

## Pengembangan Aplikasi Prediksi Pertumbuhan Ekonomi Indonesia dengan Jaringan Syaraf Tiruan *Backpropagation*

Erlangga, Sukmawati Nur Endah dan Eko Adi Sarwoko

Jurusan Ilmu Komputer / Informatika, Fakultas Sains dan Matematika, Universitas Diponegoro  
Jl. Prof. Soedarto, Tembalang, Semarang  
[erlangga.company@gmail.com](mailto:erlangga.company@gmail.com)

### Abstrak

Pertumbuhan ekonomi merupakan salah satu indikator penting dalam menganalisis pembangunan perekonomian yang terjadi di suatu negara. Dengan mengetahui pertumbuhan ekonomi di masa mendatang, dapat memberikan gambaran terhadap situasi moneter di suatu negara. Angka pertumbuhan ekonomi yang tidak selalu linier, memberi kesulitan tersendiri dalam melakukan proses prediksi. Untuk itu diperlukan suatu metode yang mampu menangani karakteristik data pertumbuhan ekonomi yang terkadang bersifat non-linier, salah satunya adalah metode *backpropagation*. Adanya data-data masa lalu mengenai pertumbuhan ekonomi, menjadikan jaringan syaraf tiruan metode *backpropagation* dapat diterapkan untuk memprediksi pertumbuhan ekonomi. Pada metode *backpropagation* terjadi penyesuaian nilai bobot dan bias yang semakin baik pada proses pelatihan, sehingga target keluaran lebih mendekati ketepatan. Berdasarkan hasil pengujian yang telah dilakukan, performa terbaik yang didapat dari perbandingan kombinasi jumlah neuron *hidden layer* dan *learning rate*, mampu menghasilkan prediksi yang mendekati angka pertumbuhan ekonomi faktual, yakni 5,86%. Hasil ini menunjukkan tingkat keakuratan sebesar 99,92% untuk prediksi pertumbuhan ekonomi Indonesia di tahun 2013 dengan data faktual sebesar 5,78%.

**Kata kunci :** *Backpropagation*, Pertumbuhan Ekonomi Indonesia, Jaringan Syaraf Tiruan

### 1. Pendahuluan

Dalam setiap tahun Anggaran Pendapatan dan Belanja Negara (APBN), pemerintah menetapkan berbagai target makro ekonomi yang salah satunya adalah pertumbuhan ekonomi. Pertumbuhan ekonomi menjadi salah satu indikator penting dalam menganalisis pembangunan perekonomian yang terjadi di suatu negara [1].

Prediksi pertumbuhan ekonomi sangat penting adanya dalam perencanaan. Dengan mengetahui pertumbuhan ekonomi di masa yang akan datang, dapat memberikan gambaran terhadap kondisi moneter suatu negara kepada pelaku ekonomi dalam mengambil keputusan untuk berinvestasi, produksi maupun konsumsi pada masa mendatang. Angka besaran pertumbuhan ekonomi juga penting bagi pemerintah salah satunya dalam menetapkan besaran penerimaan pemerintah dari sektor pajak serta besaran pengeluaran pemerintah [1]. Namun, data pertumbuhan ekonomi yang dinamis dan rumit membuat angka pertumbuhan ekonomi sulit untuk diprediksi. Dengan adanya data-data di masa lalu mengenai angka pertumbuhan ekonomi, menjadikan

jaringan syaraf tiruan metode *backpropagation* dapat diterapkan untuk melakukan prediksi pertumbuhan ekonomi Indonesia. Metode *backpropagation* selain bersifat objektif, juga memiliki dasar matematis yang kuat, objektif dan algoritma yang dapat membentuk persamaan dan nilai koefisien dalam formula dengan meminimalkan jumlah kuadrat galat *error* melalui model yang dikembangkan (*training set*) [2].

### 2. Kajian Literatur

#### 2.1. Pertumbuhan Ekonomi

Secara singkat pertumbuhan ekonomi adalah proses kenaikan output per kapita dalam jangka panjang. Pengertian tersebut menekankan pada tiga aspek, yaitu proses, output per kapita, dan jangka panjang. Dalam pengertian lebih longgar, pertumbuhan ekonomi diartikan sebagai kenaikan output total (GDP) atau Produk Domestik Bruto dalam jangka panjang tanpa memandang apakah kenaikan itu lebih kecil atau lebih besar daripada laju pertumbuhan penduduk atau apakah diikuti oleh pertumbuhan struktur perekonomian atau tidak [3].

Pertumbuhan ekonomi sebagai ukuran kuantitatif yang menggambarkan perkembangan suatu perekonomian dalam satu tahun tertentu apabila dibandingkan dengan tahun sebelumnya. Untuk perhitungan angka pertumbuhan ekonomi [4], dapat menggunakan persamaan :

$$g = \frac{GDP_n - GDP_{(n-1)}}{GDP_{(n-1)}} \times 100\%$$

- g = tingkat pertumbuhan ekonomi (persentase)
- GDP<sub>n</sub> = (Gross Domestic Product atau PDB) adalah pendapatan nasional riil
- GDP<sub>(n-1)</sub> = pendapatan nasional riil pada tahun sebelumnya, yaitu tahun (n - 1)

## 2.2. Produk Domestik Bruto (PDB)

PDB pada dasarnya merupakan jumlah nilai tambah yang dihasilkan oleh seluruh unit usaha dalam suatu negara tertentu, atau merupakan jumlah nilai barang dan jasa akhir yang dihasilkan oleh seluruh unit ekonomi pada jangka waktu tertentu, biasanya satu tahun.

Berdasarkan Badan Pusat Statistik (BPS) Indonesia, untuk menghitung angka-angka PDB dapat menggunakan pendekatan produksi yang bersumber pada seluruh unit-unit produksi yang ada pada suatu negara. Unit-unit produksi tersebut dalam penyajian ini dikelompokkan menjadi 9 lapangan usaha (sektor), yaitu :

- Pertanian, Peternakan, Kehutanan dan Perikanan
- Pertambangan dan Penggalian
- Industri Pengolahan
- Listrik, Gas dan Air Bersih
- Konstruksi
- Perdagangan, Hotel dan Restoran
- Pengangkutan dan Komunikasi
- Keuangan, Real Estate dan Jasa Perusahaan
- Jasa-jasa, termasuk jasa pelayanan pemerintah

Setiap sektor tersebut dapat dirinci lagi menjadi sub-sub sektor.

## 2.3. Jaringan Syaraf Tiruan Backpropagation

Dalam pelatihan *backpropagation* terdapat 3 fase. Fase pertama adalah fase propagasi maju (*feedforward*). Pola masukan dihitung maju

mulai dari layar masukan hingga layar keluaran menggunakan fungsi aktivasi yang ditentukan. Fase kedua adalah fase propagasi mundur (*backpropagation*). Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasi mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di layar keluaran. Fase yang ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi [5].

### Fase I : Feedforward

Selama propagasi maju, sinyal masukan (=  $x_i$ ) dipropagasikan ke layar tersembunyi menggunakan fungsi aktivasi yang ditentukan. Keluaran dari setiap unit layar tersembunyi (=  $z_j$ ) tersebut selanjutnya dipropagasikan maju lagi ke layar tersembunyi di atasnya menggunakan fungsi aktivasi yang ditentukan. Demikian seterusnya hingga menghasilkan keluaran jaringan (=  $y_k$ ).

Hasil keluaran jaringan (=  $y_k$ ) dibandingkan dengan target yang harus dicapai (=  $t_k$ ). Selisih  $|t_k - y_k|$  adalah kesalahan yang terjadi. Jika kesalahan ini lebih kecil dari batas toleransi yang ditentukan, maka iterasi dihentikan. Tetapi apabila kesalahan masih besar dari batas toleransinya, maka bobot setiap garis dalam jaringan dimodifikasi untuk mengurangi kesalahan yang terjadi.

### Fase II : Backpropagation

Berdasarkan kesalahan  $t_k - y_k$ , dihitung faktor  $\delta_k$  ( $k = 1, 2, \dots, m$ ) yang dipakai untuk mendistribusikan kesalahan di unit  $y_k$ .  $\delta_k$  juga dipakai untuk mengubah bobot garis yang berhubungan langsung dengan unit keluaran.

Dengan cara yang sama, dihitung faktor  $\delta_j$  di setiap unit layar tersembunyi sebagai dasar perubahan bobot semua garis yang berasal dari unit tersembunyi di layar bawahnya. Demikian seterusnya hingga semua faktor  $\delta$  di unit tersembunyi yang berhubungan langsung dengan unit masukan dihitung.

### Fase III : Perubahan bobot

Setelah semua faktor  $\delta$  dihitung, bobot semua garis dimodifikasi bersamaan. Perubahan bobot suatu garis didasarkan atas faktor  $\delta$  neuron

di layar atasnya. Sebagai contoh, perubahan bobot garis yang menuju ke layar keluaran didasarkan atas  $\delta_k$  yang ada di unit keluaran.

Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi dan kesalahan. Iterasi akan dihentikan jika jumlah iterasi yang dilakukan sudah melebihi jumlah maksimum iterasi yang ditetapkan, atau jika kesalahan yang terjadi sudah lebih kecil dari batas toleransi MSE (*Mean Square Error*) yang diizinkan.

#### 2.4. Preprocessing / Normalisasi

Pada proses perkiraan dengan menggunakan *backpropagation*, sebelum dilakukan proses pelatihan, data *input* dan target *output* harus dinormalisasi terlebih dahulu. Normalisasi adalah penskalaan terhadap nilai-nilai masuk ke dalam suatu range dari fungsi aktivasi yang digunakan dalam jaringan.

Jika menggunakan fungsi aktivasi sigmoid biner, data harus ditransformasikan terlebih dahulu karena range keluaran fungsi aktivasi sigmoid adalah [0,1]. Data bisa ditransformasikan ke interval [0,1]. Tapi akan lebih baik jika ditransformasikan ke interval yang lebih kecil, misal pada interval [0.1, 0.9], karena fungsi sigmoid merupakan fungsi asimtotik yang nilainya tidak pernah mencapai 0 ataupun 1 (Siang, 2004). Proses normalisasi menggunakan rumus :

$$x' = \frac{0,8(x-a)}{b-a} + 0,1$$

x = Data awal

a = Nilai minimal data awal

b = Nilai maksimal data awal

#### 2.5. Fungsi Aktivasi

Dalam *backpropagation*, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat yaitu kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun. Salah satu fungsi yang memenuhi ketiga syarat tersebut sehingga sering dipakai adalah fungsi sigmoid biner yang memiliki rentang (0,1) [5].

$$f(x) = \frac{1}{1+e^{-x}} \text{ dengan turunan } f'(x) = f(x)(1 - f(x))$$

#### 2.6. Algoritma Pelatihan Backpropagation

Algoritma pelatihan untuk jaringan dengan satu layar tersembunyi (dengan fungsi aktivasi sigmoid biner) adalah sebagai berikut [5]:

*Langkah 0* : Inisialisasi semua bobot dengan bilangan acak kecil

*Langkah 1* : Jika kondisi penghentian belum terpenuhi, seperti yang dijelaskan pada subbab 2.7.3, lakukan langkah 2-9

*Langkah 2* : Untuk setiap pasang data pelatihan, lakukan langkah 3-8

Fase I : Propagasi Maju

*Langkah 3* : Tiap masukan  $x_i$  ( $i = 0, 1, 2, \dots, n$ ) menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya, dengan  $x_0$  sebagai bias bernilai 1.

*Langkah 4* : Hitung semua keluaran di unit tersembunyi  $z_j$  ( $j = 0, 1, 2, \dots, p$ ), dengan  $z_0$  sebagai bias bernilai 1. Untuk perhitungan  $z_{netj}$  dimulai dengan  $j = 1$ .

$$z_{netj} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

$$z_j = f(z_{netj}) = \frac{1}{1+e^{-z_{netj}}}$$

*Langkah 5* : Hitung semua keluaran jaringan di unit  $y_k$  ( $k = 1, 2, \dots, m$ )

$$y_{netk} = w_{ko} + \sum_{j=1}^p z_j w_{kj}$$

$$y_k = f(y_{netk}) = \frac{1}{1+e^{-y_{netk}}}$$

Fase II : *Backpropagation*

*Langkah 6* : Hitung faktor  $\delta$  unit keluaran berdasarkan kesalahan di setiap unit keluaran  $y_k$  ( $k = 1, 2, \dots, m$ )

$$\delta k = (t_k - y_k) f'(y_{netk}) = (t_k - y_k) y_k (1 - y_k)$$

$\delta k$  merupakan unit kesalahan yang dipakai dalam perubahan bobot layar di bawah (langkah 7).

Hitung suku perubahan bobot  $w_{kj}$  (yang dipakai untuk mengubah bobot  $w_{kj}$ ) dengan laju percepatan  $\alpha$

$$\Delta w_{kj} = \alpha \delta k z_j$$

$$k = 1, 2, \dots, m \quad ; \quad j = 0, 1, \dots, p$$

*Langkah 7* : Hitung faktor  $\delta$  unit tersembunyi berdasarkan kesalahan di setiap unit

tersembunyi  $z_j$  ( $j = 1, 2, \dots, p$ )

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj}$$

Faktor  $\delta$  unit tersembunyi :

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1 - z_j)$$

Hitung suku perubahan bobot  $v_{ji}$  (yang dipakai untuk mengubah bobot  $v_{ji}$ )

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

$j = 1, 2, \dots, p$  ;  $i = 0, 1, \dots, n$

Fase III : Perubahan Bobot

Langkah 8 : Hitung semua perubahan bobot

Perubahan bobot garis yang menuju ke unit keluaran :

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

( $k = 1, 2, \dots, m$  ;  $j = 0, 1, \dots, p$ )

Perubahan bobot garis yang menuju ke unit tersembunyi :

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

( $j = 1, 2, \dots, p$  ;  $i = 0, 1, \dots, n$ )

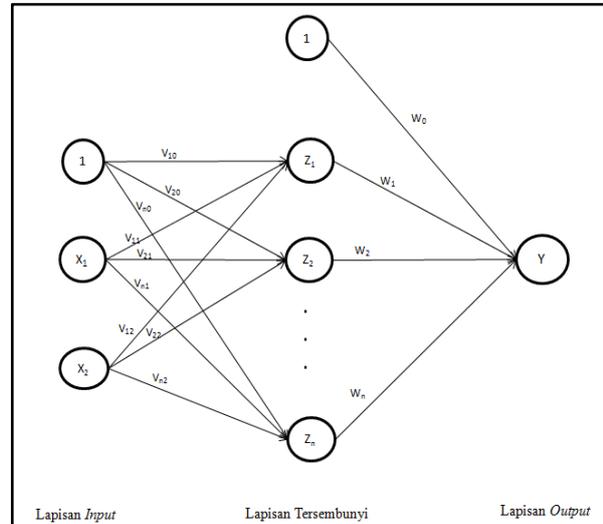
Langkah 9 : Menguji kondisi berhenti apakah sudah terpenuhi.

Setelah pelatihan selesai dilakukan, jaringan dapat dipakai untuk pengenalan pola. Dalam hal ini, hanya propagasi maju (langkah 4 dan 5) saja yang dipakai untuk menentukan keluaran jaringan. Apabila fungsi aktivasi yang dipakai bukan sigmoid biner, maka langkah 4 dan 5 harus disesuaikan sesuai dengan fungsi aktivasinya. Demikian juga turunannya pada langkah 6 dan 7.

### 3. Metode Penelitian

#### 3.1. Perancangan Arsitektur Jaringan Syaraf Tiruan Backpropagation

Perancangan arsitektur jaringan backpropagation untuk memprediksi pertumbuhan ekonomi Indonesia dapat dilihat pada Gambar 1.



Gambar 1. Arsitektur Jaringan Backpropagation Aplikasi Prediksi Pertumbuhan Ekonomi Indonesia

Dari arsitektur jaringan tersebut, dapat diformulasikan lapisan dan variabel yang digunakan :

1. Lapisan input (X) terdiri dari 2 simpul, yaitu  $X_1$  dan  $X_2$ .
2. Lapisan tersembunyi (Z) berjumlah satu lapisan, dengan jumlah simpul bervariasi.
3. Lapisan output (Y) berjumlah 1 simpul, sebagai keluaran jaringan.

#### 3.2. Metode Pengolahan Data

Dalam penelitian ini data yang digunakan merupakan data kuantitatif dari pendapatan domestik bruto (PDB) Indonesia berdasarkan pendekatan produksi yang dapat dilihat pada Tabel 1.

Pendekatan produksi terdiri dari sembilan sektor yaitu:

- a. Sektor 1 : Pertanian, Peternakan, Kehutanan dan Perikanan
- b. Sektor 2 : Pertambangan dan Penggalian
- c. Sektor 3 : Industri Pengolahan
- d. Sektor 4 : Listrik, Gas dan Air Bersih
- e. Sektor 5 : Konstruksi
- f. Sektor 6 : Perdagangan, Hotel dan Restoran
- g. Sektor 7 : Pengangkutan dan Komunikasi
- h. Sektor 8 : Keuangan, Real Estate dan Jasa Perusahaan
- i. Sektor 9 : Jasa-jasa, termasuk jasa pelayanan pemerintah

Data yang digunakan berupa data tahunan dari tahun 2000 hingga tahun 2012, sehingga masing-masing sektor memiliki 13 jumlah data. Data diperoleh secara literatur dari situs

resmi BPS yakni bps.go.id pada tahun 2014. Pola yang diperoleh dari data tersebut akan dibagi 2, yaitu 80% (9 pola) untuk data pelatihan dan 20% (2 pola) untuk data pengujian.

**Tabel 1.** Data PDB Pendekatan Produksi

Tahun\Sektor	Sektor1	Sektor2	Sektor3	Sektor4	Sektor5	Sektor6	Sektor7	Sektor8	Sektor9
2000	216831,5	167692,2	385597,9	8393,8	76573,4	224452,2	65012,1	115463	129753,8
2001	223891,5	168244,4	398323,8	9058,3	80080,4	233307,9	70275,9	123266	133957,5
2002	231613,5	169932	419387,8	9868,2	84469,8	243266,6	76173,1	131523	1338982,4
2003	240387,3	167603,8	441754,9	10349,2	89621,8	256516,6	85458,4	140374,4	145104,9
2004	247163,6	160100,5	469952,4	10897,6	96334,4	271142,2	96896,7	151123,3	152906,1
2005	253881,7	165222,6	491561,4	11584,1	103598,4	293654	109261,5	161252,2	160799,3
2006	262402,8	168031,7	514100,3	12251	112233,6	312518,7	124808,9	170074,3	170705,4
2007	271509,3	1711278,4	538084,6	13517	121808,9	340437,1	142326,7	183659,3	181706
2008	284619,1	172496,3	557764,4	14994,4	141009,6	363818,2	165905,5	198799,6	193049
2009	295883,8	180200,5	570102,5	17136,8	140267,8	368463	192198,8	209163	205434,2
2010	304777,1	187152,5	597134,9	18050,2	150022,4	400474,9	217980,4	221024,2	217842,2
2011	315036,8	190143,2	633781,9	18899,7	1559122,9	437472,9	241303	236146,6	232659,1
2012	328279,7	193115,7	670190,6	20080,7	170884,8	473110,6	265383,7	253022,7	244869,9

Contoh hasil normalisasi sektor 1 dapat dilihat pada Tabel 2.

**Tabel 2.** Normalisasi Sektor 1

Tahun	Data Asli	Data Normalisasi
2000	216831,5	0,1
2001	223891,5	0,1506
2002	231613,5	0,2061
2003	240387,3	0,269
2004	247163,6	0,3177
2005	253881,7	0,3659
2006	262402,8	0,4303
2007	271509,3	0,4924
2008	284619,1	0,5865
2009	295883,8	0,6674
2010	304777,1	0,7312
2011	315036,8	0,8049
2012	328279,7	0,9

Setelah data dinormalisasi, selanjutnya adalah menentukan pola yang digunakan pada jaringan syaraf tiruan yang dibuat, sesuai dengan arsitektur yang telah dirancang. Pola masukan jaringan dapat dilihat pada Tabel 3.

**Tabel 3.** Pola Masukan Jaringan

x-1	X	Target
0,1	0,1506	0,2061
0,1506	0,2061	0,269
0,2061	0,269	0,3177
0,269	0,3177	0,3659
0,3177	0,3659	0,4303
0,3659	0,4303	0,4924
0,4303	0,4924	0,5865
0,4924	0,5865	0,6674
0,5865	0,6674	0,7312
0,6674	0,7312	0,8049
0,7312	0,8049	0,9

Pada baris pertama, yaitu x-1 dan x adalah data pada tahun 2000 dan 2001, sedangkan targetnya adalah data pada tahun 2002. Target secara umum merupakan x+1. Pada baris ke dua, x-1 dan x adalah data pada tahun 2001 dan 2002, sedangkan targetnya adalah data pada tahun 2003. Demikian seterusnya.

Pola ini selanjutnya dibagi menjadi 2 bagian saling asing, yaitu 80% (9 pola) untuk data pelatihan dan 20% (2 pola) untuk data pengujian. Sehingga pola 1 sampai 9 digunakan sebagai data pelatihan, dan pola 10 dan 11 digunakan sebagai data pengujian.

Kemudian dilakukan inisialisasi bobot secara acak dengan angka yang berada pada range -0,5 hingga 0,5. Selanjutnya lakukan pelatihan jaringan *backpropagation*.

*Langkah 1* : Jika kondisi penghentian belum terpenuhi, lakukan langkah 2 sampai 9

*Langkah 2* : Untuk setiap pasang data pelatihan, lakukan langkah 3 sampai 8

Fase I : Propagasi Maju

*Langkah 3* : Tiap masukan  $X_i$  ( $i = 1, 2$ ) menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya.

**Langkah 4 :** Menghitung semua keluaran di unit tersembunyi  $Z_j(j = 1, 2, \dots, n$ , dengan  $n$  adalah nilai bervariasi dari masukan *user*).

$$z\_net_j = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

$$z_j = f(z\_net_j) = \frac{1}{1 + e^{-z\_net_j}}$$

**Langkah 5:** Menghitung semua keluaran jaringan di unit  $y_k(k = 1)$ .

$$y\_net = w_o + \sum_{j=1}^p z_j w_j$$

$$y = f(y\_net) = \frac{1}{1 + e^{-y\_net}}$$

**Fase II : Backpropagation**

**Langkah 6 :** Menghitung faktor  $\delta$  unit keluaran berdasarkan kesalahan di setiap unit keluaran  $y$

$$\delta = (t - y) f'(y\_net) = (t - y) y (1 - y)$$

$\delta k$  merupakan unit kesalahan yang dipakai dalam perubahan bobot layar di bawah (langkah 7).

Hitung suku perubahan bobot  $w_j$  (yang dipakai untuk mengubah bobot  $w_j$ ) dengan laju percepatan  $\alpha$

$$\Delta w_j = \alpha \delta z_j$$

$j = 0, 1, 2, \dots, n$ , dengan  $n$  adalah nilai bervariasi dari masukan *user*

**Langkah 7 :** Menghitung faktor  $\delta$  unit tersembunyi berdasarkan kesalahan di setiap unit tersembunyi  $z_j$

$$\delta\_net_j = \delta w_j$$

Faktor  $\delta$  unit tersembunyi :

$$\delta_j = \delta\_net_j f'(z\_net_j) = \delta\_net_j z_j (1 - z_j)$$

Hitung suku perubahan bobot  $v_{ji}$  (yang dipakai untuk mengubah bobot  $v_{ji}$ )

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

$i = 0, 1, 2 ; j = 1, 2, \dots, n$ , dengan  $n$  adalah nilai bervariasi dari masukan *user*

**Fase III : Perubahan Bobot**

**Langkah 8 :** Menghitung semua perubahan bobot.

Perubahan bobot garis yang menuju ke unit keluaran :

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

$j = 1, 2, \dots, n$ , dengan  $n$  adalah nilai bervariasi dari masukan *user*

Perubahan bobot garis yang menuju ke unit tersembunyi :

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

$i = 0, 1, 2 ; j = 1, 2, \dots, n$ , dengan  $n$  adalah nilai bervariasi dari masukan *user*

**Langkah 9 :** Mengecek apakah kondisi berhenti telah terpenuhi.

Setelah proses pelatihan, dilakukan proses pengujian jaringan menggunakan data pola pengujian dan bobot-bobot yang telah didapat dari hasil pelatihan. Dalam hal ini, hanya propagasi maju (langkah 4 dan 5) saja yang dipakai untuk menentukan keluaran jaringan.

Apabila fungsi aktivasi yang dipakai bukan sigmoid biner, maka langkah 4 dan 5 harus disesuaikan sesuai dengan fungsi aktivasinya. Demikian juga turunannya pada langkah 6 dan 7.

## 4. Pengujian

### 4.1. Skenario Pengujian

Untuk proses pelatihan, perhitungan disertai dengan masukan kombinasi nilai parameter-parameter pelatihan jaringan, yaitu target *error* (MSE), maksimum iterasi, laju pembelajaran / *learning rate*, dan jumlah neuron *hidden layer*. Pada pelatihan jaringan dilakukan kombinasi *learning rate* dengan nilai 0,25, 0,50 dan 0,75. Pada *neuron hidden layer* dilakukan kombinasi nilai 5, 15, 25 dan 50. Batas iterasi maksimum ditentukan sebesar 20000 iterasi. Pengujian dilakukan setelah proses pelatihan dilakukan dengan mencari nilai persentase *error* terkecil (MAPE) dari beberapa kombinasi parameter yang diujikan. Jaringan akan diuji keakuratannya dalam memprediksi pertumbuhan ekonomi Indonesiaper sektor lapangan usaha, dari sektor 1 hingga sektor 9. Arsitektur yang digunakan untuk memprediksi adalah arsitektur pada data pengujian. Bobot dan bias yang digunakan pada data pengujian adalah bobot dan bias pada data pelatihan, dengan akurasi terbaik.

### 4.2. Hasil Pengujian

Berdasarkan pengujian yang telah dilakukan, dapat dilihat bahwa seluruh hasil prediksi tiap sektor yang diambil dari kombinasi terbaik memperlihatkan nilai *error* pengujian yang kecil, sehingga bobot-bobot yang dihasilkan dapat digunakan untuk memprediksi pendapatan tiap sektor. Tingkat akurasi hasil prediksi per sektor dapat dilihat pada Tabel 4.

**Tabel 4.** Akurasi Hasil Prediksi Tahun 2013

Sektor	Error Pengujian (MAPE)	Hasil Prediksi	Data Faktual	Akurasi (%)
1	1,626	342202,7	339890,2	99,31
2	0,621	197241,9	197241,4	99,99
3	5,052	705374,3	707457,8	99,70
4	1,954	21539,29	21201	98,40
5	2,996	182340,3	182117,9	99,87
6	6,762	503933,7	501108,4	99,43
7	1,723	290380,9	292421,5	99,30
8	3,669	270193,4	272151,9	99,28
9	2,500	259237,3	258237,9	99,61
Total (PDB)		2772443,79	2770345,4	99,92

Nilai *error* terkecil yang didapat dari proses pengujian menghasilkan tingkat akurasi yang tinggi pada proses prediksi tahun 2013.

Hasil prediksi seluruh sektor dijumlahkan untuk mendapatkan angka PDB prediksi. Hasil persentase selisih PDB prediksi dengan PDB tahun sebelumnya yang didapat, merupakan pertumbuhan ekonomi Indonesia yang merupakan keluaran akhir dari aplikasi ini. Tabel 5 menunjukkan perbandingan hasil prediksi tahun 2013 dengan data faktual pertumbuhan ekonomi Indonesia tahun 2013.

**Tabel 5.** Perbandingan Hasil Prediksi Pertumbuhan Ekonomi Indonesia Tahun 2013 dengan Data Faktual

Hasil Prediksi PDB (Miliar Rupiah)	Persentase Pertumbuhan Ekonomi Hasil Prediksi	Data Faktual PDB (Miliar Rupiah)	Persentase Pertumbuhan Ekonomi Faktual	Akurasi (%)
2772443,79	5,86	2770345,4	5,78	99,92

Nilai persentase pertumbuhan ekonomi yang didapat yakni 5,86%, mampu mendekati data faktual pertumbuhan ekonomi Indonesia tahun 2013 sebesar 5,78%. Ini membuktikan bahwa metode *backpropagation* dapat digunakan untuk memprediksi pertumbuhan ekonomi.

## 5. Kesimpulan

Dari hasil yang telah dicapai, dapat disimpulkan bahwa terdapat kecenderungan jika neuron *hidden layer* yang semakin banyak akan menghasilkan nilai *error* yang semakin kecil. Sedangkan penggunaan laju pembelajaran terdapat kecenderungan bahwa dengan semakin besarnya laju pembelajaran yang digunakan, maka nilai iterasi yang dicapai semakin kecil, yang berarti jaringan semakin cepat mengenali pola. Penggunaan target *error* yang semakin kecil akan menyebabkan jaringan akan membutuhkan waktu yang sangat lama untuk mencapainya dengan jumlah iterasi yang sangat besar. Keluaran akhir pertumbuhan ekonomi yang dihasilkan yakni sebesar 5,86 %, mampu mendekati angka faktual pertumbuhan ekonomi Indonesia tahun 2013, yakni sebesar 5,78%.

## Daftar Pustaka

- [1]. Setiawan, S., Handoko, R., 2005, “*Pertumbuhan Ekonomi 2006: Suatu Estimasi dan Arah Pencapaian Pertumbuhan yang Merata dan Berkualitas*”, Kajian Ekonomi dan Keuangan, Volume 9, Nomor 4, halaman 1.
- [2]. Kiki dan Kusumadewi, S., 2004, “*Analisis Jaringan Saraf Tiruan dengan Metode backpropagation untuk Mendeteksi Gangguan Psikologi*”, halaman 13.
- [3]. Arsyad, L., 1992, “*Ekonomi Pembangunan Edisi ke – 2*”, Yogyakarta : Sekolah Tinggi Ilmu Ekonomi YPKN.
- [4]. Sukirno, S., 2006. “*Ekonomi Pembangunan Edisi ke – 2*”. Jakarta : Kencana Prenada Media Group.
- [5]. Siang, J. J., 2004. “*Jaringan Syaraf Tiruan dan Pemrograman menggunakan MATLAB*”. Yogyakarta : Andi

