

## **BAB III**

### **DESAIN SISTEM**

Pada bab ini akan diterangkan mengenai desain sistem yang digunakan sebagai panduan dalam penyusunan tugas akhir.

Perangkat lunak pada tugas akhir ini dapat menentukan lokasi kecamatan ritel yang cocok untuk digunakan sebagai acuan dalam penelitian pendahuluan atau riset pemasaran lebih lanjut. Keunggulan dari perangkat lunak ini, adalah menentukan lokasi berdasarkan jenis ritel yang diinginkan, modal usaha yang digunakan, dengan melihat aspek geografi dan demografi pada lokasi tersebut, yang berupa populasi penduduk per kecamatan di Surabaya.

Dalam penerapannya, algoritma Jaringan Saraf Tiruan (JST) propagasi balik membutuhkan data-data yang digunakan sebagai data latihan (*training set*). Data latihan yang digunakan berupa dua data utama yakni, data kependudukan serta data usaha ritel yang ada di Surabaya.

#### **3.1 Pra Pengolahan Data**

##### **3.1.1 Pengumpulan Data**

Data latihan atau *training set* di dapat dari Dinas Perdagangan dan Perindustrian (Disperindag) serta Badan Pusat Statistik (BPS). Data keseluruhan ini, kemudian dibagi atas data latihan dan data pengujian dengan perbandingan 80-20 persen dari jumlah keseluruhan data. Data dapat dilihat pada lampiran A dan B. Hal ini dilakukan untuk menguji apakah nilai error yang dihasilkan oleh perangkat lunak dapat diterima.

Dengan kata lain, melatih sistem jaringan untuk mencapai suatu keadaan dimana ia dapat merespon secara benar model input yang diberikan, dan kemampuan untuk memberikan respon atau jawaban yang masuk akal bagi data masuk yang mirip tetapi tidak identik dengan data masuk yang telah dilatih sebelumnya. Adapun detail dari data keseluruhan tersebut adalah:

### **1. Data Usaha Ritel**

Data usaha ritel di dapat dari Disperindag. Data yang digunakan sebagai *training set* adalah usaha ritel yang berdiri minimal 3 tahun hingga saat ini dan berupa, lokasi kecamatan usaha ritel itu, jenis ritel (pakaian, perhiasan, elektronik, dsb), serta besar modal usaha yang digunakan.

Total terdapat 39 jenis usaha ritel yang terdapat pada data latihan yaitu, antara lain: 1)Alat telekomunikasi, 2)Alat transportasi, 3)Alat tulis menulis dan gambar, 4)Bahan kendaraan, 5)Bahan Kimia, 6)Bahan konstruksi dari kayu, 7)Bahan konstruksi dari porselen, 8)Bahan konstruksi, 9)Barang farmasi, 10)Barang pecah belah dan perlengkapan dapur dari plastik, 11)Department store, 12)Elektronik, 13)Furnitur, 14)Hasil perikanan, 15)Hasil pertanian, 16)Hasil perternakan, 17)Jasa pengurusan transportasi, 18) Kertas, 19)Kimia dan farmasi, 20)Komputer dan mesin kantor, 21)Konsultasi bisnis dan manajemen, 22)Makanan dan minuman, 23)Makanan, minuman, dan tembakau, 24)Mesin, suku cadang, dan perlengkapannya, 25)Padi dan palawija, 26)Pakaian jadi, 27)Pakan ternak/unggas/ikan,

28)Pencetakan, penerbitan, dan perangkat lunak, 29)Peralatan dan perlengkapan rumah tangga, 30)Peralatan farmasi dan kesehatan, 31)Perhiasan, 32)Rumah makan, 33)Semen, kapur, pasir, dan batu, 34)Sepatu, sandal, dan alas kaki lainnya, 35)Sepeda motor serta suku cadang dan aksesorisnya, 36)Suku cadang dan asesoris mobil, 37)Tanaman hias, 38)Tas, dompet,koper, dan ransel, 39)Tekstil, pakaian jadi, dan kulit.

Adapun, data modal usaha yang dipakai oleh tiap jenis ritel yang telah disebutkan berkisar antara 20 juta hingga 2 milyar rupiah. Data modal yang dipakai yaitu: 1) 20.000.000, 2) 25.000.000, 3) 30.000.000, 4) 45.000.000, 5) 50.000.000, 6) 55.000.000, 7) 60.000.000, 8) 75.000.000, 9) 85.000.000, 10) 88.000.000, 11) 90.000.000, 12) 100.000.000, 13) 120.000.000, 14) 125.000.000, 15) 150.000.000, 16) 188.000.000, 17) 190.000.000, 18) 200.000.000, 19) 225.000.000, 20) 250.000.000, 21) 300.000.000, 22) 388.000.000, 23) 400.000.000, 24) 450.000.000, 25) 500.000.000, 26) 700.000.000, 27) 748.627.000, 28) 800.000.000, 29) 1.000.000.000, 30) 2.000.000.000.

Terdapat 31 kecamatan di Surabaya. Namun, lokasi kecamatan tiap usaha ritel yang disebutkan di atas apabila diakumulasi hanya berjumlah 22 kecamatan, yang terdiri atas: 1)Asemrowo, 2)Benowo, 3)Bubutan, 4)Dukuh pakis, 5)Gayungan, 6)Jambangan, 7)Karang pilang, 8)Krembengan, 9)Lakar santri, 10)Mulyorejo, 11)Pabean cantian, 12)Rungkut, 13)Sambikerep, 14)Sawahan, 15)Semampir,

16)Simokerto, 17)Sukolilo, 18)Sukomanunggal, 19)Tambaksari, 20)Wiyung, 21)Wonocolo, 22)Wonokromo.

## **2. Data Kependudukan**

Data kependudukan yang digunakan sebagai data latihan adalah pendapatan per kapita serta data populasi penduduk, yang terdiri atas populasi umur penduduk untuk tahun 2008, 2009, dan 2010. Data ini didapatkan dari BPS.

Namun, data yang terkumpul adalah data tahun 2010, sehingga penulis melakukan proses pengolahan data untuk mencari populasi umur, guna menentukan data kependudukan tahun 2008, dan 2009. Hal ini akan diterangkan lebih lanjut pada sub pengolahan data.

Menurut BPS, data yang berupa umur ini dibagi menjadi beberapa kisaran sebagai berikut: 1) 0-5 tahun, 2) 5-10 tahun, 3) 10-15 tahun, 4) 15-20 tahun, 5) 20-25 tahun, 6) 25-30 tahun, 7) 30-35 tahun, 8) 35-40 tahun, 9) 40-45 tahun, 10) 45-50 tahun, 11) 50-55 tahun, 12) 55-60 tahun, dan 13) 60 tahun keatas.

### **3.1.2 Pengolahan Data**

Setelah data latihan dikumpulkan, data tersebut mengalami proses pengolahan data terlebih dahulu.

Data pendapatan per bulan didapatkan dari hasil pembagian jumlah pendapatan per kapita tiap kecamatan dengan 12. Telah diterangkan diatas bahwa, data kependudukan yang didapatkan hanya data kependudukan (populasi jenis kelamin dan umur) tahun 2010. Proses perhitungan untuk

mencari data kependudukan tahun 2008 dan 2009 dilakukan dengan menghitung presentase jumlah tiap populasi tiap umur dari jumlah total populasi warga di kecamatan tahun 2010. Hasil presentase ini kemudian digunakan sebagai faktor seberapa banyak populasi jenis kelamin dan umur dari total populasi warga tiap kecamatan pada tahun 2008 dan 2009.

Data umur penduduk yang didapat terdiri atas 13 kisaran (seperti yang dituliskan sebelumnya). Angka ini terlalu besar dikarenakan dalam pengembangannya, tiap pola data latihan mengalami perhitungan kombinasi. Sehingga penulis membaginya kedalam 5 kisaran umur yaitu 0-15 tahun, 15-30 tahun, 30-45 tahun, 45-60 tahun, dan 60 tahun keatas.

### **3.1.3 Proses Pemetaan Data**

Sebelum melalui proses pembelajaran pada JST propagasi balik, data terlebih dahulu dipeta atau di-*mapping*, mengingat algoritma dari JST propagasi balik menggunakan perhitungan matematika. Pemetaan data dibagi menjadi dua, yakni pemetaan data berupa kategori dan pemetaan data berupa angka.

#### **3.1.3.1 Pemetaan Data Kategori**

Data yang berupa kategori yang digunakan dalam penyusunan tugas akhir ini adalah data jenis ritel dan kecamatan. Pemetaan data untuk kategori dilakukan dengan mengubah data itu menjadi data biner atau antara 0 dan 1 karena untuk perhitungan fungsi aktivasi dari algoritma JST propagasi balik yang digunakan adalah fungsi biner sigmoid.

Langkah-langkah dari proses pemetaan data yang berupa kategori dilakukan sebagai berikut:

1. Memberikan nilai numerik pada tiap-tiap kategori.
2. Membagi nilai numerik tersebut dengan 2 hingga hasil pembagiannya bernilai 1.
3. Hasil dan sisa pembagian kemudian diambil dan digunakan sebagai nilai biner dari angka numerik tersebut.
4. Jumlah maksimal unit atau bit yang digunakan dihitung dengan melakukan proses nomor 2 pada nilai numerik yang paling besar.

Seperti dalam tabel kategori jenis ritel dibawah ini, nilai numerik yang paling besar adalah 39. Nilai numerik ini kemudian mengalami proses seperti penjelasan pada langkah nomor dua. Dari proses itu menghasilkan jumlah unit atau bit sebanyak 6. Oleh karena itu, nilai numerik yang memiliki jumlah bit kurang dari 6 ini kemudian diisi dengan nilai bit 0. Seperti contoh pada nilai numerik 2, angka ini memiliki nilai biner 1 dan 0. Namun, dikarenakan jumlah maksimal bit yang digunakan adalah 6, maka di depan angka bit 1 ditambahkan nilai bit 0. Sehingga menjadi angka niner 000010.

Berikut ini adalah tabel pemetaan data yang dilakukan untuk jenis ritel, dengan variabel  $x_i$  ( $i = 1,2,\dots,n$ ) sebagai unit masukan pada JST.

**Tabel 3.1 Tabel Pemetaan Data Jenis Ritel**

Jenis Ritel	Numerik	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$
Pencetakan, penerbitan, dan perangkat lunak	1	0	0	0	0	0	1
Bahan-bahan konstruksi	2	0	0	0	0	1	0
Sepeda motor serta suku cadang dan asesorisnya	3	0	0	0	0	1	1
Alat transportasi	4	0	0	0	1	0	0
Bahan konstruksi dari kayu	5	0	0	0	1	0	1
Komputer dan mesin kantor	6	0	0	0	1	1	0
Mesin, suku cadang, dan perlengkapannya	7	0	0	0	1	1	1
Pakaian jadi	8	0	0	1	0	0	0
Semen, kapur, pasir, dan batu	9	0	0	1	0	0	1
Sepatu, sandal, dan alas kaki lainnya	10	0	0	1	0	1	0
Alat telekomunikasi	11	0	0	1	0	1	1
Peralatan dan perlengkapan rumah tangga	12	0	0	1	1	0	0
Rumah makan	13	0	0	1	1	0	1
Department store	14	0	0	1	1	1	0
Alat tulis menulis dan gambar	15	0	0	1	1	1	1
Bahan bakar kendaraan	16	0	1	0	0	0	0
Bahan konstruksi dari poselen	17	0	1	0	0	0	1
Makanan, minuman, dan tembakau	18	0	1	0	0	1	0
Hasil peternakan	19	0	1	0	1	1	0
Jasa pengurusan transportasi	20	0	1	0	1	0	0
Pakan ternak/unggas/ikan	21	0	1	0	1	0	1
Hasil pertanian	22	0	1	0	1	1	0
Konsultasi bisnis dan manajemen	23	0	1	0	1	1	1
Makanan dan minuman	24	0	1	1	0	0	0
Padi dan palawijaya	25	0	1	1	0	0	1
Tekstil, pakaian jadi, dan kulit	26	0	1	1	0	1	0
Barang farmasi	27	0	1	1	0	1	1
Peralatan farmasi dan kesehatan	28	0	1	1	1	0	0
Kimia dan farmasi	29	0	1	1	1	0	1
Bahan kimia	30	0	1	1	1	1	0
Barang pecah belah dan perlengkapan dapur dari plastik	31	0	1	1	1	1	1
Elektronik	32	1	0	0	0	0	0
Suku cadang dan Asesoris mobil	33	1	0	0	0	0	1
Furnitur	34	1	0	0	0	1	0
Perhiasan	35	1	0	0	0	1	1
Tas, dompet, dan ransel	36	1	0	0	1	0	0
Hasil perikanan	37	1	0	0	1	0	1
Kertas	38	1	0	0	1	1	0
Tanaman hias	39	1	0	0	1	1	1

Sedangkan untuk pemetaan data yang dilakukan untuk kecamatan ditunjukkan dalam tabel dibawah ini. Variabel  $y$  (1,2,..n) mewakili tiap unit keluaran pada jaringan.

**Tabel 3.2 Tabel Pemetaan Data Kecamatan**

Kecamatan	Numerik	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
Asemrowo	1	0	0	0	0	1
Benowo	2	0	0	0	1	0
Bubutan	3	0	0	0	1	1
Dukuh Pakis	4	0	0	1	0	0
Gayungan	5	0	0	1	0	1
Jambangan	6	0	0	1	1	0
Karang Pilang	7	0	0	1	1	1
Krembengan	8	0	1	0	0	0
Lakar santri	9	0	1	0	0	1
Mulyorejo	10	0	1	0	1	0
Pabean cantian	11	0	1	0	1	1
Rungkut	12	0	1	1	1	1
Sambikerep	13	1	0	0	0	0
Sawahan	14	1	0	0	0	1
Semampir	15	1	0	0	1	0
Simokerto	16	1	0	0	1	1
Sukolilo	17	1	0	1	0	0
Sukomanunggal	18	1	0	1	0	1
Tambaksari	19	1	0	1	1	0
Wiyung	20	1	0	1	1	1
Wonoloco	21	1	1	0	0	0
Wonokromo	22	1	1	0	0	1

### 3.1.3.2 Pemetaan Data Angka

Data yang berupa angka yang terdapat di dalam data latihan (*training set*) adalah modal usaha, pendapatan rata-rata per bulan (hasil olahan dari pendapatan per kapita) dan populasi umur penduduk.

Sebelum digunakan sebagai data masukan pada jaringan, data angka ( $x$ ) dipetakan dengan memasukan angka tersebut kedalam rumus pada persamaan 3.1.

Dalam penerapannya, berikut ini adalah langkah-langkah yang dilakukan:



1. Mengambil data dari Microsoft Excel.
2. Untuk tiap baris dan kolom data yang berupa angka, yakni, modal usaha, pendapatan, dan populasi umur penduduk, diambil.
3. Data tersebut kemudian dimasukkan kedalam rumus pada persamaan 3.1.
4. Data hasil pemetaan tersebut selanjutnya digunakan sebagai representasi data atau data masukan pada jaringan.

$$f_{ln}(x) = \left( \frac{1}{(1 + \exp(\ln(1 + x)))} \right)$$

**Persamaan 3.1**

Berikut ini adalah 10 contoh dari hasil pemetaan data berupa angka dalam pengembangan perangkat lunak ini. Tabel 3.3 berikut ini merupakan hasil dari pemetaan data angka yang berupa modal usaha (dalam mata uang rupiah).

**Tabel 3.3 Contoh Hasil Pemetaan Data Modal**

Modal Usaha	Hasil Pemetaan Data
100.000.000	0.09999999800000013
50.000.000	0.01999999919999997
500.000.000	0.01999999919999976
200.000.000	0.04999999949999998
750.000.000	0.018181817520661156
388.000.000	0.01333332977777782
300.000.000	0.02577319574343714
120.000.000	0.08333333194444455
250.000.000	0.03999999968000003

Sedangkan pada tabel 3.4 adalah 10 contoh dari hasil pemetaan data angka yang berupa pendapatan rata-rata per bulan (dalam mata uang rupiah).

**Tabel 3.4 Contoh Hasil Pemetaan Data Pendapatan per bulan**

<b>Pendapatan per bulan</b>	<b>Hasil Pemetaan Data</b>
7.380.928	0.1354842817910481
1.001.749	0.9982520606418159
3.540.144	0.28247422563928143
3.091.956	0.3234196583524095
1.157.190	0.864156350080108
2.562.625	0.3902245625289988
2.479.847	0.40325035919525737
1.423.400	0.7025422192746678
1.777.309	0.5626477301946594

Pada tabel 3.5 dibawah ini, merupakan 10 contoh dari hasil pemetaan data berupa populasi umur penduduk dengan kisaran umur 0-15 tahun.

**Tabel 3.5 Contoh Hasil Pemetaan Data Populasi Umur**

<b>Populasi Umur 0-15 Tahun</b>	<b>Hasil Pemetaan Data</b>
7778	0.12853470437018003
9516	0.1050640890943475
21556	0.04638649225345578
11708	0.0853970964987191
7801	0.12815583749839812
12155	0.08225713580653118
28604	0.03495770118157033
9853	0.10147133434804662
8353	0.11968880909634955
13867	0.07210325185665879

### 3.1.4 Representasi Data Pada Jaringan

Setelah melakukan proses pemetaan data seperti yang telah dijelaskan pada sub bab sebelumnya, nilai dari hasil pemetaan data tersebut kemudian digunakan sebagai data masuk (*input*) dan data keluar (*output*) atau target pada jaringan. Setiap unit pada lapisan masuk (*input layer*) terkoneksi dengan tiap

unit pada lapisan tersembunyi (*hidden layer*) yang berjumlah 25 unit, dan tiap neuron atau unit pada lapisan tersembunyi ini juga terkoneksi dengan setiap unit pada lapisan keluar (*output layer*).

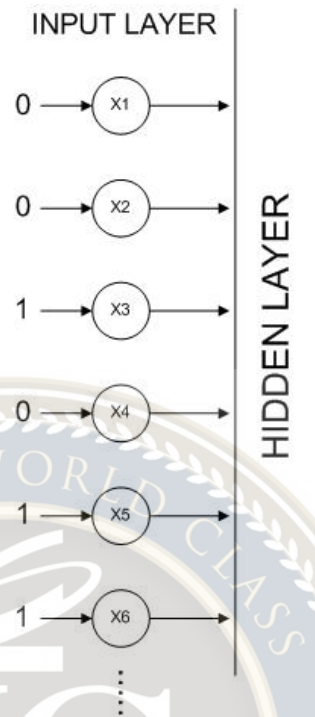
#### **3.1.4.1 Representasi Data Masuk Pada Jaringan**

Data masuk pada jaringan berupa jenis ritel, modal, pendapatan rata-rata per bulan, serta populasi umur penduduk. Dalam penerapannya, jaringan pada perangkat lunak ini menggunakan total neuron atau unit sebanyak 13 unit masuk, yang terdiri atas 6 unit masuk untuk data jenis ritel, 5 unit masuk yang mempresentasikan data populasi umur penduduk dan masing-masing 1 unit masuk berupa representasi data modal, pendapatan per bulan. Berikut ini akan dijelaskan lebih lanjut mengenai masing-masing proses representasi data pada unit masuk di dalam jaringan.

##### **1. Jenis Ritel**

Data jenis ritel yang telah dipetakan menjadi biner, memiliki maksimal 6 digit atau bit, sehingga neuron atau unit masuk yang diperlukan sebanyak 6 unit.

Gambar 3.1 dibawah ini merupakan contoh dari representasi data masuk dari hasil pemetaan biner jenis ritel rumah makan. Variabel X mewakili tiap neuron atau unit pada lapisan masuk (*input layer*).



**Gambar 3.1 Representasi Data Masuk Jenis Ritel**

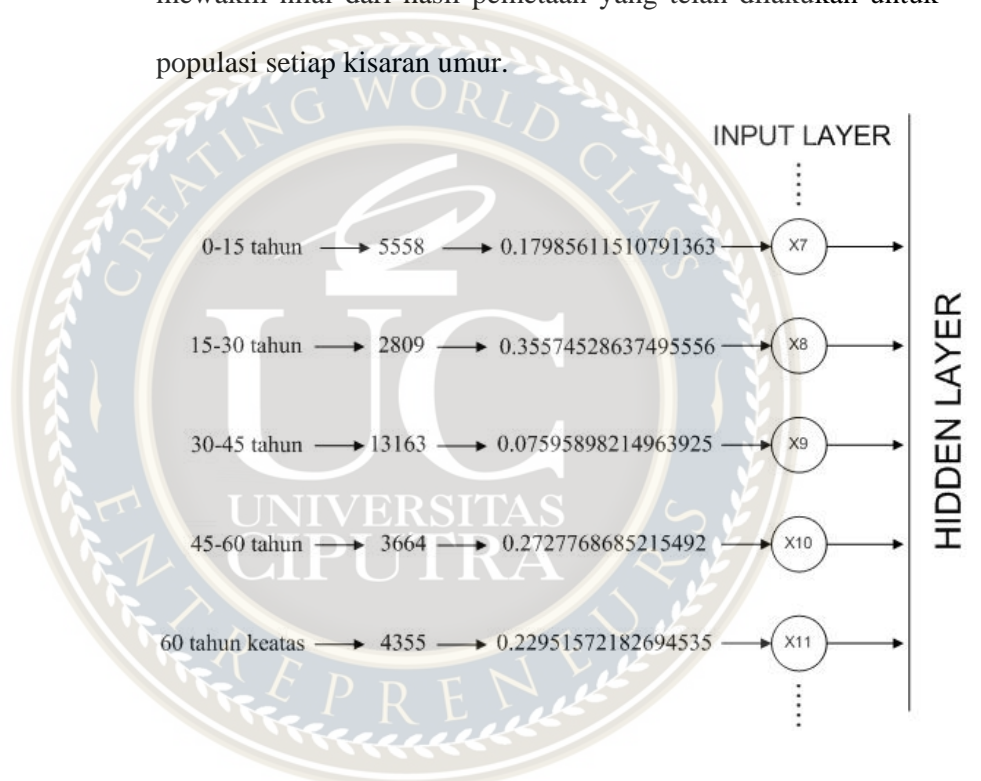
Unit masuk ke-1 hingga 6 pada jaringan digunakan sebagai representasi data masuk berupa jenis ritel. Seperti yang telah diketahui, jenis ritel rumah makan memiliki biner 001011. Tiap unit masuk mewakili satu bit yang akan menjadi nilai masuk (*input*) pada jaringan. Data masuk ini selanjutnya diteruskan ke tiap unit pada lapisan tersembunyi atau *hidden layer*.

## 2. Populasi Umur Penduduk

Data populasi umur penduduk yang diperoleh terdiri atas 5 kisaran umur, yakni populasi penduduk berusia 0-15 tahun, 15-30 tahun, 30-45 tahun, 45-60 tahun, dan 60 tahun keatas.

Sehingga diperlukan 5 unit masuk untuk mewakili nilai masuk (*input*) dari tiap kisaran umur.

Gambar 3.2 dibawah ini adalah contoh dari pemetaan populasi umur data kependudukan untuk tahun 2008. Gambar tersebut menunjukkan tiap unit masuk (X) pada jaringan mewakili nilai dari hasil pemetaan yang telah dilakukan untuk populasi setiap kisaran umur.



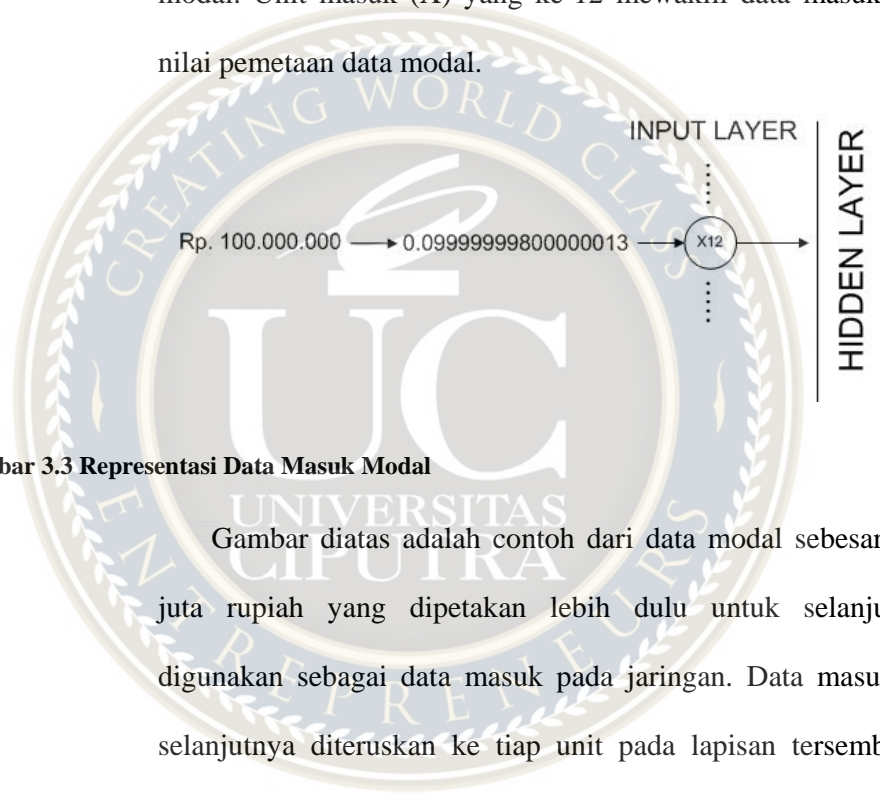
**Gambar 3.2 Representasi Data Masuk Populasi Umur**

Tiap unit masuk (X) ke-7 hingga 11 mempresentasikan data atau *input* untuk populasi setiap kisaran umur. Data masuk ini selanjutnya diteruskan ke tiap unit pada lapisan tersembunyi atau *hidden layer*.

### 3. Modal

Data berupa modal direpresentasikan menggunakan 1 unit masuk. Nilai dari data masuk tersebut merupakan hasil dari pemetaan data modal yang telah dilakukan sebelumnya.

Gambar 3.3 menunjukkan representasi data masuk berupa modal. Unit masuk (X) yang ke-12 mewakili data masuk dari nilai pemetaan data modal.



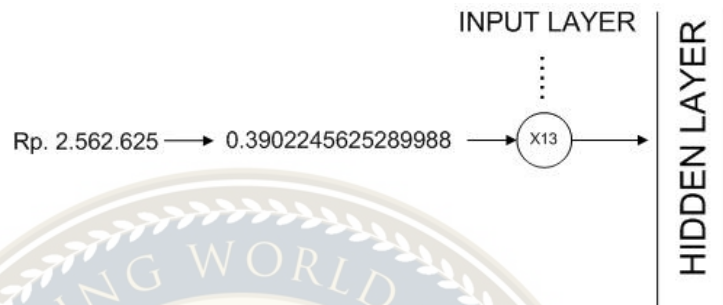
**Gambar 3.3 Representasi Data Masuk Modal**

Gambar diatas adalah contoh dari data modal sebesar 100 juta rupiah yang dipetakan lebih dulu untuk selanjutnya digunakan sebagai data masuk pada jaringan. Data masuk ini selanjutnya diteruskan ke tiap unit pada lapisan tersembunyi atau *hidden layer*.

### 4. Pendapatan per bulan

Data yang berupa pendapatan rata-rata per bulan penduduk direpresentasikan dengan menggunakan 1 unit masuk. Unit masuk ini mewakili nilai dari hasil pemetaan data pendapatan yang telah dilakukan sebelumnya.

Unit masuk ke-13 mewakili data masuk dari nilai pemetaan data modal. Gambar 3.4 menunjukkan representasi data masuk berupa modal.



**Gambar 3.4 Representasi Data Masuk Pendapatan**

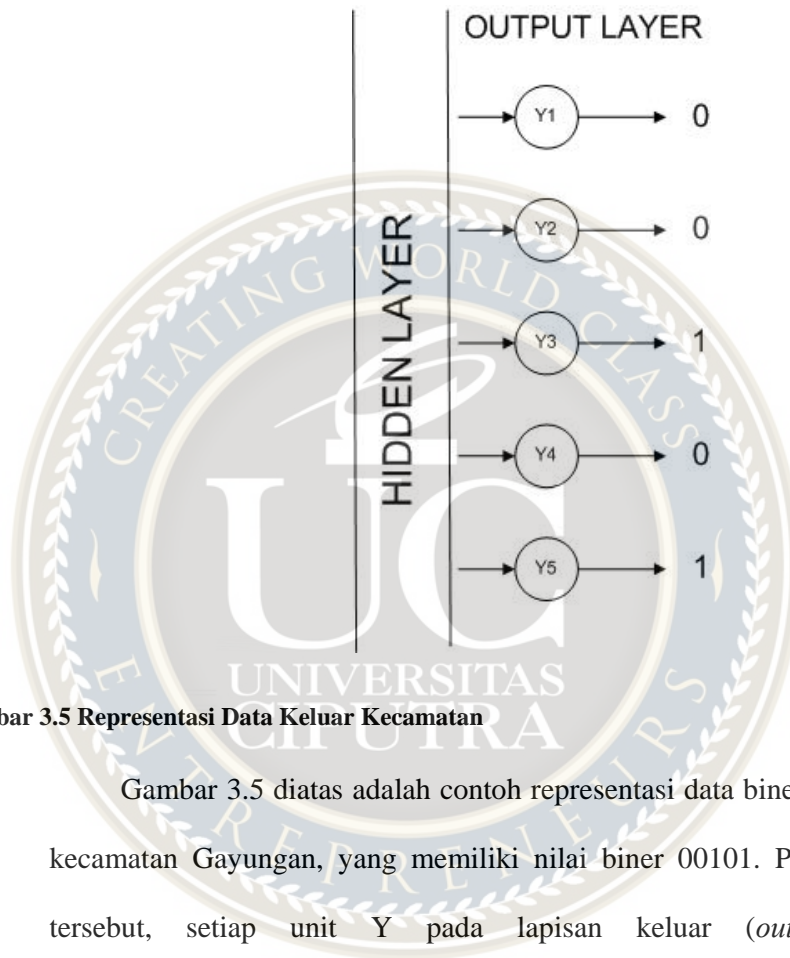
Gambar diatas adalah contoh dari pemetaan data pendapatan rata-rata per bulan penduduk di kecamatan Sukolilo pada tahun 2008. Hasil pemetaan tersebut selanjutnya digunakan sebagai representasi data pada unit masuk (X) yang ke-13 di dalam jaringan. Data masuk ini selanjutnya diteruskan ke tiap unit pada lapisan tersembunyi atau *hidden layer*.

#### **3.1.4.2 Representasi Data Keluar Pada Jaringan**

Lapisan keluar atau *output layer* pada jaringan ini terdiri atas 5 neuron atau unit keluar. Hal ini dikarenakan jumlah digit atau bit pada pemetaan biner data kecamatan memiliki jumlah bit sebanyak 5.

Tiap unit pada lapisan keluar mewakili nilai keluaran (*actual output*) yang dihasilkan oleh jaringan selama proses pembelajaran, pengujian, maupun pemakaian dengan mempresentasikan satu bit dari hasil pemetaan biner data nilai keluaran tersebut dan target yang berupa

kecamatan. Dengan kata lain, setiap unit yang ada pada lapisan keluar dalam jaringan perangkat lunak ini, mempresentasikan hasil atau nilai keluaran dan target yang diinginkan yakni, kecamatan.



**Gambar 3.5 Representasi Data Keluar Kecamatan**

Gambar 3.5 diatas adalah contoh representasi data biner dari target kecamatan Gayungan, yang memiliki nilai biner 00101. Pada gambar tersebut, setiap unit Y pada lapisan keluar (*output layer*) mempresentasikan satu nilai bit dari pemetaan biner target kecamatan itu. Dapat dilihat juga bahwa, setiap unit pada lapisan tersembunyi terkoneksi dengan tiap unit yang ada di dalam lapisan keluar.



## 3.2 Spesifikasi Perangkat Lunak

Pada sub bab ini, akan dibahas mengenai hal-hal yang dibutuhkan sebelum melakukan rancang bangun perangkat lunak. Hal ini meliputi pemilihan sistem operasi, *platform*, *framework*, *library*, dan tools.

### 1. Platform

*Platform*<sup>29</sup> yang dipakai dalam pengembangan perangkat lunak ini adalah bahasa pemrograman Java.

### 2. Sistem Operasi

Dalam pembuatan perangkat lunak penentuan lokasi ritel di Surabaya ini menggunakan komputer dengan sistem operasi Windows 7 SP 1 (32-bit), RAM 2GB, prosessor Intel Core i3 @2GHz.

### 3. Framework

*Framework*<sup>30</sup> yang digunakan dalam pengembangan perangkat lunak untuk *platform* Java adalah Netbeans IDE 6.8.

### 4. Library

Seperti yang telah diterangkan diatas, pengembangan perangkat lunak ini menggunakan bahasa pemrograman Java, berikut adalah beberapa kelas *library* di dalam Java yang digunakan:

---

<sup>29</sup> A platform is any base of technologies on which other technologies or processes are built. Definition platform?, SearchServerVirtualization, viewed 30 Mei 2011, <<http://www.searchservirtualization.com/definition/platform>>.

<sup>30</sup> In computer systems, a framework is often a layered structure indicating what kind of programs can or should be built and how they would interrelate. Some computer system frameworks also include actual programs, specify programming interfaces, or offer programming tools for using the frameworks. Framework, Whatis Dot Com, viewed 30 Mei 2011, <[http://whatis.techtarget.com/definition/0,,sid9\\_gci1103696,00.html](http://whatis.techtarget.com/definition/0,,sid9_gci1103696,00.html)>.

a. Java Swing

Java Swing berisikan komponen-komponen atau kelas-kelas pembangun tampilan antar muka atau *user interface*.

b. JFormLayout

Merupakan salah satu *class library* dalam Java yang digunakan untuk medesain tampilan atau *layout*.

c. JExcel

JExcel berisikan kelas guna membaca dan menulis ke dalam Microsoft Excel.

## 5. Tools

Alat pendukung lainnya adalah Microsoft Excel, yang digunakan sebagai sarana pencatatan data latihan (*training set*).

## 3.3 Perancangan

### 3.3.1 Arsitektur Jaringan

Tipe arsitektur dari JST propagasi balik adalah arsitektur jaringan lapisan banyak (*multilayer network*). Dalam penerapannya, arsitektur yang digunakan pada perangkat lunak ini terdiri atas satu lapisan masuk (*input layer*), satu lapisan tersembunyi (*hidden layer*), dan satu lapisan keluar (*output layer*). Dimana pada tiap lapisan tersebut terdiri atas beberapa neuron atau unit pemrosesan informasi.

Jaringan yang digunakan dalam perangkat lunak ini memiliki jumlah neuron atau unit masuk (*input unit*) sebanyak 13 unit, yang terdiri atas 6 bit

biner hasil pemetaan jenis ritel, modal dan pendapatan, serta 5 kisaran umur. Jumlah unit pada lapisan tersembunyi atau *hidden unit* berjumlah 25 unit, sedangkan unit keluar (*output unit*) yang digunakan sebanyak 5 unit, yaitu berupa hasil pemetaan data biner tiap kecamatan. Nilai data masuk dan data keluar dari jaringan telah dijelaskan pada sub bab 3.1.4.

### **3.3 4 Learning Rate**

Proses pembelajaran JST propagasi balik pada perangkat lunak ini menggunakan nilai *learning rate* yang sebesar 0.02. Hal ini dikarenakan memberikan nilai yang terlalu besar akan membuat proses belajar menjadi sangat cepat, namun memberikan dampak pada tidak selesainya pembelajaran atau tidak tercapainya nilai RMSE. Sebaliknya, menggunakan nilai *learning rate* yang terlalu kecil akan berakibat pada lamanya jumlah iterasi tetapi nilai RMSE tercapai.

### **3.3.2 Fungsi Aktifasi**

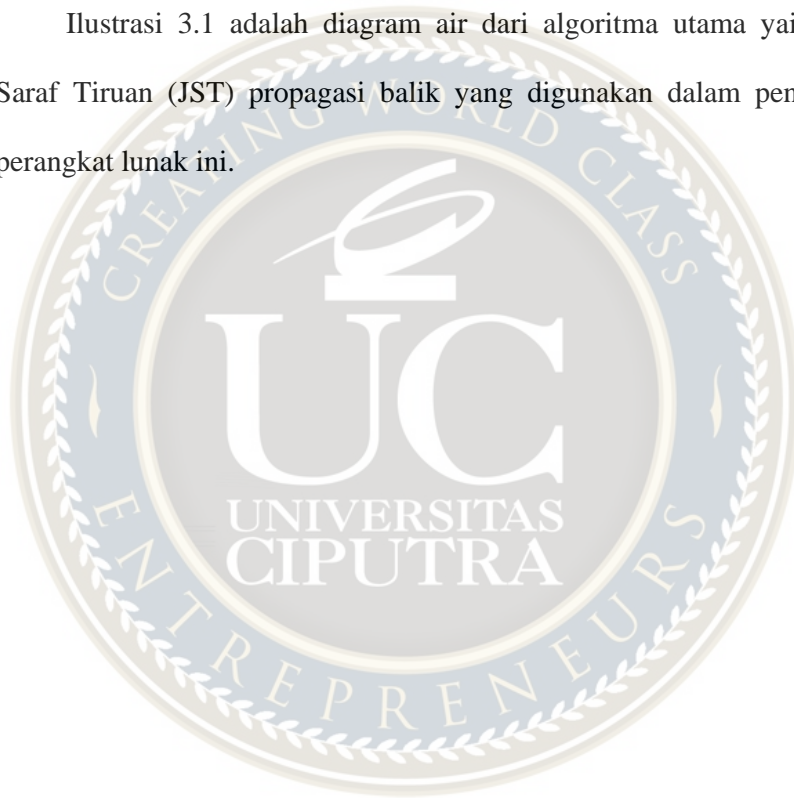
Pada tiap neuron atau unit yang terdapat pada lapisan tersembunyi dan keluar, memiliki sebuah fungsi aktifasi untuk menghitung nilai keluar yang dihasilkan oleh neuron atau unit itu pada jaringan. Dalam penerapannya, fungsi aktifasi yang digunakan adalah fungsi aktifasi biner sigmoid, yang tertulis pada persamaan 2.2.

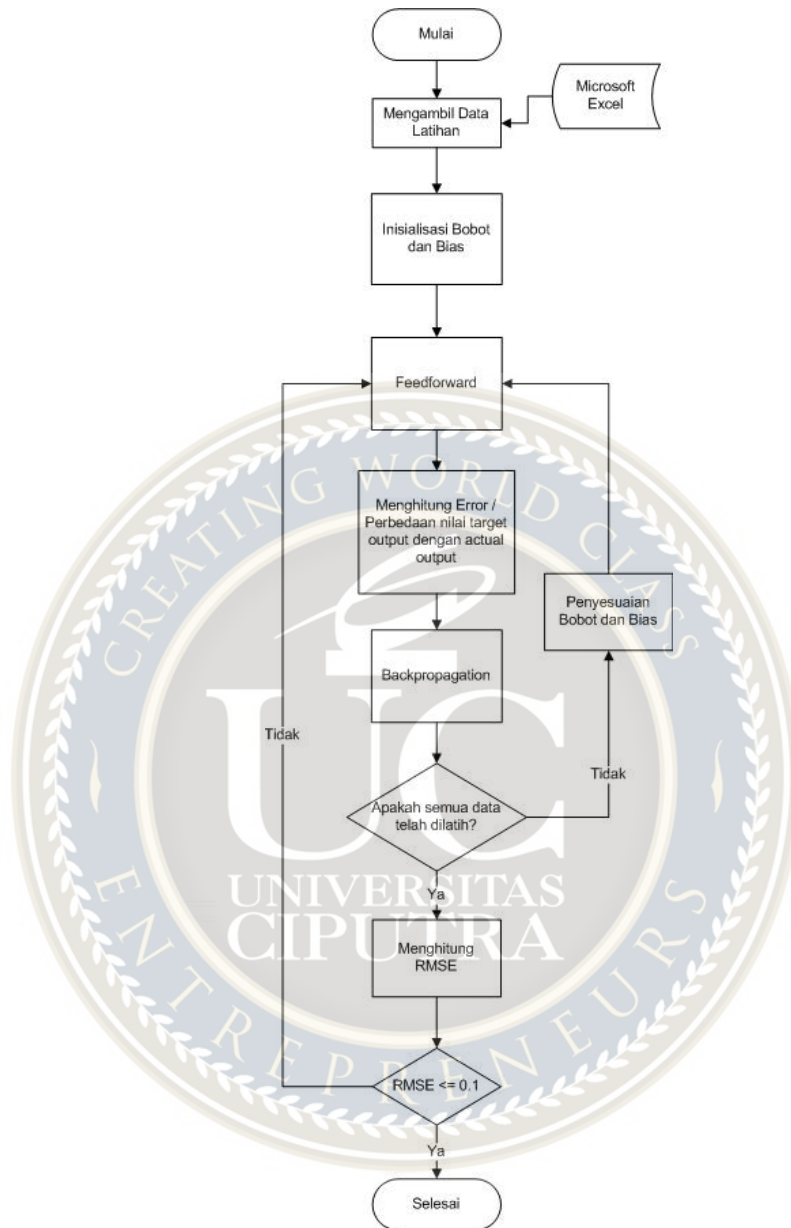
### 3.3.3 Inisialisasi Bobot dan Bias

Pada pengembangan perangkat lunak pada tugas akhir ini, proses pelatihan jaringan menggunakan inisialisasi bobot dan bias secara acak yaitu diantara -0.5 dan 0.5 dan dengan inisialisasi pendekatan *nguyen-widrow*.

### 3.3.2 Diagram Air

Ilustrasi 3.1 adalah diagram air dari algoritma utama yaitu Jaringan Saraf Tiruan (JST) propagasi balik yang digunakan dalam pengembangan perangkat lunak ini.





**Gambar 3.6** *Flowchart* Algoritma JST Propagasi Balik

### 3.2.2.1 Pemetaan Data

Pada proses ini dilakukan pengambilan dan pemetaan data. Proses pemetaan data terbagi menjadi dua, yaitu pemetaan data kategori dan pemetaan data angka

**Algoritma Pemetaan Data Kategori**

```
// Fungsi : Memetakan data berupa kategori
// Input : Sebuah file excel yang diproses
// Variabel yang digunakan : Hashmap dengan nama data
map, inputFile dengan tipe data String, compare yang
berupa arraylist, isSign dengan tipe data boolean, dan
binaryValue berupa String
MULAI
    inputFile ← membaca file yang akan diproses
    data ← mengambil array dari file
    compare ← menyimpan data yang tidak sama untuk
dipetakan
    binaryValue ← nilai biner yang didapatkan
    map ← berisi data hasil pemetaan (binaryValue)
AKHIR
```

**Segmen Algoritma 3.1 Pemetaan Data Kategori**

Langkah pertama yang dilakukan adalah mengambil data latihan yang telah disimpan dalam file excel. Data yang diambil kemudian disimpan ke dalam sebuah ArrayList dalam Java. Data tersebut selanjutnya dilakukan proses perbandingan guna mendapatkan data yang tidak kembar. Langkah berikutnya, data yang telah melalui proses perbandingan tersebut, diproses untuk dicari nilai binernya. Hasil nilai biner tersebut kemudian disimpan pemetaannya menggunakan objek HashMap.

Sedangkan untuk algoritma pemetaan data yang berupa angka dinormalisasi digambarkan pada segmen algoritma 3.2.

```

Algoritma Pemetaan Data Angka
// Fungsi : Memetakan data berupa angka
// Input : Sebuah file excel yang diproses
// Variabel yang digunakan : Hashmap dengan nama data
map, inputFile dengan tipe data String, compare yang
berupa arraylist, dan value berupa tipe data double
MULAI
    inputFile ← membaca file yang akan diproses
    data ← mengambil array dari file
    compare ← menyimpan data yang tidak sama untuk
dipetakan
    value ← nilai dari proses normalisasi
    map ← berisi data hasil pemetaan (value)
AKHIR

```

### Segmen Algoritma 3.2 Pemetaan Data Angka

Proses pemetaan data yang berupa angka, kurang lebih sama dengan proses pemetaan data kategori. Hanya saja, data dipetakan dengan menggunakan pemetaan data angka.

#### 3.2.2.2 Perambatan Maju

Langkah pertama dalam algoritma propagasi balik adalah melalui tahap perambatan maju, yang digambarkan pada segmen 3.3.

Setelah data mengalami proses pemetaan data, data tersebut selanjutnya masuk ke dalam jaringan dengan memasukannya kedalam objek pola atau pattern. Pattern menyimpan semua pola data latihan.

Untuk tiap pattern yang ada, masuk kedalam jaringan. Bila berada di unit masuk maka pattern tersebut (data masuk dan target) diteruskan ke unit selanjutnya. Pada unit tersembunyi, dilakukan proses fungsi penjumlahan untuk menghitung nilai input tiap unit, dan memproses nilai input tersebut dengan fungsi aktivasi guna menghitung nilai keluarannya.

#### **Algoritma Perambatan Maju**

```
// Fungsi : Melakukan algoritma perambatan maju
// Input : Array Pattern dengan tipe data Pattern
// Variabel yang utama yang digunakan : input
dengan tipe data double, dan output dengan tipe
data double.
MULAI
    Mengambil pattern dari file dan dipetakan
    terlebih dahulu
    Untuk tiap pattern yang ada
        Untuk tiap lapisan di dalam jaringan
        Jika berada pada unit atau neuron di
        lapisan pertama
            Set input dan output untuk unit
            pertama
        Akhir percabangan
        Jika berada pada unit di lapisan
        tersembunyi dan keluar
            Untuk tiap neuron di lapisan itu
                Input ← hasil fungsi
                penjumlahan
                Output ← hasil fungsi
                aktifasi
            Akhir perulangan
        Akhir percabangan
    Akhir perulangan
    Akhir perulangan
```

#### **Segmen Algoritma 3.3 Perambatan Maju**

Perhitungan perambatan maju terus dilakukan hingga mencapai lapisan terakhir, yaitu lapisan keluar.

#### **3.2.2.2 Perambatan Mundur**

Tahap selanjutnya adalah perambatan mundur atau algoritma balik propagasi. Segmen 3.4 menggambarkan alur dari proses perambatan mundur.



#### **Algoritma Perambatan Mundur**

```
// Fungsi : Melakukan algoritma perambatan mundur
// Input : Array Pattern dengan tipe data Pattern
// Variabel yang digunakan : error tipe double,
weights dan bias dengan tipe data array double,
actualOutput berupa data double, array output
berupa double, weightCorrectionTerm tipe data
double, biasCorrectionTerm tipe data double
Untuk tiap pattern
    Jika telah berada pada lapisan keluar atau
    lapisan tersembunyi
    Untuk tiap unit atau neuron yang berada pada
    lapisan keluar dan lapisan tersembunyi
    error ← hasil error
    weightCorrectionTerm ← menghitung besar
    koreksi untuk bobot
    biasCorrectionTerm ← menghitung besar
    koreksi untuk bias
    Memanggil fungsi pengoreksian bobot dan bias

    Akhir perulangan
    Akhir percabangan
    Akhir perulangan
```

#### **Segmen Algoritma 3.4 Perambatan Mundur**

Untuk tiap unit pada lapisan keluar (*output layer*), menghitung nilai error yang dihasilkan untuk digunakan sebagai perhitungan besar koreksi bobot dan bias untuk tiap koneksi di dalam jaringan.

#### **3.2.2.3 Perhitungan Root Mean Square Error (RMSE)**

Langkah terakhir adalah menghitung *Root Mean Square Error* atau disingkat dengan RMSE. Sebelum dapat menghitung error dari keseluruhan data latihan in, *Sum Square Error* (SSE) dihitung terlebih dahulu.

```

Algoritma Perhitungan SSE
// Fungsi : Untuk perhitungan RMSE
// Input : array error tipe data double
// Variabel yang dipakai : SSE tipe data double

Jika berada di lapisan keluar
    error ← error yang dihasilkan tiap unit
    keluar
    Untuk unit yang ada di lapisan keluar
        SSE ← perhitungan SSE untuk tiap unit
    keluar
    Akhir perulangan
    Akhir perulangan

```

### Segmen Algoritma 3.5 Perhitungan SSE

SSE dihitung setelah semua nilai error dari tiap-tiap unit pada lapisan keluar didapatkan. Nilai SSE dihitung dengan menggunakan persamaan 2.18.

```

Algoritma Perhitungan RMSE
// Fungsi : tolok ukur error yang dapat diterima
// Input : SSE tipe data double,
// errorYangDapatDiterima tipe data double
// Variabel : RMSE tipe data double
SSE ← Mengambil hasil perhitungan SSE
RMSE ← Menghitung RMSE
Jika RMSE < errorYangDapatDiterima
    END
    AKHIR Percabangan
Jika RMSE > errorYangDapatDiterima
    Memanggil fungsi learn() kembali
    AKHIR Percabangan

```

### Segmen Algoritma 3.6 Perhitungan RMSE

Setelah nilai SSE dihitung, program mengambil nilai SSE itu dan menghitung nilai RMSE dengan menggunakan persamaan 2.19.

### 3.2.4 Package Diagram

Struktur diagram dari perangkat lunak ini terdiri atas beberapa *package*. Tujuan dari pembagian *package* ini adalah untuk mempermudah pencarian kelas yang diinginkan. Berikut adalah 4 bagian utama dari *package* tersebut:

1. Neural Network

*Package* ini berisi kelas-kelas yang berfungsi sebagai mesin atau fitur utama pembelajaran algoritma JST propagasi balik pada perangkat lunak ini.

2. NNBuilder

Kelas yang berada pada *package* ini berisi representasi kelas (*object*) dari tiap jaringan (jaringan 1, jaringan 2, dan jaringan 3). Dari tiap *object* tersebut dapat memanggil fungsi-fungsi pada kelas yang berada di *package neural network*.

Selain itu, juga terdapat kelas yang berguna untuk membaca dan menulis data dari dan ke Microsoft Excel.

3. Mapping

*Package* mapping berisi kelas-kelas yang berfungsi sebagai proses pemetaan data latihan.

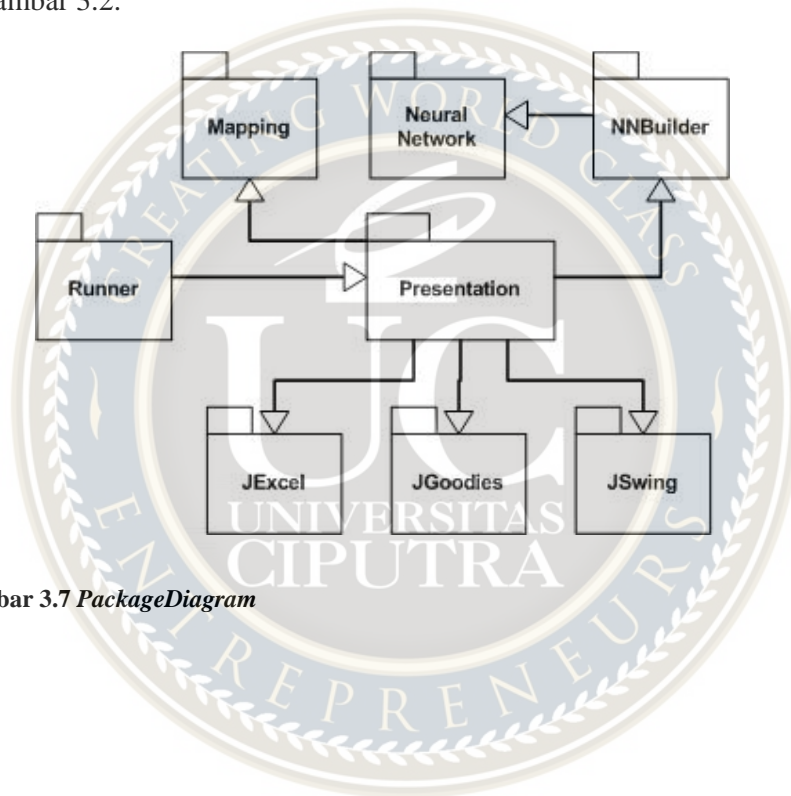
4. Presentation

*Package presentation* adalah *package* yang berisikan kelas untuk membuat tampilan perangkat lunak.

## 5. Runner

*Runner* merupakan *package* yang hanya memiliki 1 *class* yang berfungsi memanggil dan menampilkan tampilan antar muka perangkat lunak.

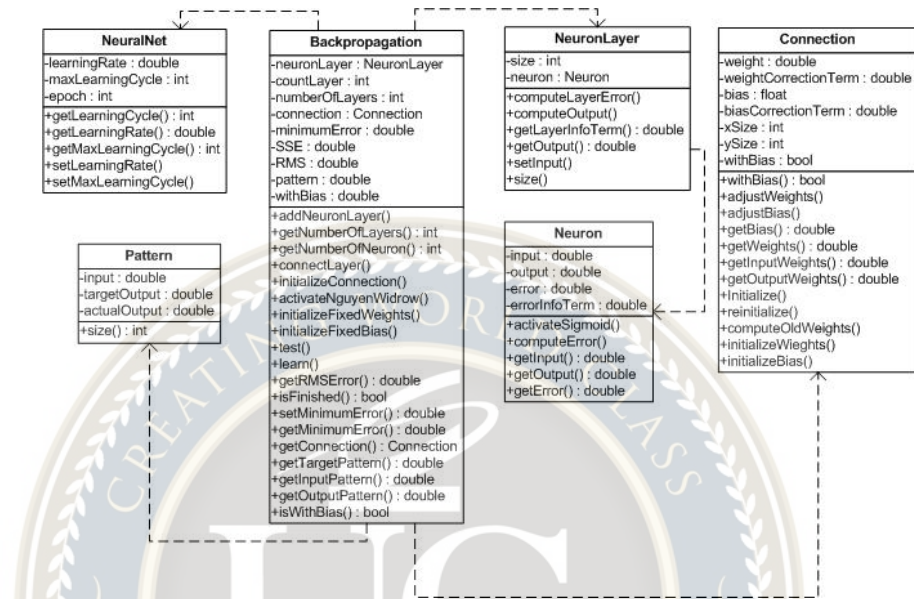
Ilustrasi dari *package* yang telah disebutkan diatas diterangkan pada gambar 3.2.



Gambar 3.7 PackageDiagram

### 3.2.5 Class Diagram

#### 3.2.5.1 Package Neural Network



Gambar 3.8 Package Neural Network

Dalam tabel berikut akan dijelaskan fungsi dari tiap kelas yang ada di package ini.

Tabel 3.6 Penjelasan fungsi kelas pada Package Neural Network

Nama Class	Keterangan
Backpropagation	Merupakan kelas untuk menjalankan fitur pembelajaran dan pengujian
Neuron	Merupakan kelas untuk melakukan fungsi penjumlahan dan fungsi aktifasi serta perhitungan error
NeuronLayer	Merupakan kelas untuk menyimpan objek neuron ke dalam beberapa lapisan
Connection	Merupakan kelas untuk menyimpan bobot dan bias sebuah jaringan
Pattern	Menyimpan pola atau pattern data latihan
NeuralNet	Merupakan sebuah <i>abstract class</i>

### 3.2.5.2 Package NNBuilder



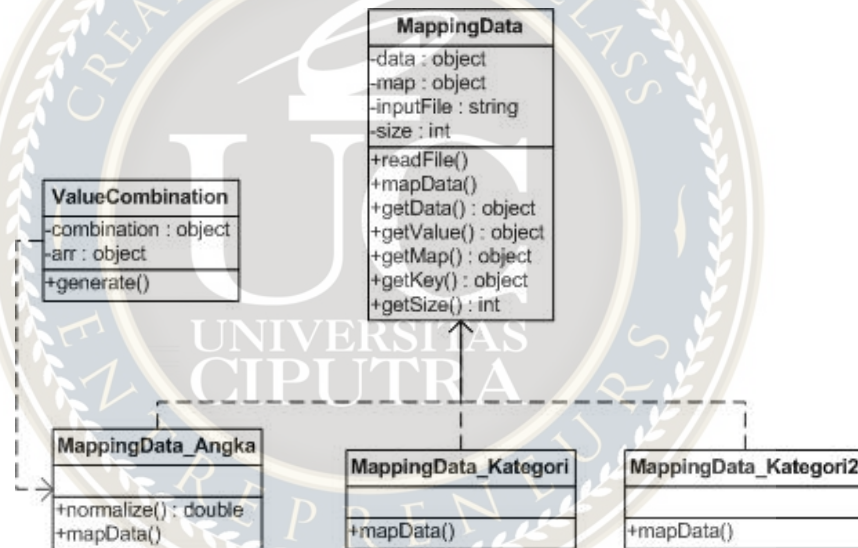
Gambar 3.9 Package NNBuilder

Tabel berikut menjelaskan fungsi dari tiap kelas yang ada pada package ini.

**Tabel 3.7** Penjelasan fungsi kelas pada *Package NNBuilder*

Nama Class	Keterangan
ExcelWrite	Merupakan kelas yang berfungsi untuk menulis data ke dalam file excel
ExcelRead	Merupakan kelas yang berfungsi untuk membaca data dari file excel
NN_Builder	Merupakan kelas yang memanggil fungsi-fungsi dari kelas-kelas yang berada di <i>package neural network</i>
NN_3	Merupakan kelas yang berfungsi melatih dan menguji pembelajaran pada jaringan

### 3.2.5.3 Package Mapping



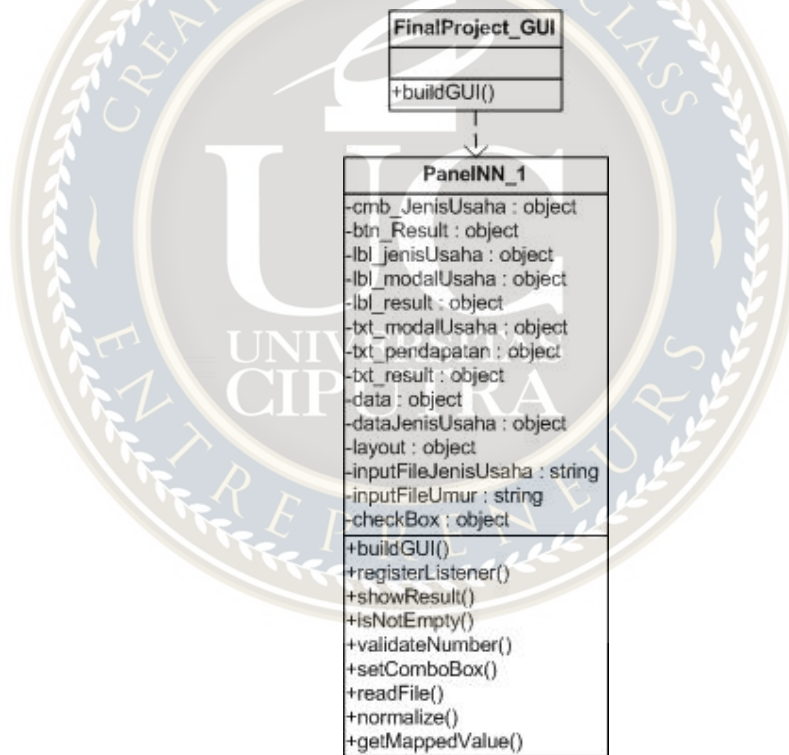
**Gambar 3.10** Package Mapping

Tabel berikut menjelaskan fungsi dari tiap kelas *package mapping*.

**Tabel 3.8** Penjelasan fungsi kelas pada *Package Mapping*

Nama Class	Keterangan
MappingData_Angka	Merupakan kelas yang berfungsi untuk memetakan data latihan yang berupa angka
MappingData_Kategori	Merupakan kelas yang berfungsi untuk memetakan data latihan kecamatan
MappingData_Kategori2	Merupakan kelas yang berfungsi untuk memetakan data latihan jenis ritel
MappingData	Merupakan <i>abstract class</i>
ValueCombination	Merupakan kelas untuk menghasilkan data kombinasi (jenis kelamin dan umur)

### 3.2.5.4 Package Presentation



**Gambar 3.11** *Package Presentation*

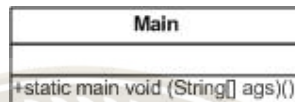
Tabel dibawah ini menerangkan tiap kelas yang berada di *package presentation*.



**Tabel 3.9** Penjelasan fungsi kelas pada *Package Presentation*

<b>Nama Class</b>	<b>Keterangan</b>
PanelNN_3	Merupakan kelas panel tampilan antar muka pada perangkat lunak ini.
FinalProject_GUI	Merupakan kelas untuk memanggil panel tampilan jaringan.

### 3.2.5.5 Package Runner



```
classDiagram
    class Main {
        +static main void (String[] args)()
    }
```

<b>Main</b>
+static main void (String[] args)()

**Gambar 3.12** *Package Runner*

Penjelasan untuk kelas main dituliskan pada tabel berikut.

**Tabel 3.10** Keterangan fungsi kelas pada *Package Runner*

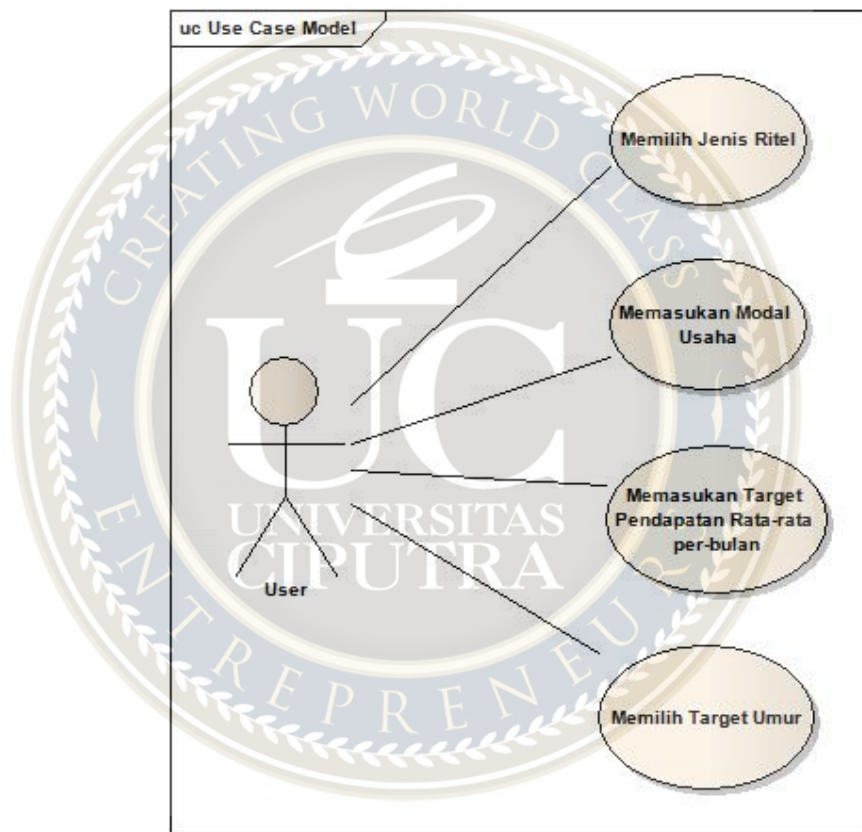
<b>Nama Class</b>	<b>Keterangan</b>
Main	Merupakan kelas yang menjalankan tampilan antar muka perangkat lunak

### 3.2.6 Use Case Diagram

Gambar 3.8 adalah use case diagram untuk jaringan 3. Langkah-langkah dari use case ini adalah sebagai berikut.

1. Start perangkat lunak
2. User memilih jenis ritel yang diinginkan
3. User memasukkan modal usaha yang digunakan
4. User memasukkan target pendapatan rata-rata per bulan
5. User memilih target umur yang diinginkan
6. User menekan tombol hasil
7. Jika modal dan pendapatan kosong atau kurang dari satu atau berisi tipe data String, sistem akan menampilkan sebuah peringatan.

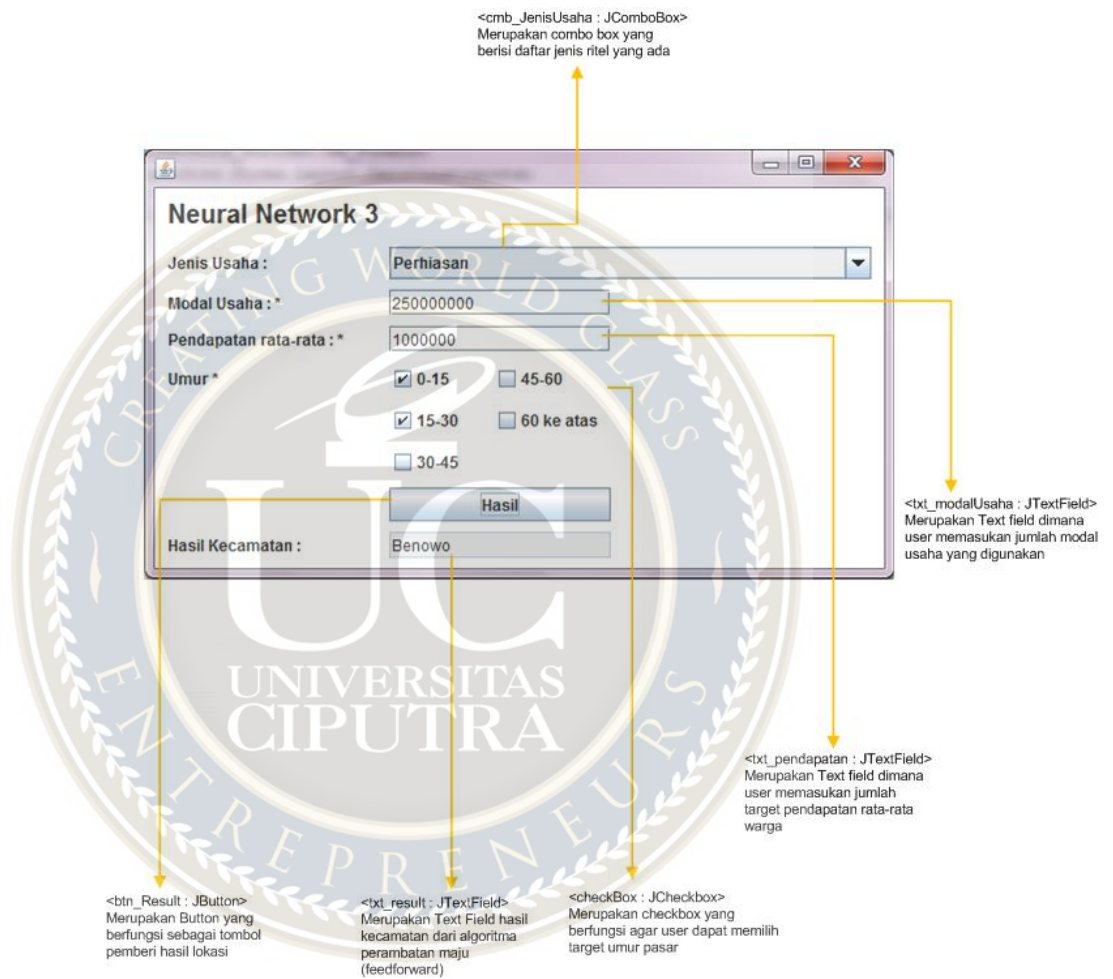
8. Jika user belum memilih target umur yang telah disediakan, maka sistem akan menampilkan sebuah peringatan.
9. Jika modal dan pendapatan serta target umur ada yang dipilih atau memenuhi persyaratan maka sistem akan menampilkan hasil lokasi (kecamatan).



**Gambar 3.13** Use case diagram Jaringan

### 3.3.7 Tampilan Antar Muka

Tampilan yang digunakan pada perangkat lunak ini ditunjukkan pada gambar dibawah ini.



Gambar 3.14 Tampilan antar muka