

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Setiap diadakan penelitian baru tidak lepas dari kajian hasil penelitian-penelitian terdahulu karena dijadikan sebagai referensi atau bahan perbandingan atas apa yang akan dilakukan oleh peneliti. Adapun hasil-hasil penelitian yang dijadikan perbandingan tidak terlepas dari topik penelitian yaitu pengembangan sistem jaringan syaraf tiruan menggunakan algoritma backpropagation dalam beberapa bidang dan salah satu diantaranya adalah bidang peramalan sebagai acuan dari penelitian.

JST merupakan salah satu representasi buatan otak manusia yang selalu mencoba untuk mensimulasi proses pembelajaran pada otak manusia (Aprijani dan Sufandi, 2011; Lai, 2006) sedangkan Li dan Liu (2006); Warsito, et al (2008) memodelkan JST sebagai sistem yang memiliki *input* dan *output* berdasarkan saraf biologi.

Beberapa penelitian di berbagai bidang menggunakan metode *problem solving* JST diantaranya seperti permasalahan *Traveling Salesman* (Puspitorini, 2008). *Problem solving* untuk pengenalan pola (Hidayatno et al, 2008). Optimalisasi hasil deteksi pola pada gambar tertentu (EL-Bakry, 2006; Chickerur dan M Kumar, 2011; Jing He et al, 2009). Permasalahan pada bidang elektro (Mismar dan AbuBaker, 2010; Frianto dan Rivai, 2008; Wang et al, 2007; Qin dan Zimmermann, 2007). Dapat menyelesaikan masalah dalam bidang kesehatan

(Kanth et al, 2011; Yuwono, 2009) dan *speech recognition* (Maheswari et al, 2009).

Salah satu bidang dimana JST dapat diaplikasikan dengan baik adalah bidang peramalan (Setiawan, 2008) dan salah satu teknik peramalan yang sering digunakan dalam JST adalah backpropagation. Teknik peramalan banyak digunakan untuk proses perencanaan dan pengambilan keputusan. Peramalan menunjukkan perkiraan yang akan terjadi pada suatu keadaan tertentu sebaliknya perencanaan menggunakan peramalan untuk membantu para pengambil keputusan dalam memilih alternatif keputusan yang terbaik. Dengan demikian, suatu ramalan mencoba untuk memperkirakan apa yang akan terjadi.

Santoso, et al (2007) menerapkan model backpropagation dengan struktur jaringan 2-5-4-1-2 untuk meramalkan banyaknya permintaan karet sebagai komoditas pada PT. Perkebunan Nusantara XII Surabaya dan perbandingan tingkat akurasi metode peramalan dengan data pengujian didapatkan persentase kesalahan absolute (MAPE) adalah 17,54%.

Metode backpropagation digunakan oleh Andrijasa dan Mistianingsih (2010) untuk memprediksi jumlah pengangguran di Provinsi Kalimantan Timur dan hasil pengujian di peroleh prediksi jumlah pengangguran Tahun 2009 adalah 133.104 sedangkan hasil prediksi pengangguran Tahun 2009 yang dilakukan oleh BPS Provinsi Kalimantan Timur adalah 139.830.

Pengembangan aplikasi dengan menggunakan metode backpropagation digunakan juga oleh Mulyana (2008) untuk meramalkan tingkat penjualan dan diperoleh tingkat penyimpangan rata-rata sebesar 3.3%.

Setiawan (2008) memprediksi harga saham menggunakan algoritma backpropagation namun hasil dari prediksi menggunakan algoritma tersebut dibandingkan dengan metode *Exponential Smoothing* didapat persentasi toleransi MAPE 3.97%. Susanti, et al (2010) menggunakan *recurrent neural network* dikombinasikan dengan algoritma backpropagation *through time* untuk memprediksi harga saham. Lubis, et al (2005) menggunakan Hybrid Neural Network yaitu : algoritma backpropagation, algoritma self organizing, maps kohonen untuk memprediksi harga saham. Hasil dari penelitian ini membuktikan bahwa algoritma hybrid dapat memberikan keakuratan prediksi harga saham yang lebih tinggi dibandingkan dengan algoritma backpropagation maupun algoritma self organizing maps kohonen. Demikian halnya dengan Kuswati (2008) menggunakan *multilayer perceptron feed forward neural network* untuk meramalkan time series harga saham, dan hasil dari penelitian ini adalah perlu memperhatikan nilai tiap *unit hidden layer*, *learning rate*, dan momentum karena parameter-parameter tersebut dapat menentukan besar kecilnya error pada saat pelatihan maupun pengujian. Hasil peramalan menunjukkan bahwa keakuratannya rata-rata di atas 90%, semakin banyak data maka semakin kecil erornya.

Kuncoro dan Dalimi (2005) mengembangkan aplikasi untuk meramalkan beban tenaga listrik jangka panjang pada sistem kelistrikan Indonesia, menurut perhitungan dengan metode JST diperoleh hasil bahwa perkiraan beban puncak di Indonesia pada tahun 2005, 2010, 2015, 2020, dan 2025 adalah masing-masing 16.516 MW, 24.482 MW, 36.157 MW, 56.060 MW, dan 85.584 MW. Ismail, et al (2011) juga menggunakan JST untuk memprediksi jumlah listrik yang keluar dan

masuk melalui pipa, penelitian ini akan membantu untuk menghilangkan ketidakpastian pada meteran listrik untuk tujuan kestabilan dan keakuratan penagihan listrik. Adapun tujuan dari penelitiannya mengembangkan model JST untuk konsumsi listrik dan menganalisis kinerja model tersebut dan hasil menunjukkan bahwa JST mampu menunjukkan kinerja yang memadai. Sari (2006) salah satu peramalan beban listrik adalah menggunakan JST, adapun variabel yang digunakan untuk melakukan penelitian data beban listrik dan data temperatur lingkungan dimana temperatur lingkungan ini merupakan salah satu hal yang mempengaruhi konsumsi beban, hasil dari penelitian diperoleh dengan menambahkan rata-rata data hari-hari similar yang telah dipilih dan persentasi kesalahan rata-rata absolute antara hasil listrik peramalan menggunakan data beban listrik wilayah Jateng dan DIY dengan beban aktual sebesar 2.39

Purnama (2007) pada penelitiannya menggunakan metode backpropagation untuk memprediksi inflasi harga Jakarta dan Surabaya, adapun hasil dari penelitiannya adalah learning rate 0.05, nilai momentum 0.8 dengan arsitektur 19.1 memberikan hasil yang baik.

Sutikno, et al (2007) memprediksi risiko kredit menggunakan JST backpropagation. Dari prediksi JST backpropagation dengan 1 *hidden layer* dan 100 jumlah sel serta menggunakan fungsi aktivasi satlin mampu memprediksi risiko kredit dengan persentasi kebenaran 70%, JST backpropagation dengan 2 *hidden layer* dan 100 jumlah sel pertama, 300 jumlah sel kedua dengan kombinasi fungsi aktivasi logsig-satlin mampu memprediksi risiko kredit dengan persentasi

100% sehingga melalui penelitian ini, peneliti merekomendasikan JST sebagai metode yang efektif pada sistem prediksi risiko kredit.

Suhartono (2007) melakukan penelitian terkait dengan peramalan data runtun waktu yang adalah salah satu bidang pemodelan statistik. Estimasi parameter yang digunakan menerapkan algoritma backpropagation pada optimisasi *nonlinear least squares*. Hasil dari penelitian ini kajian perbandingan ketepatan ramalan pada kasus runtun waktu yang multivariate pada data produksi minyak, menunjukkan bahwa algoritma backpropagation memberikan hasil ramalan yang lebih baik dibandingkan dengan model GSTAR (*Generalized Space-Time Autoregressive*) VAR (*Vector Autoregressive*).

Warsito (2006) membandingkan metode backpropagation dan GRNN (*General Regression Neural Network*) untuk memprediksi nilai tukar mata uang Yen Jepang terhadap dolar AS. Dan hasil penelitian ini menunjukkan bahwa model GRNN relatif lebih unggul daripada model backpropagation.

Anurag dan Chaturvedi (2011) dalam penelitian mereka mendemonstrasikan peramalan lintasan pesawat menggunakan metode backpropagation, untuk melatih sistem jaringan syaraf mereka menggunakan serangkaian lintasan dan hasil menunjukkan bahwa jaringan syaraf berhasil diterapkan.

Berdasarkan beberapa uraian penelitian diatas, menekankan pada analisis pemodelan backpropagation untuk melakukan prediksi atau peramalan dengan berbagai objek penelitiannya masing-masing. Penelitian yang akan dikembangkan ini merupakan sistem prediksi untuk menentukan jumlah dokter keluarga

bertujuan untuk membantu pihak PT Askes (Persero) dalam memberikan pemerataan pelayanan kesehatan. Adapun sistem yang dikembangkan berbasis desktop (*Microsoft Visual Basic 6.0* sebagai perangkat lunak untuk membangun sistem prediksi dan *Microsoft Access 2007* sebagai perangkat lunak tambahan untuk menyimpan data ke dalam *database*) dengan sifatnya yang lintas *platform* sehingga dapat di *install* pada berbagai *platform* komputer tanpa melakukan proses kompilasi berulang kali. Pada Tabel 2.1 dibawah ini menunjukkan perbandingan penelitian sebelumnya dengan penelitian yang dikembangkan oleh penulis.

Tabel 2. 1 Perbandingan Penelitian

| Peneliti | Rekayasa Objek Penelitian | Algoritma | Fase Penelitian | | | |
|-----------------------------|--|--|-----------------|-------------|--------------|-----------|
| | | | Analisis | Perancangan | Implementasi | Pengujian |
| Santoso, et al | Permintaan Karet | Backpropagation | √ | | | √ |
| Andrijasa dan Mistianingsih | Pengangguran di Provinsi KALTIM | Backpropagation | √ | | | √ |
| Mulyana | Tingkat Penjualan | Propagasi balik | | | √ | √ |
| Setiawan | Harga Saham | Backpropagation | | | √ | √ |
| Susanti, et al | Harga Saham | Backpropagation Through Time | √ | √ | | √ |
| Lubis, et al | Harga Saham | Hybrid Neural Network (Backpropagation, Self Organizing, Maps Kohonen) | | | | √ |
| Kuswati | Harga Saham | Multilayer Perceptron Feed Forward Neural Network | √ | √ | √ | |
| Ismail, et al | Konsumsi Energi | MultiLayer Feedforward | √ | | | |
| Sari | Kebutuhan Beban Jangka Pendek | Backpropagation | √ | √ | √ | √ |
| Purnama | Inflasi harga | Backpropagation | √ | | | |
| Sutikno | Risiko kredit | Backpropagation | | √ | | √ |
| Warsito | Nilai Tukar Mata Uang Yen dan Dolar AS | Backpropagation dan General Regression Neural Network | √ | | | |
| Anurag dan Chaturvedi | Lintasan pesawat | Backpropagation | √ | | | |
| Penulis | Jumlah Dokter Keluarga | Backpropagation | √ | √ | √ | √ |

2.2 LANDASAN TEORI

2.2.1 Kecerdasan Buatan

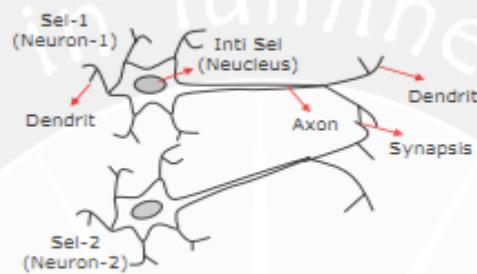
Kecerdasan buatan merupakan bagian ilmu pengetahuan yang digunakan untuk menyelesaikan masalah manusia dengan cara memahami, memprediksi dan memanipulasi (Russell dan Norvig, 2010), kecerdasan yang dibuat untuk sistem menggunakan algoritma tertentu sehingga sistem seolah-olah dapat berpikir seperti manusia (Coppin, 2004). Definisi lain mengatakan bahwa kecerdasan buatan merupakan cabang teknologi informasi yang berusaha mengimitasi kecerdasan atau cara berpikir manusia untuk diaplikasikan pada komputer (Fatta, 2007).

2.2.2 Jaringan Syaraf Tiruan

2.2.2.1 Gambaran Umum Jaringan Syaraf Tiruan

Jaringan syaraf tiruan merupakan bagian dari sistem kecerdasan buatan (Russell dan Norvig, 2010) digunakan untuk memproses informasi yang didesain dengan menirukan cara kerja otak manusia dalam menyelesaikan masalah dengan melakukan proses belajar melalui perubahan bobot sinapsisnya (Hermawan, 2006; Siang, 2009; Septiani, 2005). Pada gambar 2.1 menunjukkan salah satu contoh syaraf secara biologis dimana setiap sel syaraf (*neuron*) akan memiliki satu inti sel yang bertugas untuk melakukan pemrosesan informasi. Informasi yang datang akan diterima oleh *dendrit*. Selain menerima informasi, *dendrit* juga menyertai *axon* sebagai keluaran dari suatu pemrosesan informasi. Informasi hasil olahan ini akan menjadi masukan bagi *neuron* lain dimana antar *dendrit* kedua sel tersebut dipertemukan dengan sinapsisnya. Informasi yang dikirimkan antar *neuron* ini

berupa rangsangan yang dilewatkan melalui *dendrit*. Informasi yang datang dan diterima oleh *dendrit* akan dijumlahkan dan dikirim melalui *axon* lain. Informasi ini akan diterima oleh *neuron* lain jika memenuhi batasan tertentu dikenal dengan nilai ambang (*threshold*) yang dikatakan teraktivasi (Fausett, 1994) dalam Purnawati, (2010).



Gambar 2. 1 Syaraf Secara Biologi (Fausett, 1994)

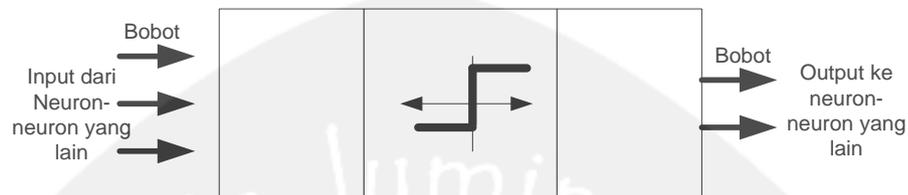
Karakteristik jaringan syaraf ditentukan oleh beberapa hal (Hermawan, 2006) yaitu :

- a. Pola hubungan antar *neuron* yang disebut dengan arsitektur jaringan;
- b. Metode penentuan bobot-bobot sambungan yang disebut dengan pelatihan atau proses belajar jaringan;
- c. Fungsi aktivasi.

2.2.2.2 Komponen-komponen Jaringan Syaraf Tiruan

Seperti halnya otak manusia, jaringan syaraf juga terdiri dari beberapa *neuron*, dan terdapat hubungan antara *neuron-neuron* tersebut. *Neuron-neuron* tersebut akan mentransformasikan informasi yang diterima melalui sambungan keluarnya menuju ke *neuron-neuron* yang lain. Pada jaringan syaraf hubungan ini dikenal dengan nama bobot. Informasi tersebut disimpan pada suatu nilai tertentu

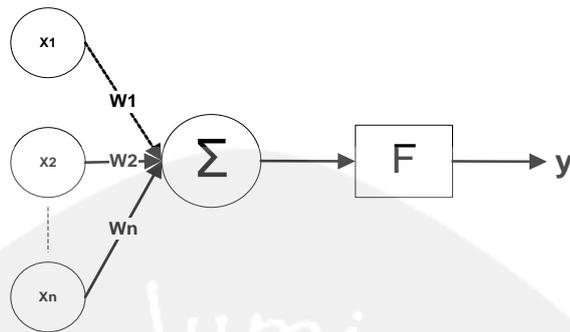
pada bobot tersebut, Gambar 2.2 menunjukkan *neuron* pada jaringan syaraf (Kusumadewi, 2010).



Gambar 2. 2 Struktur *Neuron* Jaringan Syaraf Tiruan (Kusumadewi, 2010)

Informasi yang disebut dengan masukan dikirim ke *neuron* dengan bobot kedatangan tertentu. Masukan ini diproses oleh suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang akan datang. Hasil penjumlahan ini kemudian dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap *neuron*. Apabila masukan tersebut melewati suatu nilai ambang tertentu maka *neuron* tersebut akan diaktifkan. Apabila *neuron* diaktifkan, maka *neuron* tersebut akan mengirimkan keluaran melalui bobot-bobot keluaran ke semua *neuron* yang berhubungan dengannya demikian selanjutnya.

Pada jaringan syaraf, *neuron-neuron* akan dikumpulkan dalam lapisan-lapisan yang disebut dengan lapisan *neuron*. Biasanya *neuron* pada satu lapisan akan dihubungkan dengan lapisan sebelum atau sesudahnya terkecuali lapisan masukan dan lapisan keluaran. Informasi yang diberikan pada jaringan syaraf akan dirambatkan dari lapisan ke lapisan, mulai dari lapisan masukan sampai lapisan keluaran melalui lapisan tersembunyi. Algoritma pembelajaran menentukan informasi akan dirambatkan kearah mana (Kusumadewi, 2010), Gambar 2.3 menunjukkan *neuron* jaringan syaraf sederhana dengan fungsi aktivasi F.



Gambar 2. 3 Model Neuron Sederhana (Kusumadewi, 2010)

Pada Gambar 2.3 sebuah *neuron* akan mengolah N masukan ($X_1, X_2, X_3, \dots, X_n$) yang masing-masing memiliki bobot $W_1, W_2, W_3, \dots, W_n$, dengan rumus :

$$y_{in} = \sum_{i=1}^n x_i w_i \quad (2.1)$$

Kemudian, fungsi aktivasi F akan mengaktivasi y_{in} menjadi keluaran jaringan y . Untuk Jaringan syaraf dengan jumlah *neuron* pada lapisan keluaran sebanyak m buah maka proses pengolahan data pada *neuron* adalah :

$$y_{in_j} = \sum_{i=1}^n x_i w_{ij} ; j=1, \dots, m \quad (2.2)$$

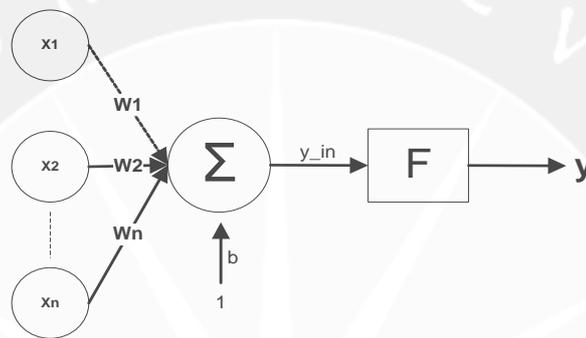
Dengan W_{ij} adalah bobot yang menghubungkan masukan ke- i menuju *neuron* ke- j . Namun, terkadang jaringan syaraf tiruan tidak mampu mengakomodasi informasi yang ada melalui data-data masukan maupun bobot-bobotnya. Untuk itu biasanya ditambahkan bias yang senantiasa bernilai 1 ditunjukkan pada Gambar 2.4. Pengaruh bias terhadap *neuron* ditunjukkan dengan bobot bias (b). Apabila pada jaringan syaraf dilengkapi dengan bias, maka proses komputasi pada *neuron* menjadi :

$$y_{in} = \sum_{i=1}^n x_i w_{ij} + b \quad (2.3)$$

Jaringan syaraf dengan jumlah *neuron* pada lapisan keluaran sebanyak *m* buah, maka proses pengolahan data pada *neuron* ke-*j* adalah :

$$y_{in_j} = \sum_{i=1}^n x_i w_{ij} + b_j; j=1, \dots, m \quad (2.4)$$

W_{ij} adalah bobot yang menghubungkan masukan ke-*i* menuju ke *neuron* ke-*j*, dan b_j adalah bobot bias yang menuju ke *neuron* ke-*j*.



Gambar 2. 4 Model *Neuron* Sederhana Dengan Bias (Kusumadewi, 2010)

2.2.2.3 Arsitektur Jaringan Syaraf Tiruan

Hubungan antar *neuron* pada jaringan syaraf tiruan mengikuti pola tertentu, tergantung pada arsitektur jaringan syarafnya. Pada dasarnya terdapat tiga macam arsitektur yaitu (Kusumadewi, 2010) :

1. Jaringan Syaraf dengan Lapisan Tunggal

Jaringan tipe ini hanya memiliki satu lapisan dengan bobot-bobot terhubung, menerima masukan kemudian secara langsung mengolah menjadi keluaran tanpa harus melalui lapisan tersembunyi. Ciri-ciri arsitektur ini hanya terdiri atas satu lapisan masukan dan satu lapisan keluaran tanpa adanya lapisan tersembunyi. Gambar 2.3 dan Gambar 2.4 merupakan salah satu contoh model *neuron* dengan satu lapisan masukan.

2. Jaringan Syaraf dengan Banyak Lapisan

Arsitektur tipe ini memiliki satu atau lebih lapisan yang terletak di antara lapisan masukan dan lapisan keluaran, memiliki juga satu atau lebih lapisan tersembunyi. Umumnya, ada lapisan bobot-bobot yang terletak antara dua 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 juga. Gambar 2.5 merupakan salah satu contoh model *neuron* dengan banyak lapisan dan hanya memiliki satu lapisan tersembunyi. V_{ij} adalah bobot-bobot yang menghubungkan antara *neuron-neuron* pada lapisan masukan dengan lapisan tersembunyi. Sedangkan W_{jk} adalah bobot-bobot yang menghubungkan antara *neuron-neuron* pada lapisan tersembunyi dengan lapisan keluaran. z_in_j adalah hasil pengolahan data pada lapisan tersembunyi dengan fungsi aktivasi F1 untuk menghasilkan z_j ($j=1, \dots, k$);

$$z_in_j = \sum_{i=1}^n x_i w_{ij} \quad (2.5)$$

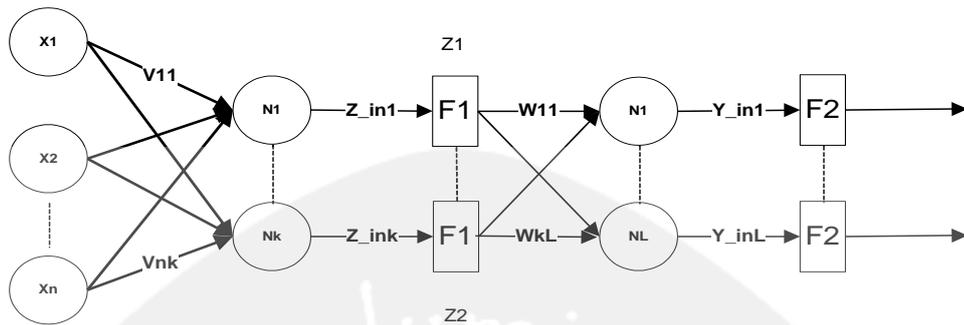
$$z_j = F1(z_in_j) \quad (2.6)$$

Sedangkan y_in_k adalah hasil pengolahan data pada lapisan keluaran dengan fungsi aktivasi F2 untuk menghasilkan keluaran jaringan.

$$y_k ; (k = 1, \dots, L) \quad (2.7)$$

$$y_in_k = \sum_{j=1}^k z_j w_{jk} \quad (2.8)$$

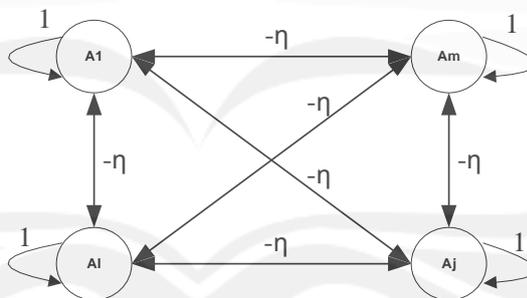
$$z_j = F2(y_in_k) \quad (2.9)$$



Gambar 2. 5 Model *Neuron* dengan Banyak Lapisan (Kusumadewi, 2010)

3. Jaringan Syaraf dengan Lapisan Kompetitif

Arsitektur tipe ini tampak pada Gambar 2.6 dimana memiliki pengaturan bobot yang telah ditetapkan dan tidak memiliki proses pelatihan. Digunakan untuk mengetahui *neuron* pemenang dari sejumlah *neuron* yang ada. Nilai bobot untuk diri sendiri dari setiap *neuron* adalah 1, dan *neuron* lain adalah bobot acak negatif dengan bobot $-\eta$.



Gambar 2. 6 Model *Neuron* dengan Lapisan Kompetitif (Kusumadewi, 2010)

2.2.2.4 Fungsi Aktivasi

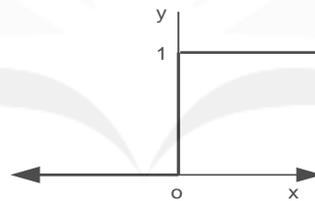
Ada beberapa fungsi aktivasi yang sering digunakan dalam jaringan syaraf tiruan antara lain:

1. Fungsi Sigmoid Biner

Fungsi ini digunakan untuk jaringan syaraf yang dilatih menggunakan metode backpropagation. Fungsi sigmoid biner yang tampak pada Gambar 2.6 memiliki nilai antara 0 sampai 1. Karena itu, fungsi ini sering digunakan untuk jaringan syaraf yang membutuhkan nilai keluaran yang terletak pada interval 0 sampai 1. Fungsi sigmoid biner dirumuskan sebagai berikut:

$$y = f(x) = \frac{1}{1+e^{-\sigma x}} \quad (2.10)$$

$$f'(x) = \sigma f(x) [1 - f(x)] \quad (2.11)$$



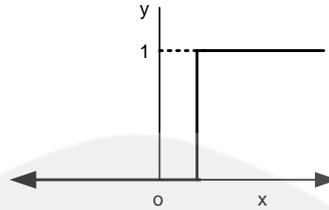
Gambar 2. 7 Fungsi Aktivasi Sigmoid Biner (Kusumadewi, 2010)

2. Fungsi Sigmoid Bipolar

Fungsi ini hampir sama dengan fungsi sigmoid biner, hanya saja keluaran dari fungsi ini antara 1 sampai -1 terlihat pada Gambar 2.7. Sedangkan fungsi sigmoid bipolar dirumuskan seperti berikut ini :

$$y = f(x) = \frac{1-e^{-x}}{1+e^{-\sigma x}} \quad (2.12)$$

$$f'(x) = \frac{\sigma}{2} [1 + f(x)][1 - f(x)] \quad (2.13)$$

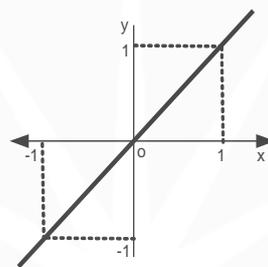


Gambar 2. 8 Fungsi Aktivasi Sigmoid Bipolar (Kusumadewi, 2010)

3. Fungsi Linear (Identitas)

Fungsi linear memiliki nilai keluaran yang sama dengan nilai masukannya terlihat pada Gambar 2.8. Fungsi linear dirumuskan sebagai berikut :

$$y = x \quad (2.14)$$



Gambar 2. 9 Fungsi Aktivasi Linear (Kusumadewi, 2010)

2.2.2.5 Algoritma Pembelajaran

Tujuan utama proses pembelajaran adalah melakukan pengaturan terhadap bobot-bobot yang ada pada jaringan syaraf, sehingga diperoleh bobot akhir yang tepat sesuai dengan pola data yang dilatih. Selama proses pembelajaran akan terjadi perbaikan bobot-bobot berdasarkan algoritma tertentu. Nilai bobot akan bertambah jika informasi yang diberikan oleh *neuron* bersangkutan tersampaikan, sebaliknya jika informasi tidak tersampaikan maka bobot akan dikurangi. Pada dasarnya ada dua metode pembelajaran yaitu metode pembelajaran terawasi dan pembelajaran tidak terawasi.

1. Pembelajaran Terawasi

Metode pembelajaran disebut terawasi bila keluaran yang diharapkan telah diketahui sebelumnya. Misalkan dimiliki jaringan syaraf untuk mengenali pasangan pola dengan operasi AND, pada proses pembelajaran satu pola masukan akan diberikan ke satu *neuron* pada lapisan masukan. Pola dirambatkan di sepanjang jaringan syaraf sampai dengan *neuron* pada lapisan keluaran. Lapisan keluaran ini akan membangkitkan pola keluaran yang nantinya akan dicocokkan dengan pola keluaran targetnya. Apabila terjadi perbedaan antara pola keluaran hasil pembelajaran dengan pola target maka terjadi *error*, apabila nilai *error* masih cukup besar mengindikasikan masih perlu dilakukan pembelajaran lagi.

2. Pembelajaran Tidak Terawasi

Metode ini tidak memerlukan target keluaran karena tidak dapat ditentukan hasil yang seperti apakah yang diharapkan selama proses pembelajaran. Saat proses pembelajaran, nilai bobot disusun dalam suatu *range* tertentu tergantung pada nilai keluaran yang diberikan, apabila nilai *error* masih cukup besar mengindikasikan masih diperlukan proses pembelajaran.

2.2.2.5.1 Algoritma Pembelajaran Backpropagation

Algoritma backpropagation merupakan bagian dari algoritma pembelajaran terawasi yang biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan *neuron-neuron* yang ada pada lapisan tersembunyi. Algoritma ini menggunakan *error* keluaran untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk

mendapatkan *error* ini tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu. Saat perambatan maju, *neuron-neuron* diaktifkan dengan menggunakan fungsi aktivasi yang dapat dideferensiasikan seperti sigmoid :

$$y = f(x) = \frac{1}{1+e^{-\sigma x}} \quad (2.15)$$

$$f'(x) = \sigma f(x) [1 - f(x)] \quad (2.16)$$

Atau seperti tangen sigmoid :

$$y = f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.17)$$

$$f'(x) = [1 + f(x)][1 - f(x)] \quad (2.18)$$

Pelatihan Backpropagation dilakukan melalui langkah-langkah berikut ini :

Langkah ke-0 : Inisialisasi bobot;

Langkah ke-1 : Selama kondisi berhenti bernilai salah, kerjakan langkah 2-9;

Langkah ke-2 : Untuk setiap data *training*, lakukan langkah 3-8.

Umpan Maju (*Feedforward*)

Langkah ke-3 : Setiap unit *input* ($X_i, i = 1, \dots, n$) menerima sinyal *input* x_i dan menyebarkan sinyal tersebut ke seluruh unit tersembunyi.

Langkah ke-4 : Pada setiap unit tersembunyi ($Z_j, j = 1, \dots, p$), menjumlahkan sinyal-sinyal *input* yang sudah berbobot (termasuk biasanya)

$$z_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

Lalu menghitung sinyal *output* dari unit tersembunyi dengan menggunakan fungsi aktivasi yang telah ditentukan :

$$z_j = f(z_in_j)$$

Sinyal *output* ini selanjutnya dikirim ke seluruh unit pada unit atas (unit *output*).

Langkah ke-5 : Tiap-tiap unit *output* ($Y_k, k = 1, \dots, m$), menjumlahkan bobot sinyal *input* :

$$y_in_k = w_{0k} + \sum_{i=1}^n z_i w_{jk}$$

Lalu menghitung sinyal *output* dari unit *output* bersangkutan dengan menggunakan fungsi aktivasi yang telah ditentukan

$$y_k = f(y_in_k)$$

Sinyal *output* ini selanjutnya dikirim ke seluruh unit pada *output*.

Umpan Mundur/ Propagasi Error (*Backpropagation of Error*)

Langkah ke-6 : Setiap unit *output* ($Y_k, k = 1, \dots, m$) menerima suatu pola target yang sesuai dengan pola *input* pelatihan, untuk menghitung kesalahan (*error*) antara target dengan *output* yang dihasilkan jaringan

$$\delta_k = (t_k - y_k) f'(y_in_k)$$

Faktor δ_k digunakan untuk menghitung koreksi *error* (Δw_{jk}) yang nantinya akan dipakai untuk memperbaiki w_{jk} , dimana

$$\Delta w_{jk} = \alpha \delta_k z_j$$

Selain itu juga dihitung koreksi bias Δw_{0k} yang nantinya akan dipakai untuk memperbaiki w_{0k} , dimana

$$\Delta w_{0k} = \alpha \delta_k$$

Faktor δ_k kemudian dikirimkan ke lapisan yang berada pada langkah ke-7.

Langkah ke-7 : Setiap unit tersembunyi ($Z_j, j = 1, \dots, p$) menerima *input* delta (dari langkah ke-6) yang sudah berbobot

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk}$$

Kemudian hasilnya dikalikan dengan turunan dari fungsi aktivasi yang digunakan jaringan untuk menghitung informasi kesalahan *error* δ_j , dimana

$$\delta_j = \delta_{in_j} f'(z_{in_j})$$

Kemudian hitunglah koreksi bobot (untuk memperbaiki v_{ij})

$$\Delta v_{ij} = \alpha \delta_j x_i$$

Setelah itu hitung koreksi *bias* (digunakan untuk memperbaiki v_{0j})

$$\Delta v_{0j} = \alpha \delta_j$$

Update Bobot dan Bias (Adjustment)

Langkah ke-8 : Setiap unit *output* ($Y_k, k = 1, \dots, m$) memperbaiki bobot dan bias dari setiap unit tersembunyi ($j = 0, \dots, p$)

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$$

Demikian pula untuk setiap unit tersembunyi ($Z_j, j = 1, \dots, p$) akan memperbaharui bobot dan bias dari setiap unit *input* ($i = 0, \dots, n$)

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

Langkah ke-9 : Tes kondisi berhenti apabila *error* ditemukan

Jika kondisi STOP telah terpenuhi, maka pelatihan jaringan dapat dihentikan.

Untuk memeriksa kondisi STOP, biasanya digunakan kriteria MSE (*Mean Square Error*) berikut ini :

$$\begin{aligned}MSE &= 0.5 \\ &\times \{(t_{k1} - y_{k1})^2 + (t_{k2} - y_{k2})^2 + \dots \\ &+ (t_{km} - y_{km})^2\}\end{aligned}$$

Pengujian Backpropagation

Pengujian backpropagation dilakukan melalui *feedforward* langkah-langkahnya sebagai berikut :

Langkah 0 : Inisialisasikan bobot (dari hasil pelatihan)

Langkah 1 : Untuk setiap vektor *input*, kerjakan langkah 2-4

Langkah 2 : Untuk $i=1, \dots, n$: set aktivasi unit *input* x_i

Langkah 3 : Untuk $j=1, \dots, p$:

$$z_{in_j} = v_{0j} + \sum_i x_i v_{ij}$$

$$z_j = f(z_{in_j})$$

Langkah 4 : Untuk $k=1, \dots, m$:

$$y_{in_k} = w_0 + \sum_i z_j w_{jk}$$

$$y_k = f(y_{in_k})$$

2.2.2.5.2 Contoh Perhitungan Manual Algoritma Backpropagation

Misalnya sebuah jaringan terdiri atas dua unit *input*, satu unit tersembunyi, dan satu unit keluaran. Fungsi aktivasi yang digunakan adalah Sigmoid Biner, *learning rate* / *alpha* (α) = 0.01, toleransi *error* yang diperkenankan adalah 0.41. Jaringan digunakan untuk menyelesaikan fungsi XOR. Pada Gambar 2.9 menunjukkan arsitektur jaringan yang akan dilatih.



Gambar 2. 10 Arsitektur Jaringan yang Dilatih Pada Contoh

Adapun data *training* yang digunakan terdiri atas empat pasang masukan dan keluaran yakni :

Tabel 2. 2 Data Training Contoh

| No | Masukkan 1 | Masukkan 2 | Keluaran |
|----|------------|------------|----------|
| 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 1 |
| 3 | 1 | 0 | 1 |
| 4 | 1 | 1 | 0 |

Langkah-langkah pada proses *training* adalah sebagai berikut :

Langkah 0 : Inisialisasi Sembarang bobot dan bias, misalnya

$$\begin{aligned}
 v_{01} &= 1,718946 \\
 v_{11} &= -1,263178 \\
 v_{21} &= -1,083092 \\
 w_{01} &= -0,541180 \\
 w_{11} &= 0,543960
 \end{aligned}$$

Langkah 1 : Dengan bobot sembarang tersebut, tentukan *error* untuk data *training* secara keseluruhan dengan rumus sebagai berikut :

$$z_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

$$z_j = f(z_in_j)$$

$$z_in_{11} = 1,718946 + \{(0 \times -1,263178) + (0 \times -1,083092)\}$$

$$= 1,718946$$

$$z_{11} = f(z_in_{11}) = 0,847993$$

$$z_in_{12} = 1,718946 + \{(0 \times -1,263178) + (1 \times -1,083092)\}$$

$$= 0,635854$$

$$z_{12} = f(z_in_{12}) = 0,653816$$

$$z_in_{13} = 1,718946 + \{(1 \times -1,263178) + (0 \times -1,083092)\}$$

$$= 0,455768$$

$$z_{13} = f(z_in_{13}) = 0,612009$$

$$z_in_{14} = 1,718946 + \{(1 \times -1,263178) + (1 \times -1,083092)\}$$

$$= -0,627324$$

$$z_{14} = f(z_in_{14}) = 0,348118$$

Dimana indeks z_{jn} berarti bobot untuk unit tersembunyi ke- j dan data *training* ke- n

$$y_in_k = w_{0k} + \sum_{j=1}^n z_j w_{jk}$$

$$y_k = f(y_in_k)$$

$$y_in_{11} = -0,541180 + (0,847993 \times 0,543960) = 0,079906$$

$$y_{11} = f(y_in_{11}) = 0,480034$$

$$y_in_{12} = -0,541180 + (0,653816 \times 0,543960) = -0,185530$$

$$y_{12} = f(y_in_{12}) = 0,453750$$

$$y_in_{13} = -0,541180 + (0,612009 \times 0,543960) = 0,208271$$

$$y_{13} = f(y_in_{13}) = 0,448119$$

$$y_in_{14} = -0,541180 + (0,348118 \times 0,543960) = -0,351818$$

$$y_{14} = f(y_in_{14}) = 0,412941$$

Sehingga,

$$E = 0,5 \times \{(0 - 0,480034)^2 + (1 - 0,453750)^2 + (1 - 0,448119)^2$$

$$+ (0 - 0,412941)^2\}$$

$$= 0,501957$$

Langkah 2 : Karena, data *error training* masih lebih besar dari toleransi yakni

0.41. Maka, pelatihan dilanjutkan pada langkah 3-8

Langkah 3 : $x_1 = 0, x_2 = 0$; (*Training* untuk data pertama)

$$\text{Langkah 4 : } z_{in_{11}} = 1,718946 + \{(0 \times -1,263178) + (0 \times -1,083092)\} = 1,718946$$

$$z_{11} = f(z_{in_{11}}) = 0,847993$$

$$\text{Langkah 5 : } y_{in_{11}} = -0,541180 + (0,847993 \times 0,543960) = 0,079906$$

$$y_{11} = f(y_{in_{11}}) = 0,480034$$

$$\text{Langkah 6 : } \delta_1 = (0 - 0,480034)f'(0,079906) = -0,119817$$

$$\Delta w_{11} = 0,01 \times -0,119817 \times 0,847993 = -0,001016$$

$$\Delta w_{01} = 0,01 \times -0,119817 = -0,00119817$$

$$\text{Langkah 7 : } \delta_{in_1} = -0,119817 \times 0,543960 = -0,065176$$

$$\delta_1 = -0,065176 \times f'(1,718946) = -0,008401$$

$$\Delta v_{11} = 0,01 \times -0,008401 \times 0 = 0$$

$$\Delta v_{21} = 0,01 \times -0,008401 \times 0 = 0$$

$$\Delta v_{01} = 0,01 \times -0,008401 = -0,00008401$$

$$\text{Langkah 8 : } w_{01}(\text{baru}) = -0,541180 + (-0,00119817) = -0,542378$$

$$w_{11}(\text{baru}) = 0,543960 + (-0,001016) = 0,542944$$

$$v_{01}(\text{baru}) = 1,718946 + (-0,00008401) = 1,718862$$

$$v_{11}(\text{baru}) = -1,263178 + 0 = -1,263178$$

$$v_{21}(\text{baru}) = -1,083092 + 0 = -1,083092$$

Setelah langkah 3-8 untuk data *training* pertama dikerjakan, ulangi kembali langkah 3-8 untuk data *training* ke-2,3 dan 4. Setelah seluruh data *training* dikerjakan itu berarti satu iterasi telah diproses. Bobot yang dihasilkan pada iterasi pertama untuk data *training* ke-2,3, dan 4 adalah :

Data *Training* ke-2

$$w_{01} = -0,541023$$

$$w_{11} = 0,543830$$

$$v_{01} = 1,718862$$

$$v_{11} = -1,263178$$

$$v_{21} = -1,083092$$

Data *Training* ke-3

$$w_{01} = -0,539659$$

$$w_{11} = 0,544665$$

$$v_{01} = 1,719205$$

$$v_{11} = -1,263002$$

$$v_{21} = -1,082925$$

Data Training ke-4

$$w_{01} = -0,540661$$

$$w_{11} = 0,544316$$

$$v_{01} = 1,719081$$

$$v_{11} = -1,263126$$

$$v_{21} = -1,083049$$

Setelah sampai pada data *training* ke-4, maka iterasi pertama selesai dikerjakan. Proses *training* dilanjutkan pada langkah ke-9 yaitu memeriksa kondisi STOP dan kembali pada langkah ke-2. Demikian seterusnya sampai *error* yang dihasilkan memenuhi toleransi *error* yang ditentukan. Setelah proses *training* selesai, bobot akhir yang diperoleh untuk contoh XOR adalah sebagai berikut :

$$w_{01} = -5,018457$$

$$w_{11} = 5,719889$$

$$v_{01} = 12,719601$$

$$v_{11} = -6,779127$$

$$v_{21} = -6,779127$$

Jika terdapat masukkan baru, misalnya $x_1 = 0,2$ dan $x_2 = 0,9$ maka keluarannya dapat dicari dengan menggunakan langkah-langkah umpan maju berikut ini :

Langkah 0 : Bobot yang digunakan adalah bobot akhir hasil pelatihan di atas.

Langkah 1 : Perhitungan dilakukan pada langkah 2 – 4.

Langkah 2 : Dalam contoh ini, bilangan yang digunakan telah berada dalam interval 0 dan 1, jadi tidak perlu diskalakan lagi.

$$\begin{aligned} \text{Langkah 3 : } z_{in_1} &= 12,719601 + \{(0,2 \times -6,779127) + (0,9 \times -6,779127)\} \\ &= 5,262561 \end{aligned}$$

$$z_1 = f(z_{in_1}) = 0,994845$$

$$\begin{aligned} \text{Langkah 4 : } y_{in_1} &= -5,018457 + (5,719889 \times 0,994845) = 0,671944 \\ y_1 &= f(y_{in_1}) = 0,661938 \end{aligned}$$

Jadi, jika *input* data adalah $x_1 = 0,2$ dan $x_2 = 0,9$; *output* jaringan yang dihasilkan adalah 0,661938.

Dengan menunjukkan perhitungan manual algoritma backpropagation, sangat tidak memungkinkan untuk melakukan perhitungan manual pada penelitian prediksi jumlah dokter keluarga dengan menggunakan 78 data *training* dan 13 data *testing*, karena itu diperlukan perangkat lunak untuk dapat melakukan proses komputasi pada penelitian ini.

2.2.3 Prediksi

2.2.3.1 Definisi Prediksi

Prediksi atau peramalan merupakan studi terhadap data historis dengan tujuan untuk menemukan hubungan kecenderungan dan pola sistematis (Sutono, 2008).

2.2.3.2 Metode dan Jenis Prediksi

Menurut Makridakit, et al (1999) dalam Sutono, (2008) metode prediksi dapat diklasifikasikan ke dalam dua kategori utama yaitu:

1. Metode Kualitatif

Masukkan yang digunakan tergantung pada metode tertentu dan umumnya berdasarkan pada pendapat ahli dan digunakan saat rekaman data historis tidak banyak tersedia. Jenis prediksi tipe ini adalah *judgement model*, model prediksi yang dilakukan berdasarkan pengalaman dan survei serta cenderung bersifat subyektif.

2. Metode Kuantitatif

Metode ini menggunakan ekspresi matematika untuk memperlihatkan hubungan antara variabel bebas dengan variabel tidak bebas juga memerlukan rekaman data historis dimana data dapat dikuantitatifkan sebagai data numerik dan berasumsi pola data masa lalu akan berlanjut pada masa mendatang. Metode kuantitatif dikelompokkan dalam dua jenis model yaitu :

a) Metode Prediksi Deret-Berskala

Metode Deret berskala merupakan metode prediksi yang menggunakan sekumpulan data berdasarkan nilai data masa lalu dalam interval waktu tertentu. Data masa lalu dianalisis untuk menemukan pola yang tepat kemudian menggunakan pola data tersebut untuk memprediksi sesuatu nilai masa depan.

b) Metode Prediksi Kausal

Metode ini mengasumsikan bahwa faktor yang diprediksi menunjukkan adanya hubungan sebab akibat dengan satu atau lebih variabel lain disebut variabel bebas. Penggunaan metode kausal memungkinkan adanya faktor subyektifitas dalam memprediksi yaitu saat penentuan variabel bebas apa saja yang akan dipertimbangkan.

2.2.3.3 Data Sebagai Komponen Utama Peramalan dan Identifikasi Pola Data

Prinsip *garbage in garbage out* berlaku pada tahapan pertama dalam melakukan peramalan yaitu pengumpulan data. Apabila data yang dikumpulkan kurang tepat atau kurang memadai, hasil peramalan akan menjadi kurang akurat.

Menurut Supranto, (1993) dalam Sutono, (2008) untuk memperoleh data yang baik diperlukan beberapa syarat, yaitu :

- 1 Data harus bersifat obyektif, dimana data tersebut harus sesuai dengan keadaan yang sebenarnya;
- 2 Data yang diperoleh berdasarkan penelitian *sample*, harus dapat mewakili keseluruhan populasi (*representative*);
- 3 Data sebagai parameter perkiraan harus mempunyai *standard error* atau *sampling error* minimum;
- 4 Data harus tepat waktu;
- 5 Data harus memiliki realasi dengan persoalan yang akan dipecahkan (*relevant*).

2.2.4 Akurasi Prediksi

Kesalahan atau *error* dalam memprediksi (e_i) merupakan perbedaan antara nilai variabel yang sesungguhnya (Y_i) dengan nilai variabel yang diestimasi dengan persamaan (\hat{Y}_i), dapat ditulis sebagai berikut :

$$e_i = Y_i - \hat{Y}_i$$

Terdapat berbagai parameter untuk menghitung kesalahan peramalan, antara lain :

- a. *Mean Absolute Deviation* (MAD)

$$MAD = \frac{\sum_{i=1}^n |e_i|}{n}$$

- b. *Mean Squared Error* (MSE)

$$MSE = \frac{\sum_{i=1}^n e_i^2}{n}$$

c. *Mean Percentage Error (MPE)*

$$MPE = \frac{\sum_{i=1}^n \frac{e_i}{Y_i} \times 100}{n}$$

d. *Mean Absolute Percentage Error (MAPE)*

$$MAPE = \frac{\sum_{i=1}^n \frac{|e_i|}{Y_i} \times 100}{n}$$

2.2.5 Kesehatan Primer

Sejak tahun 1978 ketika Organisasi Kesehatan Dunia (WHO) mencanangkan program “*Health for All in 2000*”, pelayanan kesehatan primer menjadi hal utama dalam pengembangan perencanaan pemerintah. Program tersebut menitikberatkan kepada pelayanan kesehatan yang komprehensif. Dan pada tahun 2008 Laporan kesehatan dunia WHO dengan judul “*Primary health care now more than ever*” mobilitas pengetahuan layanan kesehatan primer merupakan bagian paling penting dari perawatan kesehatan yang efektif (Kringos, *et al.*, 2010). Dokter Praktek di Tingkat Primer harus memiliki kompetensi yang memadai dalam pelayanan individu dan mampu mengintegrasikan pelayanan kesehatan, keluarga dan komunitas.

Pada Januari 1995 WHO dan Organisasi Dokter Keluarga Dunia (WONCA) merumuskan Visi Global dan rencana tindakan untuk meningkatkan kesehatan individu dan masyarakat yang tertuang dalam tulisan “*Making Medical Practice and Education More Relevant to People’s Needs: The Role of Family Doctor*”. Sehingga dalam acara pembukaan Temu Ilmiah Akbar dengan Tema Kursus Penyegar dan Penambah Ilmu Kedokteran (TIA-KPPIK) 2002 di Jakarta, Menteri Kesehatan Achmad Sujudi menyatakan bahwa Visi dan Misi kurikulum

pendidikan dokter di Indonesia seyogianya diarahkan untuk menghasilkan dokter keluarga, tidak lagi dokter komunitas atau dokter Puskesmas seperti sekarang. Hal ini sesuai dengan Peraturan Menteri Kesehatan Nomor 916/Menkes/Per/VIII/1997 tentang Pelayanan Dokter Umum yang diarahkan menjadi pelayanan dokter keluarga.

Berdasarkan pernyataan diatas konsep Dokter Keluarga berangkat dari suatu metode pelayanan kesehatan primer. Pelayanan kesehatan primer begitu kompleks dalam penerapannya menurut Kringos, *et al.*, (2010) wawasan kompleksitas perawatan primer merupakan wujud dari sistem multidimensi. Kringos *et al* mengidentifikasi sepuluh dimensi inti yang membentuk sistem layanan kesehatan primer, yaitu : struktur kesehatan primer terdiri dari tiga dimensi 1). Pemerintah; 2). Kondisi Ekonomi; 3). Tenaga Kerja Pembangunan. Pengelohan primer terdiri dari empat dimensi 4). Akses; 5). Kesiambungan Pelayanan; 6). Koordinasi Perawatan; 7). Kelengkapan Pengobatan. Hasil dari sistem layanan kesehatan primer mencakup tiga dimensi yaitu 8). Kualitas Pelayanan; 9). Efisiensi Perawatan; 10). Ekuitas di Bidang Kesehatan. Terdapat cukup bukti bahwa kontribusi layanan kesehatan primer melalui dimensi kinerja sistem kesehatan secara keseluruhan dapat mencapai kesehatan masyarakat yang baik. Sepaham dengan Miller, *et al.*, (2010) dayaguna praktek layanan kesehatan primer dipahami sebagai sistem adaptif kompleks yang terdiri atas praktek sebagai kunci dari sumber daya, struktur organisasi dan proses fungsional.

Pelayanan kesehatan primer menjadi hal yang krusial dalam masyarakat menurut Neumark, *et al.*, (2008) karena pelayanan kesehatan primer merupakan

pertolongan pertama untuk penderita infeksi saluran pernapasan, Johansen, *et al.*,(2010); Renang dan Marchira., (2009); Jedenius, *et al.*, (2008) sepaham bahwa pelayanan kesehatan primer adalah tonggak awal solusi untuk para penderita penyakit mental di Norwegia.

2.2.6 Dokter Keluarga

Dokter keluarga adalah dokter yang bertanggung jawab melaksanakan pelayanan kesehatan personal, terpadu, berkesinambungan, dan proaktif memiliki sifat pelayanan yang meliputi peningkatan derajat kesehatan (promotif), pencegahan (preventif), pengobatan (kuratif) dan rehabilitatif (Nadesul, 2010). Menurut Nadesul, 2010 apabila suatu masalah khusus tidak dapat ditanggulangi maka dokter keluarga bertindak sebagai koordinator dalam merencanakan konsultasi atau rujukan yang diperlukan kepada dokter spesialis yang lebih sesuai.

Penerapan Sistem Pelayanan Dokter Keluarga (SPDK) yang berarti penerapan pendekatan kedokteran keluarga akan menjadi kebutuhan dunia. Badan Kesehatan Dunia (WHO) dan Organisasi Dokter Keluarga Sedunia (WONCA) pada tahun 1994 menyusun rekomendasi pengimplementasian SPDK di setiap Negara (Wonodirekso, 2009). Dibanyak Negara SPDK terbukti mampu meningkatkan kepercayaan dan kepuasan pasien dalam pelayanan dan penghematan biaya kesehatan (Starfield *et al.*, 2005). Di Indonesia sistem ini sementara tahap penyesuaian karena Indonesia kekurangan dokter keluarga yang andal (Soetono, 2010) di sisi lain masih banyak masyarakat belum mengenal konsep dokter keluarga (Rusady, 2010) kendala lain yang dihadapi di beberapa wilayah tertentu pemerintah setempat belum menyambut baik program ini,

sehingga sosialisasi tentang pelayanan berbasis dokter keluarga kepada masyarakat, para dokter termasuk puskesmas, Dinas Kesehatan dan Pemerintah setempat perlu terus dilakukan. Oleh karena itu sebaiknya sistem pelayanan kesehatan perlu diarahkan agar lebih terstruktur dan berjenjang dan ditingkatkan mutunya melalui penerapan pelayanan kedokteran keluarga sebagai bentuk dari strata pertama yang dapat menjamin efektifitas, efisiensi, pemerataan, dan kesinambungan pelayanan kesehatan (Croft, 2010).

Di Indonesia kebijakan kedokteran keluarga tertuang dalam Pelaksanaan Undang-Undang Praktek Kedokteran No.29 Tahun 2004 yaitu upaya untuk memperbaiki kualitas pelayanan dasar dan kualitas dokter praktek umum di Indonesia, pada SKN 2004 digariskan bahwa upaya kesehatan perorangan strata pertama memakai konsep dokter keluarga, dan Pelaksanaan Undang-Undang Sistem Jaminan Sosial Nasional (SJSN) Nomor 40 Tahun 2004 salah satunya adalah jaminan kesehatan akan menjadi payung perlindungan sosial setiap rakyat, khususnya akses terhadap pelayanan kesehatan yang terstruktur dan berjenjang.

Kebutuhan masyarakat akan pelayanan kesehatan yang bermutu dan berjangkau merupakan sesuatu yang esensial, dengan penyelenggaraan pelayanan kesehatan model dokter keluarga diharapkan menjadi *private good* dokter menjadi bagian dari keluarga (Moeloek, 2005) atau berfungsi sebagai *gate keeper* (Rusady, 2010) dokter keluarga sebagai ujung tombak dalam pelayanan kedokteran tingkat pertama, yang dapat berkolaborasi dengan pelayanan kedokteran tingkat kedua dan bersinergi dengan sistem yang lain. Menurut

Rusady (2010) terdapat peningkatan yang signifikan untuk manfaat pelayanan dokter keluarga seperti yang terlihat pada tabel 2.2 dibawah ini.

Tabel 2. 3 Manfaat Jenis Pelayanan Sebelum dan Sesudah

| Jenis Pelayanan | Sebelum | Sesudah |
|------------------------------------|------------------------------|--|
| Pelayanan Medik | Fokus pada pelayanan kuratif | Fokus pada upaya promotif dan preventif |
| Pelayanan Obat | Apotik dan <i>Dispensing</i> | Apotik dan <i>Dispensing</i> |
| Pelayanan Laboratorium Sederhana | Tidak ada | Darah dan uring rutin |
| Besaran Kapitasi | Rp 2.500 | Rp 5.500,- s/d Rp 6.500 sesuai hasil kredensialing dan komposisi peserta terdaftar |
| Pencatatan | Manual | Komputerisasi |
| Rekrutmen | Sesuai permintaan | Penilaian Kapitasi sarana dan prasarana sesuai standar PDKI |
| Standar Kompetensi Dokter Keluarga | Tidak ada | <i>Workshop</i> dan modul Dokter Keluarga |

2.2.7 Jaringan Pelayanan Kesehatan PT Askes (Persero)

Jaringan pelayanan merupakan salah satu unsur yang memiliki peran strategis dalam memenuhi kebutuhan peserta Askes terhadap pelayanan kesehatan. Informasi terkait dengan keberadaan, jumlah maupun kapasitas pelayanan dari tiap-tiap pemberi pelayanan kesehatan yang bekerja sama dengan PT Askes (Persero) sangat dibutuhkan untuk memberikan pilihan kepada peserta Askes. Selain itu, informasi tersebut sangat diperlukan untuk mendukung upaya PT Askes (Persero) dalam menyempurnakan jasa layanan yang diberikan kepada pesertanya.

Ditinjau dari sisi akuntabilitas pelayanan, informasi jaringan pelayanan yang *up to date*, valid, dan akuntabel sangat diperlukan sebagai dasar evaluasi terhadap pembiayaan pelayanan kesehatan bagi peserta. Keseluruhan kondisi tersebut merefleksikan perlunya dilakukan penataan serta pembinaan secara

berkesinambungan terhadap jaringan pelayanan yang ada untuk mendukung struktur pelayanan yang efisien dan efektif.

Sejalan dengan Visi PT Askes untuk menjadi “Spesialis dan Pusat Unggulan Asuransi Kesehatan di Indonesia” serta komitmen untuk memberikan “Pelayanan Melampaui Harapan Pelanggan” kepada peserta, PT Askes melalui Direktori Jaringan Pelayanan Kesehatan edisi Tahun 2010-2011 sebagai sarana untuk memenuhi kebutuhan pelanggannya. Hal ini dilakukan untuk memberikan gambaran nyata tentang jumlah, serta sebaran jaringan pelayanan yang ada yang tersebar di seluruh Indonesia.

Adapun sebaran jaringan pelayanan tersebut adalah puskesmas perawatan, puskesmas non perawatan, klinik dan balai pengobatan, rumah sakit pemerintah, rumah sakit swasta, rumah sakit TNI/POLRI, rumah sakit khusus/jiwa, apotik, optik, laboratorium, PMI, PPK hemodialisa dan dokter keluarga. Pada tabel 2 menunjukkan pembagian jaringan pelayanan kesehatan yang tersebar di seluruh Indonesia dibentuk dalam 12 Regional yaitu :

Tabel 2. 4 Pembagian Jaringan Pelayanan Kesehatan PT.Askes (Persero)

| DAFTAR PPK | KANTOR CABANG |
|---|---|
| <p style="text-align: center;">REGIONAL I : NAD (ACEH), DAN SUMATERA UTARA</p> | <ol style="list-style-type: none"> 1. Cabang Banda Aceh 2. Cabang Aceh Timur 3. Cabang Aceh Utara 4. Cabang Aceh Barat 5. Cabang Medan 6. Cabang Pematang Siantar 7. Cabang Karo 8. Cabang Sibolaga 9. Cabang Tanjung Balai. |
| <p style="text-align: center;">REGIONAL II :</p> | <ol style="list-style-type: none"> 1. Cabang Pekanbaru |

| | |
|---|---|
| <p>RIAU, KEPULAUAN RIAU, SUMATERA BARAT, DAN JAMBI</p> | <ol style="list-style-type: none"> 2. Cabang Duri 3. Cabang Batam 4. Cabang Padang 5. Cabang Solok 6. Cabang Bukittinggi 7. Cabang Jambi 8. Cabang Bungo. |
| <p>REGIONAL III : SUMATERA SELATAN, BANGKA BELITUNG, BENGKULU DAN LAMPUNG.</p> | <ol style="list-style-type: none"> 1. Cabang Palembang 2. Cabang Pangkal Pinang/Bangka Belitung 3. Cabang Prabumulih 4. Cabang Lubuk Linggau 5. Cabang Bengkulu 6. Cabang Bandar Lampung 7. Cabang Kotabumi 8. Cabang Metro. |
| <p>REGIONAL IV : DKI JAKARTA, BANTEN, DAN KALIMANTAN</p> | <ol style="list-style-type: none"> 1. Cabang Jakarta Pusat 2. Cabang Jakarta Selatan 3. Cabang Jakarta Timur 4. Cabang Jakarta Barat 5. Cabang Jakarta Utara 6. Cabang Tangerang 7. Cabang Pontianak 8. Cabang Singkawang 9. Cabang Sintang. |
| <p>REGIONAL V : JAWA BARAT</p> | <ol style="list-style-type: none"> 1. Cabang Bandung 2. Cabang Sukabumi 3. Cabang Bogor 4. Cabang Bekasi 5. Cabang Karawang 6. Cabang Sumedang 7. Cabang Cirebon 8. Cabang Tasikmalaya |
| <p>REGIONAL VI : JAWA TENGAH, DAN DAERAH ISTIMEWA YOGYAKARTA</p> | <ol style="list-style-type: none"> 1. Cabang Semarang 2. Cabang Tegal 3. Cabang Banyumas 4. Cabang Magelang |

| | |
|---|--|
| | <ol style="list-style-type: none"> 5. Cabang Boyolali 6. Cabang Surakarta 7. Cabang Kudus 8. Cabang Yogyakarta. |
| <p align="center">REGIONAL VII : JAWA TIMUR</p> | <ol style="list-style-type: none"> 1. Cabang Surabaya 2. Cabang Bojonegoro 3. Cabang Madiun 4. Cabang Kediri 5. Cabang Malang 6. Cabang Pasuruan 7. Cabang Jember 8. Cabang Banyuwangi 9. Cabang Sumenep. |
| <p align="center">REGIONAL VIII : KALIMANTAN TIMUR, KALIMANTAN SELATAN DAN KALIMANTAN TENGAH</p> | <ol style="list-style-type: none"> 1. Cabang Samarinda 2. Cabang Balikpapan 3. Cabang Tarakan 4. Cabang Banjarmasin 5. Cabang Hulu Sungai Tengah 6. Cabang Palangkaraya 7. Cabang Kotawaringin Timur 8. Cabang Barito Utara. |
| <p align="center">REGIONAL IX : SULAWESI SELATAN, SULAWESI BARAT DAN SULAWESI TENGGARA.</p> | <ol style="list-style-type: none"> 1. Cabang Makassar 2. Cabang Bantaeng 3. Cabang Bone 4. Cabang Parepare 5. Cabang Luwu 6. Cabang Kendari 7. Cabang Buton 8. Cabang Majene |
| <p align="center">REGIONAL X : SULAWESI UTARA, GORONTALO, SULAWESI TENGAH DAN MALUKU UTARA</p> | <ol style="list-style-type: none"> 1. Cabang Manado 2. Cabang Gorontalo 3. Cabang Palu 4. Cabang Poso 5. Cabang Maluku Utara. |
| <p align="center">REGIONAL XI : BALI, NUSA TENGGARA BARAT DAN NUSA TENGGARA TIMUR</p> | <ol style="list-style-type: none"> 1. Cabang Denpasar 2. Cabang Klungkung 3. Cabang Mataram |

| | |
|------------------------------------|---|
| | 4. Cabang Kupang 5. Cabang Sikka 6. Cabang Ende 7. Cabang Sumba Timur. |
| REGIONAL XII : MALUKU DAN PAPUA | 1. Cabang Ambon 2. Cabang Jayapura 3. Cabang Sorong 4. Cabang Biak Numfor. |

2.2.8 Administrasi Profil Wilayah Kota Palu

Kota Palu merupakan bagian dari wilayah administrasi Propinsi Sulawesi Tengah dengan luas wilayah 395.06 km² dibagi dalam empat kecamatan dan empat puluh tiga kelurahan. Adapun batas-batas administratif Kota Palu adalah sebagai berikut :

Tabel 2. 5 Batas Wilayah Administratif

| Geografi | Kota Palu |
|-----------------|--------------------|
| Sebelah Utara | Teluk Palu |
| Sebelah Selatan | Kecamatan Binangga |
| Sebelah Timur | Kecamatan Biromeru |
| Sebelah Barat | Bandara Mutiara |

Tabel 2. 6 Pola Penyebaran Wilayah Kota Palu untuk Kecamatan Palu Barat dengan luas wilayah 57.47 km²

| No | Daftar Nama Kelurahan / Desa |
|----|------------------------------|
| 1 | Kelurahan / Desa Baru |
| 2 | Kelurahan / Desa Boyaoge |
| 3 | Kelurahan / Desa Lere |
| 4 | Kelurahan / Desa Siranindi |
| 5 | Kelurahan / Desa Nunu |
| 6 | Kelurahan / Desa Ujuna |
| 7 | Kelurahan / Desa Kamonji |

| | |
|----|----------------------------|
| 8 | Kelurahan / Desa Duyu |
| 9 | Kelurahan / Desa Balaroa |
| 10 | Kelurahan / Desa Donggala |
| 11 | Kelurahan / Desa Kabonena |
| 12 | Kelurahan / Desa Silae |
| 13 | Kelurahan / Desa Buluri |
| 14 | Kelurahan / Desa Tipo |
| 15 | Kelurahan / Desa watusampu |

Tabel 2.7 Pola Penyebaran Wilayah Kota Palu untuk Kecamatan Palu Selatan dengan luas wilayah 61.35 km²

| No | Daftar Nama Kelurahan / Desa |
|----|-----------------------------------|
| 1 | Kelurahan / Desa Birobuli Selatan |
| 2 | Kelurahan / Desa Birobuli Utara |
| 3 | Kelurahan / Desa Petobo |
| 4 | Kelurahan / Desa Kawatuna |
| 5 | Kelurahan / Desa Tanamodindi |
| 6 | Kelurahan / Desa Lolu Selatan |
| 7 | Kelurahan / Desa Lolu Utara |
| 8 | Kelurahan / Desa Tatura Selatan |
| 9 | Kelurahan / Desa Tatura Utara |
| 10 | Kelurahan / Desa Tawanjuka |
| 11 | Kelurahan / Desa Palupi |
| 12 | Kelurahan / Desa Pengawu |

Tabel 2.8 Pola Penyebaran Wilayah Kota Palu untuk Kecamatan Palu Timur dengan luas wilayah 186.55 km²

| No | Daftar Nama Kelurahan / Desa |
|----|--------------------------------|
| 1 | Kelurahan / Desa Besusu Barat |
| 2 | Kelurahan / Desa Besusu Tengah |
| 3 | Kelurahan / Desa Besusu Timur |
| 4 | Kelurahan / Desa Layana Indah |

| | |
|---|-------------------------------|
| 5 | Kelurahan / Desa Poboya |
| 6 | Kelurahan / Desa Lasoani |
| 7 | Kelurahan / Desa TaipaTalise |
| 8 | Kelurahan / Desa MamboroTondo |

Tabel 2.9 Pola Penyebaran Wilayah Kota Palu untuk Kecamatan Palu Utara dengan luas wilayah 89.69 km²

| No | Daftar Nama Kelurahan / Desa |
|----|----------------------------------|
| 1 | Kelurahan / Desa Lambara |
| 2 | Kelurahan / Desa Panau |
| 3 | Kelurahan / Desa Baiya |
| 4 | Kelurahan / Desa Pantoloan |
| 5 | Kelurahan / Desa Kaymalue Pajeko |
| 6 | Kelurahan / Desa Kayumalue Ngapa |
| 7 | Kelurahan / Desa Taipa |
| 8 | Kelurahan / Desa Mamboro |

Tabel 2.10 Pola Penyebaran Wilayah Kota Palu dan Jumlah Penduduk

| Kecamatan | Laki-laki | Perempuan | Laki-laki + perempuan | Sex ratio |
|------------------|----------------|----------------|-----------------------|------------|
| (1) | (2) | (3) | (4) | (5) |
| Palu Barat | 49.743 | 49.048 | 98.791 | 101 |
| Palu Selatan | 61.379 | 60.524 | 121.903 | 101 |
| Palu Timur | 38.470 | 37.262 | 75.732 | 103 |
| Palu Utara | 19.615 | 19.256 | 38.871 | 102 |
| Kota Palu | 169.207 | 166.090 | 335.297 | 102 |