

**DETEKSI PENYAKIT GINJAL KRONIS  
MENGUNAKAN JARINGAN SYARAF TIRUAN  
*RADIAL BASIS FUNCTION***



**PROPOSAL TUGAS AKHIR**

**Disusun Sebagai Salah Satu Syarat  
Untuk Memperoleh Gelar Sarjana Komputer  
Pada Departemen Ilmu Komputer/ Informatika**

**Disusun Oleh:**

**MUHAMMAD LUQMAN FIKRI**

**24010313120043**

**DEPARTEMEN ILMU KOMPUTER/ INFORMATIKA  
FAKULTAS SAINS DAN MATEMATIKA  
UNIVERSITAS DIPONEGORO**

**2017**

## **Halaman Pengesahan**

Yang bertandatangan di bawah ini menyatakan bahwa Proposal Tugas Akhir yang berjudul:

**DETEKSI PENYAKIT GINJAL KRONIS MENGGUNAKAN JARINGAN  
SYARAF TIRUAN *RADIAL BASIS FUNCTION***

Dipersiapkan dan disusun oleh:

Nama : Muhammad Luqman Fikri

NIM : 24010313120043

Telah disahkan sebagai Proposal Tugas Akhir yang merupakan salah satu syarat  
untuk memperoleh gelar Sarjana Komputer.

Mengetahui,  
Ketua Departemen Ilmu Komputer/Informatika

Menyetujui,  
Pembimbing

Ragil Saputra, S.Si, M.Cs  
NIP 19801021 200501 1 003

Helmie Arif Wibawa, S.Si, M.Cs  
NIP 19780516 200312 1 001

## Daftar Isi

Halaman Pengesahan .....	ii
Daftar Isi .....	iii
Daftar Gambar .....	v
Daftar Tabel .....	vi
BAB I PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	3
1.3. Tujuan dan Manfaat .....	3
1.4. Ruang Lingkup .....	3
BAB II Tinjauan Pustaka .....	4
2.1. Ginjal .....	4
2.2. Penyakit Ginjal Kronis .....	4
2.2.1. Pemeriksaan Laboratorium .....	5
2.3. Jaringan Syaraf Tiruan .....	6
2.3.1. Komponen Jaringan Syaraf Tiruan .....	6
2.3.2. Arsitektur Jaringan Syaraf Tiruan .....	7
2.4. <i>Radial Basis Function</i> (RBF) .....	8
2.4.1. Arsitektur RBF .....	8
2.4.2. Algoritma RBF .....	9
2.5. Penerapan jaringan syaraf tiruan dalam penelitian lain .....	10
2.6. Evaluasi Kerja <i>Classifier</i> .....	11
2.6.1. <i>K-Fold Cross Validation</i> .....	11
2.6.2. <i>Confusion Matrix</i> .....	12
2.7. Model Pengembangan Perangkat Lunak <i>Waterfall</i> .....	13
2.8. Pemodelan Analisis .....	14
2.8.1. Pemodelan Data .....	15
2.8.2. Pemodelan Fungsional .....	16
2.9. Pengujian Perangkat Lunak .....	19
BAB III Metode Penelitian .....	20
3.1. Metodologi .....	20

3.1.1. Studi Pustaka.....	20
3.1.2. Eksperimental .....	20
3.2. Garis Besar Penyelesaian Masalah.....	20
3.4. Jadwal.....	23
Daftar Pustaka .....	24
Lampiran-Lampiran.....	26

## Daftar Gambar

Gambar 2.1 Struktur Neuron Jaringan Syaraf Tiruan.....	6
Gambar 2.2 Arsitektur <i>Single Layer Network</i> .....	7
Gambar 2.3 Arsitektur <i>Multi Layer Network</i> .....	8
Gambar 2.4 Arsitektur <i>Radial Basis Function Neural Network</i> .....	9
Gambar 2.5 Model Proses Waterfall (Sommerville, 2011) .....	13
Gambar 2.6 Relasi satu ke satu ( <i>one to one</i> ) .....	15
Gambar 2.7 Relasi satu ke banyak ( <i>one to many</i> ) .....	15
Gambar 2.8 Relasi banyak ke banyak ( <i>many to many</i> ) .....	16
Gambar 3.1 Garis Besar Penyelesaian Masalah .....	21

## Daftar Tabel

Tabel 2.1 Penelitian lain menggunakan metode jaringan syaraf tiruan.....	10
Tabel 2.2 Tabel <i>Confusion Matrix</i> .....	12
Tabel 2.3 Contoh format tabel SRS.....	14
Tabel 2.4Tabel Notasi Pemodelan Data .....	16
Tabel 2.5 Notasi yang digunakan untuk pemodelan fungsional.....	18
Tabel 3.1 Jadwal Kegiatan Penelitian.....	23

# **BAB I**

## **PENDAHULUAN**

Bab ini menyajikan latar belakang, rumusan masalah, tujuan dan manfaat, dan ruang lingkup dalam pembuatan penelitian mengenai Deteksi Penyakit Ginjal Kronis Menggunakan Metode Radial Basis Function Neural Network.

### **1.1. Latar Belakang**

Penyakit ginjal kronis merupakan keadaan dimana terjadi kerusakan pada ginjal seseorang. Ginjal merupakan organ penting bagi tubuh manusia. Ginjal berfungsi sebagai pengatur jumlah air dalam tubuh, pengatur tekanan darah, serta pengatur pembuatan sel darah merah dan lain sebagainya. Ginjal yang rusak tidak dapat menjaga kesehatan tubuh manusia. Ginjal tidak dapat melakukan penyaringan darah dan tidak dapat bekerja menjalankan fungsinya sehingga, tubuh akan merasakan sakit akibat tidak berfungsinya ginjal (National Kidney Foundation, 2013).

Pada akhir tahun 2004 angka kejadian gagal ginjal kronis diseluruh dunia meningkat sehingga mencapai jumlah 1.371.000 pasien yang menjalani terapi hemodialisis (Grassmann, Giobere, Moeller, & Brown, 2005). Di Amerika Serikat, insiden penyakit gagal ginjal kronik terjadi 268 kasus baru per satu juta populasi setiap tahunnya (Black & Hawks, 2005). Berdasarkan data tahun 2013 dengan menggunakan unit analisa individu menunjukkan bahwa secara nasional 0.2% penduduk Indonesia menderita penyakit ginjal kronis (Kemenkes, 2013).

Deteksi penyakit ginjal kronis sangat penting dilakukan karena manifestasi klinis dari penyakit ini cukup sulit untuk diketahui, hal ini dapat dibuktikan dengan masih banyaknya kejadian ginjal kronis yang sudah lambat dalam penanganan dan berujung kematian. Ketika ginjal kronis semakin lambat terdeteksi maka akan memerlukan biaya yang lebih besar dalam pengobatannya serta membutuhkan tenaga medis yang lebih ahli dalam penanganannya dengan peluang penyembuhan yang semakin kecil (Francesco, 2002).

Untuk melakukan deteksi terhadap suatu penyakit dalam dunia informatika terdapat beberapa metode yang dapat digunakan. Salah satu metode untuk melakukan deteksi terhadap penyakit dapat menggunakan model jaringan syaraf tiruan. Dalam penelitian Deteksi penyakit pengeroposan tulang (Mardianto & Pratiwi, 2008) menggunakan jaringan

syaraf tiruan, dalam penelitian tersebut memiliki tingkat akurasi yang tinggi dalam deteksi penyakit berbasis data. Dengan adanya model jaringan syaraf tiruan ini diharapkan dapat membantu petugas medis dalam menetapkan diagnosa terhadap terhadap pasien menggunakan data laboratorium dari pasien sehingga dapat dilakukan pemeriksaan lebih lanjut menggunakan prosedur medis yang lebih tinggi agar dapat menghambat tingkat kerusakan ginjal

Jaringan syaraf tiruan memiliki berbagai macam jenis arsitektur. Arsitektur dalam jaringan syaraf tiruan adalah bentuk hubungan antara masukan, keluaran dan lapisan tersembunyi. Beberapa arsitektur jaringan syaraf tiruan antara lain : *Single Layer Perceptron*, *Multi Layer Perceptron*, *Backpropagation*, *Radial Basis Function*, dan lain sebagainya. Salah satu arsitektur dari jaringan syaraf tiruan adalah arsitektur *radial basis function* (Kusumadewi, 2003). *Radial basis function* merupakan arsitektur jaringan syaraf tiruan yang menggunakan fungsi aktivasi radial basis (Haykin, 2009).

Algoritma *radial basis function* telah digunakan dalam beberapa penelitian. Beberapa contoh penelitian menggunakan *radial basis function* dalam deteksi penyakit sebagai berikut : klasifikasi penderita kanker payudara menggunakan data *micro array* (Hanim dan Puteh, 2012) dan diagnosa serangan jantung berdasarkan data medik pasien (Mashail dkk, 2014). Beberapa penelitian menunjukkan bahwa performa *Radial Basis Function* lebih baik daripada beberapa metode lain. Dalam penelitian Peramalan cuaca (Santhanam dan Subhajini, 2011) yang menggunakan metode *backpropagation* dan *Radial Basis Function* perbandingan hasil klasifikasi antara algoritma *backpropagation* dengan *Radial Basis Function* didapatkan hasil berupa tingkat akurasi antara lain 81.99% dan 88.49%. Selain itu dalam penelitian lain yang membandingkan performa antara *Radial Basis Function* dengan *backpropagation* (Sug, 2009) memiliki hasil akurasi 58.5% dan 61.9%. Penelitian lain yang dilakukan (Riama, 2015) dalam klasifikasi Berat Bayi Lahir Rendah dilakukan perbandingan dengan metode klasifikasi regresi logistik biner dan *Radial Basis Function* didapatkan hasil akurasi 81.7% dan 92.96%.

Dalam tugas akhir ini akan melakukan penelitian untuk mencari model jaringan syaraf tiruan menggunakan *Radial Basis Function* pada data penyakit ginjal kronis.



## **1.2. Rumusan Masalah**

Berdasarkan latar belakang yang telah dituliskan maka dapat disusun rumusan masalah yaitu :

1. Bagaimana model jaringan syaraf tiruan dengan arsitektur *Radial Basis Function* pada deteksi penyakit ginjal kronis ?
2. Bagaimana kinerja model jaringan syaraf tiruan dengan arsitektur *Radial Basis Function* pada deteksi penyakit ginjal kronis ?

## **1.3. Tujuan dan Manfaat**

Tujuan yang dicapai dari penelitian ini adalah menghasilkan model jaringan syaraf tiruan *Radial Basis Function* yang dapat melakukan deteksi penyakit ginjal kronis. Adapun manfaat dilakukan penelitian Tugas Akhir ini adalah sebagai berikut :

1. Model yang dihasilkan diharapkan dapat digunakan untuk membantu petugas medis dalam mendukung diagnosa terhadap pasien penyakit ginjal kronis.
2. Menambah wawasan mengenai penerapan teknik jaringan syaraf tiruan pada deteksi dini penyakit ginjal kronis.

## **1.4. Ruang Lingkup**

Dalam penyusunan penelitian ini, diberikan ruang lingkup yang jelas agar pembahasan lebih terarah dan tidak menyimpang dari tujuan penulisan. Adapun ruang lingkup dalam penelitian ini adalah sebagai berikut:

1. Data set pasien ginjal kronis diambil dari Alagappa University ([https://archive.ics.uci.edu/ml/datasets/Chronic\\_Kidney\\_Disease](https://archive.ics.uci.edu/ml/datasets/Chronic_Kidney_Disease)) yaitu sejumlah 400 data.
2. Metode perhitungan yang digunakan adalah jaringan syaraf tiruan dengan arsitektur *Radial Basis Function*
3. *Output* dari sistem ini klasifikasi berupa 2 buah kelas yaitu kelas ya atau tidak.

## **BAB II**

### **Tinjauan Pustaka**

Bab ini dipaparkan mengenai tinjauan pustaka, serta menyajikan teori-teori yang membantu dalam pembuatan model jaringan syaraf tiruan *radial basis function* pada deteksi penyakit ginjal kronis.

#### **2.1. Ginjal**

Ginjal adalah organ utama untuk membuang produk sisa metabolisme yang tidak diperlukan oleh tubuh. Produk ini meliputi: urea (dari metabolisme asam amino), kreatinin (dari kreatin otot), asam urat (dari asam nukleat), produk akhir dari pemecahan hemoglobin (bilirubin) dan metabolit dari berbagai hormon. Ginjal juga membuang toksin dan zat asing lainnya yang diproduksi oleh tubuh dan pencernaan seperti pestisida, obat-obatan dan makanan tambahan (Guyton & Hall, 2007). Fungsi utama ginjal adalah menyingkirkan buangan metabolisme normal, mengekskresi xenobiotik dan metabolitnya dan fungsi non ekskretori. Urin adalah jalur utama ekskresi toksikan sehingga ginjal mempunyai volume aliran darah yang tinggi, mengkonsentrasikan toksikan pada filtrat, membawa toksikan melalui sel tubulus (Lu, 1994).

#### **2.2. Penyakit Ginjal Kronis**

Penyakit ginjal kronis adalah suatu keadaan dimana ginjal sudah tidak mampu mempertahankan lingkungan internal yang konsisten dengan kehidupan dan pemulihan fungsi sudah tidak dimulai (Long, 1996).

Secara laboratorik gagal ginjal dapat dinilai dari tes klirens kreatinin yang dianggap mendekati laju filtrasi glomerulus/GFR. Sesuai dengan nilai TKK, gagal ginjal kronik dibagi sebagai berikut :

- 100 – 75 ml/mol : Insufisiensi berkurang
- 75 – 26 ml/mol : Insufisiensi ginjal kronik
- 25 – 0 ml/mol : Gagal ginjal kronik
- < 5 ml/mol : Gagal ginjal terminal

### 2.2.1. Pemeriksaan Laboratorium

Untuk mendapatkan diagnosis yang tepat pada penyakit ginjal sudah tentu diperlukan kelengkapan data-data yang saling mendukung satu dengan lainnya. Untuk itu diperlukan pemeriksaan penunjang yang tepat dan terarah sehingga diagnosis penyakit ginjal yang tepat dapat dipenuhi.

Pemeriksaan penunjang pada penyakit gagal ginjal kronik menurut Doenges (2000) antara lain :

#### 1. Urine

- a. Volume, biasanya kurang dari 400 ml/24 jam (oliguria) atau urine tidak ada (anuria).
- b. Warna, secara abnormal urine keruh mungkin disebabkan oleh pus, bakteri, lemak, pertikel koloid, fosfat atau urat.
- c. Berat jenis urine, kurang dari 1,015 (menetap pada 1,010 menunjukkan kerusakan ginjal berat)
- d. Klirens kreatinin, mungkin menurun
- e. Natrium, lebih besar dari 40 meq/L karena ginjal tidak mampu mereabsorpsi natrium.
- f. Protein, derajat tinggi proteinuria (3-4 +) secara kuat menunjukkan kerusakan glomerulus.

#### 2. Darah

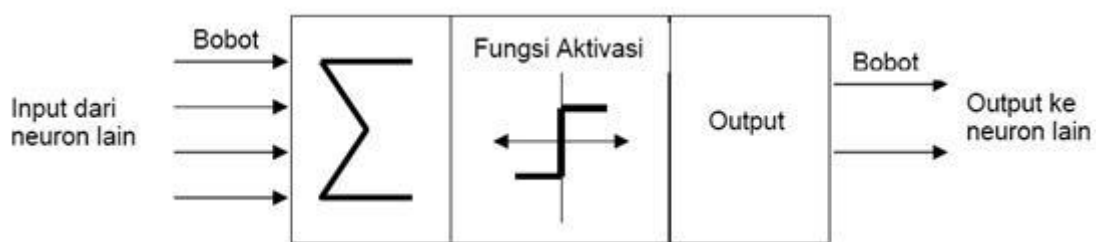
- a. Hitung darah lengkap, Hb menurun pada adanya anemia, Hb biasanya kurang dari 7-8 gr
- b. Sel darah merah, menurun pada defisiensi eritropoetin seperti azotemia.
- c. GDA, pH menurun, asidosis metabolik (kurang dari 7,2) terjadi karena kehilangan kemampuan ginjal untuk mengekskresi hydrogen dan amonia atau hasil akhir katabolisme protein, bikarbonat menurun, PaCO<sub>2</sub> menurun.
- d. Kalium, peningkatan berhubungan dengan retensi sesuai perpindahan seluler (asidosis) atau pengeluaran jaringan)
- e. Magnesium fosfat meningkat
- f. Kalsium menurun
- g. Protein (khusus albumin), kadar serum menurun dapat menunjukkan kehilangan protein melalui urine, perpindahan cairan, penurunan pemasukan atau sintesa karena kurang asam amino esensial.
- h. Osmolaritas serum: lebih besar dari 285 mOsm/kg, sering sama dengan urin.

## 2.3. Jaringan Syaraf Tiruan

Jaringan syaraf tiruan merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk menstimulasikan proses pembelajaran pada otak manusia tersebut (Kusumadewi, 2003).

### 2.3.1. Komponen Jaringan Syaraf Tiruan

Ada beberapa tipe jaringan syaraf, namun demikian, hampir semuanya memiliki komponen-komponen yang sama. Seperti halnya otak manusia, jaringan syaraf juga terdiri dari beberapa *neuron*, dan ada hubungan antara *neuron-neuron* tersebut. *Neuron-neuron* tersebut akan mentransformasikan informasi yang diterima melalui sambungan menuju ke *neuron-neuron* yang lain. pada jaringan syaraf, hubungan ini disebut dengan nama bobot. informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut. Gambar 2.1 menunjukan struktur *neuron* pada jaringan syaraf (Kusumadewi, 2003).



Gambar 2.1 Struktur Neuron Jaringan Syaraf Tiruan

*Neuron* buatan ini sebenarnya mirip dengan sel *neuron* biologis. *Neuron-neuron* buatan tersebut bekerja dengan cara yang sama pula dengan *neuron-neuron* biologis. Informasi (disebut dengan : *input*) akan dikirim ke *neuron* dengan bobot kedatangan tertentu. *Input* ini akan diproses oleh suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang datang. Hasil penjumlahan ini kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivitas setiap *neuron*. Apabila *input* tersebut melewati suatu nilai ambang tertentu, maka *neuron* tersebut akan diaktifkan. Apabila *neuron* tersebut diaktifkan, maka *neuron* tersebut akan mengirimkan output melalui bobot-bobot *output-nya* ke semua *neuron* yang berhubungan dengannya (Kusumadewi, 2003).

Pada jaringan syaraf, *neuron-neuron* akan dikumpulkan dalam lapisan-lapisan (*layer*) yang disebut dengan lapisan *neuron* (*neuron layers*). Biasanya *neuron-neuron* pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya (kecuali lapisan

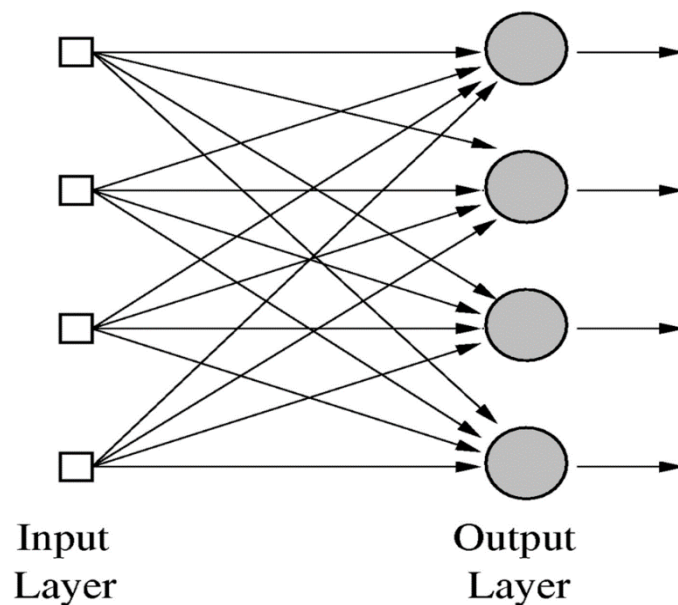
*input* dan lapisan *output*). Informasi yang diberikan pada jaringan syaraf akan dirambatkan lapisan ke lapisan, mulai dari lapisan input sampai ke lapisan *output* melalui lapisan yang lainnya, yang sering dikenal dengan nama lapisan tersembunyi (*hidden layer*). Tergantung pada algoritma pembelajarannya, bisa jadi informasi tersebut akan dirambatkan secara mundur pada jaringan (Kusumadewi, 2003).

### 2.3.2. Arsitektur Jaringan Syaraf Tiruan

Menurut Kusumadewi ada beberapa arsitektur syaraf, antara lain :

#### 1. Jaringan dengan lapisan tunggal (*single layer net*)

Jaringan dengan lapisan tunggal hanya memiliki satu lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima input kemudian secara langsung akan mengolahnya menjadi *output* tanpa harus melalui lapisan tersembunyi (gambar 2.2) pada gambar 2.2 tersebut, lapisan *input* memiliki 4 *neuron*, sedangkan pada lapisan *output* memiliki 4 *neuro*. *Neuron-neuron* pada lapisan kedua lapisan saling berhubungan. Seberapa besar hubungan antara 2 *neuron* ditentukan oleh bobot yang bersesuaian. Semua unit *input* akan dihubungkan dengan setiap unit *output*.

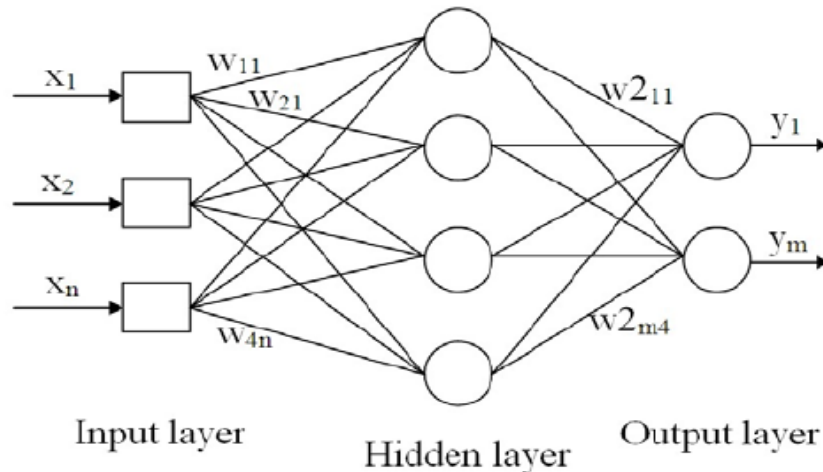


Gambar 2.2 Arsitektur *Single Layer Network*

#### 2. Jaringan dengan banyak lapisan

Jaringan dengan banyak lapisan memiliki 1 atau lebih lapisan yang terletak diantara lapisan *input* dan *output* (memiliki 1 atau lebih lapisan tersembunyi), seperti terlihat pada gambar 2.3. Umumnya, ada lapisan bobot-bobot yang terletak antara 2 lapisan yang

bersebelahan. Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit daripada lapisan dengan lapisan tunggal, tentu saja dengan pembelajaran yang lebih rumit. Namun demikian, pada banyak kasus, pembelajaran pada jaringan dengan banyak lapisan ini lebih sukses dalam menyelesaikan masalah.



Gambar 2.3 Arsitektur *Multi Layer Network*

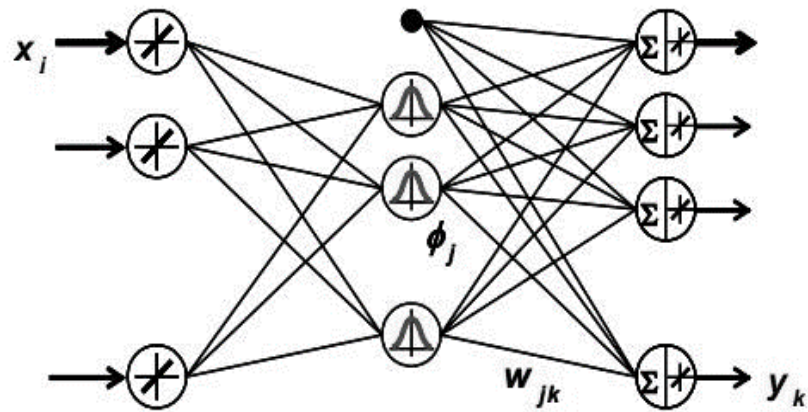
#### 2.4. *Radial Basis Function (RBF)*

Model jaringan saraf tiruan Radial Basis Function merupakan salah satu bentuk multilayer perceptron yang memperbaiki nilai-nilai bobot, nilai tengah, dan jarak antar data agar mengurangi kesalahan yang terjadi pada keluaran jaringan. Pada model ini, Jaringan Saraf Tiruan menggunakan fungsi aktivasi basis (Gaussian) pada lapisan tersembunyi (David, 2005).

Dalam penerapannya untuk mendapatkan model jaringan saraf tiruan Radial Basis Function terbaik diperlukan kombinasi yang tepat antara jumlah variabel masukan, jumlah node (cluster) pada unit lapisan tersembunyi, nilai tengah serta standar deviasi (skala atau lebar data) dari variabel masukan pada setiap node, yang berimplikasi pada jumlah parameter yang optimal (David, 2005).

##### 2.4.1. *Arsitektur RBF*

Jaringan saraf tiruan Radial Basis Function tersusun dalam tiga lapisan (lapisan masukan, lapisan tersembunyi, dan lapisan keluaran). Jaringan saraf tiruan Radial Basis Function dapat digambarkan seperti pada Gambar 2.4 (David, 2005).



Gambar 2.4 Arsitektur *Radial Basis Function Neural Network*

Neuron-neuron lapisan tersembunyi pada jaringan saraf tiruan Radial Basis Function melakukan transformasi nonlinear dan memetakan masukan pada neuron masukan ke neuron tersembunyi tanpa parameter yang diubah-ubah. Selanjutnya neuron-neuron di lapisan keluaran melakukan kombinasi linear terhadap neuron tersembunyi dengan parameter yang diubah-ubah yakni bobot hubungan antara neuron di lapisan tersembunyi dengan neuron-neuron di lapisan keluaran. Nonlinearitas dalam jaringan saraf tiruan Radial Basis Function dapat dipilih dari beberapa fungsi nonlinear yang ada (David, 2005).

#### 2.4.2. Algoritma RBF

Menurut Kusumadewi (2004), algoritma perhitungan RBF adalah sebagai berikut :

1. Menentukan titik pusat secara acak berdasarkan data.
2. Mencari jarak Euclidean dari titik pusat dengan data.

$$D = \sqrt{\sum_{j=1}^n (x_{ij} - x_{kj})^2}$$

3. Mencari nilai aktivasi dari D.

$$\varphi = e^{-(D_{ik})^2}$$

4. Mencari bobot lapisan dan bobot bias dengan menggunakan persamaan matriks berikut :

$$G W = T$$

$$G = \begin{bmatrix} \varphi_{11} & \cdots & \varphi_{1n} & 1 \\ \vdots & \ddots & \vdots & 1 \\ \varphi_{n1} & \cdots & \varphi_{nn} & 1 \\ 1 & 1 & \cdots & 1 \end{bmatrix}$$

$$W = \begin{bmatrix} w_{11} \\ \vdots \\ w_{n1} \\ b \end{bmatrix}$$

$$T = \begin{bmatrix} T_{11} \\ \vdots \\ T_{n1} \end{bmatrix}$$

Sehingga :

$$W = G^{-1}T$$

Dimana :

W = bobot

G = nilai aktivasi masukan

T = Target

5. Pengujian dapat menggunakan rumus berikut dimana Y merupakan keluaran :

$$Y_i = \sum_{k=1}^n \varphi_{ik} w_{ik} + b$$

## 2.5. Penerapan jaringan syaraf tiruan dalam penelitian lain

Penerapan jaringan syaraf telah banyak dilakukan oleh banyak peneliti. Beberapa contoh penelitian menggunakan jaringan syaraf tiruan adalah deteksi penyakit, prediksi dengan data time series, klasifikasi data dll. Objek penelitian yang diteliti pun beragam, ada deteksi penyakit dengan berbasis data, deteksi penyakit dengan berbasis citra, serta klasifikasi dengan objek berupa suara. Pada tabel 2.1 menampilkan penelitian lain menggunakan metode jaringan syaraf tiruan.

Tabel 2.1 Penelitian lain menggunakan metode jaringan syaraf tiruan

Penelitian	Metode	Objek	Hasil
Is mardianto dan Dian Pratiwi (2008)	Jaringan syaraf tiruan (backpropagation)	Data set penyakit pengeroposan tulang belakang	Kelas ya atau tidak penyakit pengeroposan tulang belakang
	Radial Basis		
Ferry Tan dkk (2012)	Function dengan Pembelajaran Hybrid	Data time series harga Saham	Prediksi harga saham



Penelitian	Metode	Objek	Hasil
Sherif E. Hussein dkk. (2013)	Wavelet anlysis dan Jaringan syaraf tiruan	Data set citra mata	Kelas ya atau tidak adanya penyakit ginjal
Tommaso Di Noia,dkk. (2013)	Jaringan syaraf tiruan (MLP)	Data set penyakit ginjal stadium akhir	Kelas ya atau tidak penyakit ginjal stadium akhir
Istatik Rohmana. (2014)	Jaringan syaraf tiruan (backpropagation) dan Naïve bayes	Data set penyakit Stroke	Kelas ya atau tidak penyakit Stroke
Tangguh Gradhianta dkk (2015)	Radial Basis Function	Data set musik berbagai macam genre	Klasifikasi genre music
Yohanes Tanjung Sarwono	K-means, Radial Basis Function	Data set citra otak	Kelas ya atau tidak kelainan pada otak
Metode yang diajukan	Radial Basis Function	Data set penyakit ginjal kronis	Kelas ya atau tidak penyakit ginjal kronis

## 2.6. Evaluasi Kerja Classifier

Ukuran kinerja dari model pada test set seringkali berguna karena ukuran tersebut memberikan estimasi yang tidak bias dari *error* generalisasinya. Akurasi dari tingkat *error* yang dihitung dari data set pengujian dapat digunakan untuk membandingkan kinerja dari *classifier-classifier* pada domain yang sama. Evaluasi kinerja *classifier* digunakan untuk mengevaluasi tingkat akurasi pada proses klasifikasi.

### 2.6.1. K-Fold Cross Validation

*K-Fold Cross Validation* merupakan salah satu metode yang umumnya digunakan untuk mengevaluasi kinerja classifier. Metode *K-Fold Cross Validation* dilakukan dengan membagi dataset secara acak menjadi *k* himpunan bagian (Kohavi, 1995). *K-Fold Cross Validation* melakukan iterasi sebanyak *k* kali untuk data pelatihan dan pengujian. Setiap

iterasi, satu subset digunakan untuk pengujian sedangkan subset sisanya digunakan untuk pelatihan. Metode *K-Fold Cross Validation* berguna untuk memvalidasi keakuratan sebuah prediksi atau klasifikasi terhadap suatu data yang belum muncul dalam dataset.

Dataset dibagi menjadi  $k$  subset secara acak yang masing-masing subset memiliki jumlah instance dan perbandingan jumlah kelas yang sama. Pembagian data ini digunakan pada proses iterasi klasifikasi. Iterasi dilakukan sesuai nilai  $k$ . Setiap iterasi satu subset digunakan untuk pengujian sedangkan subset-subset lainnya digunakan untuk pelatihan. Kelebihan dari metode ini adalah data pelatihan digunakan lebih banyak sehingga dapat memberikan generalisasi yang lebih baik. Metode *K-Fold Cross Validation* pada umumnya menggunakan  $k = 10$ . Hasil uji coba metode ini dengan  $k = 10$  menghasilkan rata-rata akurasi yang cukup tinggi yaitu 97% lebih baik dibandingkan metode Naïve Bayesian yang menghasilkan rata-rata akurasi 96.24% (Zamani dkk, 2012).

### 2.6.2. *Confusion Matrix*

Evaluasi dari kinerja klasifikasi didasarkan pada banyak test record yang diprediksi secara benar dan secara tidak benar oleh sebuah model. Test record ini ditabulasikan dalam sebuah tabel yang dikenal sebagai confusion matrix. Confusion matrix berisi informasi tentang klasifikasi aktual dan prediksi yang dilakukan oleh sistem klasifikasi (Kohavi & Provost, 1998). Kinerja sistem tersebut umumnya dievaluasi dengan menggunakan data dalam matriks.

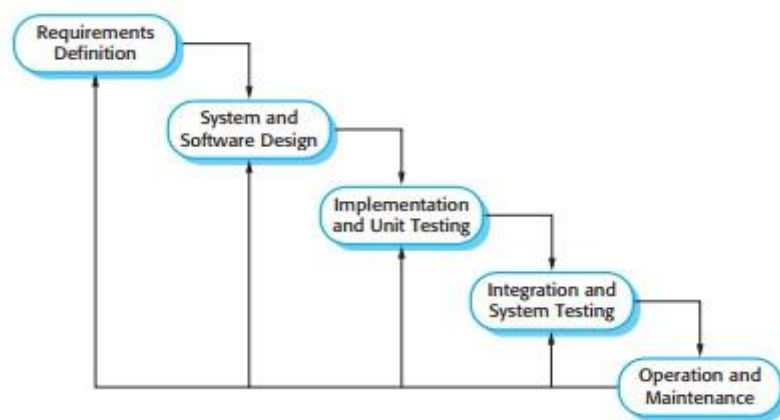
Tabel 2.2 Tabel *Confusion Matrix*

Actual Class	Predicted Class	
	Kelas = 1	Kelas = 2
Kelas = 1	$K_{11}$	$K_{12}$
Kelas = 2	$K_{21}$	$K_{22}$

Setiap entri  $C_{ij}$  dalam tabel ini menyatakan banyaknya record dari kelas  $i$  yang diprediksi menjadi kelas  $j$ . Sebagai contoh,  $C_{01}$  adalah banyaknya record dari kelas 0 yang secara tidak benar diprediksi sebagai kelas 1. Informasi dalam confusion matrix diperlukan untuk menentukan kinerja model klasifikasi. Ringkasan informasi ini memberikan nilai untuk membandingkan kinerja dari model-model yang berbeda.

## 2.7. Model Pengembangan Perangkat Lunak *Waterfall*

Sommerville (2011) menuturkan, bahwa model proses *waterfall* merupakan model pengembangan perangkat lunak pertama yang ada, dimana diturunkan dari proses rekayasa sistem. Tahapan perancangan dan pembuatan sistem dengan model *waterfall* menurut Sommerville (2011) dapat dilihat pada Gambar 2.5. Hasil dari setiap tahapan adalah dokumen, dimana tahapan selanjutnya hanya dapat dikerjakan jika tahapan sebelumnya telah selesai, dikarenakan tahapan setelahnya memerlukan hasil dari tahapan yang sedang dikerjakan.



Gambar 2.5 Model Proses Waterfall (Sommerville, 2011)

Proses yang dilakukan dengan model *waterfall* menurut Sommerville (2011) adalah sebagai berikut :

### 1. *Requirement Definition*

Merupakan langkah dimana *developer* bertemu dengan klien, menentukan batasan, tujuan, dan apa yang dilakukan oleh perangkat lunak, serta mengidentifikasi *requirement* yang dibutuhkan dari perangkat lunak. Menurut Sommerville (2011), spesifikasi sistem yang telah didapatkan didokumentasikan dalam bentuk *System Requirement Specification* (SRS).

SRS adalah dokumen yang berisi deskripsi lengkap mengenai kemampuan *software*. SRS diperlukan karena banyak kesalahan yang timbul pada tahap *requirement* dan tidak terdeteksi sejak dini serta untuk menghemat biaya perbaikan. Standar penulisan SRS yang digunakan adalah IEEE/ANSI 830-1998. Karakteristik dari SRS adalah benar sesuai spesifikasi yang diinginkan, lengkap, tidak ambigu, bisa diverifikasi, bisa dimodifikasi,

dapat ditelusuri, konsisten, *testability*, jelas, dan mungkin. Setiap fungsional *requirement* dalam SRS diberikan *unique identifier label* (Agarwal, 2009). Oleh karena itu fungsional SRS dapat disajikan dengan format tabel seperti pada Tabel 2.3.

Tabel 2.3 Contoh format tabel SRS

SRS-ID	Deskripsi
SRS-XXXX-F-YY	.....

Keterangan :

SRS : *Software Requirement Specification*  
 XXXX : *Nickname* sistem yang dibangun  
 FXX : F adalah fungsional dapat *optional* berupa NF (Non Fungsional)  
 YY : Nomor urut SRS

## 2. System and Software Design

Pada tahapan ini, dilakukan proses identifikasi dan menggambarkan abstraksi mendasar dari sistem perangkat lunak serta hubungan didalamnya.

## 3. Implementation and Unit Testing

Pada tahapan ini dilakukan pengembangan perangkat lunak serta pengujian apakah sistem telah memenuhi spesifikasi.

## 4. Integration and System Testing

Setiap unit bagian dari sistem digabungkan kemudian dilakukan pengujian sebagai suatu kesatuan, untuk memastikan kebutuhan sistem telah terpenuhi. Setelah dilakukan pengujian sistem diberikan kepada pelanggan.

## 5. Operation and Maintenance

Merupakan tahapan yang paling lama, dimana sistem digunakan dalam keperluan sehari-hari. *Maintenance* melibatkan membenarkan kesalahan yang belum teridentifikasi, serta menambahkan kemampuan dari sistem jika ditemukan *requirement* baru.

## 2.8. Pemodelan Analisis

Pemodelan analisis merupakan gabungan dari beberapa model yang pertama kali merepresentasikan sistem secara teknis. Terdapat dua pemodelan, yaitu pemodelan data dan pemodelan fungsional.

### 2.8.1. Pemodelan Data

Mentransformasi model domain informasi yang dibuat selama analisis ke dalam struktur data yang akan diperlukan untuk mengimplementasi perangkat lunak. Objek dan hubungannya digambarkan dalam bentuk ERD (*Entity Relationship Diagram*). Keluaran yang dihasilkan adalah struktur basis data (Gelinas, 1990).

ERD merupakan suatu *graph* yang menyajikan entitas, atribut dan hubungannya yang bertujuan untuk mengetahui hubungan antar entitas.

Pada dasarnya ada tiga simbol yang digunakan, yaitu (Gelinas, 1990):

#### 1. Entitas

Entitas merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari entitas ini biasanya digambarkan dengan persegi panjang. Properti dari entitas antara lain: nama entitas, daftar atribut, dan *primary key*.

#### 2. Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Simbol dari relasi ini biasanya digambarkan dengan belah ketupat. Relasi yang terjadi di antara dua himpunan entitas (misalnya A dan B) dalam satu basis data yaitu:

##### a. Satu ke satu (*One to one*)

Hubungan satu ke satu yaitu setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B. Hubungan satu ke satu dapat dilihat pada gambar 2.6.



Gambar 2.6 Relasi satu ke satu (*one to one*)

##### b. Satu ke banyak (*One to many*)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A. Hubungan satu ke banyak dapat dilihat pada gambar 2.7.



Gambar 2.7 Relasi satu ke banyak (*one to many*)

c. Banyak ke banyak (*Many to many*)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B. Hubungan banyak ke banyak dapat dilihat pada gambar 2.8.



Gambar 2.8 Relasi banyak ke banyak (*many to many*)

### 3. Atribut

Properti yang dimiliki setiap entitas yang akan disimpan datanya. Berikut ini adalah tabel notasi pemodelan data menggunakan notasi Peter Chen dapat dilihat pada tabel 2.4.

Tabel 2.4 Tabel Notasi Pemodelan Data

No	Nama	Gambar	Deskripsi
1	Entitas		Entitas adalah sebuah objek yang dapat dibedakan dengan objek lain
2	Atribut		Atribut berfungsi untuk mendiskripsikan karakter entitas.
3	Relasi		Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda
4	Relation		Garis sebagai penghubung antara relasi dengan entitas, dan entitas dengan atribut
5	Kardinalitas	<div>1 : 1</div> <div>1 : N</div> <div>M : N</div>	Kardinalitas merupakan angka yang menunjukkan banyaknya kemunculan suatu objek terkait dengan kemunculan objek lain pada relasi

#### 2.8.2. Pemodelan Fungsional

Untuk memodelkan seluruh fungsi yang tercakup dalam sistem ini, digunakan *Data Context Diagram* (DCD) serta *Data Flow Diagram* (DFD). DCD dapat juga dikatakan

sebagai DFD Level-0. Untuk DFD merupakan penjabaran lebih lanjut dari DCD. DFD berguna untuk menggambarkan fungsi-fungsi yang mentransformasikan data, serta berguna untuk menggambarkan bagaimana data ditransformasikan pada perangkat lunak (Kadir, 2003).

DCD merupakan tingkatan tertinggi dalam diagram aliran data yang hanya memuat satu proses, dan menunjukkan sistem secara keseluruhan. Proses tersebut diberi nomor nol. Semua entitas eksternal yang ditunjukkan pada diagram konteks berikut aliran-aliran data utama menuju dan dari sistem. Diagram tersebut tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan, entitas-entitas eksternal serta aliran-aliran data menuju dan dari sistem diketahui oleh seorang analis dari wawancara dengan pengguna sebagai hasil analisis dokumen (Kadir, 2003).

DFD adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data di mana komponen-komponen tersebut, asal, tujuan, dan penyimpanan dari data tersebut. Kita dapat menggunakan DFD untuk dua hal utama, yaitu untuk membuat dokumentasi dari sistem informasi yang ada atau untuk menyusun dokumentasi untuk sistem informasi yang baru. DFD mempunyai empat komponen utama, yaitu *external entity*, *data flow*, proses, dan *data store*. Penjelasan dari masing-masing elemen tersebut adalah sebagai berikut (Kadir, 2003):

#### 1. *External Entity*

*External entity* digunakan untuk menggambarkan suatu entitas eksternal (bagian lain, sebuah perusahaan, seseorang atau sebuah mesin) yang dapat mengirim data atau menerima data dari sistem. Entitas ini disebut juga sumber atau tujuan data dan dianggap eksternal terhadap sistem yang sedang digambarkan. Setiap entitas diberi label dengan sebuah nama yang sesuai. Meskipun berinteraksi dengan sistem, namun dianggap di luar batas-batas sistem. Entitas-entitas tersebut harus diberi nama dengan suatu kata benda, entitas yang sama dapat digunakan lebih dari sekali atas suatu diagram aliran data tertentu untuk menghindari persilangan antara jalur-jalur aliran.

#### 2. *Data Flow* (Arus Data)

*Data flow* menunjukkan perpindahan data dari satu titik ke titik yang lain, dengan kepala tanda panah mengarah ke tujuan data. Karena sebuah tanda panah menunjukkan seseorang, tempat atau sesuatu, maka harus digambarkan dalam kata benda.

### 3. Proses

Proses digunakan untuk menunjukkan adanya proses transformasi. Proses - proses tersebut selalu menunjukkan suatu perubahan dalam di dalam atau perubahan data. Jadi, aliran data yang meninggalkan suatu proses selalu diberi label yang berbeda dari aliran data yang masuk. Proses-proses yang menunjukkan hal itu didalam sistem dan harus diberi nama menggunakan salah satu format berikut ini. Sebuah nama yang jelas memudahkan untuk memahami proses apa yang sedang dilakukan. Pemberian nama pada proses (Kadir, 2003):

- a. Menetapkan nama sistem secara keseluruhan saat menamai proses pada level yang lebih tinggi. Contoh: sistem kontrol inventaris.
- b. Menamai suatu subsistem utama, menggunakan nama-nama seperti: Sistem Pelaporan Inventaris atau Sistem Pelayanan Konsumen Internet.
- c. Menggunakan format kata kerja dan kata benda untuk proses-proses yang mendetail. Kata kerja yang menggambarkan jenis kegiatan yang seperti ini, misalnya menghitung, memverifikasi, menyiapkan, mencetak, atau menambahkan. Contoh-contoh nama proses yang lengkap adalah: menghitung pajak penjualan, memverifikasi status rekening konsumen, menyiapkan *invoice* pengapalan, dan menambah *record* inventaris.


### 4. Data Store

*Data store* digunakan untuk menunjukkan penyimpanan data. Penyimpanan data menandakan penyimpanan manual, seperti lemari *file* atau sebuah *file* atau basis data terkomputerisasi. Karena penyimpanan data mewakili seseorang, tempat atau sesuatu, maka diberi nama dengan sebuah kata benda. Penyimpanan data sementara seperti kertas catatan atau sebuah *file* komputer sementara tidak dimasukkan ke dalam diagram aliran data.


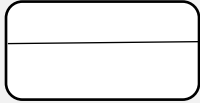


### 5. Split / Merge

*Split* digunakan untuk membagi aliran data menjadi beberapa aliran yang mengirimkan data ke tujuan yang berbeda, atau menggabungkan aliran data dari berbagai tujuan yang berbeda menjadi satu. Adapun notasi DFD yang digunakan adalah notasi Gane dan Sarson gambar beserta keterangan setiap notasi DFD dapat dilihat pada tabel 2.5.

Tabel 2.5 Notasi yang digunakan untuk pemodelan fungsional.

No	Notasi	Keterangan
1		<i>External Entity</i> (Entitas Eksternal)



No	Notasi	Keterangan
2		<i>Data Flow</i> (Aliran Data)
3		<i>Process</i> (Proses)
4		<i>Data Store</i>
5		<i>Split/Merge</i>

## 2.9. Pengujian Perangkat Lunak

Menurut IEEE, pengujian perangkat lunak adalah proses sistem operasi atau komponen menurut kondisi tertentu, pengamatan atau pencatatan hasil dan mengevaluasi beberapa aspek sistem atau komponen, proses analisis item perangkat lunak untuk mendeteksi perbedaan antara kondisi yang ada dengan yang diinginkan dan mengevaluasi fitur item perangkat lunak. Tujuan pengujian perangkat lunak (Peranginangin, 2006):

### 1. Tujuan langsung

- Identifikasi dan menemukan beberapa kesalahan yang mungkin ada dalam perangkat lunak yang diuji.
- Setelah perangkat lunak dibetulkan, diidentifikasi lagi kesalahan dan dites ulang untuk menjamin kualitas level penerimaan.
- Membentuk tes yang efisien dan efektif dengan anggaran dan jadwal yang terbatas.

### 2. Tujuan tidak langsung

Mengumpulkan daftar kesalahan untuk digunakan dalam daftar pencegahan kesalahan. Digunakan metode *Black box (functionality) testing* yaitu dengan mengidentifikasi kesalahan yang berhubungan dengan kesalahan fungsionalitas perangkat lunak yang tampak dalam kesalahan *output*. Definisi menurut IEEE, adalah pengujian yang mengabaikan mekanisme internal sistem atau komponen dan fokus semata-mata pada *output* yang dihasilkan yang merespon *input* yang dipilih dan kondisi eksekusi. Pengujian yang dilakukan untuk mengevaluasi pemenuhan sistem atau komponen dengan kebutuhan fungsional tertentu.

## **BAB III**

### **Metode Penelitian**

Bab ini menyajikan metodologi, garis besar penyelesaian masalah yang akan digunakan pada implementasi model Jaringan Syaraf Tiruan *Radial Basis Function* dalam Deteksi penyakit ginjal kronis.

#### **3.1. Metodologi**

Penelitian ini menggunakan metode studi kepustakaan dan eksperimental. Penjelasan lebih lanjut mengenai metodologi dijelaskan pada subbab selanjutnya.

##### **3.1.1. Studi Pustaka**

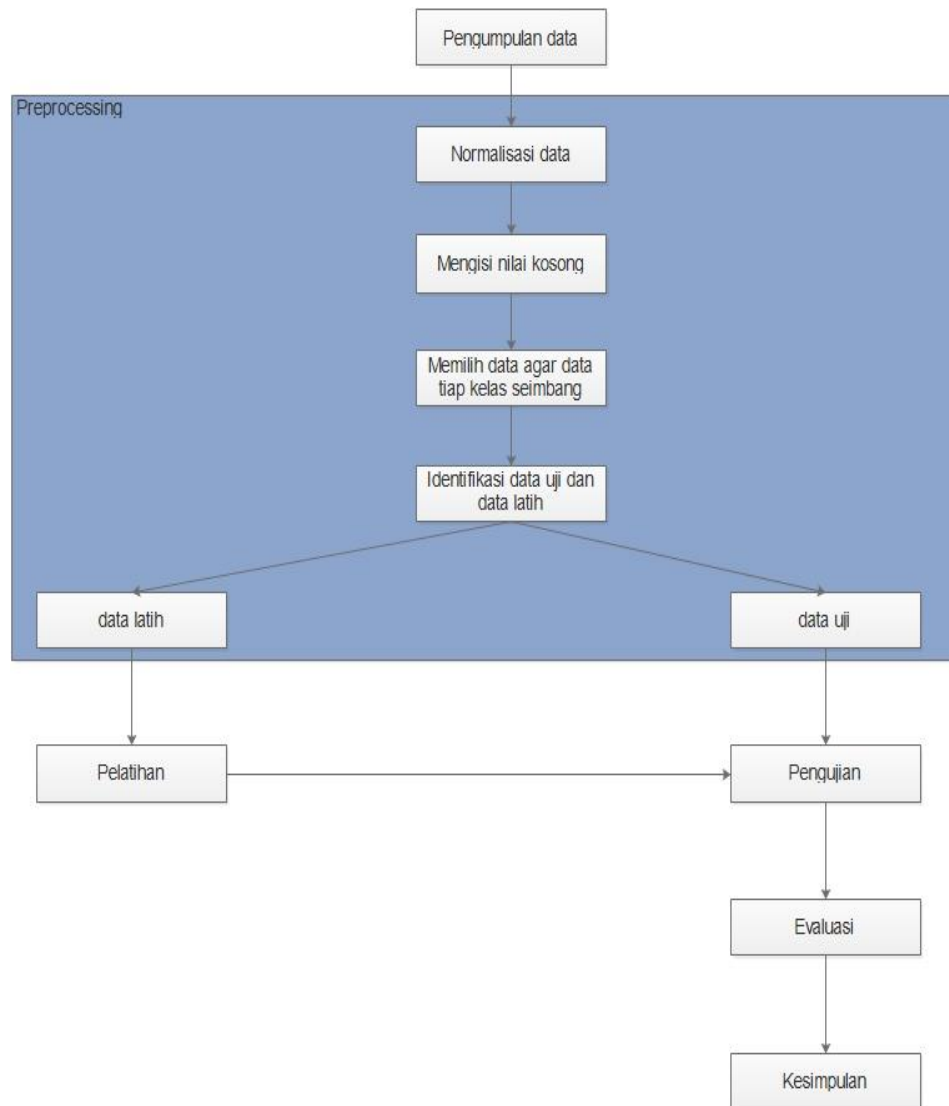
Studi pustaka merupakan langkah awal dalam penelitian ini. Metode studi pustaka digunakan untuk mengumpulkan dan mempelajari teori-teori dalam literatur antara lain buku, jurnal maupun artikel yang relevan dengan permasalahan yang diambil dalam tugas akhir ini.

##### **3.1.2. Eksperimental**

Eksperimen adalah observasi di bawah kondisi buatan (*artificial condition*) di mana kondisi tersebut dibuat dan diatur oleh si peneliti. Sedangkan penelitian eksperimental adalah penelitian yang dilakukan dengan mengadakan manipulasi terhadap objek penelitian serta adanya kontrol (Nazir, 2005). Pada penelitian ini membagi data menjadi data latih dan data uji yang mana kedua data digunakan dalam proses percobaan untuk menentukan bagaimana model jaringan yang terbaik untuk data kasus penyakit ginjal kronis

#### **3.2. Garis Besar Penyelesaian Masalah**

Pada sub bab ini akan dibahas penjelasan mengenai garis besar penyelesaian masalah beserta tahapan penelitian tugas akhir. Garis besar penyelesaian masalah disajikan dalam bentuk gambar alur pada Gambar 3.1.



Gambar 3.1 Garis Besar Penyelesaian Masalah

Berikut adalah penjelasan mengenai setiap langkah pada diagram garis besar penyelesaian masalah di atas :

### 1. Pengumpulan data

Pengumpulan data untuk penelitian ini dilakukan dengan mengambil data dari UCI. Data yang diambil adalah data set penyakit ginjal kronis dari Alagappa University

### 2. Normalisasi data

Setelah data didapatkan maka dilakukan normalisasi data. Normalisasi data adalah memetakan data kedalam range yang sama, range yang digunakan adalah [0 1].

### 3. Pengisian Nilai Kosong

Tahapan pengisian nilai kosong dilakukan untuk melengkapi *field* data yang kosong. Untuk melakukan pengisian nilai kosong menggunakan metode knn dan rata-rata data.

#### 4. Seleksi Data

Tahapan seleksi data dilakukan agar data masukan memiliki jumlah kelas yang seimbang. Sehingga, model yang dihasilkan dapat merepresentasikan kedua kelas secara seimbang.

#### 5. Pembentukan data latih dan data uji

Tahapan identifikasi data latih dan data uji dilakukan untuk membagi data penelitian menjadi data pelatihan dan data pengujian. Data latih (*training set*) digunakan dalam proses pembangunan model/ proses pelatihan, sementara data uji (*test set*) digunakan untuk validasi model yang berupa pengujian keakuratan dari model yang dilatih.

Untuk membagi seluruh data menjadi data latih dan data uji digunakan *k-fold cross validation*. *K-fold cross validation* menggunakan keseluruhan data yang ada sebagai *training set* dan *test set*. Menggunakan *K-Fold Cross Validation*, data akan dipartisi menjadi subset berjumlah K, di mana tiap subset memiliki jumlah data yang sama. Pembagian data uji dan data latih dilakukan secara iteratif tergantung pada nilai K yang digunakan

#### 6. Pelatihan Radial Basis Function

Setelah data dibagi menjadi data latih dan data uji, selanjutnya dilakukan pelatihan. Pelatihan dilakukan pada tiap-tiap *fold*, sehingga pelatihan dilakukan sejumlah K kali. Sebelum pelatihan dimulai pertama menetapkan titik tengah pada data yang digunakan sebagai center dan kemudian pelatihan dapat dimulai.

#### 7. Pengujian

Tahapan pengujian dilakukan untuk menguji kinerja model yang digunakan untuk pelatihan. Untuk menghitung akurasinya, digunakan *Mean Square Error* (MSE), sehingga hasil dari tahapan pengujian adalah nilai MSE terbaik dan arsitektur terbaik dari jaringan tersebut.

#### 8. Evaluasi

Evaluasi dilakukan untuk menentukan bagaimana model jaringan yang terbaik. Evaluasi dilakukan dengan membandingkan nilai MSE. Model jaringan yang memiliki nilai MSE terbaik merupakan model jaringan yang terbaik dalam mengenali pola.

### 3.4. Jadwal

Subbab ini menyajikan jadwal kegiatan mulai dari tahap awal hingga tahap akhir. Dengan adanya jadwal ini akan dapat memberikan gambaran mengenai tahapan yang akan dilakukan. Jadwal kegiatan penelitian dapat dilihat pada tabel 3.1

Tabel 3.1 Jadwal Kegiatan Penelitian

Kegiatan	Apr	Mar	Mei				Jun				Jul				Ags				Sep			
	2017	2017	2017				2017				2017				2017				2017			
	1-4	1-4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Pencarian Jurnal	■																					
Penyusunan Proposal TA1	■	■																				
Seminar Proposal TA1			■																			
Analisis Kebutuhan Sistem			■	■	■																	
Desain				■	■	■	■	■														
Implementasi							■	■	■	■	■	■										
Pengujian sistem												■	■	■								
Penyusunan Laporan TA2													■	■	■	■						
Persiapan Sidang TA2																		■				
Sidang TA2																			■	■		
Revisi Laporan TA2																				■	■	■

## Daftar Pustaka

- Agarwal, B. B. 2009. Software Engineering. New Delhi: Firewall.
- Alsalamah, Mashail dkk. 2014. Diagnosis of Heart Disease by Using a Radial Basis Function Network Classification Technique on Patients' Medical Records. IEEE 978-1-4799-5447-6/14.
- Black, J.M. & Hawks, J.H. 2005. *Medical Surgical Nursing; Clinical Management for Positive Outcomes. 7th Edition*, St. Louis: Elsevier Saunders.
- Doenges, Marilyn E. 2002. Rencana Asuhan keperawatan. Ed. 3. Jakarta: EGC.
- Francesco, Locatelli dkk. 2002. *The importance of early detection of chronic kidney disease*. Oxford : Oxford University Press.
- Frank, C.Lu. 1994. *Toksikologi Dasar*. Edisi Kedua. Genewa
- Gelinas, U.J., Oram, A.E., Wiggins, W.P. 1990. "Accounting Information System". PWS-KENT: Publishing Company.
- Grassmann, Aileen dkk. 2005. *ESRD Patients In 2004: Global Overview Of Patient Numbers, Treatment Modalities And Associated Trends*. Oxford : Oxford University Press
- Guyton, Arthur C. & John E. Hall. 2007. *Buku Ajar Fisiologi Kedokteran, Edisi 11*. Jakarta: EGC.
- Haykin Simon. 2009. Neural Network and Machine Learning Third Edition. New Jersey : Pearson Education inc.
- Is, mardianto dan Dian, Pratiwi. 2008. Sistem Deteksi Penyakit Pengeroposan Tulang Dengan Metode Jaringan Syaraf Tiruan Backpropagation Dan Representasi Ciri Dalam Ruang Eigen. *CommIT, Vol. 2* hlm. 69 - 80
- Kadir, A. 2003. "Pengenal Sistem Informasi". Yogyakarta: Penerbit Andi.
- Kementerian Kesehatan. 2013. Infodatin Hipertensi. Jakarta : Pusat Data Dan Informasi Kementerian Kesehatan.
- Kriesel, David. 2005. A brief introduction to neural network. Diakses 04 april 2017. [http://www.dkriesel.com/en/science/neural\\_networks](http://www.dkriesel.com/en/science/neural_networks)
- Kohavi, R & Provost, F. 1998. *Glossary of terms, Machine Learning*. Vol. 30, No. 2/3, pp. 271-274.

- Kohavi, Ron. 1995. *A Study of Cross Validation and Bootstrap for Accurary Estimation and Model Selection*. International Joint Conference of Artificial Intelegence (IJCAI).
- Kusumadewi, Sri. *Artificial Intellegence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu. 2003.
- Long, Barbara C, 1996, *Perawatan Medikal Bedah, (Volume 2)*, Bandung: Yayasan Alumni Pendidikan Keperawatan.
- Mazlan, Umi Hanim dan Puteh, Saad. 2012. *Classification Of Breast Cancer Microarray Data Using Radial Basis Function Network*. International Conference on Statistics in Science, Business and Engineering (ICSSBE)
- Moh. Nazir. Ph.D, 2005, *Metode Penelitian*. Ghalia Indonesia. Bogor
- National Kidney Foundation. 2013. *About Chronic Kidney Disease : A Guide For Patients*. New York : National Kidney Foundation, inc.
- Peranginangin, K. 2006. “Aplikasi Web dengan PHP dan MySQL”. Yogyakarta: Penerbit Andi.
- Riama Oktaviyani Samosir. 2015. *Perbandingan Metode Klasifikasi Regresi Logistik Biner dan Radial Basis Function Network Pada Berat Bayi Lahir Rendah (Studi Kasus: Puskesmas Pamenang Kota Jambi)*. Skripsi tidak diterbitkan. Semarang : Universitas Diponegoro.
- Santhanam Tiruvenkadam dan Subahjini A.C. 2011. *An Efficient Weather Forecasting System using Radial Basis Function Neural Network*. *Journal of Computer Science* 7 (7) hlm 962-966.
- Sommerville, I. 2011. *Software Engineering (Rekayasa Perangkat Lunak) (6 ed.)*. Jakarta: Erlangga.
- Sug Hyontai. 2011. *Performance Comparison of RBF networks and MLPs for Classification*. Proceedings of the 9th WSEAS International Conference on Applied Informatics And Communications (AIC '09).
- Zamani, M. 2011. *Informatic Engineering and Information Science*. Proceeding of International Conference ICEIS, Kuala Lumpur, Malaysia.

## **Lampiran-Lampiran**



**Lampiran 1.** Tabel dataset penyakit ginjal kronis