



## Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

## BAB II

### LANDASAN TEORI

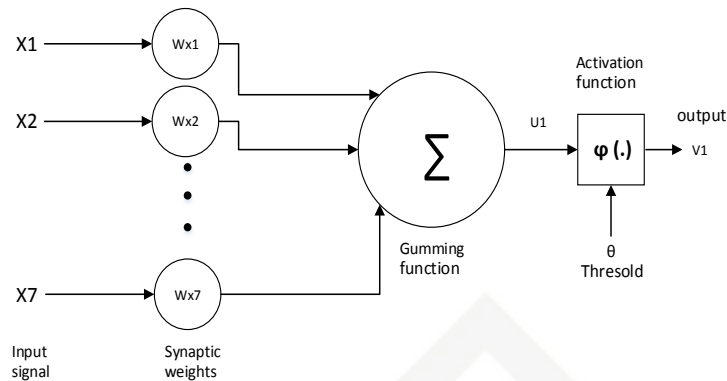
#### 2.1 Jaringan Syaraf Tiruan

Jaringan Syaraf Tiruan (JST) merupakan salah satu representasi dari sebuah kinerja otak manusia yang disimulasikan ke sebuah sistem atau aplikasi tertentu dan mengembangkan simulasi kerja otak manusia ke sebuah sistem yang akan dibuat (Kusumadewi, 2004).

Hal yang ingin dicapai dengan melatih JST adalah untuk mencapai keseimbangan antara kemampuan memorisasi dan generalisasi. Yang dimaksud kemampuan memorisasi adalah kemampuan JST untuk mengambil kembali secara sempurna sebuah pola yang telah dipelajari. Kemampuan generalisasi adalah kemampuan JST untuk menghasilkan respon yang bisa diterima terhadap pola-pola yang sebelumnya telah dipelajari. Hal ini sangat bermanfaat bila pada suatu saat ke dalam JST itu di masukkan informasi baru yang belum pernah dipelajari, maka JST itu masih akan tetap dapat memberikan tanggapan yang baik, memberikan keluaran yang mendekati (Puspaningrum, 2006).

##### 2.1.1 Model *Neuron*

Satu sel syaraf terdiri dari 3 bagian, yaitu: fungsi penjumlahan (*summing function*), fungsi aktivasi (*activation function*), dan keluaran (*output*).



**Gambar 2.1 Model *Neuron* (Hartanto, 2009)**

Jika dilihat, *neuron* buatan diatas mirip dengan sel *neuron* biologis. Informasi (masukan) akan dikirim ke *neuron* dengan bobot tertentu. Masukan ini akan diproses oleh suatu fungsi yang akan menjumlahkan nilai-nilai bobot yang ada. Hasil penjumlahan kemudian akan 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, jika tidak, maka *neuron* tersebut akan mengirimkan *output* melalui bobot-bobot *output* ke semua *neuron* yang berhubungan dengannya.

Berdasarkan dari penjelasan sebelumnya sehingga dapat disimpulkan bahwa *neuron* terdiri dari 3 elemen pembentuk yaitu:

1. Himpunan unit-unit yang dihubungkan dengan jalur koneksi. Jalur-jalur tersebut memiliki bobot yang berbeda-beda. Bobot yang bernilai positif akan memperkuat sinyal yang dibawa. Jumlah struktur dan pola hubungan antar unit-unit tersebut akan menentukan arsitektur jaringan.
2. Suatu unit penjumlahan yang akan menjumlahkan masukan sinyal yang sudah dikalikan dengan bobotnya.
3. Fungsi aktivasi yang menentukan apakah sinyal dari masukan *neuron* akan diteruskan ke *neuron* lain atau tidak.



#### Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

### 2.1.2 Konsep Dasar Jaringan Syaraf Tiruan

Setiap pola-pola informasi *input* dan *output* yang diberikan kedalam JST diproses dalam *neuron*. *Neuron-neuron* tersebut terkumpul di dalam lapisan-lapisan yang disebut *neuron layers* (IJDAR,2003 dikutip oleh Sinaga, 2012). Lapisan-lapisan penyusun JST tersebut dapat dibagi menjadi 3, yaitu:

1. Lapisan *Input*, unit-unit di dalam lapisan *input* disebut unit-unit *input*. Unit-unit *input* tersebut menerima pola masukan data dari luar yang menggambarkan suatu permasalahan.
2. Lapisan Tersembunyi, unit-unit di dalam lapisan tersembunyi disebut unit-unit tersembunyi. Dimana *outputnya* tidak dapat secara langsung diamati.
3. Lapisan *Output*, unit-unit di dalam lapisan *output* disebut unit-unit *output*. *Output* dari lapisan ini merupakan solusi JST terhadap suatu permasalahan.

### 2.1.3 Arsitektur Jaringan Syaraf Tiruan

Pada JST, *neuron-neuron* akan dikumpulkan dalam lapisan-lapisan (*layer*) yang disebut dengan lapisan *neuron* (*neuron layers*). *Neuron-neuron* pada satu lapisan akan dihubungkan dengan lapisan-lapisan sebelum dan sesudahnya. Informasi yang diberikan pada jaringan syaraf akan dirambatkan lapisan ke lapisan, mulai dari lapisan masukan sampai ke lapisan keluaran melalui lapisan tersembunyi (*hidden layer*). Faktor terpenting dalam menentukan kelakuan suatu *neuron* adalah fungsi aktivasi dan pola bobotnya. Umumnya *neuron-neuron* yang terletak pada lapisan yang sama akan memiliki keadaan yang sama sehingga pada setiap lapisan yang sama *neuron-neuron* memiliki fungsi aktivasi yang sama. Bila *neuron-neuron* pada suatu lapisan (misal lapisan tersembunyi) akan dihubungkan dengan *neuron-neuron* pada lapisan lain (misal lapisan keluaran) maka setiap *neuron* pada lapisan tersebut (lapisan tersembunyi) juga harus dihubungkan dengan setiap *neuron* pada lapisan lainnya (lapisan keluaran). Terdapat 3 macam arsitektur JST, yaitu (Kusumadewi, 2004):



- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

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

Jaringan ini hanya memiliki 1 lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima masukan kemudian secara langsung akan mengolahnya menjadi keluaran tanpa harus melalui lapisan tersembunyi. Pada gambar berikut *neuron-neuron* pada kedua lapisan saling berhubungan. Seberapa besar hubungan antara 2 *neuron* ditentukan oleh bobot yang bersesuaian. Semua unit masukan akan dihubungkan dengan setiap unit keluaran.

### 2. Jaringan dengan banyak lapisan (*multilayer net*)

Jaringan ini memiliki 1 atau lebih lapisan yang terletak diantara lapisan masukan dan lapisan keluaran. 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 tunggal, tentu saja dengan pembelajaran yang lebih rumit. Pada banyak kasus, pembelajaran pada jaringan dengan banyak lapisan ini lebih sukses dalam menyelesaikan masalah.

### 3. Jaringan dengan lapisan kompetitif (*competitive layer net*)

Pada jaringan ini sekumpulan *neuron* bersaing untuk mendapatkan hak menjadi aktif. Umumnya hubungan antar *neuron* pada lapisan kompetitif ini tidak diperlihatkan pada diagram arsitektur.

## 2.1.4 Normalisasi

Normalisasi adalah proses transformasi nilai menjadi kisaran 0 dan 1. Normalisasi merupakan proses penskalaan nilai atribut dari data sehingga bisa jatuh pada range tertentu (Elvia Budianita dan Widodo Prijodiprodjo, 2013). Tujuan dari normalisasi data adalah untuk mendapatkan data dengan ukuran yang lebih kecil, mewakili data asli tanpa kehilangan karakteristiknya. Normalisasi terbagi menjadi 5 jenis metode pencarian, yaitu : *min-max*, *z-score*, *decimal scaling*, *sigmoidal* dan *softmax*. Metode pencarian yang paling banyak digunakan adalah *min-max*.



1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Metode *min-max* adalah metode normalisasi dengan melakukan transformasi linier terhadap data asli. Berikut adalah Persamaan (2.1) dari metode pencarian *min-max* adalah sebagai berikut :

$$X^* = \frac{X - \min(X)}{\max(X) - \min(X)} \tag{2.1}$$

Keterangan:

- X\* = nilai setelah dinormalisasi
- X = nilai sebelum dinormalisasi
- Min(X) = nilai minimum dari fitur
- Max(X) = nilai maksimum dari fitur

### 2.1.5 Fungsi Aktivasi

Fungsi aktivasi (batas ambang atau *threshold*) merupakan fungsi untuk menentukan nilai keluaran (*output*) berdasarkan nilai total masukan (*input*) pada *neuron* (Kusumadewi, 2004). Fungsi aktivasi pada jaringan syaraf tiruan diantaranya adalah hard limit, purelin dan sigmoid. Syarat-syarat fungsi aktivasi pada metode BPNN adalah kontinu, terdiferensial dengan mudah dan fungsi aktivasi yang tidak turun. Fungsi aktivasi yang memenuhi syarat-syarat tersebut adalah :

#### 1. Fungsi Sigmoid Biner

Pada fungsi sigmoid biner, nilai *output* terletak pada interval range 0 sampai dengan 1. Dapat didefinisikan pada persamaan (2.2) dan persamaan (2.3):

$$y = f(x) = \frac{1}{1 + e^{-x}} \tag{2.2}$$

Dengan turunan

$$y' = f'(x) = f(x)[1 - f(x)] \tag{2.3}$$



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

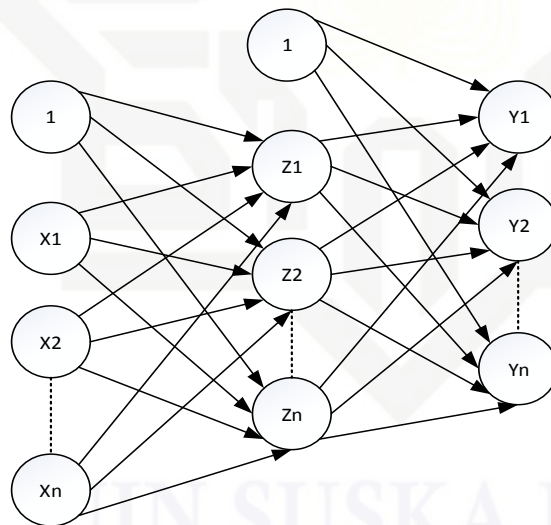
keluaran akan disebarkan mundur (*backward*) pada lapisan tersembunyi kemudian dari lapisan tersembunyi menuju lapisan masukan.

Tahap pelatihan ini merupakan langkah untuk melatih suatu jaringan syaraf tiruan, yaitu dengan cara melakukan perubahan bobot, sedangkan penyelesaian masalah akan dilakukan jika proses pelatihan tersebut telah selesai, fase ini disebut fase pengujian (Puspaningrum, 2006).

### 2.2.2 Arsitektur Jaringan Backpropagation neural network

Setiap unit dari layer *input* pada jaringan BPNN selalu terhubung dengan setiap unit yang berada pada *layer* tersembunyi, demikian juga setiap unit *layer* tersembunyi selalu terhubung dengan unit pada *layer output*. Jaringan BPNN terdiri dari banyak lapisan (*multilayer network*) yaitu:

1. Lapisan *input* (1 buah), yang terdiri dari 1 hingga n unit *input*.
2. Lapisan *hidden* (minimal 1 buah), yang terdiri dari 1 hingga p unit *hidden*.
3. Lapisan *output* (1 buah), yang terdiri dari 1 hingga m unit *output*.



Gambar 2.2 Arsitektur jaringan *Backpropagation neural network* (Jong, 2004)



### Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

### 2.2.3 Algoritma Pelatihan BPNN

Aturan pelatihan jaringan BPNN terdiri dari 2 tahapan, *feedforward* dan *backward propagation*. Pada jaringan diberikan sekumpulan contoh pelatihan yang disebut set pelatihan. Set pelatihan ini digambarkan dengan sebuah vektor feature yang disebut dengan *vektor input* yang diasosiasikan dengan sebuah *output* yang menjadi target pelatihannya. Dengan kata lain set pelatihan terdiri dari vektor *input* dan juga vektor *output* target. Keluaran dari jaringan berupa sebuah vektor *output* aktual. Selanjutnya dilakukan perbandingan antara *output* aktual yang dihasilkan dengan *output* target dengan cara melakukan pengurangan diantara kedua *output* tersebut. Hasil dari pengurangan berupa *error*. *Error* dijadikan sebagai dasar dalam melakukan perubahan dari setiap bobot dengan mempropagationkannya kembali.

Setiap perubahan bobot yang terjadi dapat mengurangi *error*. Siklus perubahan bobot (*epoch*) dilakukan pada setiap set pelatihan sehingga kondisi berhenti dicapai, yaitu bila mencapai jumlah *epoch* yang diinginkan atau hingga sebuah nilai ambang yang ditetapkan terlampaui.

Algoritma pelatihan jaringan BPNN terdiri dari 3 tahapan yaitu:

1. Tahap umpan maju (*Feedforward*).
2. Tahap umpan mundur (*Backforward*).
3. Tahap pembaharuan bobot dan bias.

Secara rinci algoritma pelatihan jaringan BPNN dapat diuraikan sebagai berikut:

1. Langkah 0 : Inisialisasi nilai bobot dengan nilai acak yang kecil.
2. Langkah 1 : Selama kondisi berhenti tidak terpenuhi (bernilai salah), maka lakukan langkah sebagai berikut :
  - a. Untuk tiap-tiap pasangan pelatihan, lakukan :
 

Tahap Perambatan Maju (*Feedforward Propagation*)

    1. Tiap-tiap unit *input* ( $X_i$ ,  $i = 1, 2, 3, \dots, n$ ) menerima sinyal  $x_i$  dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (*hidden layer*).



## Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2. Tiap-tiap unit tersembunyi ( $Z_j$ ,  $j = 1, 2, 3, \dots, p$ ) menjumlahkan sinyal-sinyal *input* berbobot dengan persamaan (2.6) :

$$z\_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (2.6)$$

dimana  $v_0$  = bias dan  $v$  = bobot.

3. Gunakan fungsi aktivasi untuk menghitung sinyal *output* dengan persamaan (2.7):

$$z_j = f(z\_in_j) \quad (2.7)$$

dan kirimkan sinyal tersebut ke semua unit lapisan atasnya (unit-unit *output*).

4. Tiap-tiap unit *output* ( $Y_k$ ,  $k=1, 2, 3, \dots, m$ ) menjumlahkan sinyal-sinyal *input* berbobot dengan persamaan (2.8):

$$y\_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk} \quad (2.8)$$

dimana  $w_0$  = bias dan  $w$  = bobot.

Gunakan fungsi aktivasi untuk menghitung sinyal *output* dengan persamaan (2.19):

$$y_k = f(y\_in_k) \quad (2.9)$$

Dan kirimkan sinyal tersebut ke semua unit di lapisan sisanya (unit-unit *output*).

- b. Untuk tiap-tiap pasangan pelatihan, lakukan :

Tahap Perambatan-Balik (*Backward propagation*)

1. Tiap-tiap unit *output* ( $Y_k, k = 1, 2, 3, \dots, m$ ) menerima pola target yang berhubungan dengan pola *input* pelatihan, kemudian hitung *error*-nya dengan persamaan berikut dengan persamaan (2.10):

$$\delta_k = (t_k - y_k) f' (y\_in_k) \quad (2.10)$$

dimana  $t$  = target *output*.

$f'$  = turunan dari fungsi aktivasi



#### Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperoleh nilai  $w_{jk}$ ) dengan persamaan (2.11):

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (2.11)$$

dimana  $\alpha$  = learning rate

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $w_{0k}$ ) dengan persamaan (2.12):

$$\Delta w_{0k} = \alpha \delta_k \quad (2.12)$$

Kirimkan  $\delta_k$  ini ke unit-unit yang ada di lapisan paling kanan.

2. Tiap-tiap unit tersembunyi ( $Z_j$ ,  $j=1,2,3,\dots,p$ ) menjumlahkan delta *inputnya* (dari unit-unit yang berada pada lapisan di kanannya) dengan persamaan (2.13):

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (2.13)$$

Kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi *error* dengan persamaan (2.14):

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (2.14)$$

1. Kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{jk}$ ) dengan persamaan (2.15):

$$\Delta v_{jk} = \alpha \delta_j x_i \quad (2.15)$$

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{0j}$ ) dengan persamaan (2.16):

$$\Delta v_{0j} = \alpha \delta_j \quad (2.16)$$

- c. Perubahan bobot dan bias :

1. Tiap-tiap unit tersembunyi ( $Z_j$ ,  $j=1,2,3,\dots,p$ ) memperbaiki bias dan bobotnya ( $i = 0,1,2,\dots,n$ ) dengan persamaan (2.17):



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \tag{2.17}$$

2. Tiap-tiap unit *output* ( $Y_k, k=1,2,3,\dots,m$ ) memperbaiki bias dan bobotnya ( $j = 0,1,2,\dots,p$ ) dengan persamaan (2.18):

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \tag{2.18}$$

- d. Uji apakah kondisi berhenti terpenuhi.

### 2.2.4 Algoritma Pengujian BPNN

Dengan didapatnya nilai *output* yang paling mendekati target, maka bobot dan bias akhir dari pelatihan disimpan dan dilakukan proses pengujian. Algoritma pengujian *backpropagation* adalah sebagai berikut :

1. Langkah 0 : Inisialisasi bobot
2. Langkah 1 : Untuk setiap *vektor* masukan
3. Langkah 2 : for  $i=1\dots n$ : atur aktivasi unit masukan  $x$
4. Langkah 3 : for  $j=1\dots p$ :

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

$$z_j = f(z_{in_j}) \tag{2.19}$$

5. Langkah 4 : for  $k=1\dots m$ :

$$y_{in_k} = w_{0k} + \sum_{j=1}^p z_j w_{jk}$$

$$y_k = f(y_{in_k}) \tag{2.20}$$

6. Langkah 5 : jika  $Y_k \geq 0.5$  maka  $Y_k = 1$ , else  $Y_k = 0$ .

### 2.3 Prediksi

Prediksi adalah ramalan atau perkiraan. Menurut Kamus Besar Bahasa Indonesia (KBBI), prediksi adalah hasil dari kegiatan memprediksi atau meramal atau memperkirakan. Prediksi bisa berdasarkan metode ilmiah ataupun subjektif belaka. Prediksi menggunakan metode ilmiah, contohnya dalam sebuah universitas





Mahasiswa Baru (PMB) Undangan, PMB Jalur Tulis dan PMB Mahasiswa Luar Negeri.

SNMPTN dan SBMPTN merupakan penerimaan mahasiswa baru diselenggarakan oleh Kementerian Riset, Teknologi dan Pendidikan Tinggi (Kemenristekdikti) dan mata pelajaran yang di ujikan hanya Program Studi (Prodi) umum. SNMPTN indikator penerimaan mahasiswa berdasarkan prestasi akademik yang dilihat dari nilai rapor. SBMPTN indikator penerimaan mahasiswa berdasarkan nilai dari hasil ujian yang dilaksanakan secara nasional.

SPAN-PTKIN dan UM-PTKIN adalah penerimaan mahasiswa baru yang diselenggarakan oleh Kementerian Agama dan mata pelajaran yang di ujikan hanya Program Studi (Prodi) Agama. SPAN-PTKIN indikator penerimaan mahasiswa berdasarkan prestasi akademik yang dilihat dari nilai rapor. UM-PTKIN indikator penerimaan mahasiswa berdasarkan nilai dari hasil ujian yang dilaksanakan secara nasional.

PMB merupakan penerimaan mahasiswa yang diselenggarakan UIN Suska Riau dan mata pelajaran yang diujikan adalah semua Prodi. PMB Undangan indikator penerimaan mahasiswa berdasarkan prestasi akademik yang dilihat dari nilai rapor. PMB Jalur Tulis indikator penerimaan mahasiswa berdasarkan hasil ujian tertulis. PMB Mahasiswa Luar Negeri seleksi penerimaan mahasiswa yang dilaksanakan khusus bagi siswa yang berasal dari luar negeri.

## 2.6 Penelitian Terkait

Penelitian terkait yaitu penelitian sebelumnya yang memiliki hubungan dengan penelitian ini, sehingga ada perbandingan atau sumber yang lebih jelas bagaimana penelitian ini dibuat. Perbandingan dilakukan untuk mencari mana metode yang paling baik diterapkan untuk studi kasus prediksi masa studi mahasiswa. Berikut adalah rangkuman dari referensi yang penulis gunakan pada penelitian ini dapat di lihat pada Tabel 2.1:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Tabel 2.1 Penelitian terkait

NO	Penulis	Judul	Tahun	Metode	Hasil
1	Risky Meliawati, Oni Soesanto, Dwi Kartini	Penerapan Metode <i>Learning Vector Quantization (Lvq)</i> Pada Prediksi Jurusan Di Sma Pgri 1 Banjarbaru	2016	<i>Learning Vector Quantization (LVQ)</i>	Hasil pengujian pada sistem menunjukkan bahwa akurasi yang mendekati dengan hasil sebenarnya dengan jumlah iterasi yang berbeda adalah akurasi 79,31% untuk iterasi= 60 dan 90. Pada penelitian dengan nilai alpha yang berubah didapat akurasi yang mendekati dengan hasil sebenarnya adalah 75,86% dengan nilai alpha ( $\alpha$ ) = 0,14.
2	Elvia Budianita, Ulti Desi Arni	Penerapan <i>Learning Vector Quantization</i> Penentuan Bidang Konsentrasi Tugas Akhir (Studi Kasus: Mahasiswa Teknik Informatika UIN Suska Riau)	2015	<i>Learning Vector Quantization (LVQ)</i>	Nilai parameter yang digunakan meliputi nilai <i>learning rate</i> ( $\alpha$ ) = 0,9, nilai minimal <i>learning rate</i> (Mina) = 0,01, dan nilai pengurangan $\alpha$ adalah 0,1, merupakan nilai parameter yang sudah cukup efektif dan efisien dalam



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

					melakukan penentuan bidang konsentrasi studi tugas akhir mahasiswa mencapai tingkat akurasi 80%.
3	Candra Dewi, M. Muslikh	Perbandingan Akurasi Backpropagation Neural Network dan Adaptive Neuro Fuzzy Inference System (ANFIS) untuk memprediksi cuaca	2013	<i>Backpropagation Neural Network dan Adaptive Neuro Fuzzy Inference System(ANFIS)</i>	Berdasarkan hasil uji coba menunjukkan bahwa metode BPNN menghasilkan nilai RMSE yang lebih kecil jika dibandingkan dengan ANFIS. Dan, BPNN memiliki tingkat akurasi yang lebih tinggi jika dibandingkan dengan ANFIS
4	Finki Dona Marleny dan Vincent Suhartono	Prediksi Indeks Harga Saham Gabungan (IHSG) Menggunakan Backpropagation Neural Network	2013	<i>Backpropagation Neural Network</i>	Pada penelitian ini hasil terbaik yang didapatkan menggunakan Levenberg Marquardt (trainlm) dengan akurasi sebesar 95,93%,setelah dibandingkan dengan performance maksimum dari algoritma BPNN yang lain.



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

<p>Maria Agustin, Toni Prahasto</p>	<p>Penggunaan Jaringan Syaraf Tiruan Backpropagation Untuk Seleksi Penerimaan Mahasiswa Baru Pada Jurusan Teknik Komputer Di Politeknik Negeri Sriwijaya</p>	<p>2012</p>	<p><i>Backpropagation Neural Network</i></p>	<ol style="list-style-type: none"> <li>1. Jaringan syaraf tiruan dengan 1 <i>hidden layer</i>, dengan jumlah <i>neuron</i> 50, iterasi 1000 dengan</li> <li>2. Jaringan syaraf tiruan dengan 2 <i>hidden layer</i>, dengan jumlah <i>neuron</i> 50, iterasi 4000 dengan fungsi aktivasi <i>tansig</i> mampu mendekati regresi 0.7944.</li> <li>3. Jaringan syaraf tiruan dengan 3 <i>hidden layer</i>, dengan jumlah <i>neuron</i> 35, iterasi 5000 dengan fungsi aktivasi <i>tansig</i> mampu mendekati regresi 0.8563.</li> </ol>
-------------------------------------	--	-------------	--	---

<p>Nazla Nurmila, Aris Sugiharto, Dan Eko Adi Sarwoko</p>	<p>Algoritma Back Propagation Neural Network Untuk Pengenalan Pola Karakter Huruf Jawa</p>	<p>2010</p>	<p><i>Backpropagation Neural Network</i></p>	<p>Rata-rata keakuratan <i>BPNN</i> dalam Mengenali pola karakter huruf Jawa adalah sebesar 99.563% untuk data sampel berupa data pelatihan, 61.359% untuk data sampel diluar Data pelatihan, dan 75% untuk data sampel data pelatihan dan di luar data pelatihan.</p>
---	--	-------------	--	--

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.