pada Gambar 4.16 untuk dimasukkan kedalam neural networks.

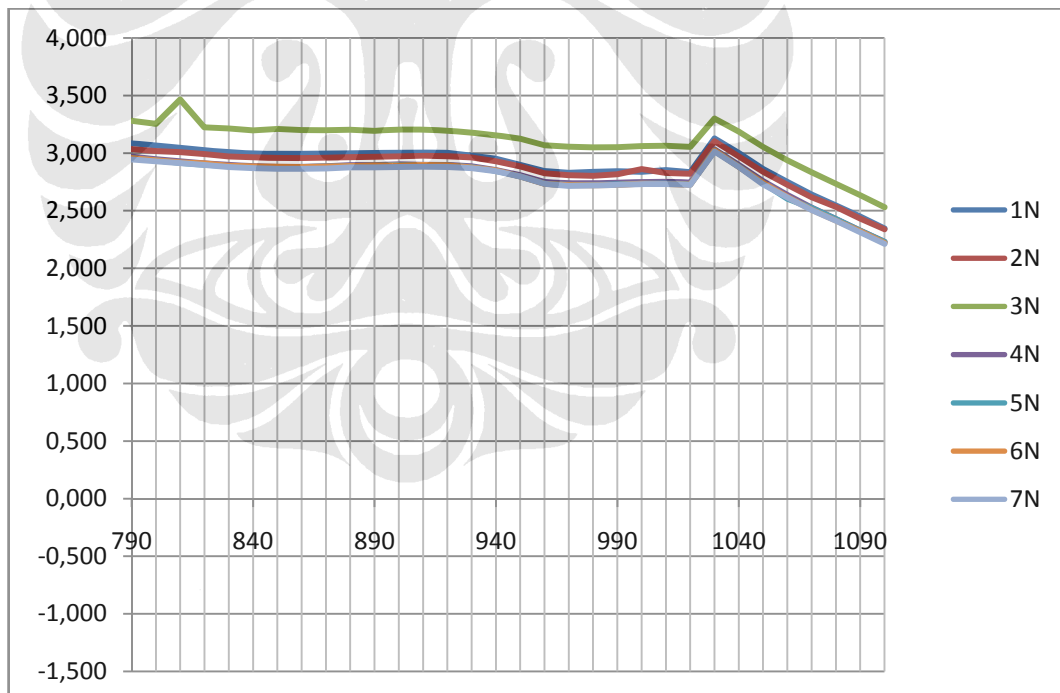
4.2.1 Absorbansi 1100 s/d 790 nm (Near Infra Red)



Gambar 4.1 Absorbansi Dengue 1100 s/d 790 nm

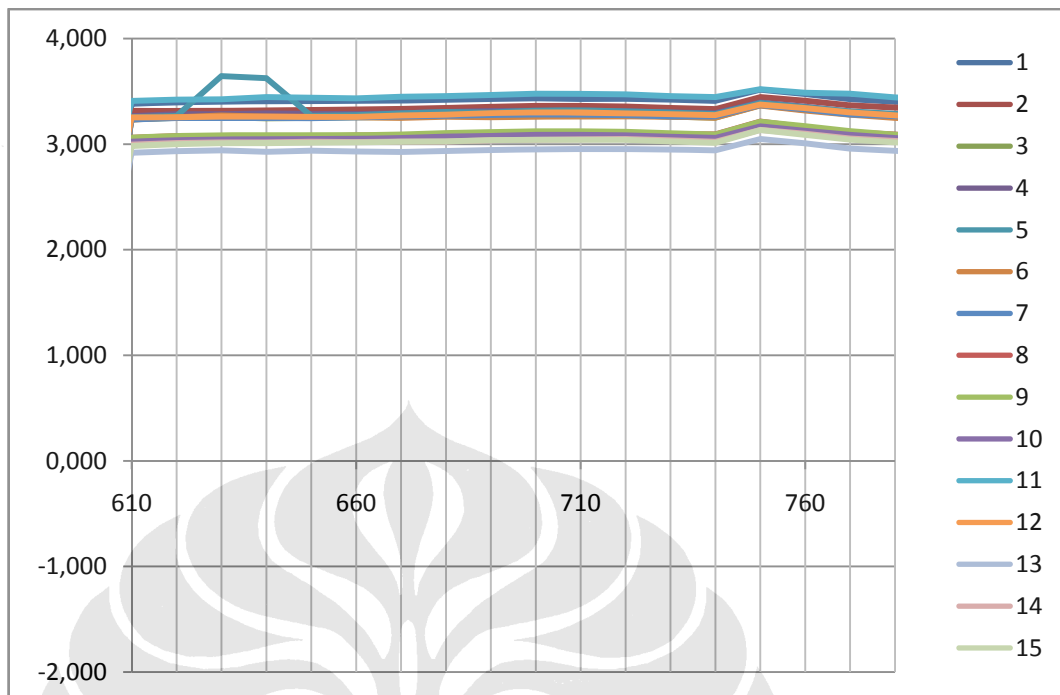


Gambar 4.2 Absorbansi Non Dengue 1100 s/d 790 nm

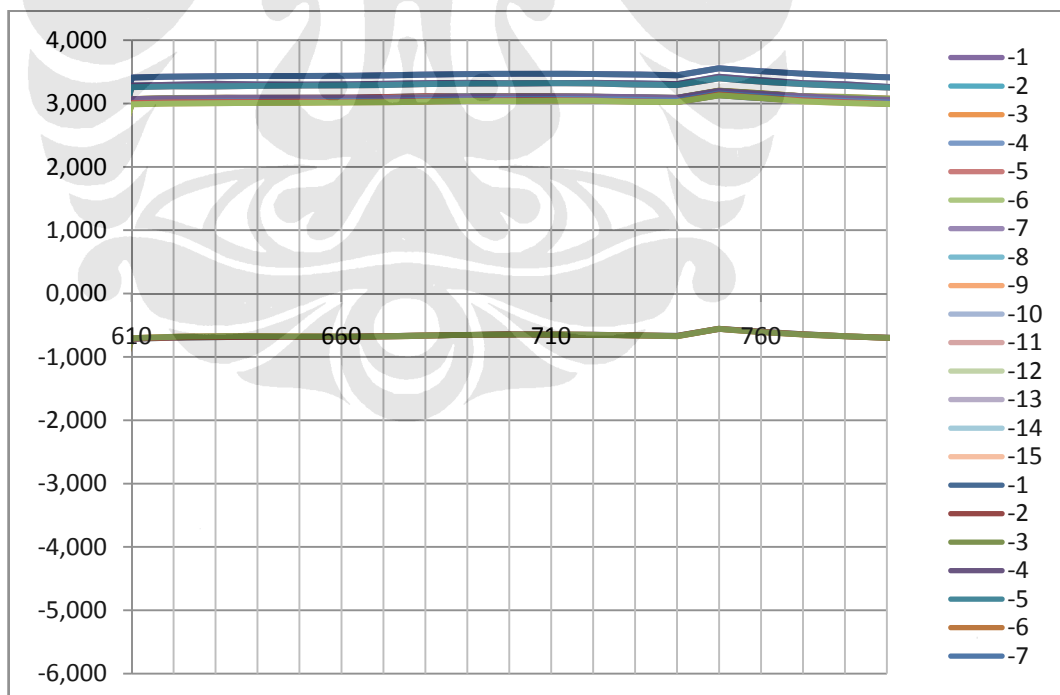


Gambar 4.3 Absorbansi Normal 1100 s/d 790 nm

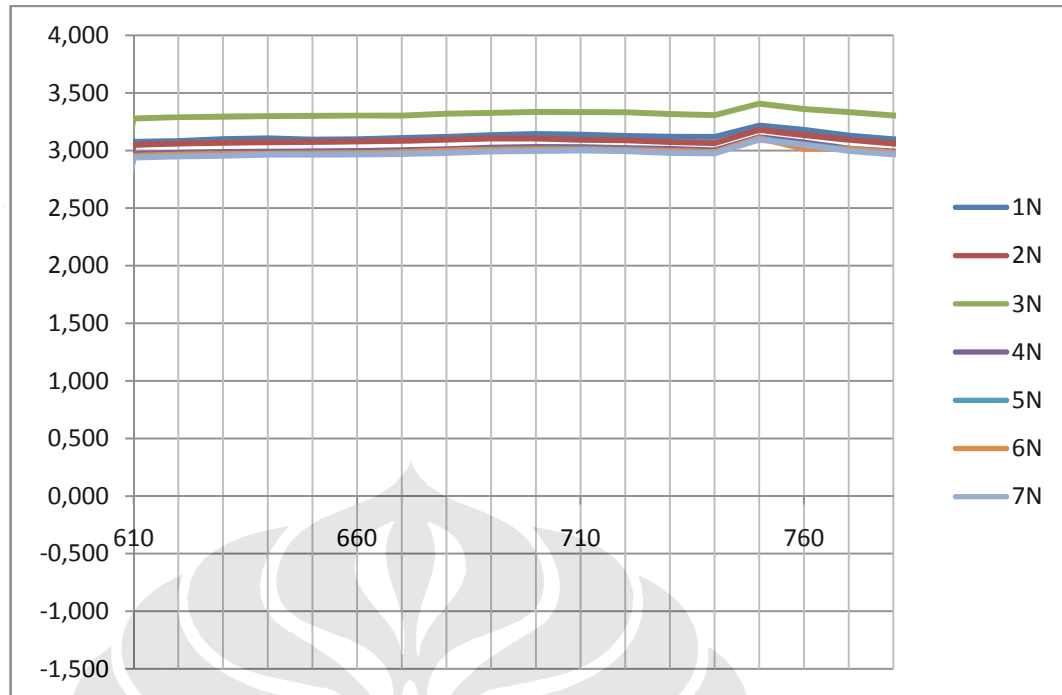
4.2.2 Absorbansi 780 s/d 610 nm (Visible)



Gambar 4.4 Absorbansi Dengue 780 s/d 610 nm

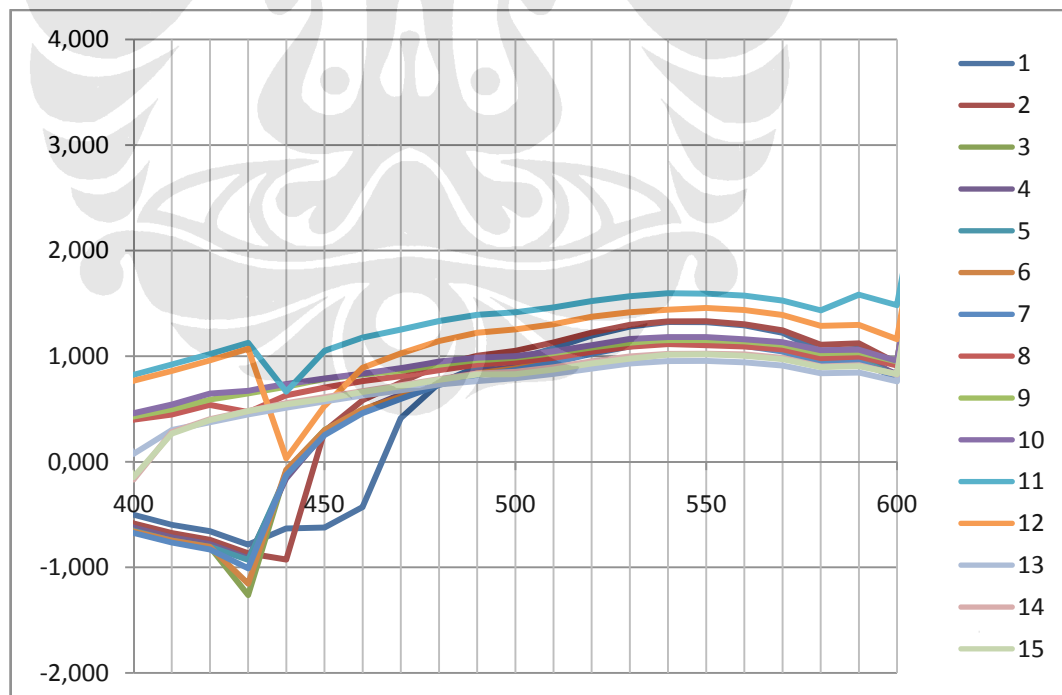


Gambar 4.5 Absorbansi Non Dengue 780 s/d 610 nm

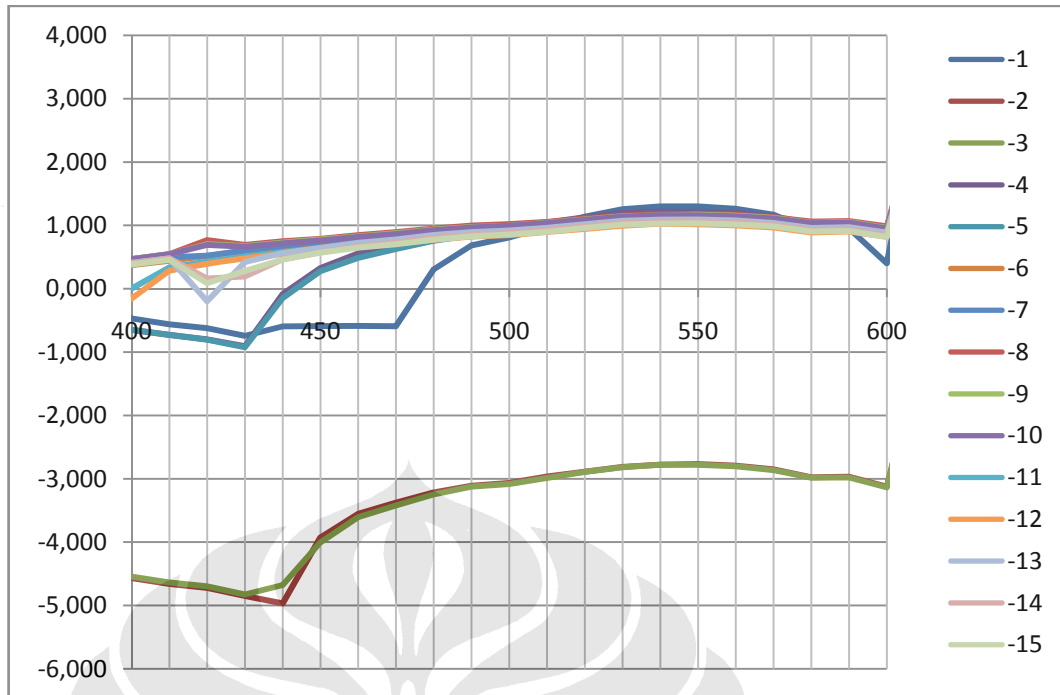


Gambar 4.6 Absorbansi Normal 780 s/d 610 nm

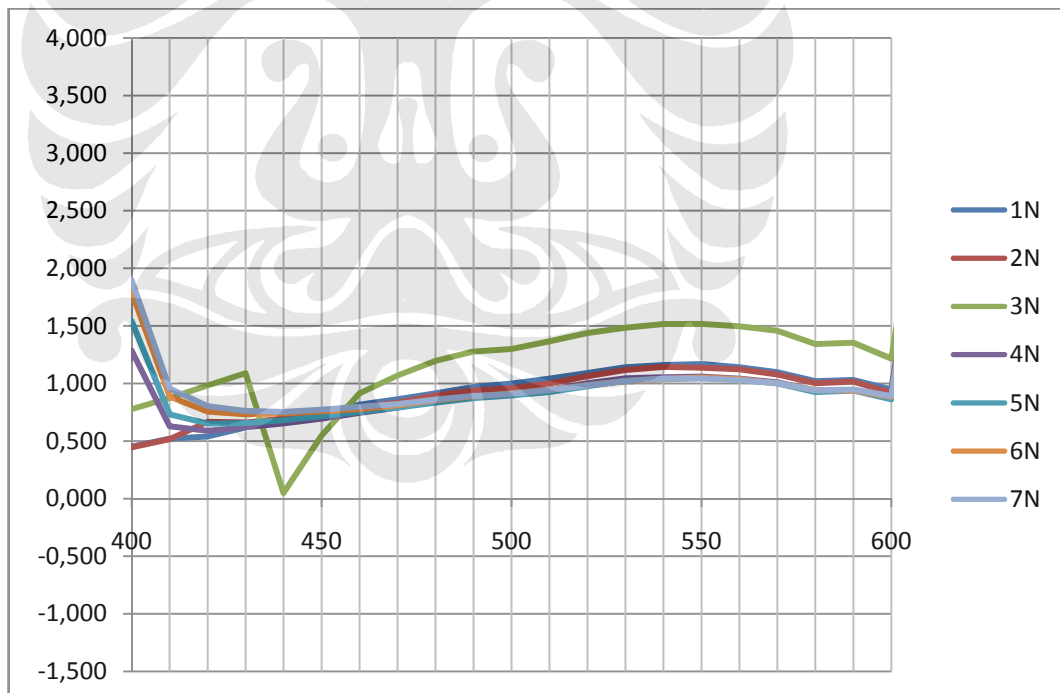
4.2.3 Absorbansi 600 s/d 400 nm (Visible)



Gambar 4.7 Absorbansi Dengue 400 s/d 600 nm

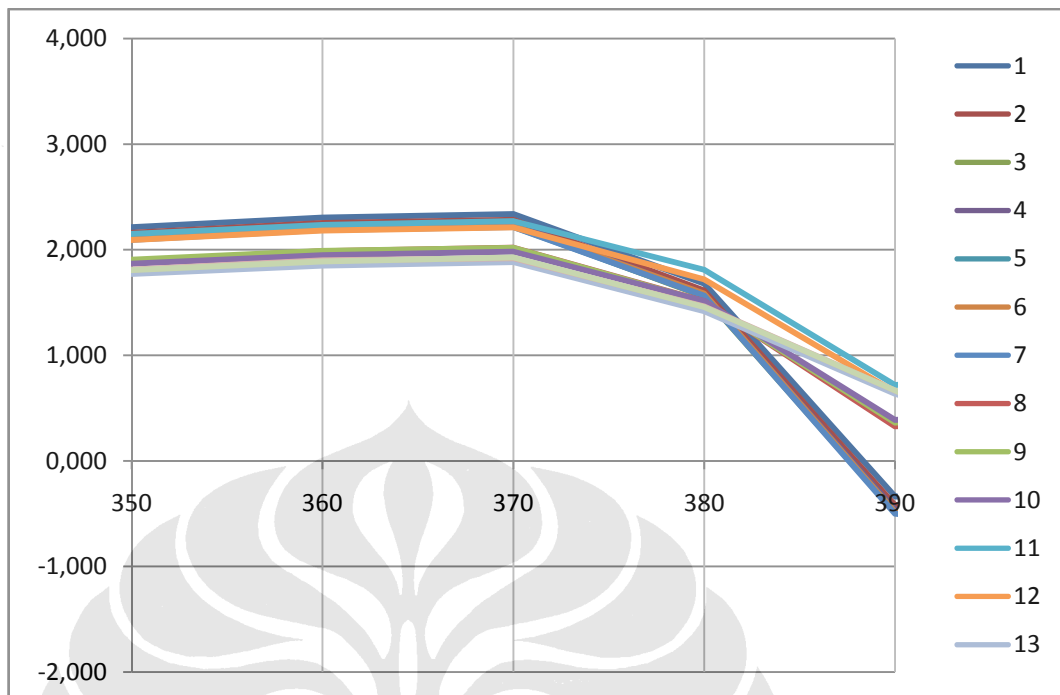


Gambar 4.8 Absorbansi Non Dengue 400 s/d 600 nm

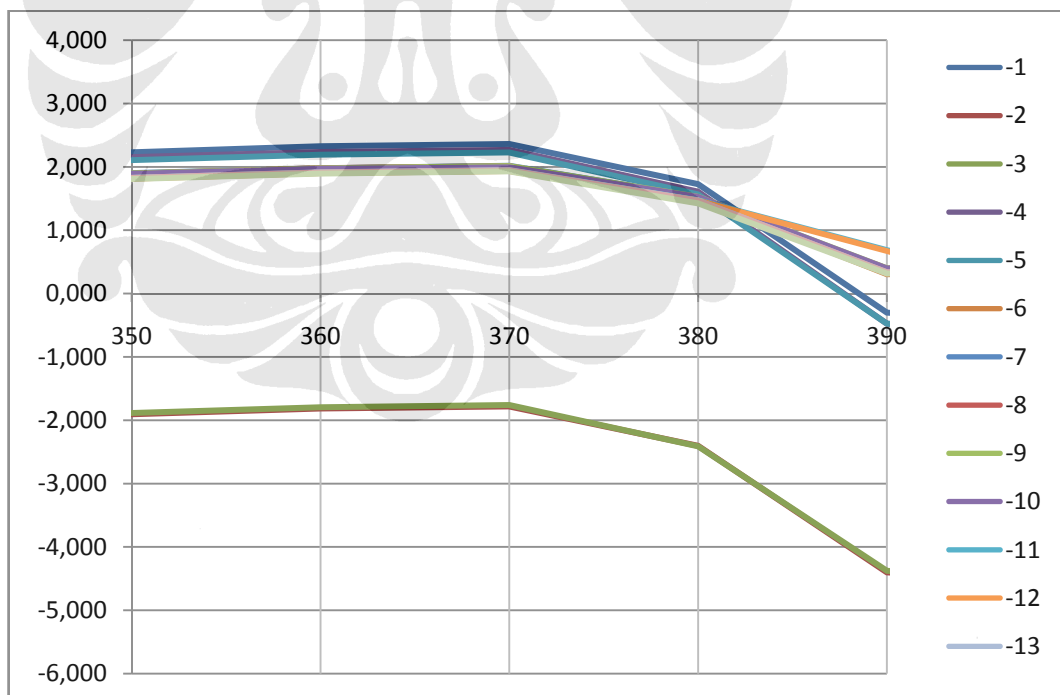


Gambar 4.9 Absorbansi Normal 400 s/d 600 nm

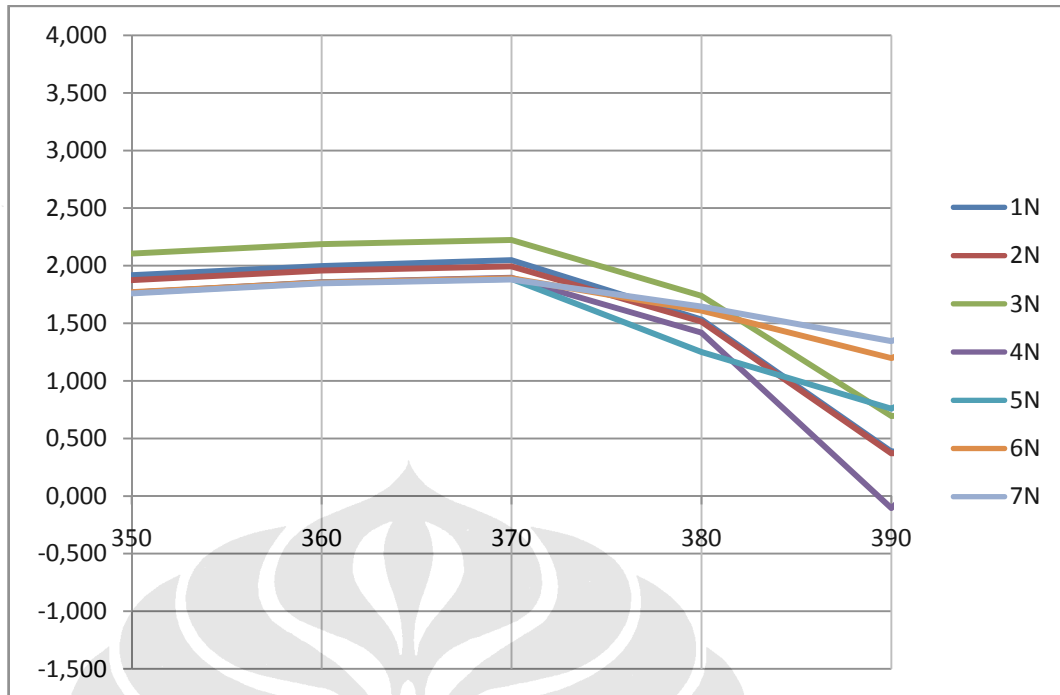
4.2.4 Absorbansi 390 s/d 350 nm (Near UV)



Gambar 4.10 Absorbansi Dengue 350 s/d 390 nm

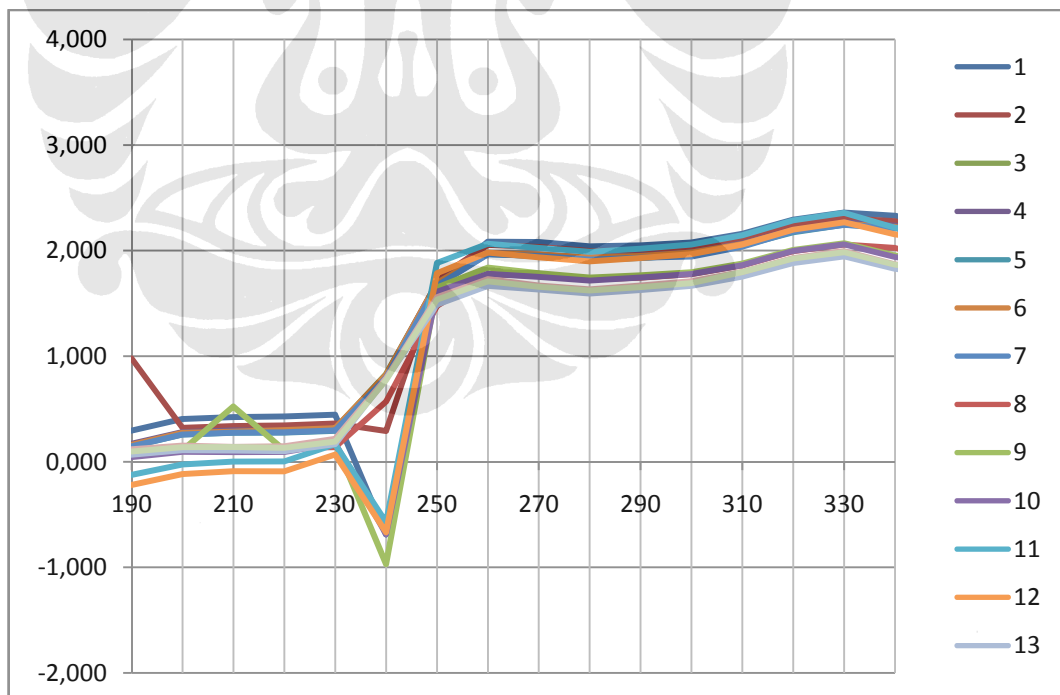


Gambar 4.11 Absorbansi Non Dengue 350 s/d 390 nm

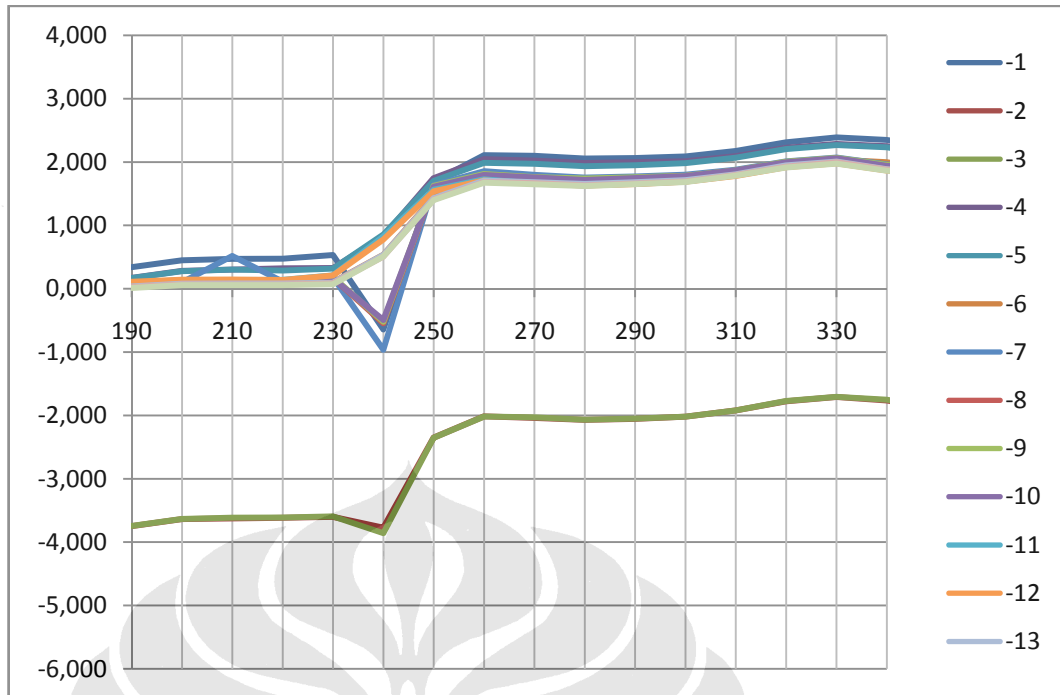


Gambar 4.12 Absorbansi Normal 350 s/d 390 nm

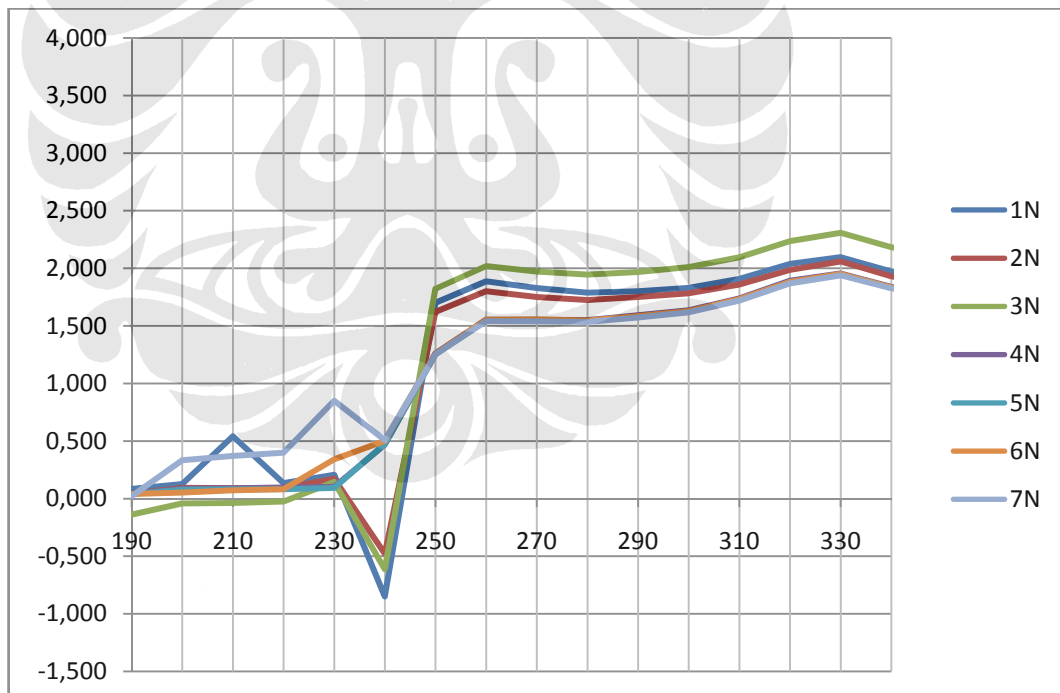
4.2.5 Absorbansi 340 s/d 190 nm (Near UV)



Gambar 4.13 Absorbansi Dengue 190 s/d 340 nm



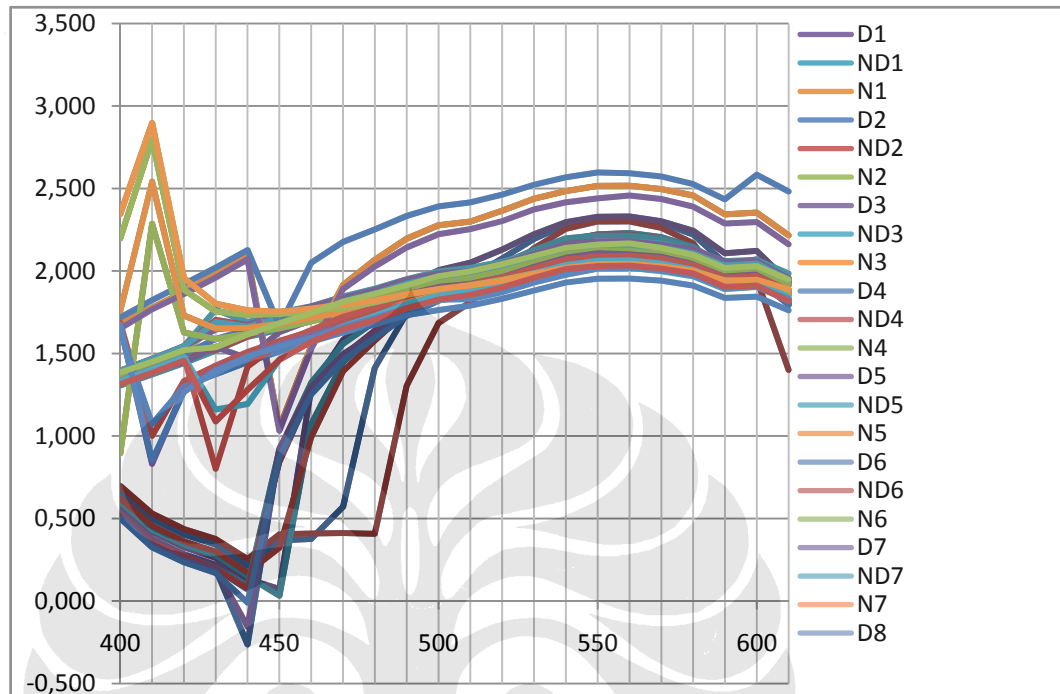
Gambar 4.14 Absorbansi Non Dengue 190 s/d 340 nm



Gambar 4.15 Absorbansi Normal 190 s/d 340 nm

4.3 Jaringan Saraf Tiruan

4.3.1 Data input yang digunakan untuk Jaringan Saraf Tiruan



Gambar 4.16 Data yang digunakan

Data yang digunakan untuk diproses pada Jaringan Saraf Tiruan menggunakan data dari panjang gelombang 400 nm s/d 610 nm sebanyak data untuk 45 sampel yang terdiri dari 3 kelas, yaitu Dn sebagai Dengue ke-n, NDn sebagai Non Dengue ke-n, Nn sebagai Normal ke-n. Data tersebut digunakan sebagai input untuk Self Organizing Maps (SOM).

4.4 Self Organizing Maps

4.4.1 Algoritma Self Organizing Maps

Proses pelatihan dijalankan dengan langkah-langkah berikut:

1. Melakukan proses inialisasi untuyk setiap bobot
2. Set parameter laju pembelajaran α (learning rate)

→ parameter ini akan menentukan kecepatan pembelajaran. Semakin kecil α maka semakin lama waktu pelatihan yang dibutuhkan, demikian sebaliknya.

3. Menghitung Euclidian distance

$$D = \sqrt{\sum_{i=1}^n (X_i - W_i)^2} \quad (5.1)$$

dimana X adalah vector pada unit masukan dan W adalah vector bobot

4. Mencari jarak terkecil

→ unit pada lapisan keluaran yang mempunyai jarak terkecil (D_{\min}) disebut *Best Matching unit* (BMU).

$$D_{\min} = \min[D(m)] \quad (5.2)$$

5. Penyesuaian bobot unit

$$W(t + 1) = W(t) + \alpha(X(t) - W(t)) \quad (5.3)$$

t adalah jumlah iterasi yang telah terjadi (*time stamp*), α adalah laju pembelajaran yang akan menjadi semakin kecil untuk setiap iterasi.

6. Penyesuaian laju pembelajaran

Untuk iterasi selanjutnya, laju pembelajaran α akan disesuaikan seperti berikut

$$\alpha_n = \frac{\alpha_{n-1}}{2} \quad (5.4)$$

7. Melanjutkan iterasi

→ Iterasi dilanjutkan dengan mengulangi langkah 3-7 sampai batas laju pembelajaran yang diinginkan (penambahan *epoch*).

Untuk tahap pengujian, data pengujian dalam ruang eigen akan dimasukkan sebagai unit masukan dan dicari jarak terdekat dengan menggunakan *Euclidian distance* pada masing-masing bobot keluaran (vector perwakilan) seperti pada tahap pelatihan, kemudian unit keluaran dengan jarak terkecil akan dijadikan kelas dari unit masukan tersebut.

4.4.2 Hasil Self Organizing Maps

SOM	alpha	dikenali	recognition rate
1	0.3	5	33%
2	0.3	5	33%
3	0.3	5	33%
4	0.3	5	33%
5	0.3	5	33%
6	0.4	5	33%
7	0.4	5	33%
8	0.4	5	33%
9	0.4	5	33%
10	0.4	5	33%
11	0.5	5	33%
12	0.5	5	33%
13	0.5	5	33%
14	0.5	5	33%
15	0.5	5	33%
16	0.6	5	33%
17	0.6	5	33%
18	0.6	5	33%
19	0.6	5	33%
20	0.6	5	33%

4.5 Principle Component Analysis

4.5.1 Algoritma Principle Component Analysis

Langkah – langkah PCA

1. Tentukan sebuah data set yang ingin dianalisis
2. Ubah data dengan metode z-score
3. Hitung matriks covariance.

$$C = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix} \quad (5.5)$$

4. Hitung eigenvektor dan eigenvaluesnya
5. Pilih komponen dan bentuk feature vektor → pengurangan dimensi

$$\rightarrow \text{FeatureVector} = (\text{eig1 eig2 eig3 ... eign})$$

→ Perlu diperhatikan bahwa eigenvektor yang memiliki eigenvalue tertinggi adalah komponen dasar (*principle component*) dari data set. Oleh karena itu eigenvalue yang terkecil dapat dihilangkan (akan terjadi kehilangan informasi, namun karena eigenvalue-nya kecil, tidak akan terlalu signifikan). Sehingga akan didapatkan sebuah data set yang memiliki dimensi lebih kecil daripada dataset sebelumnya.

6. Memperoleh data set yang baru

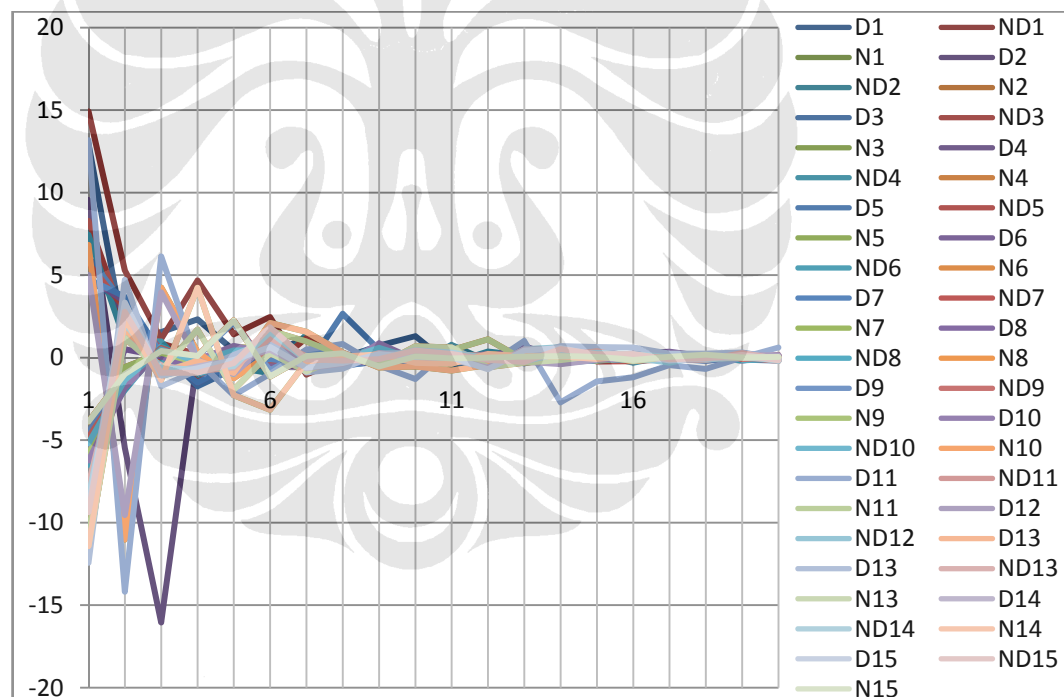
→ Merupakan langkah terakhir dan paling mudah dalam PCA

→ $\text{FinalData} = \text{DataAwal} \times \text{RowFeatureVector}$

dimana:

$\text{RowFeatureVector} = \text{transpose FeatureVector}$

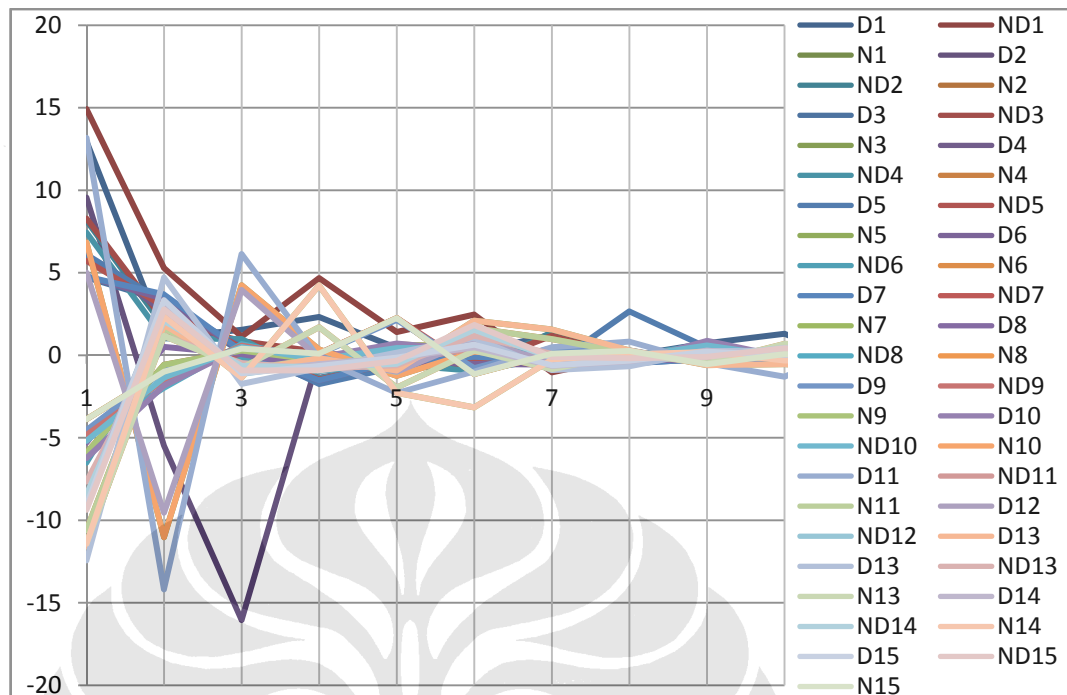
4.5.2 Data output PCA 20 dimensi



Gambar 4.17 Data output Principle Component Analysis dengan 20 data

Gambar 4.17 menunjukkan data output Principle Component Analysis dengan 20 data yang diambil dari 92 data panjang gelombang dari 45 kelompok sampel (3 klasifikasi x 15 sampel)

4.5.3 Data output PCA 10 dimensi



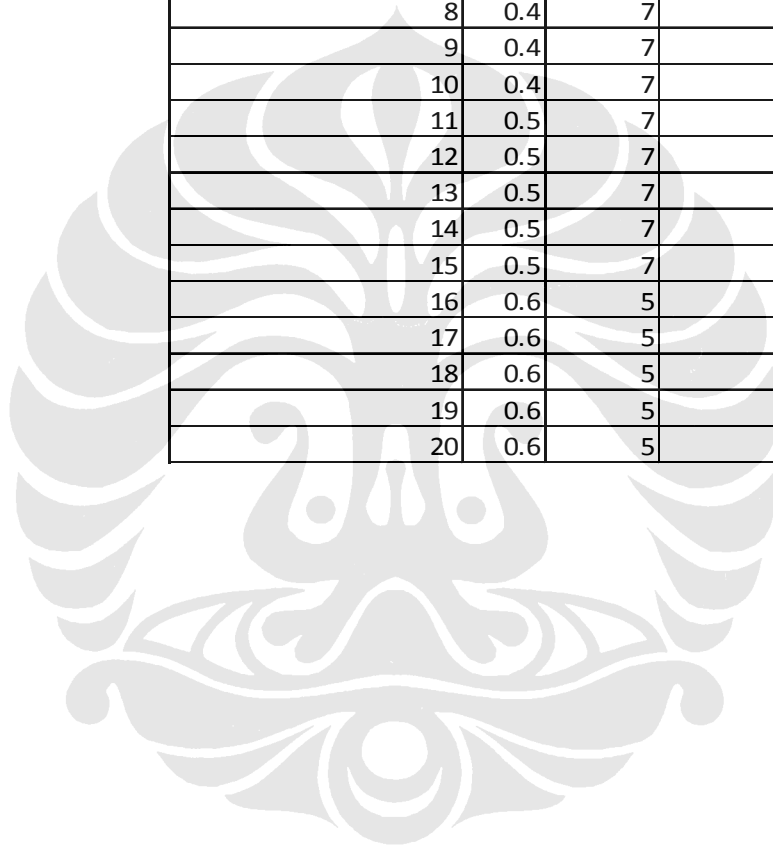
Gambar 4.18 Data output Principle Component Analysis dengan 10 data

Gambar 4.18 menunjukkan data output Principle Component Analysis dengan 10 data yang diambil dari 92 data panjang gelombang dari 45 kelompok sampel (3 klasifikasi x 15 sampel)

4.6 Self Organizing Maps dengan Data PCA

4.6.1 SOM dengan 20 Data PCA

PCA + SOM 20 dimensi	alpha	dikenali	recognition rate
1	0.3	7	47%
2	0.3	7	47%
3	0.3	7	47%
4	0.3	7	47%
5	0.3	7	47%
6	0.4	7	47%
7	0.4	7	47%
8	0.4	7	47%
9	0.4	7	47%
10	0.4	7	47%
11	0.5	7	47%
12	0.5	7	47%
13	0.5	7	47%
14	0.5	7	47%
15	0.5	7	47%
16	0.6	5	33%
17	0.6	5	33%
18	0.6	5	33%
19	0.6	5	33%
20	0.6	5	33%



4.6.2 SOM dengan 10 Data PCA

PCA + SOM 10 dimensi	alpha	dikenali	recognition rate
1	0.3	7	47%
2	0.3	7	47%
3	0.3	7	47%
4	0.3	7	47%
5	0.3	7	47%
6	0.4	7	47%
7	0.4	7	47%
8	0.4	7	47%
9	0.4	7	47%
10	0.4	7	47%
11	0.5	7	47%
12	0.5	7	47%
13	0.5	7	47%
14	0.5	7	47%
15	0.5	7	47%
16	0.6	5	33%
17	0.6	5	33%
18	0.6	5	33%
19	0.6	5	33%
20	0.6	5	33%

