



TESIS- SM 092306

**PREDIKSI JUMLAH PENDUDUK MISKIN  
DI PROVINSI KALIMANTAN SELATAN  
MENGUNAKAN JARINGAN SYARAF TIRUAN  
*BACKPROPAGATION***

Farida Amina  
NRP 1211 201 203

Dosen Pembimbing  
Prof. Dr. Muhammad Isa Irawan, M.T

PROGRAM MAGISTER  
JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2014



**FINAL PROJECT- SM 092306**

**PREDICTION OF POOR POPULATION  
IN SOUTH KALIMANTAN PROVINCE  
USING BACKPROPAGATION NEURAL NETWORK**

Farida Amina  
NRP 1211 201 203

Supervisor  
Prof. Dr. Muhammad Isa Irawan, M.T

PROGRAM OF MAGISTER  
DEPARTMENT OF MATHEMATICS  
FACULTY OF MATHEMATICS AND SCIENCE  
SEPULUH NOPEMBER INSTITUTE OF TECHNOLOGY  
SURABAYA  
2014

# **PREDIKSI JUMLAH PENDUDUK MISKIN DI PROVINSI KALIMANTAN SELATAN MENGUNAKAN JARINGAN SYARAF TIRUAN *BACKPROPAGATION***

Nama Mahasiswa : Farida Amina  
NRP : 1211201203  
Dosen Pembimbing : Prof. Dr. M. Isa Irawan, M.T.

## **ABSTRAK**

Masalah kemiskinan merupakan salah satu persoalan mendasar yang menjadi pusat perhatian pemerintah di negara manapun. Hampir setiap tahunnya terjadi peningkatan jumlah penduduk miskin. Menurut Badan Pusat Statistik (BPS), penduduk miskin adalah penduduk yang memiliki rata-rata pengeluaran per kapita per bulan dibawah Garis Kemiskinan. Namun, apabila kita melihat realita dalam kehidupan masyarakat, dapat diasumsikan bahwa terdapat beberapa indikator lain yang berkaitan dengan jumlah penduduk miskin, diantaranya adalah nilai Produk Domestik Regional Bruto (PDRB), inflasi, ekspor-impor, luas panen padi dan palawija, jumlah industri manufaktur besar dan sedang, jumlah pengangguran, serta jumlah penduduk. Pemilihan indikator-indikator tersebut didasari pada kenyataan bahwa adanya keterkaitan langsung dengan tingkat pendapatan penduduk. Untuk itu, dalam penelitian ini, guna mengetahui indikator yang lebih berkaitan dengan jumlah penduduk miskin di Kalimantan Selatan, digunakan metode *Principal Component Analysis* (PCA) dan regresi berganda.

Indikator-indikator yang berhubungan erat dengan jumlah penduduk miskin di Kalimantan Selatan yaitu PDRB sektor jasa – jasa atas dasar harga konstan, jumlah pengangguran, pengeluaran rata-rata perkapita sebulan untuk barang tahan lama, dan pengeluaran rata-rata perkapita sebulan untuk kelompok buah. Koefisien regresi yang positif pada setiap variabel bebas menunjukkan bahwa jumlah penduduk miskin akan semakin tinggi jika indikator-indikator tersebut semakin tinggi. Namun, Prediksi dengan penginputan hasil prediksi *time series* variabel bebas ke persamaan regresi untuk mendapatkan hasil prediksi jumlah penduduk miskin setiap bulannya pada tahun 2012 memiliki nilai MAPE yang lebih besar yaitu 9,98 % sehingga lebih kecil keakurasiannya dibandingkan prediksi dengan JST *time series* dari jumlah penduduk miskin yaitu sebesar 8,74%. Hasil estimasi jumlah penduduk miskin pada bulan ke-1 tahun 2012 adalah sebesar 244.466 orang. Sedangkan pada data sebenarnya berjumlah 250.568 orang. Sehingga nilai MAPE estimasi tersebut sebesar 2,43 %.

**Kata kunci :** jaringan syaraf tiruan *backpropagation*, kemiskinan, metode PCA, regresi berganda, prediksi.

# **PREDICTION OF POOR POPULATION IN SOUTH KALIMANTAN PROVINCE USING BACKPROPAGATION NEURAL NETWORK**

By : Farida Amina  
Student Identity Number : 1211201203  
Supervisor : Prof. Dr. M. Isa Irawan, M.T.

## **ABSTRACT**

Poverty is one of the fundamental problems that become the center of attention of the government in any country. Nearly every year number of poor people have increasing. According to Badan Pusat Statistik (BPS), the poor are people who have an average monthly per capita expenditure below the poverty line . However , if we look at the reality in people's lives, it can be assumed that there are some other indicators related to poverty, such as the value of Gross Domestic Product (GDP), inflation, export, rice and pulses harvested area, the number of large and medium manufacturing industries, the number of unemployment, and total population . The selection of these indicators is based on the fact that the existence of a direct relationship with the level population income. Therefore, in this study, to determine the indicators which more related to the number of poor people in South Kalimantan, have been used Principal Component Analysis (PCA) and multiple regression method .

The indicators are closely related to the number of poor people in South Kalimantan, namely GDP services sector at constant prices, the number of unemployment, the average expenditures per capita a month for durable goods, and the average expenditure per capita a month for the fruit group. Positive regression coefficient for each independent variable indicates that the number of poor people would be higher if these indicators are higher. However, predictions by inputting the results predicted time series of independent variables to the regression equation with back propagation neural network predictions have larger accuracy value MAPE than back propagation neural network with time series of the number of poor people. The first MAPE is 9,98 %, and the latter is only 8,74%. Estimation result for the number of poor people at first month in 2012 is 244.466 people. On the other hand, the actual data is 250.568 people. So the MAPE estimation is 2,43 %.

**Keywords** : backpropagation neural network, poverty, PCA method, multiple regression, prediction.

**PREDIKSI JUMLAH PENDUDUK MISKIN  
DI PROVINSI KALIMANTAN SELATAN MENGGUNAKAN  
JARINGAN SYARAF TIRUAN *BACKPROPAGATION***

**Tesis ini disusun untuk memenuhi salah satu syarat memperoleh gelar  
Magister Sains (M. Si)  
di  
Institut Teknologi Sepuluh Nopember**

**Oleh :  
FARIDA AMINA  
NRP. 1211 201 203**

**Tanggal Ujian : 18 Januari 2014  
Periode Wisuda : Maret 2014**

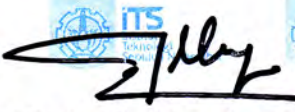
**Disetujui Oleh :**

  
**1. Prof. Dr. Muhammad Isa Irawan, M.T  
NIP. 19631225 198903 1 001**

**( Pembimbing )**

  
**2. Dr. Imam Mukhlash, S.Si, M.T  
NIP. 19700831 199403 1 003**

**( Penguji )**

  
**3. Dr. Dwi Ratna Sulistyningrum, S.Si, M.T  
NIP. 19690405 199403 2 003**

**( Penguji )**

  
**4. Dr. Mardijah, M.T  
NIP. 19670114 199102 2 001**

**( Penguji )**

  
**Direktur Program Pascasarjana,**

  
**Prof. Dr. Adi Suprijanto, M.T  
NIP. 19640405 199002 1 001**

## KATA PENGANTAR

*Bismillahirrahmanirrahim*

Alhamdulillahirabbil'alamiin, puji syukur penulis panjatkan kehadirat Allah SWT yang telah memberikan rahmat serta hidayah-Nya sehingga penulis dapat menyelesaikan tesis dengan judul: **Prediksi Jumlah Penduduk Miskin di Provinsi Kalimantan Selatan Menggunakan Jaringan Syaraf Tiruan Backpropagation**. Sholawat dan salam penulis haturkan kepada Rasulullah Shollallahu'alaihiwasalam. Penulis menyadari bahwa tesis ini dapat diselesaikan dengan baik atas bimbingan, arahan dan dukungan moril maupun materiil dari berbagai pihak. Untuk itu penulis mengucapkan terimakasih dan penghargaan yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. Muhammad Isa Irawan, M.T selaku dosen pembimbing yang telah memberikan bantuan, bimbingan dan arahan kepada penulis dalam menyelesaikan tesis.
2. Dr. Imam Mukhlash, S.Si, M.T., selaku dosen wali sekaligus dosen penguji yang telah memberikan bantuan, bimbingan dan arahan kepada penulis mulai dari penulis pertama kali menempuh pendidikan S2 hingga dapat menyelesaikannya dan memperoleh gelar Magister Sains.
3. Bapak Dr. Subiono, M.Sc, selaku Koordinator Program Studi S2 Matematika ITS yang telah memberikan kesempatan penulis untuk menyelesaikan S2 hingga dinyatakan lulus.
4. Ibu Dr. Erna Apriliani, M.Si., selaku Ketua Jurusan Matematika FMIPA-ITS.
5. Ibu Dr. Dwi Ratna Sulistyaningrum, S.Si, M.T. dan Ibu Dr. Mardlijah, M.T., sebagai dosen penguji.
6. Segenap dosen pengajar yang telah memberikan ilmu, arahan, dan pengalaman kepada penulis selama menempuh pendidikan S2 di jurusan matematika ITS.
7. Seluruh dosen, karyawan, serta staf Jurusan Matematika ITS.

8. Keluarga tercinta: Alm. Abah Rachimi Dharma dan Mama Mariani, Ading Karima Azkiya, serta Suami Mustafid Amna atas kasih sayang yang begitu besar, dukungan dan do'anya.
9. Sahabat-sahabatku, mahasiswa S2 Matematika angkatan semester genap tahun 2011, terima kasih atas kebersamaan dan kekeluargaannya selama di Surabaya, dan terima kasih pula atas bantuannya yang tak ternilai selama sama-sama menempuh pendidikan ini.
10. Sahabat-sahabatku, mahasiswa asal Banjarmasin, yang telah bersedia membantu banyak hingga tesis ini bisa selesai.
11. Kepala Kantor, serta keluarga besar BPS Kota Banjarmasin atas izin dan dukungannya dalam menyelesaikan tesis.
12. Seluruh pihak yang telah memberikan dukungan dan motivasi, terima kasih banyak atas semuanya.

Penulis menyadari takkan mampu membalas semua kebaikan tersebut, semoga Allah SWT memberikan kebaikan yang barokah. Akhirnya, penulis berharap tesisi ini dapat memberikan manfaat bagi para pembaca dan seluruh pihak yang terkait. Semoga Allah SWT senantiasa melimpahkan rahmat-Nya sehingga kita dapat terus berkreasi dan berinovasi positif untuk kemaslahatan bersama.

Banjarmasin, Januari 2014

**Farida Amina**

## DAFTAR ISI

	Hal
<b>HALAMAN PENGESAHAN</b> .....	i
<b>ABSTRAK</b> .....	iii
<b>ABSTRACT</b> .....	v
<b>KATA PENGANTAR</b> .....	vii
<b>DAFTAR ISI</b> .....	ix
<b>DAFTAR GAMBAR</b> .....	xi
<b>DAFTAR TABEL</b> .....	xiii
<b>BAB 1 PENDAHULUAN</b> .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	4
<b>BAB 2 KAJIAN PUSTAKA DAN DASAR TEORI</b> .....	5
2.1 Kemiskinan .....	5
2.2 Jaringan Syaraf Tiruan .....	7
2.3 Algoritma Pembelajaran Backpropagation .....	9
2.4 <i>Principal Component Analysis (PCA)</i> .....	14
2.5 Analisis Regresi Linear Berganda.....	15
2.6 Interpolasi.....	17
2.7 Penelitian Relevan.....	20
<b>BAB 3 METODE PENELITIAN</b> .....	22
3.1 Pengumpulan Data .....	23
3.2 <i>Data Preprocessing</i> /Normalisasi Data .....	25
3.3 Pemilihan Data.....	26
3.4 Pembuatan Program dengan MATLAB Menggunakan Arsitektur Jaringan .....	26
3.5 <i>Training</i> /Pelatihan Jaringan Syaraf Tiruan .....	27



3.6 <i>Testing</i> /Pengujian Jaringan Syaraf Tiruan .....	27
3.7 <i>Data Postprocessing</i> /Denormalisasi .....	28
<b>BAB 4 HASIL PENELITIAN DAN PEMBAHASAN</b> .....	31
4.1 Pengolahan Data .....	31
4.2 Pemilihan Data .....	31
4.3 Prediksi Jumlah Penduduk Miskin Menggunakan JST <i>Time Series</i> dan Regresi .....	40
4.4 Prediksi Jumlah Penduduk Miskin Menggunakan JST Multivariat .....	52
<b>BAB 5 KESIMPULAN DAN SARAN</b> .....	57
5.1 Kesimpulan .....	57
5.2 Saran .....	57
<b>DAFTAR PUSTAKA</b> .....	59
<b>LAMPIRAN</b> .....	61

## DAFTAR GAMBAR

	Hal
Gambar 2.1 Jaringan syaraf sederhana.....	8
Gambar 2.2 JST <i>backpropagation</i> dengan satu <i>hidden layer</i> .....	9
Gambar 2.3 Sigmoid biner, interval (0, 1).....	11
Gambar 2.4 Sigmoid bipolar , interval (-1, 1).....	11
Gambar 3.1 Arsitektur JST.....	27
Gambar 3.2 Diagram penelitian.....	29
Gambar 4.1 Uji normalitas P-P Plot .....	39
Gambar 4.2 Arsitektur JST <i>backpropagation</i> penelitian.....	43
Gambar 4.3 Perbandingan hasil target (o) dan output (-) pada variabel PDRB sektor jasa.....	46
Gambar 4.4 Grafik data aktual dan hasil prediksi pada variabel PDRB sektor jasa .....	46
Gambar 4.5 Perbandingan hasil target (o) dan output (-) pada variabel pengangguran.....	47
Gambar 4.6 Grafik data aktual dan hasil prediksi pada variabel pengangguran.....	47
Gambar 4.7 Perbandingan hasil target (o) dan output (-) pada variabel pengeluaran barang tahan lama.....	48
Gambar 4.8 Grafik data aktual dan hasil prediksi pada variabel pengeluaran barang tahan lama.....	48
Gambar 4.9 Perbandingan hasil target (o) dan output (-) pada variabel pengeluaran buah.....	49
Gambar 4.10 Grafik data aktual dan hasil prediksi pada variabel pengeluaran buah .....	49
Gambar 4.11 Perbandingan hasil target (o) dan output (-) pada variabel jumlah penduduk miskin .....	50
Gambar 4.12 Grafik data aktual dan hasil prediksi pada variabel jumlah penduduk miskin.....	50

Gambar 4.13	Arsitektur JST Multivariat.....	52
Gambar 4.14	Perbandingan hasil target (o) dan output (-) pada JST multivariat.....	54

## DAFTAR TABEL

		Hal
Tabel 4.1	<i>Rotated component matrix</i> .....	32
Tabel 4.2	Uji F dengan ANOVA. ....	34
Tabel 4.3	Uji t pada <i>coefficients</i> .....	35
Tabel 4.4	Uji kecocokan model pada <i>model summary</i> .....	36
Tabel 4.5	Uji multikolinieritas dengan VIF.....	37
Tabel 4.6	Uji heteroskedastisitas pada <i>coefficients</i> .....	38
Tabel 4.7	Uji autokorelasi Durbin Watson pada <i>model summary</i> .....	38
Tabel 4.8	Uji Autokorelasi LM Test.....	39
Tabel 4.9	One-Sample Kolmogorov-Smirnov Test.....	40
Tabel 4.10	Uji linieritas pada ANOVA.....	40
Tabel 4.11	Perubahan nilai uji coba data pelatihan dengan perubahan nilai <i>time window</i> .....	41
Tabel 4.12	Pergeseran unit input dan target pada tahap pelatihan .....	42
Tabel 4.13	Pergeseran unit input dan target pada tahap pengujian.....	42
Tabel 4.14	Perubahan nilai uji coba data pelatihan dengan perubahan nilai <i>learning rate</i> .....	43
Tabel 4.15	Perubahan nilai uji coba data pelatihan dengan perubahan nilai <i>hidden layer</i> .....	44
Tabel 4.16	Pergeseran unit input dan target pada tahap prediksi.....	45
Tabel 4.17	Data hasil prediksi dengan penggunaan variabel hasil PCA dan regresi tahun 2012.....	50
Tabel 4.18	Perbandingan hasil prediksi dari JST <i>time series</i> jumlah penduduk miskin tahun 2002 – 2011 dan data sebenarnya tahun 2012.....	51
Tabel 4.19	Perubahan nilai uji coba data pelatihan dengan perubahan nilai <i>learning rate</i> pada JST multivariat.....	52
Tabel 4.20	Perubahan nilai uji coba data pelatihan dengan perubahan nilai <i>hidden layer</i> pada JST multivariat .....	53

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Masalah kemiskinan merupakan salah satu persoalan mendasar yang menjadi pusat perhatian pemerintah di negara manapun. Salah satu aspek penting untuk mendukung strategi penanggulangan kemiskinan adalah tersedianya data kemiskinan yang akurat dan tepat sasaran. Pengukuran kemiskinan yang dapat dipercaya dapat menjadi instrumen tangguh bagi pengambil kebijakan dalam memfokuskan perhatian pada kondisi hidup orang miskin.

Komposisi pengeluaran penduduk dapat dijadikan salah satu ukuran tingkat kesejahteraan masyarakat suatu wilayah. Semakin besar pengeluaran penduduk untuk konsumsi makanan merupakan indikasi tingkat kesejahteraan yang semakin baik. Begitu juga halnya dengan pengeluaran untuk non makanan, karena masyarakat perkotaan cenderung memiliki pengeluaran lebih besar untuk kelompok ini. Pengeluaran untuk konsumsi makanan dan bukan makanan berkaitan erat dengan tingkat pendapatan masyarakat.

Menurut Badan Pusat Statistik (BPS), penduduk miskin adalah penduduk yang memiliki rata-rata pengeluaran per kapita per bulan dibawah Garis Kemiskinan. Garis Kemiskinan Makanan (GKM) merupakan nilai pengeluaran kebutuhan minimum makanan yang disetarakan dengan 2100 kkalori per kapita per hari. Paket komoditi kebutuhan dasar makanan diwakili oleh 52 jenis komoditi (padi-padian, umbi-umbian, ikan, daging, telur dan susu, sayuran, kacang-kacangan, buah-buahan, minyak, dan lemak, dll). Sedangkan Garis Kemiskinan Non Makanan (GKNM) adalah kebutuhan minimum untuk perumahan, sandang, pendidikan, dan kesehatan. Paket komoditi kebutuhan dasar non makanan diwakili oleh 51 jenis komoditi di perkotaan dan 47 jenis komoditi di perdesaan.

Apabila kita melihat realita dalam kehidupan masyarakat, dapat diasumsikan bahwa terdapat beberapa indikator yang berkaitan dengan tingkat kemiskinan, selain pengeluaran rata-rata perkapita sebulan yang menjadi dasar BPS dalam penentuan jumlah penduduk miskin. Beberapa diantaranya adalah

nilai Produk Domestik Regional Bruto (PDRB), inflasi, ekspor-impor, produksi padi, jumlah industri manufaktur besar dan sedang, serta jumlah pengangguran. Pemilihan indikator-indikator tersebut didasari pada kenyataan bahwa adanya keterkaitan langsung dengan tingkat pendapatan penduduk. Selain itu, jumlah penduduk seluruhnya juga diasumsikan memberi pengaruh pada seberapa besar masyarakat dapat memenuhi kebutuhan anggota keluarganya yang semakin bertambah.

Namun untuk melihat secara jelas keterkaitan indikator-indikator yang ada dengan jumlah penduduk miskin, akan dilakukan pemilihan menggunakan data yang ada dengan metode *Principal Component Analysis* (PCA). PCA itu sendiri merupakan suatu teknik untuk mereduksi data multivariat atau multivariabel yang saling berkorelasi dengan mengambil ciri penting dari sekumpulan data set dengan mereduksi data tersebut menjadi data yang tidak saling berkorelasi. Dengan kata lain, PCA digunakan untuk meminimumkan masalah multikolinearitas tanpa harus mengeluarkan variabel bebas yang terlibat hubungan kolinear. Sasaran PCA adalah menjelaskan suatu kumpulan data dengan variabel yang lebih kecil. Selain itu, analisis regresi berganda juga dapat membantu untuk mengetahui indikator apa saja yang paling berkaitan dengan penduduk miskin dan mencari angka jumlah penduduk miskin menggunakan persamaan linier yang terbentuk.

Aplikasi jaringan syaraf tiruan pada peramalan sendiri menarik minat banyak kalangan profesional di luar ahli komputasi, yaitu antara lain profesional dari dunia keuangan atau asuransi, karena model pasar keuangan menggunakan model matematika masih banyak yang tidak berhasil. Cara lain dengan menggunakan jaringan syaraf tiruan cukup menjanjikan tetapi masih dalam masa perkembangan dan memiliki beberapa masalah, baik praktik maupun teorinya.

Oleh karena itu, peneliti juga ingin mencoba melakukan peramalan dalam bidang sosial ekonomi dan kependudukan. Dalam hal ini peneliti ingin memprediksi jumlah penduduk miskin di Provinsi Kalimantan Selatan menggunakan jaringan syaraf tiruan dengan algoritma pembelajaran Backpropagation.

## **1.2 Perumusan Masalah**

Berdasarkan latar belakang yang ada, maka perumusan masalah dalam penelitian ini dapat diberikan sebagai berikut:

- a. Bagaimana melakukan pemilihan indikator yang lebih berkaitan dengan jumlah penduduk miskin di Kalimantan Selatan menggunakan metode *Principal Component Analysis* (PCA) dan analisis regresi berganda?
- b. Bagaimana memprediksi jumlah penduduk miskin di Kalimantan Selatan pada tahun 2012 setiap bulannya menggunakan jaringan syaraf tiruan *backpropagation*?

## **1.3 Batasan Masalah**

Untuk membatasi kajian agar lebih spesifik dan terarah, maka penelitian ini dibatasi pada pembahasan sebagai berikut:

- a. Data yang digunakan merupakan data bulanan dari tahun 2002-2011.
- b. Pemilihan indikator yang dijadikan input dalam jaringan syaraf tiruan menggunakan metode PCA dan analisis regresi berganda.
- c. Algoritma pembelajaran jaringan syaraf tiruan yang digunakan adalah *Backpropagation*.

## **1.4 Tujuan Penelitian**

Dari perumusan masalah maka dapat dirumuskan tujuan penelitian ini sebagai berikut:

- a. Melakukan pemilihan indikator yang lebih berkaitan dengan jumlah penduduk miskin di Kalimantan Selatan menggunakan metode PCA dan analisis regresi berganda.
- b. Memprediksi jumlah penduduk miskin di Kalimantan Selatan pada bulan tahun 2012 setiap bulannya menggunakan jaringan syaraf tiruan *backpropagation*.

## **1.5 Manfaat Penelitian**

Manfaat penelitian yang diperoleh dari penelitian ini adalah sebagai berikut:

- a. Menetapkan APBD dengan lebih menitikberatkan pada beberapa indikator yang berkaitan dengan jumlah penduduk miskin.
- b. Mengevaluasi kebijakan pemerintah yang sudah ada terhadap kemiskinan.
- c. Sebagai bahan acuan dalam pengambilan kebijakan penanggulangan kemiskinan selanjutnya.



## **BAB 2**

### **KAJIAN PUSTAKA DAN DASAR TEORI**

#### **2.1 Kemiskinan**

Definisi tentang kemiskinan telah mengalami perluasan, seiring dengan semakin kompleksnya faktor penyebab, indikator maupun permasalahan lain yang melingkupinya. Kemiskinan tidak lagi hanya dianggap sebagai dimensi ekonomi melainkan telah meluas hingga ke dimensi sosial, kesehatan, pendidikan dan politik. Menurut Badan Pusat Statistik, kemiskinan adalah ketidakmampuan memenuhi standar minimum kebutuhan dasar yang meliputi kebutuhan makan maupun non makan.

Berbagai definisi dan konsep yang ada untuk kesejahteraan dan kemiskinan, salah satunya yaitu apakah rumah tangga atau individu memiliki sumber daya yang cukup atau kemampuan untuk memenuhi kebutuhan mereka saat ini. Definisi ini didasarkan pada perbandingan pendapatan setiap individu, konsumsi, pendidikan, atau hal-hal lain dengan beberapa ambang batas yang didefinisikan. Selain itu, merupakan hasil dari ketimpangan dalam distribusi pendapatan, konsumsi, atau hal lain dalam suatu populasi. Hal ini didasarkan pada pandangan umum kebanyakan orang mengenai posisi individu atau rumah tangga dalam lingkungan masyarakat merupakan aspek penting dari tingkat kesejahteraan mereka. (Coudouel, 2002).

Kemiskinan sangat berhubungan erat dengan kesejahteraan keluarga. Pola konsumsi rumah tangga merupakan salah satu indikator kesejahteraan keluarga. Selama ini berkembang pengertian bahwa besar kecilnya proporsi pengeluaran untuk konsumsi makanan terhadap seluruh pengeluaran rumah tangga dapat memberikan gambaran kesejahteraan rumah tangga tersebut. Konsumsi rumah tangga merupakan penyumbang terbesar dalam penggunaan Produk Domestik Bruto (PDB) Indonesia (Direktorat Diseminasi Statistik BPS, 2009).

Bagi pemerintah, statistik ekspor-impor berguna dalam merumuskan kebijakan dan memantau kinerja perekonomian. Di samping itu, statistik tersebut

dipakai pula dalam penyusunan PDB dan Neraca Pembayaran (*Balance of Payment/BOP*). Bagi swasta dan akademisi, statistik tersebut dapat dipakai untuk berbagai analisis dalam penelitian ekonomi dan sosial.

Selain itu pertumbuhan ekonomi juga berpengaruh pada kesejahteraan penduduk. Berdasarkan data strategis BPS tahun 2012, beberapa faktor terkait penurunan jumlah dan persentase penduduk miskin selama Maret 2011-Maret 2012 adalah inflasi umum relatif rendah, adanya kenaikan NTP (Nilai Tukar Petani), perekonomian Indonesia triwulan I-2012 yang meningkat disertai dengan meningkatnya pengeluaran konsumsi rumah tangga tumbuh, dan penghasilan yang cukup untuk memenuhi kebutuhan sehari-hari dalam sebulan baik di daerah perkotaan maupun pedesaan.

Pemenuhan kebutuhan pangan yang layak masih menjadi persoalan bagi masyarakat miskin. Terbatasnya kecukupan dan kelayakan pangan berkaitan dengan rendahnya gizi baik nutrisi maupun kalori, dan terbatasnya diversifikasi pangan dengan pola konsumsi yang cenderung mengandalkan beras. Pada umumnya kesulitan pemenuhan pangan ini disebabkan oleh rendahnya daya beli, tata niaga yang tidak efisien, dan kesulitan stok pangan di beberapa daerah yang terjadi pada musim tertentu. Masalah kecukupan pangan bukan hanya terkait dengan produksi bahan pangan, tetapi juga masalah peningkatan pendapatan karena mayoritas petani miskin harus membeli bahan makanan mereka. Berbagai upaya telah dilakukan untuk menyediakan stok pangan yang cukup ditingkat nasional, namun masalah keterbatasan akses masyarakat miskin terhadap pangan belum terpecahkan sepenuhnya.

Produksi pada dasarnya merupakan hasil kali luas panen dengan produktivitas per hektare lahan, sehingga seberapa besar produksi suatu wilayah sangat tergantung berapa luas panen pada tahun yang bersangkutan atau berapa tingkat produktivitasnya. Tingkat produktivitas per satuan luas, merupakan cerminan tingkat penerapan teknologi usaha tani, baik penggunaan bibit, luas lahan, tenaga kerja, dan pemupukan. Produksi pertanian merupakan salah satu upaya untuk menjaga ketahanan pangan agar penduduk Indonesia yang pada dasarnya berprofesi sebagai petani dan membutuhkan konsumsi pangan dapat terpenuhi.

Pemerintah sampai saat ini terus melakukan upaya-upaya dalam peningkatan laju pertumbuhan ekonomi nasional. Sebagai wujud dari upaya tersebut, pemerintah terus berperan aktif sebagai fasilitator dan dinamisator dalam menciptakan iklim usaha yang makin kondusif melalui penetapan berbagai kebijakan ekonomi yang harus berdampak positif terhadap sektor riil maupun moneter. Mengingat pentingnya peran sektor industri manufaktur terhadap PDB nasional, maka diperlukan indikator dini untuk mengamati perkembangan industri manufaktur. Salah satu indikator tersebut adalah pertumbuhan produksi industri manufaktur besar dan sedang (IBS).

Ada hubungan yang erat sekali antara tingginya tingkat pengangguran, luasnya kemiskinan, dan distribusi pendapatan yang tidak merata. Bagi sebagian besar mereka, yang tidak mempunyai pekerjaan yang tetap atau hanya bekerja paruh waktu (*part time*) selalu berada diantara kelompok masyarakat yang sangat miskin. Mereka yang bekerja dengan bayaran tetap disektor pemerintah dan swasta biasanya termasuk diantara kelompok masyarakat kelas menengah ke atas. Namun demikian, adalah salah jika beranggapan bahwa setiap orang yang tidak mempunyai pekerjaan adalah miskin, sedang yang bekerja secara penuh adalah orang kaya. Hal ini karena kadangkala ada pekerja di perkotaan yang tidak bekerja secara sukarela karena mencari pekerjaan yang lebih baik yang lebih sesuai dengan tingkat pendidikannya. Mereka menolak pekerjaan yang mereka rasakan lebih rendah dan mereka bersikap demikian karena mereka mempunyai sumber lain yang bisa membantu masalah keuangan mereka. (Arsyad, 1997)

BPS pertama kali melakukan penghitungan jumlah dan persentase penduduk miskin pada tahun 1984. Pada saat itu, penghitungan jumlah dan persentase penduduk miskin mencakup periode 1976-1981 dengan menggunakan data Survei Sosial Ekonomi Nasional (Susenas) modul konsumsi. Sejak itu, setiap tiga tahun sekali BPS secara rutin mengeluarkan data jumlah dan persentase penduduk miskin yang disajikan menurut daerah perkotaan dan pedesaan. Sejak tahun 2003, BPS secara rutin mengeluarkan data jumlah dan persentase penduduk miskin setiap tahun. Hal tersebut bisa terwujud karena sejak tahun 2003 BPS mengumpulkan data Susenas Panel Modul Konsumsi setiap bulan Februari atau Maret. Data kemiskinan yang diproduksi BPS diterbitkan secara tahunan dalam

publikasi yang berjudul "Analisis dan Penghitungan Tingkat Kemiskinan". (BPS, 2012)

## 2.2 Jaringan Syaraf Tiruan (JST)

Fausett (1994) menyatakan Jaringan Syaraf Tiruan (*Artificial Neural Network*) adalah pemrosesan sistem informasi pada karakteristik tertentu dalam keadaan yang berhubungan dengan jaringan syaraf biologi. Jaringan syaraf tiruan dibangun berdasarkan generalisasi dari model matematika pada manusia atau syaraf biologi, didasarkan pada asumsi :

1. Pemrosesan informasi terjadi pada banyak element yang disebut neuron.
2. Sinyal berjalan diantara neuron yang terkoneksi jaringan.
3. Masing masing jaringan koneksi dihubungkan dengan bobot, yang mana didalam jaringan syaraf khusus, melipatgandakan transmisi sinyal.
4. Masing masing neuron mempergunakan fungsi aktivasi (biasanya nonlinier) ke jaringan inputnya untuk menentukan sinyal output.

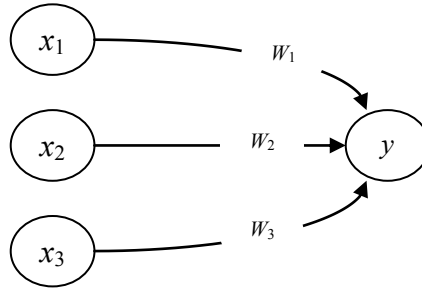
Jaringan syaraf dikarakteristikan oleh (1) polanya terkoneksi diantara neuron (disebut arsitektur), (2) metode untuk menentukan bobot dari koneksi (disebut training atau learning, algoritma), dan (3) fungsi aktivasi.

Jaringan syaraf terdiri dari banyak sekali elemen pemroses yang disebut dengan neuron, unit, atau node. Masing-masing terhubung dengan neuron yang lain oleh alat komunikasi jaringan secara langsung dan masing-masing terhubung dengan bobot. Bobot menggambarkan informasi yang digunakan oleh jaringan untuk memecahkan masalah. Jaringan syaraf dapat diaplikasikan pada bermacam-macam permasalahan, seperti penyimpanan dan pemanggilan ulang data atau pola, pengklasifikasian pola, melakukan pemetaan umum dari pola input ke pola output, pengelompokkan pola yang mirip, atau menemukan solusi permasalahan secara maksimal.

Masing-masing neuron mempunyai bagian internal, disebut aktivasi atau level aktivasi, dimana fungsinya adalah menerima input. Secara khusus, neuron mengirimkan aktivasi berupa sinyal kepada beberapa neuron yang lain. Hal penting untuk dicatat bahwa neuron dapat mengirim satu sinyal pada satu waktu, walaupun sinyal ini mengirim ke beberapa neuron.

Contoh, anggap sebuah neuron  $y$ , dilustrasikan pada Gambar 2.1, ia menerima input dari neuron  $x_1$ ,  $x_2$  dan  $x_3$ . Aktivasi pada neuron ini adalah  $x_1$ ,  $x_2$  dan  $x_3$  tentunya. Bobot dari hubungan antara  $x_1$ ,  $x_2$  dan  $x_3$  ke neuron  $y$  adalah  $w_1$ ,  $w_2$  dan  $w_3$ . net input,  $y\_in$ , ke neuron  $y$  adalah jumlah dari bobot sinyal dari neuron  $x_1$ ,  $x_2$  dan  $x_3$ , yaitu:

$$y\_in = w_1x_1 + w_2x_2 + w_3x_3 \quad (2.1)$$



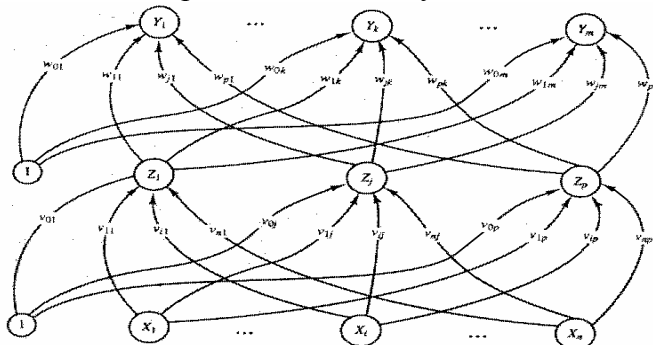
Gambar 2.1 Jaringan syaraf sederhana (Fausett, 1994)

### 2.3 Algoritma Pembelajaran Backpropagation

Backpropagation menggunakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya. (Kusumadewi, 2004)

#### Arsitektur

Jaringan syaraf tiruan *multi-layer* dengan satu *layer* pada unit *hidden* (unit  $Z$ ) ditunjukkan pada Gambar 2.2. unit-unit *output* (unit  $Y$ ) dan unit *hidden* juga memiliki bias. Bias pada unit *output*  $Y_k$  dilambangkan dengan  $w_{0k}$ , bias pada unit *hidden*  $Z_j$  dilambangkan dengan  $v_{0j}$ . Pada fase *feedforward* hanya satu arah informasi yang mengalir dari operasi. Selama fase *backpropagation* pada proses *learning*, sinyal dikirimkan pada arah sebaliknya.



Gambar 2.2 JST *backpropagation* dengan satu *hidden layer* (Fausett, 1994)

Seperti yang telah dijelaskan sebelumnya, *training* jaringan dengan *backpropagation* meliputi tiga langkah: propagasi maju (*feedforward*) dari pola *inputtraining*, perhitungan *error* (*backpropagationoferror*) dan penyesuaian bobot.

Selama *feedforward*, tiap unit *input* ( $X_i$ ) menerima sebuah sinyal *input* dan menyebarkannya ke tiap unit *hidden*  $Z_1 \dots Z_p$ . Kemudian tiap unit *hidden* menghitung aktivasinya dan mengirimkan sinyalnya ( $z_j$ ) ke setiap unit *output*. Tiap unit *output* ( $Y_k$ ) menghitung aktivasinya ( $y_k$ ) untuk membentuk respon dari net untuk memberikan pola *input*.

Selama proses *training*, tiap unit *output* membandingkan aktivasinya yang telah dihitung  $y_k$  dengan nilai targetnya  $t_k$  untuk menentukan hubungan *error* dengan pola tersebut dengan unit yang bersangkutan. Berdasarkan nilai *error* ini, faktor  $\delta_k$  ( $k = 1 \dots m$ ) dihitung.  $\delta_k$  digunakan untuk mendistribusikan *error* pada unit *output*  $Y_k$  kembali ke semua unit pada *layer* sebelumnya (unit *hidden* yang berhubungan ke  $Y_k$ ).  $\delta_k$  juga digunakan untuk meng-*update* bobot antara *layer output* dan *hidden layer*. Dengan cara yang sama, faktor  $\delta_j$  digunakan untuk meng-*update* bobot antara *hidden layer* dan *layer input*.

Setelah semua faktor  $\delta$  telah ditentukan, bobot-bobot pada semua *layer* disesuaikan secara simultan. Penyesuaian bobot  $w_{jk}$  (dari unit *hidden*  $Z_j$  ke unit *output*  $Y_k$ ) didasarkan pada faktor  $\delta_k$  dan aktivasi  $z_j$  dari unit *hidden*  $Z_j$ . Penyesuaian bobot  $v_{ij}$  (dari unit *input*  $X_i$  ke unit *hidden*  $Z_j$ ) didasarkan pada faktor  $\delta_j$  dan aktivasinya  $x_i$  dari unit *input*. (Fausett, 1994)

### **Fungsi Aktivasi**

Fungsi aktivasi untuk jaringan *backpropagation* memiliki beberapa karakteristik yang penting. Fungsi aktivasi tersebut kontinu, dapat diturunkan dan tidak turun secara monoton. Untuk efisiensi perhitungan, turunan dari fungsi aktivasi tersebut juga mudah dihitung. Fungsi aktivasi yang paling umum digunakan adalah nilai dari turunannya (pada nilai tertentu dari variabel independen) dapat diekspresikan dalam bentuk nilai dari fungsi aktivasi tersebut (pada nilai variabel independen).

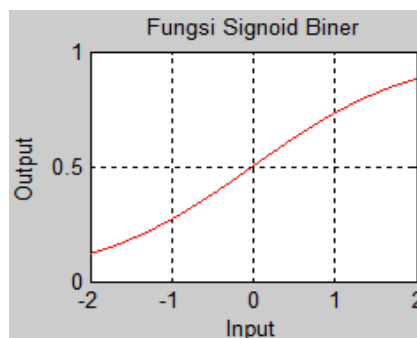
Salah satu tipe fungsi aktivasi yang paling banyak digunakan adalah fungsi sigmoid biner, yang mempunyai interval (0,1) dan didefinisikan sebagai berikut:

$$f_1(x) = \frac{1}{1 + \exp(-x)} \quad (2.2)$$

dan dengan

$$f_1'(x) = f_1(x)[1 - f_1(x)] \quad (2.3)$$

Fungsi ini diilustrasikan pada Gambar 2.3 berikut:



Gambar 2.3 Sigmoid biner, interval (0,1) (Kusumadewi, 2004)

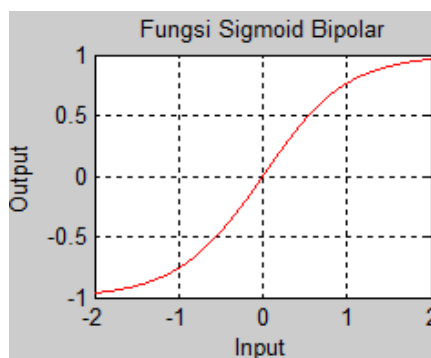
Fungsi aktivasi lain yang umum digunakan adalah fungsi bipolar sigmoid, yang mempunyai *range* (-1,1) dan didefinisikan sebagai berikut:

$$f_2(x) = \frac{2}{1 + \exp(-x)} - 1 \quad (2.4)$$

dengan

$$f_2'(x) = \frac{1}{2} [1 + f_2(x)][1 - f_2(x)] \quad (2.5)$$

Fungsi ini diilustrasikan pada Gambar 2.4 berikut:



Gambar 2.4 Sigmoid bipolar, interval (-1, 1) (Kusumadewi, 2004)

Fungsi sigmoid bipolar ini berhubungan erat dengan fungsi

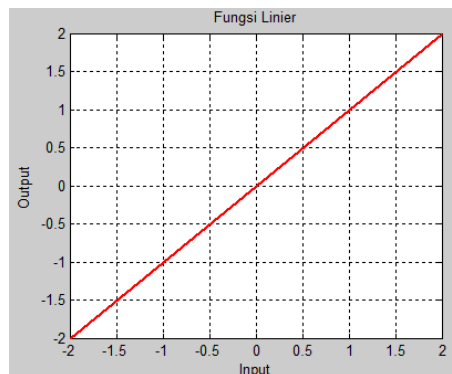
$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.6)$$

Kedua fungsi aktivasi yang telah dijelaskan sebelumnya merupakan fungsi yang dapat dideferensialkan. Fungsi yang dapat dideferensialkan ini merupakan salah satu syarat yang harus dimiliki apabila ingin menggunakan algoritma *backpropagation*. Selain kedua fungsi tersebut, terdapat pula fungsi linier yang dapat dideferensialkan. Fungsi linier, atau yang biasa disebut dengan fungsi identitas, memiliki nilai output yang sama dengan nilai inputnya. Fungsi linier dirumuskan sebagai

$$f_3(x) = x \quad (2.7)$$

dengan

$$f_3'(x) = 1 \quad (2.8)$$



Gambar 2.5 Linier atau Identitas (Kusumadewi, 2004)

### Algoritma *Training*

Adapun algoritma *training backpropagation* adalah sebagai berikut :

Langkah 0 : Inisialisasi nilai bobot dengan nilai acak yang kecil.

Langkah 1 : Selama kondisi berhenti masih belum terpenuhi, kerjakan langkah 2 - 9.

Langkah 2 : Untuk tiap pasangan pelatihan (s:t), kerjakan langkah 3 - 8.

#### **Feedforward :**

Langkah3: Setiap unit *input* ( $X_i, i=1, \dots, n$ ) menerima sinyal *input*  $x_i$  dan menyebarkan sinyal itu ke seluruh unit pada *layer* berikutnya (*hidden layer*)



Langkah 4: Setiap unit dalam ( $Z_j, j=1, \dots, p$ ) hitung nilai *input* dengan menggunakan nilai bobotnya

$$z\_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij} \quad (2.9)$$

Selanjutnya hitung nilai *output* dengan menggunakan fungsi aktivasi yang dipilih

$$z_j = f(z\_in_j) \quad (2.10)$$

Hasil fungsi tersebut dikirim ke semua unit pada *layer* berikutnya (unit *output*).

Langkah 5: Untuk tiap unit *output* ( $Y_k, k=1, \dots, m$ ) hitung nilai *input* dengan menggunakan nilai bobotnya

$$y\_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk} \quad (2.11)$$

Kemudian hitung nilai *output* dengan menggunakan fungsi aktivasi

$$y_k = f(y\_in_k) \quad (2.12)$$

### Backpropagation Of Error

Langkah 6: Setiap unit *output* ( $Y_k, k=1, \dots, m$ ) menerima pola target yang bersesuaian dengan pola *input* dan kemudian hitung error informasi

$$\delta_k = (t_k - y_k) f'(y_{in_k}) \quad (2.13)$$

Kemudian hitung koreksi nilai bobot yang akan digunakan untuk memperbaharui nilai bobot  $w_{jk}$  :

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (2.14)$$

Hitung koreksi nilai bias yang akan digunakan untuk memperbaharui nilai  $w_{0k}$  :

$$\Delta w_{0k} = \alpha \delta_k \quad (2.15)$$

dan nilai  $\delta_k$  dikirim ke unit pada *layer* sebelumnya.

Langkah 7: Setiap unit dalam ( $Z_j, j=1, \dots, p$ ) hitung delta *input* yang berasal dari unit pada *layer* di atasnya :

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (2.16)$$

Kemudian nilai tersebut dikalikan dengan nilai turunan dari fungsi aktivasi untuk menghitung error informasi :

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (2.17)$$

Hitung koreksi nilai bobot yang kemudian digunakan untuk memperbaharui nilai  $v_{ij}$

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (2.18)$$

dan hitung nilai koreksi bias yang kemudian digunakan untuk memperbaharui  $v_{oj}$

$$\Delta v_{oj} = \alpha \delta_j \quad (2.19)$$

#### Update nilai bobot dan bias :

Langkah 8 : Setiap unit *output* ( $Y_k, k=1, \dots, m$ ) update bias dan bobotnya ( $j=0, \dots, p$ )

$$w_{jk}(\text{new}) = w_{jk}(\text{old}) + \Delta w_{jk} \quad (2.20)$$

Setiap unit *hidden* ( $Z_j, j=1, \dots, p$ ) update bias dan bobotnya ( $i=0, \dots, n$ )

$$v_{ij}(\text{new}) = v_{ij}(\text{old}) + \Delta v_{ij} \quad (2.21)$$

Langkah 9 : Menguji apakah kondisi berhenti sudah terpenuhi.

Satu *epoch* adalah satu putaran melalui seluruh set vektor *training*. Pada umumnya, beberapa *epoch* diperlukan untuk *training* jaringan syaraf *backpropagation*. Algoritma sebelumnya melakukan pembaharuan bobot sesudah masing-masing pola *training* ditampilkan. Variasi umum muncul dari beberapa *updating*, dengan bobot hasil *updating* dikumpulkan melebihi semua *epoch* (atau beberapa pola persentase yang lain) sebelum digunakan.

## 2.4 Principal Component Analysis (PCA)

*Principal Component Analysis* (PCA) merupakan suatu teknik untuk mereduksi data multivariat atau multivariabel yang saling berkorelasi dengan mengambil ciri penting dari sekumpulan data set dengan mereduksi data tersebut

menjadi data yang tidak saling berkorelasi. Dengan kata lain, PCA digunakan untuk meminimumkan masalah multikolinearitas tanpa harus mengeluarkan variable bebas yang terlibat hubungan kolinear. Sasaran PCA adalah menjelaskan suatu kumpulan data dengan variabel yang lebih kecil.

Proyeksi pada PCA merupakan representasi himpunan data  $X$  ke dalam bentuk vektor eigen dari matriks varian-kovarian dari  $X$ . Vektor eigen dengan nilai eigen yang besar mempunyai peranan penting dalam proses transformasi dimensi. Hal ini disebabkan semakin tinggi nilai eigen maka semakin tinggi pula lebar distribusi data. Sehingga mereduksi dimensi dengan cara membuang nilai eigen yang mendekati nol tidak akan membuat kita kehilangan informasi data atau hanya kehilangan informasi yang minimum. (Ulfah, 2011)

PCA dapat digunakan untuk mengurangi kompleksitas variabel input apabila terdapat banyak informasi yang berkaitan sehingga diperoleh variabel yang memiliki interpretasi lebih baik. *Principal Components* (PC) dapat ditentukan oleh persamaan berikut:

$$Z_i = a_{i1}X_1 + a_{i2}X_2 + \dots + a_{ip}X_p \quad (2.22)$$

dengan  $Z_i$  adalah PC,  $a_i$  adalah vektor eigen terkait dan  $X_i$  adalah variabel input. Informasi ini diperoleh dengan menyelesaikan persamaan:

$$|R - I\lambda| = 0 \quad (2.23)$$

dengan  $I$  adalah matriks satuan,  $R$  adalah matriks varian-kovarian dan  $\lambda$  adalah nilai eigen. (Noori, 2011)

Dalam PCA, dicari kombinasi linier standar dari matriks data  $X$  berdasarkan dekomposisi *Jordan* dari matriks varian-kovarian  $\Sigma$  dari  $X$  atau ekuivalen dengan matriks korelasi  $\Phi$  dari  $X$ . Sehingga akan dinyatakan mean dari observasi sebagai  $\mu_{1 \times p}$ .

Dimisalkan  $x_{1 \times p} = (x_1, x_2, \dots, x_p)$  menyatakan vektor acak observasi dalam matriks data (yaitu transpos dari setiap baris dari matriks data  $n \times p$ ), dengan mean  $\mu_{1 \times p}$  dan matriks kovarian  $\Sigma$ . Sebuah komponen utama adalah transformasi dari bentuk  $x_{1 \times p} \rightarrow y_{1 \times p} = (x - \mu)_{1 \times p} \Gamma_{1 \times p}$ , dengan  $\Gamma$  diperoleh dari dekomposisi *Jordan*  $\Sigma$ , yaitu  $\Gamma^T \Sigma \Gamma = \Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_p)$  dengan  $\lambda_{is}$

menjadi nilai-nilai eigen dari dekomposisi. Setiap elemen  $y_{1 \times p}$  adalah kombinasi linear dari elemen  $x_{1 \times p}$ . Selain itu, masing-masing elemeny saling bebas.

Jadi akan didapatkan  $p$  komponen utama bebas sesuai dengan  $p$  nilai eigen dari dekomposisi *Jordan*  $\Sigma$ . Umumnya, hanya akan digunakan beberapa komponen utama pertama untuk regresi. (Maitra, 2008)

## 2.5 Analisis Regresi Linear Berganda

Analisis regresi merupakan teknik statistika yang digunakan untuk mengidentifikasi dan membuat model mengenai kaitan antar variabel. Kaitan tersebut adalah antara satu variabel tak bebas dengan satu atau lebih variabel bebas. Jika model regresi tersebut melibatkan lebih dari satu variabel bebas dengan kaitan bersifat linear maka disebut model regresi liner berganda. Model umum matematisnya adalah sebagai berikut:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon \quad (2.24)$$

yang terdiri dari satu variabel tak bebas ( $y$ ) dan  $k$  buah variabel bebas ( $x_i$ ). Parameter-parameter  $\beta_j$  dengan  $j = 0, 1, \dots, k$  disebut koefisien regresi. Pengertian linear digunakan untuk menunjukkan fungsi linear dari paramater  $\beta_j$  nya. Sedangkan  $\varepsilon$  menunjukkan pengaruh variabel lain yang tidak diamati. Model regresi tersebut akan diduga oleh model regresi yang diformulasikan berdasarkan contoh data yang diamati.

Dalam melakukan estimasi persamaan linier dengan menggunakan metode OLS, maka asumsi-asumsi dari OLS harus dipenuhi. Apabila asumsi tidak terpenuhi, maka tidak akan dapat menghasilkan nilai parameter yang BLUE (*Best Linier Unbiased Estimator*). Asumsi BLUE (Gujarati, 2004) yaitu :

- a. Nilai harapan dari rata-rata kesalahan adalah 0 (nol).
- b. Variansnya tetap (*homoskedastisity*).
- c. Tidak ada autokorelasi dalam gangguan.
- d. Variabel yang menjelaskan adalah nonstokastik (yaitu tetap dalam penyempelan berulang) atau jika stokastik didistribusikan secara independen dari gangguan variabel lain.
- e. Tidak ada multikolinearitas diantara variabel yang menjelaskan.

- f. Variabel didistribusikan secara normal dengan rata-rata dan varians yang diberikan oleh asumsi 1 dan 2.

Untuk mengetahui apakah model tersebut memenuhi asumsi BLUE atau tidak, perlu dilakukan beberapa pengujian yaitu uji multikolinearitas, uji autokorelasi, uji heteroskedastisitas dan juga uji normalitas untuk memastikan bahwa data terdistribusi secara normal. Cara yang digunakan untuk menguji penyimpangan asumsi klasik adalah sebagai berikut :

1. Uji multikolinieritas, menguji besar tidaknya hubungan antar variabel bebas. Multikolinieritas artinya terjadi hubungan linier yang tinggi antar variabel bebas.
2. Uji heteroskedastisitas, melihat apakah ada ketidaksamaan varians dari residual suatu pengamatan ke pengamatan lain. Jika terjadi heteroskedastisitas, maka kondisi variansi errornya tidak tetap atau tidak identik.
3. Uji autokorelasi, mengetahui apakah residual atau errornya tidak bebas atau berkaitan dari satu pengamatan ( $Y_i$ ) ke pengamatan lainnya ( $Y_j$ ), sehingga bisa menyebabkan antar pengamatan terjadi keterkaitan.
4. Uji normalitas, menguji apakah data sampel yang diambil telah mengikuti sebaran distribusi normal.
5. Uji linieritas, mengetahui apakah model yang dihasilkan bersifat linier atau tidak.

## 2.6 Interpolasi

Interpolasi adalah proses pencarian dan perhitungan nilai suatu fungsi yang grafiknya melewati sekumpulan titik yang diberikan. Titik-titik tersebut merupakan hasil penelitian atau diperoleh dari sebuah fungsi yang diketahui. Fungsi interpolasi biasanya dipilih dari sekelompok fungsi tertentu, salah satunya adalah fungsi polinomial yang paling banyak dipakai.

Interpolasi polinomial adalah sebuah metode untuk memperkirakan nilai di antara titik-titik data yang tepat. Persamaan polinomial adalah persamaan aljabar yang hanya mengandung jumlah dari variabel  $x$  berpangkat bilangan bulat. Bentuk umum persamaan polinomial order  $n$  adalah:

$$f(x) = a_0 + a_1x + a_2x^2 + \dots + ax^n \quad (2.25)$$

dengan  $a_0, a_1, a_2, \dots, a_n$  adalah parameter yang akan dicari berdasarkan titik data,  $n$  adalah derajat (order) dari persamaan polinomial, dan  $x$  adalah variabel bebas. Untuk  $(n + 1)$  titik data, hanya terdapat satu atau kurang polinomial order  $n$  yang melalui semua titik. Misalnya, hanya ada satu garis lurus (polinomial order 1) yang menghubungkan dua titik (interpolasi linier), demikian juga tiga buah titik dapat dihubungkan oleh fungsi parabola (interpolasi kuadrat) sedang untuk 4 titik (interpolasi kubik).

Interpolasi polinomial terdiri atas penentuan polinomial unik orde ke- $n$  yang cocok dengan  $n+1$  titik data. Walaupun hanya terdapat satu polinomial orde ke- $n$  yang cocok dengan  $n+1$  titik, terdapat beragam bentuk matematika untuk pengungkapan polinomial tersebut.

### **Bentuk Umum Interpolasi Polinomial**

Prosedur seperti dijelaskan diatas dapat digunakan untuk membentuk polinomial order  $n$  dari  $(n + 1)$  titik data. Bentuk umum polinomial order  $n$  adalah:

$$f_n(x) = b_0 + b_1(x - x_0) + \dots + b_n(x - x_0)(x - x_1) \dots (x - x_{n-1}) \quad (2.26)$$

Seperti yang dilakukan interpolasi linier dan kuadrat, titik-titik data dapat dilakukan dengan evaluasi koefisien  $b_0, b_1, \dots, b_n$ . Untuk polinomial order  $n$ , diperlukan  $(n + 1)$  titik data  $x_0, x_1, x_2, \dots, x_n$ . Dengan menggunakan titik-titik data tersebut, maka persamaan berikut digunakan untuk mengevaluasi koefisien  $b_0, b_1, \dots, b_n$ .

$$b_0 = f(x_0)$$

$$b_1 = f(x_1, x_0)$$

$$b_2 = f(x_2, x_1, x_0)$$

$$b_n = f(x_n, x_{n-1}, \dots, x_2, x_1, x_0) \quad (2.27)$$

Dengan definisi fungsi berkurung ( $[ \dots ]$ ) adalah pembagian beda hingga. Misalnya, pembagian beda hingga pertama adalah:

$$f[x_i, x_j] = \frac{f(x_i) - f(x_j)}{x_i - x_j}$$

Pembagian beda hingga kedua adalah:

$$f[x_i, x_j, x_k] = \frac{f[x_i, x_j] - f[x_j, x_k]}{x_i - x_j}$$

Pembagian beda hingga ke  $n$  adalah:

$$f[x_n, x_{n-1}, \dots, x_2, x_1, x_0] = \frac{f[x_n, x_{n-1}, \dots, x_1] - f[x_{n-1}, x_{n-2}, \dots, x_0]}{x_n - x_0}$$

Bentuk pembagian beda hingga tersebut dapat digunakan untuk mengevaluasi koefisien-koefisien yang kemudian disubstitusikan untuk mendapatkan interpolasi polinomial order  $n$ .

$$f_n(x) = f(x_0) + f[x_1, x_0](x - x_0) + f[x_2, x_1, x_0](x - x_0)(x - x_1) + \dots + f[x_n, x_{n-1}, \dots, x_2, x_1, x_0](x - x_0)(x - x_1) \dots (x - x_{n-1}) \quad (2.28)$$

### Konversi Frekuensi Data

Metode konversi frekuensi data dari frekuensi rendah ke frekuensi tinggi (misal: data tahunan menjadi data bulanan) pada program *EViews 7* adalah sebagai berikut:

- a. *Constant - match average* dan *constant - match sum*.

Kedua metode tersebut menetapkan nilai yang sama untuk semua observasi dalam deret frekuensi tinggi yang terkait dengan periode frekuensi rendah tertentu. Pada metode pertama, nilai tersebut dipilih sehingga rata-rata observasi frekuensi tinggi sesuai dengan observasi frekuensi rendah (nilainya diulang). Pada metode kedua, nilai tersebut dipilih sehingga jumlah observasi frekuensi tinggi sesuai dengan observasi frekuensi rendah (nilai dibagi dengan jumlah observasi).

- b. *Quadratic - match average* dan *quadratic - match sum*

Kedua metode ini sesuai untuk polinomial kuadrat lokal untuk setiap observasi dari deret frekuensi rendah, kemudian polinomial ini digunakan untuk mengisi semua observasi dari seri frekuensi tinggi yang berkaitan dengan periode. Kuadrat polinomial dibentuk dengan mengambil tiga titik yang berdekatan dari deret awal dan mencocokkan kuadratnya sehingga baik rata-rata maupun jumlah titik dari frekuensi tinggi sesuai dengan data frekuensi rendah yang sebenarnya. Untuk titik terbanyak, satu titik sebelum dan satu titik sesudah periode yang diinterpolasi tersebut digunakan untuk menetapkan tiga titik. Untuk titik ujung, kedua periode itu diambil dari satu sisi dimana data tersedia.

Metode ini adalah metode lokal murni. Kurva interpolasi yang dihasilkan tidak dibatasi terus menerus pada batas antara periode yang berdekatan. Sehingga

metode ini lebih cocok untuk situasi di mana titik data relatif sedikit yang diinterpolasi dan sumber data yang cukup halus. Untuk mencari nilai data bulanan dari data tahunan dapat digunakan metode *quadratic - match sum*, yaitu

$$q_s = \frac{1}{12} \left[ q_t - \left( \frac{s - 6,5}{12} \right) (q_s - q_{t-1}) \right], \quad s = 1, 2, 3, \dots, 12 \quad (2.29)$$

Keterangan:

$q_s$  = data bulan ke- $s$

$q_t$  = data tahun  $t$

$q_{t-1}$  = data tahun  $t-1$  (sebelumnya)

$s$  = bulan

c. *Linear - match last*

Metode ini memberikan setiap nilai dalam deret frekuensi rendah ke observasi frekuensi tinggi terakhir yang terkait dengan periode frekuensi rendah, kemudian menempatkan semua titik menengah pada garis lurus yang menghubungkan titik-titik.

d. *Cubic - match last*

Metode ini memberikan setiap nilai dalam deret frekuensi rendah ke observasi frekuensi tinggi terakhir yang terkait dengan periode frekuensi rendah, kemudian menempatkan semua titik menengah pada spline kubik alami menghubungkan semua poin.

## 2.7 Penelitian – Penelitian yang Relevan

Pada bagian ini diuraikan beberapa hasil penelitian yang relevan tentang prediksi/peramalan berdasarkan jaringan syaraf tiruan, antara lain sebagai berikut:

Noori (2011) melakukan penelitian memprediksi aliran sungai bulanan menggunakan Support Vector Machine (SVM). Ada beberapa teknik memilih input yang digunakan, salah satunya menggunakan PCA. Dari delapan belas variabel input awal, direduksi jumlahnya menggunakan PCA menjadi lima variabel input. Hasil penelitian menunjukkan bahwa *preprocessing* variabel masukan dengan cara teknik PCA meningkatkan model operasi SVM.

Ulfah (2011) menggunakan Jaringan Syaraf Tiruan untuk memprediksi risiko pemberian kredit dalam manajemen risiko keuangan. Implementasi JST



dipadukan dengan metode analisis komponen utama atau *Principal Component Analysis* (PCA) untuk meningkatkan performansi. Hasil simulasi terhadap dataset kredit yang direduksi dengan PCA menggunakan software Matlab 7.8 menunjukkan batas waktu komputasi selama proses *training* lebih singkat dari dimensi input yang asli.

Kheirkhah (2012) membuat algoritma baru untuk mengestimasi dan memprediksi konsumsi listrik masyarakat setiap bulannya. Penelitian ini menyajikan pendekatan yang menggunakan JST, PCA, DEA, dan metode ANOVA. Metode JST digunakan dengan algoritma pembelajaran MLP. Sedangkan PCA digunakan sebagai metode untuk memilih input. Algoritma-algoritma tersebut terbukti dapat menghasilkan solusi yang akurat untuk menyelesaikan masalah prediksi konsumsi listrik.

Penggunaan algoritma Backpropagation dalam peramalan dilakukan oleh Utomo (2007) menggunakan jaringan syaraf tiruan untuk memprediksi datatime series dan wavelet untuk mendekomposisi dan merekonstruksi data sehingga dapat mengurangi banyaknya epoch. Pada *Wavelet-Artificial Neural Network* (WANN) ada tiga tahapan untuk mendapatkan hasil prediksi data times yaitu *pre-processing, prediction, dan post-processing*. Hasil penelitian membuktikan bahwa Jaringan Syaraf Tiruan Backpropagation memiliki keakuratan yang tinggi untuk memprediksi atau mengenali pola data yang diinputkan.

Pino (2007) dalam penelitiannya tentang perkiraan harga energi listrik per jam hari berikutnya di Spanyol, digunakan jaringan syaraf tiruan *Multi Layer Perceptron* (MLP). Perkiraan ini dibandingkan dengan yang diperoleh dari metode *Box-Jenkins* peramalan ARIMA. Hasil penelitian menunjukkan bahwa MLP lebih baik dari model ARIMA dari segi keakuratan dan kecepatan.

Faradisa (2009) melakukan penelitian untuk mendapatkan algoritma training pada FeedForward Neural Network (FNN) yang efisien serta memiliki akurasi yang tinggi pada permasalahan forecasting. Pada penelitian ini dilakukan penggabungan/hybrid antara algoritma *Backpropagation* dan *Particle Swarm Optimization with Constriction* (PSOC), yaitu PSOC-BP. Pada algoritma baru ini, training dimulai dari nilai bobot yang telah dekat dengan nilai global optimum

sehingga algoritma akan konvergen ke nilai optimum dengan lebih cepat serta diperoleh penyelesaian dengan nilai akurasi yang lebih tinggi.

Sheng-qiang, dkk (2009) dalam mengembangkan algoritma *Backpropagation* untuk membentuk model Jaringan Syaraf Tiruan dalam peramalan ledakan batubara dan gas. Dalam penelitian ini dipilih lima faktor dominan pada *grey correlation analysis* sebagai parameter inputnya. Untuk melakukan *training*, digunakan sampel pembelajaran dari ledakan batubara dan gas pada tambang yang ada di Cina.

Juhos (2008) memprediksi konsentrasi NO dan NO<sub>2</sub> empat hari berikutnya, salah satunya menggunakan *Multi Layer Perceptron* (MLP). Prediksi dilakukan sebagian atas dasar nilai-nilai sebelumnya, dan sebagian lain berdasarkan data kecepatan suhu, kelembaban dan angin. Untuk mengurangi dimensi dari data input, digunakan analisis *time series* yaitu *Principal Component*.

Masalah peramalan yang berhubungan dengan kemiskinan yaitu tentang konsumsi rumah tangga dilakukan oleh Al-Foul (2012) dalam penelitiannya di Yordania. Data yang digunakan adalah data tahunan selama periode 1976-2008 dengan metode *JST Backpropagation*. Sedangkan variabel bebas sebagai variabel input yaitu PDB, populasi, ekspor, dan impor. Hasil penelitian menunjukkan bahwa terjadi peningkatan konsumsi energi dari tahun 2015, 2020 hingga 2025. Sehingga para pembuat kebijakan dapat menggunakan ini sebagai bahan pertimbangan dalam pembuatan keputusan dekade berikutnya.

## **BAB 3**

### **METODA PENELITIAN**

Tahapan penelitian dalam prediksi jumlah penduduk miskin di Provinsi Kalimantan Selatan menggunakan Jaringan Syaraf Tiruan adalah sebagai berikut:

#### **3.1 Pengumpulan Data**

Data yang digunakan dalam penelitian ini sebanyak 62 jenis data, terdiri dari:

1. Jumlah penduduk miskin.
2. Pengeluaran rata-rata perkapita sebulan untuk kelompok makanan yang terdiri dari 13 kelompok, yaitu:
  - a. Padi-padian,
  - b. Umbi-umbian,
  - c. Ikan,
  - d. Daging,
  - e. Telur dan susu,
  - f. Sayur-sayuran,
  - g. Buah-buahan,
  - h. Minyak dan lemak,
  - i. Bahan minuman,
  - j. Konsumsi lainnya,
  - k. Makanan jadi,
  - l. Minuman beralkohol,
  - m. Tembakau dan sirih.
3. Pengeluaran rata-rata perkapita sebulan untuk kelompok non makanan yang terdiri dari 8 kelompok, yaitu:
  - a. Perumahan, bahan bakar, penerangan dan air,
  - b. Aneka barang dan jasa,
  - c. Pendidikan,
  - d. Kesehatan,
  - e. Pakaian, alas kaki dan tutup kepala,

- f. Barang yang tahan lama,
  - g. Pajak pemakaian dan premi asuransi,
  - h. Keperluan pesta dan upacara.
4. Produk Domestik Regional Bruto (PDRB) yang terbagi menjadi Atas Dasar Harga Berlaku (ADHK) dan Atas Dasar Harga Konstan 2000(ADHK) pada setiap sektornya sejumlah 9 sektor, yaitu:
- a. PDRB Sektor Bangunan,
  - b. PDRB Sektor Industri Pengolahan,
  - c. PDRB Sektor Jasa-Jasa,
  - d. PDRB Sektor Keuangan, Persewaan dan Jasa Perusahaan,
  - e. PDRB Sektor Listrik, Gas dan Air Bersih,
  - f. PDRB Sektor Pengangkutan dan Komunikasi,
  - g. PDRB Sektor Perdagangan, Hotel dan Restoran,
  - h. PDRB Sektor Pertambangan dan Penggalian,
  - i. PDRB Sektor Pertanian.
5. Inflasi yang terdiri dari 7 kelompok, yaitu:
- a. Kelompok bahan makanan,
  - b. Kelompok makanan jadi, minuman, rokok, dan tembakau,
  - c. Kelompok perumahan, air, listrik, gas, dan bahan bakar,
  - d. Kelompok sandang,
  - e. Kelompok kesehatan,
  - f. Kelompok pendidikan, rekreasi, dan olahraga,
  - g. Kelompok transpor, komunikasi, dan jasa keuangan.
6. Nilai ekspor menurut komoditi terbagi menjadi 5 komoditi, yaitu:
- a. Kelapa sawit,
  - b. Karet alam,
  - c. Produk kayu,
  - d. Produk perikanan,
  - e. Produk tambang.
7. Luas panen tanaman padi dan palawija perbulan yang terdiri dari dari 7 kelompok, yaitu:
- a. Padi,

- b. Jagung,
  - c. Kedelai,
  - d. Kacang tanah,
  - e. Kacang hijau,
  - f. Ubi kayu,
  - g. Ubi jalar.
8. Jumlah industri manufaktur besar dan sedang.
  9. Jumlah pengangguran.
  10. Jumlah penduduk.

Data yang digunakan merupakan data dari tahun 2002 hingga 2011 dan merupakan data bulanan. Namun, ada beberapa data yang merupakan data tahunan. Oleh karena itu, digunakan teknik interpolasi untuk mengubah data tersebut menjadi data bulanan.

Yang merupakan data tahunan adalah data jumlah penduduk, jumlah penduduk miskin, pengeluaran rata-rata per kapita sebulan, PDRB, nilai ekspor, jumlah industri manufaktur besar dan sedang, dan jumlah pengangguran. Data ini akan diambil dari data Badan Pusat Statistik (BPS) Provinsi Kalimantan Selatan, terkecuali data nilai ekspor serta jumlah industri manufaktur besar dan sedang yang diambil dari data Dinas Perindustrian dan Perdagangan (Disperindag) Kalimantan Selatan.

### **3.2 Data Preprocessing/Normalisasi Data**

Sebelum data input dan target yang diimplementasikan ke dalam jaringan saraf tiruan harus terlebih dahulu melalui preprosesing data berupa penskalaan yaitu normalisasi data. Tujuannya agar jaringan saraf dapat mengenali data yang akan menjadi masukan bobot-bobotnya. Data akan bernilai antara 0 sampai 1, hal ini sesuai dengan fungsi aktivasi yang akan digunakan.

$$N = \frac{(D - Dk)}{(Db - Dk)} \quad (3.1)$$

Keterangan:

N = Data yang sudah ternormalisasi,

D = Data yang akan dinormalisasi,

Dk = Data terkecil dari sekumpulan data,

Db = Data terbesar dari sekumpulan data.

### 3.3 Pemilihan Data

Dari 62 indikator yang diasumsikan berkaitan dengan jumlah penduduk miskin, dilakukan pemilihan variabel yang berhubungan dengan variabel jumlah penduduk miskin menggunakan metode PCA dan metode regresi stepwise, kemudian indikator tersebut diprediksi data bulanan tahun 2012 dan dimasukkan pada persamaan regresi linier berganda untuk mencari angka jumlah penduduk miskin bulanan tahun 2012.

### 3.4 Pembuatan Program dengan MATLAB Menggunakan Arsitektur Jaringan

Untuk melakukan proses selanjutnya, akan dibuat program yang memuat tahapan pelatihan dan pengujian dengan menggunakan *software* MATLAB R2010a, dan proses running dilakukan pada komputer dengan *processor* Intel® Pentium® dual-core CPU T4400 2,2 GHz.

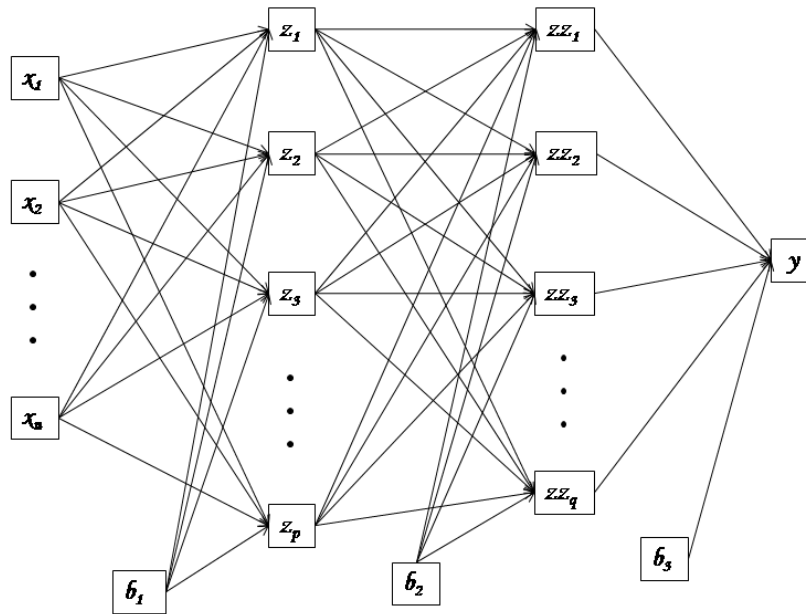
Arsitektur jaringan terdiri dari *input layer* sejumlah  $n$  neuron, *hidden layer* sejumlah  $p$  neuron, dan *output layer* terdiri dari 1 neuron. Jumlah input ditentukan dengan mencari *time window* yang tepat dengan cara *trial and error*, sedangkan jumlah *hidden* ditentukan dengan cara *trial and error* pula, dalam arti hasil pembelajaran yang tercepat dan terbaik yang akan menentukan jumlah *time window* dan *hidden layer* tersebut. JST yang digunakan di sini adalah JST *time series*, sehingga baik input maupun target merupakan data variabel itu sendiri.

Fungsi aktivasi yang digunakan pada *hidden layer* adalah fungsi sigmoid

$$f(x) = \frac{1}{1 + \exp(-x)} \quad (3.2)$$

Sedangkan fungsi aktivasi pada *output layer* sudah merupakan *default* pada fungsi *newff* pada Matlab R2010a yaitu fungsi identitas (*purelin*).

$$f(x) = x \quad (3.3)$$



Gambar 3.1 Arsitektur JST

### 3.5 Training/Pelatihan Jaringan Syaraf Tiruan

Proses pelatihan jaringan syaraf tiruan menggunakan data yang ada dengan mengatur *time window*, jumlah *hidden layer*, dan parameter sehingga menghasilkan arsitektur neural network yang berbeda-beda. Setiap arsitektur neural network akan menghasilkan bobot pelatihan yang terakhir yang akan digunakan sebagai bobot awal saat melakukan pengujian.

### 3.6 Testing/Pengujian Jaringan Syaraf Tiruan

Proses pengujian jaringan syaraf tiruan menggunakan data dengan *time window*, jumlah *hidden layer*, dan parameter yang menghasilkan arsitektur jaringan terbaik berdasarkan nilai kriteria informasi yang dihasilkan.

Setelah dilakukan *training* dan *testing* pola-pola yang dilatih, akan diperoleh hasil bahwa *test* terhadap pola-pola tersebut benar (akurat).

Perhitungan akurasi jaringan pada proses testing adalah:

$$MSE = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{N} \quad (3.4)$$

$$MAE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{N} \quad (3.5)$$

$$\begin{aligned}
 &MAPE \\
 &= \frac{\sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i}}{N}
 \end{aligned}
 \tag{3.6}$$

dengan

$y_i$  = nilai aktual data

$\hat{y}_i$  = nilai prediksi

$N$  = jumlah nilai *testing*

### 3.7 Data Postprocessing/Denormalisasi

Data postprocessing adalah pengubahan dari data hasil jaringan syaraf tiruan yang mempunyai range antara 0 sampai 1 menjadi data seperti nilai aslinya.

$$P = O \times (Db - Dk) + Dk \tag{3.7}$$

dengan

$P$  = Data postprocessing atau data yang sudah denormalisasi,

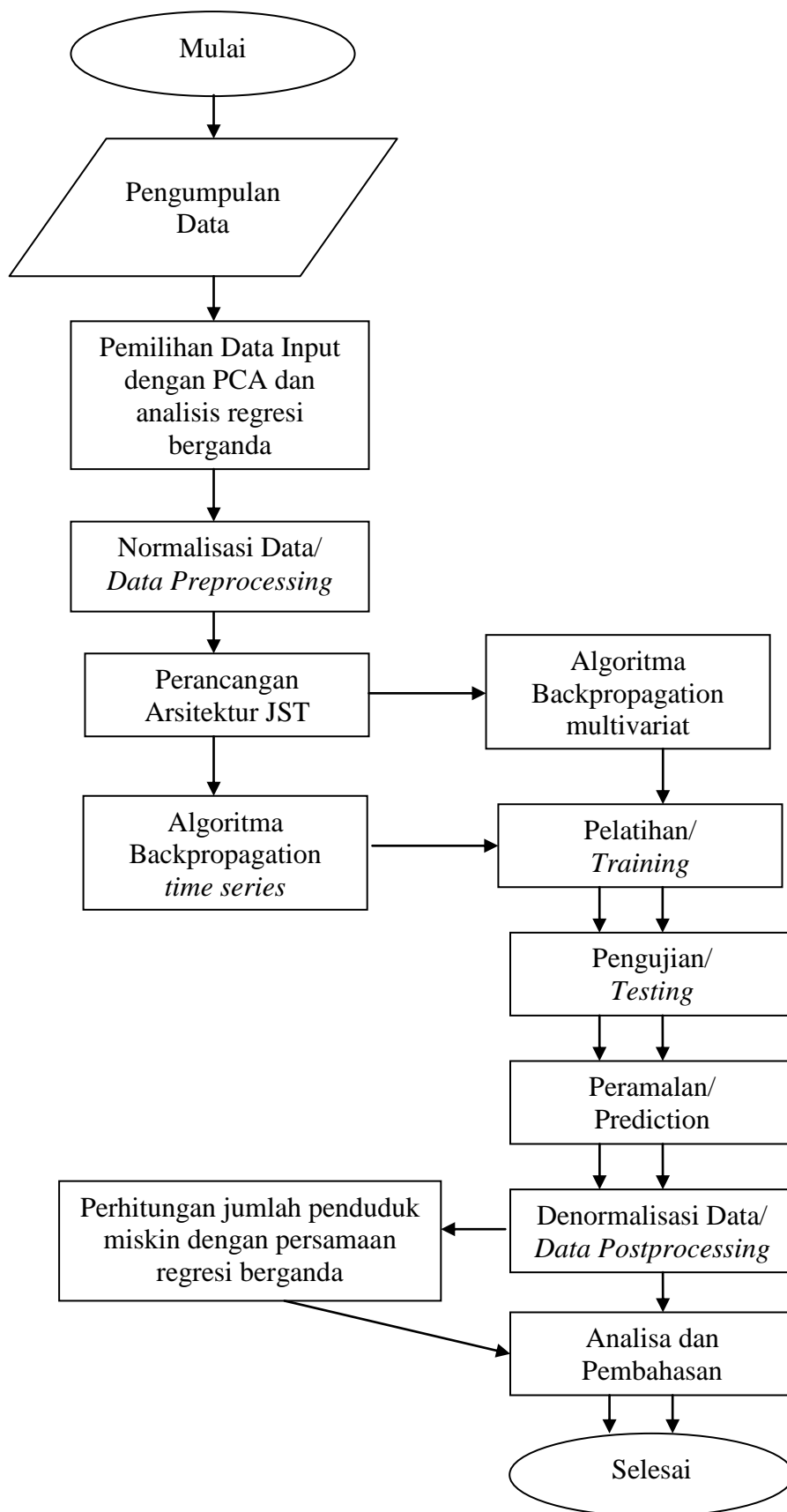
$O$  = Data output jaringan,

$Db$  = Data terbesar dari sekumpulan data,

$Dk$  = Data terkecil dari sekumpulan data.

Diagram alir sistem yang menggambarkan tahapan penelitian sesuai dengan yang telah diuraikan di atas adalah sebagai berikut





Gambar 3.2 Diagram penelitian



## **BAB 4**

### **HASIL PENELITIAN DAN PEMBAHASAN**

#### **4.1 Pengolahan Data**

Proses pertama yang dilakukan dalam penyelesaian penelitian ini setelah semua data sekunder didapatkan adalah interpolasi data yang bersifat tahunan menjadi data bulanan. Interpolasi dilakukan dengan program *Eviews 7*. Terdapat dua metode interpolasi yang digunakan pada penelitian ini yaitu:

- a. *Linear-match last*, digunakan untuk data jumlah penduduk, jumlah penduduk miskin, pengeluaran rata-rata perkapita sebulan untuk kelompok makanan dan non makanan, jumlah industri manufaktur besar dan sedang, dan jumlah pengangguran.
- b. *Quadratic match sum*, digunakan untuk data PDRB dan nilai ekspor.

#### **4.2 Pemilihan Data**

Pada pengumpulan data sebelumnya, dicari semua data yang diasumsikan berpengaruh pada prediksi jumlah penduduk miskin. Dari semua data yang diasumsikan dilakukan uji multikolinieritas. Terbukti bahwa nilai VIF masih ada yang lebih dari 10, artinya telah terjadi multikolinieritas yang serius di dalam model regresi ini. Oleh karena itu, sebelum memprediksi, dilakukan metode pemilihan variabel dari 62 data variabel yang ada direduksi sehingga hanya beberapa data yang dianggap mewakili data – data lainnya agar tidak terjadi multikolinieritas antar variabel bebas. Jumlah data yang dipakai berdasarkan pada jumlah komponen yang memiliki nilai eigen dan variansi terbesar. Dari 62 data yang ada, terbentuk 11 komponen yang memiliki nilai eigen lebih dari 1 dengan nilai eigen terkecil 1,010 dan yang terbesar 39,932. Metode pemilihan yang digunakan adalah *Principal Analysis Component (PCA)*. PCA digunakan untuk melihat variabel-variabel yang menjadi faktor pembentuk komponen - komponen melalui nilai korelasi tertinggi pada masing - masing komponennya.

Tabel 4.1 *Rotated Component Matrix*

Komponen	Variabel	Korelasi dengan variabel sekomponen
1	PDRB sektor jasa atas dasar harga konstan	0.996
2	Pengeluaran barang tahan lama	0.637
3	Pengeluaran buah	0.854
4	Ekspor ikan	- 0.813
5	Luas Panen Ubi Jalar	0.681
6	LuasPanen KacangTanah	0.843
7	Inflasi Makanan Jadi	0.782
8	Pengangguran	0.704
9	Inflasi Sandang	0.824
10	Inflasi Pendidikan DII	0.860
11	Inflasi Kesehatan	0.839

Extraction Method: Principal Component Analysis.

Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 10 iterations.

Berdasarkan *total variance explained*, terdapat 11 komponen yang memiliki total *initial eigen value* atau nilai eigen yang bernilai lebih dari 1 sehingga komponen tersebut dapat menjelaskan variabel dengan baik. Komponen-komponen tersebut juga dapat menjelaskan varian yang cukup besar.

Variabel yang memiliki nilai tertinggi pada masing – masing komponen dilihat dari *rotated component matrix* yang merupakan hasil rotasi sebanyak 10 kali terhadap analisis korelasi *component matrix* yaitu:

1. PDRB sektor jasa-jasa atas dasar harga konstan 2000,
2. Pengeluaran rata-rata perkapita sebulan untuk buah,
3. Nilai ekspor produk perikanan,
4. Luas panen ubi jalar,
5. Luas panen kacang tanah,
6. Inflasi kelompok makanan jadi, minuman, rokok, dan tembakau,
7. Jumlah pengangguran,
8. Pengeluaran rata-rata perkapita sebulan untuk barang tahan lama,
9. Inflasi kelompok sandang
10. Inflasi kelompok pendidikan, rekreasi, dan olahraga,
11. Inflasi kelompok kesehatan.

## Pendefinisian Variabel

Pemilihan model regresi terbaik pada penelitian ini menggunakan metode Stepwise Regression karena relatif memiliki kelebihan yaitu sifat penilaiannya reversible terhadap variabel penjelas yang akan masuk persamaan regresi, maka pemeriksaan yang dilakukan relatif lebih ketat (Wibisono, 2002). Dengan menggunakan metode Stepwise regression diperoleh model regresi terbaik dengan menggunakan 4 variabel penjelas yang ditunjukkan dengan persamaan regresi linier berganda sebagai berikut:

$$U = -71339.476 + 1.107u_1 + 0.092u_2 + 0.151u_3 + 0.435u_4 \quad (4.1)$$

Keterangan:

$U$  = Penduduk Miskin

$u_1$  = PDRB sektor jasa – jasa atas dasar harga konstan

$u_2$  = Pengangguran

$u_3$  = Pengeluaran rata-rata perkapita sebulan untuk barang tahan lama

$u_4$  = Pengeluaran rata-rata perkapita sebulan untuk kelompok buah

Interpretasi model regresi tersebut yaitu:

1. Koefisien regresi yang positif pada setiap variabel bebas menunjukkan bahwa jumlah penduduk miskin akan semakin tinggi jika PDRB sektor jasa – jasa atas dasar harga konstan, jumlah pengangguran, pengeluaran rata-rata perkapita sebulan untuk barang tahan lama, dan pengeluaran rata-rata perkapita sebulan untuk kelompok buah semakin tinggi,
2. Jika untuk satu variabel bebas berubah sebesar 1 satuan pada saat variabel bebas lainnya tetap maka pengaruhnya terhadap jumlah penduduk miskin adalah meningkat 1,107 satuan jika PDRB sektor jasa – jasa atas dasar harga konstan meningkat 1 satuan, meningkat 0,092 satuan jika jumlah pengangguran meningkat 1 satuan, meningkat 0,151 satuan jika pengeluaran rata-rata perkapita sebulan untuk barang tahan lama meningkat 1 satuan, dan meningkat 0,435 satuan jika pengeluaran rata-rata perkapita sebulan untuk kelompok buah meningkat 1 satuan.

## Uji Hipotesis

Uji hipotesis merupakan uji signifikansi untuk menentukan apakah terdapat hubungan linear antara variabel tak bebas  $y$  dengan variabel bebas  $u_1, u_2, u_3$  dan  $u_4$ . Uji hipotesis ini terdiri dari dua bagian, yaitu uji signifikansi secara umum mengenai hubungan linear antara variabel tak bebas dengan variabel bebas dengan menggunakan statistik uji F dan uji signifikansi individual untuk hubungan variabel tak bebas dengan masing-masing variabel bebas dengan menggunakan statistik uji t.

### 1. Uji F untuk uji signifikansi persamaan regresi

Uji signifikansi yang digunakan untuk menentukan hubungan antara variabel tak bebas dengan variabel bebas secara umum ini menggunakan formulasi hipotesis sebagai berikut:

$$H_0: \beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$$

$$H_1: \beta_j \neq 0 \text{ untuk paling sedikit satu } j$$

Statistik uji yang digunakan adalah Uji F dengan rumus perhitungannya adalah

$$F = \frac{MSR}{MSE} \quad (4.2)$$

Hasil perhitungan dengan menggunakan perangkat lunak SPSS, nilai F hitungnya adalah sebagai berikut:

Tabel 4.2 Uji F dengan ANOVA

Model	Sum of Squares	Df	Mean Square	F	Sig.
Regression	222223150 601.801	4	55555787650. 450	12210.668	.000(d)
Residual (Error)	523224094 .066	115	4549774.731		
Total	222746374 695.867	119			

Karena Signif. F-nya relatif kecil atau  $F_{hitung} (12210,668) > F_{0,05;4,120} (2,45)$  maka tolak  $H_0$ . Sehingga dapat disimpulkan bahwa ada kaitan linear yang signifikan antara jumlah penduduk miskin dengan variabel-variabel bebas yang meliputi PDRB sektor jasa – jasa atas dasar harga konstan, jumlah pengangguran, pengeluaran rata-rata perkapita sebulan untuk barang tahan lama, dan pengeluaran rata-rata perkapita sebulan untuk kelompok buah.

## 2. Uji t untuk masing-masing koefisien regresi

Hipotesis untuk menguji signifikansi masing-masing koefisien regresi, yaitu  $\beta_j$ , adalah sebagai berikut:

$$H_0: \beta_j = 0$$

$$H_1: \beta_j \neq 0$$

Jika  $H_0: \beta_j = 0$  tidak ditolak atau artinya tidak ada kaitan antara satu variabel  $x_i$ , maka variabel tersebut dapat dihilangkan dari model regresinya. Statistik ujinya adalah :

$$t_0 = \frac{\beta_j}{\sqrt{\sigma^2 C_{jj}}} \quad (4.3)$$

Hasil perhitungan dengan menggunakan SPSS adalah:

Tabel 4.3 Uji t pada *coefficients*

Model	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta		
(Constant)	-71339.476	1482.396		-48.124	.000
PDRB_Jasa2	1.107	.008	.941	130.882	.000
Pengangguran	.092	.009	.048	9.847	.000
Peng.Barang_Tahan_Lama	.151	.022	.048	7.012	.000
Peng.Buah	.435	.110	.019	3.952	.000

Berdasarkan nilai statistik uji t untuk masing-masing koefisien regresi di atas maka dapat disimpulkan sebagai berikut :

1. Berdasarkan nilai signif t-nya maka semua variabel bebas memiliki tingkat signifikansi hubungan linear yang tinggi dengan variabel tak bebas.
2. Jika menggunakan perbandingan dengan nilai  $t_{0,05,120}$  sebesar 1,65765 pada  $\alpha = 0,10$ , semua nilai t hitung lebih besar daripada t tabel, sehingga semua variabel bebas menunjukkan adanya kaitan linear yang signifikan dengan variabel tak bebasnya yaitu jumlah penduduk miskin.

### Uji Kecocokan Model

Mengevaluasi kecocokan model merupakan bagian terpenting dari masalah regresi berganda. Beberapa metoda yang digunakan untuk evaluasi

tersebut meliputi koefisien determinansi, plot residual, metoda pengukuran residual, dan sebagainya. Penggunaan metoda-metoda tersebut disini akan lebih dititikberatkan pada menilai kecocokkan model, identifikasi data pencilan, dan pengaruh data pencilan serta cara penanganannya.

Untuk menguji apakah model regresi linear berganda yang telah diformulasikan tersebut cocok untuk contoh data yang dianalisis, digunakan koefisien korelasi ( $R$ ), koefisien determinan ( $R^2$ ) dan *adjusted R<sup>2</sup>*. Nilai-nilai tersebut yang dihitung dengan menggunakan SPSS adalah sebagai berikut:

Tabel 4.4 Uji kecocokan model pada *model summary*

R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
.999(d)	.998	.998	2133.02010	.044

$R$  dengan nilai sebesar 0,999 merupakan koefisien korelasi berganda antara variabel tak bebas  $y$  dengan himpunan variabel bebas  $u_1, u_2, u_3$  dan  $u_4$  yang menunjukkan ukuran hubungan linier diantara kedua jenis variabel tersebut. Nilai  $R^2$  sebesar 0.998 menunjukkan bahwa sekitar 99,8 persen keragaman variabel tak bebas dapat diterangkan oleh keragaman variabel-variabel bebasnya. Atau jika dihubungkan dengan jenis variabel yang diamati, keragaman jumlah penduduk miskin dapat diterangkan 99,8 persen oleh keragaman variabel PDRB sektor jasa – jasa atas dasar harga konstan, jumlah pengangguran, pengeluaran rata-rata perkapita sebulan untuk barang tahan lama, dan pengeluaran rata-rata perkapita sebulan untuk kelompok buah.

Nilai *adjusted R<sup>2</sup>* sebesar 0,998 menunjukkan cara interpretasi yang sama secara umum menunjukkan bahwa pada model tersebut semua variabel bebas secara signifikan berkaitan linier dengan variabel tak bebas.

Berdasarkan ukuran-ukuran di atas maka dapat disimpulkan bahwa model regresi berganda yang sudah diformulasikan tersebut cocok digunakan untuk melihat kaitan antara variabel tak bebas  $y$  dan 4 buah variabel bebas  $u_i$ .

### Uji Asumsi Klasik

#### 1. Uji Multikolinearitas



Pengujian multikolinearitas bertujuan untuk melihat apakah ke 4 variabel bebas tidak berkorelasi satu sama lain (atau saling bebas). Multikolinieritas pada setiap variabel dalam analisis regresi linier berganda dengan SPSS dapat diuji dengan melihat nilai signifikansi pada uji t. Apabila bernilai lebih besar dari 0.05 (atau lebih besar dari nilai  $\alpha$  lainnya), maka terjadi multikolinieritas. Pada model ini dapat dilihat dari uji t sebelumnya bahwa signifikansinya 0,000, sehingga terbebas dari multikolinieritas.

Selain itu, uji multikolinearitas juga dapat dilakukan dengan menggunakan nilai *variance inflation factor* (VIF). Model dinyatakan terbebas dari gangguan multikolinearitas jika mempunyai nilai VIF di bawah 10 atau *tolerance* di atas 0,1. Berikut adalah nilai VIF dan *tolerance* hasil SPSS dalam penelitian ini:

Tabel 4.5 Uji multikolinieritas dengan VIF

Model	Collinearity Statistics	
	Tolerance	VIF
(Constant)		
PDRB_Jasa2	.395	2.531
Pengangguran	.857	1.167
Peng.Barang_Tahan_Lama	.442	2.261
Peng.Buah	.888	1.127

Tabel di atas memberikan semua nilai VIF di bawah 10 atau nilai tolerance di atas 0,1. Berarti tidak terdapat gejala multikolinearitas pada model dalam penelitian ini.

## 2. Uji Heteroskedastisitas

Uji heteroskedastisitas bertujuan untuk menguji apakah dalam model regresi terjadi ketidaksamaan varians dari residual satu pengamatan ke pengamatan yang lain. Model regresi yang baik adalah yang homoskedastisitas atau tidak terjadi heteroskedastisitas. Gejala heteroskedastisitas lebih sering terjadi pada data cross section (Imam Ghazali, 2005).

Uji heteroskedastisitas dengan metode glejser menunjukkan jika asymp sig. pada masing-masing variabel independen  $> 5\%$ , maka data tidak mengalami heteroskedastisitas, dan sebaliknya.

Tabel 4.6 Uji heteroskedastisitas pada *coefficients*

Model	Unstandardized Coefficients		Standardized Coefficients	t	Sig.
	B	Std. Error	Beta	Zero-order	Partial
1 (Constant)	-164.013	1466.199		-.112	.911
PDRB_Jasa2	.002	.008	.033	.224	.823
Pengangguran	.001	.009	.006	.061	.951
Peng.Barang_Tahan_Lama	-.019	.021	-.128	-.920	.359
Peng.Buah	.008	.106	.008	.080	.937

a Dependent Variable: absresid

### 3. Uji Autokorelasi

Menurut Imam Ghozali (2005), uji autokorelasi digunakan untuk mengetahui apakah dalam model regresi linear ada korelasi antara kesalahan pengganggu pada periode  $t$  dengan kesalahan pengganggu pada periode  $t-1$  (sebelumnya), dimana jika terjadi korelasi dinamakan ada problem autokorelasi. Autokorelasi muncul karena observasi yang berurutan sepanjang waktu berkaitan satu sama lainnya. Masalah ini timbul karena residual (kesalahan pengganggu) tidak bebas dari satu observasi ke observasi lainnya. Hal ini sering ditemukan pada data runtut waktu (time series).

Berikut adalah nilai Durbin-Watson pada model dalam penelitian ini:

Tabel 4.7 Uji autokorelasi Durbin Watson pada *model summary*

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
4	.999(d)	.998	.998	2133.02010	.044

d Predictors: (Constant), PDRB\_Jasa2, Pengangguran, Peng.Barang\_Tahan\_Lama, Peng.Buah

e Dependent Variable: Penduduk\_Miskin

Adapun nilai  $d$  untuk 4 buah variabel bebas dengan 120 data pada taraf 5% adalah sebesar 0.044, sedangkan nilai  $dL = 1,6339$  dan  $dU = 1,7715$ . Tampak bahwa  $0 < d < dL$  sehingga terjadi gangguan autokorelasi. Namun, dikarenakan jumlah data lebih dari 100, maka ada kemungkinan hasil dari uji Durbin Watson tidak tepat. Oleh karena itu dilakukan pengujian kembali menggunakan *Lagrange Multiplier Test* untuk ukuran sampel cukup besar, lebih dari 100 data dan dengan derajat autokorelasi lebih dari satu. Pada uji ini dilakukan transformasi pada residualnya, kemudian dijadikan tambahan variabel bebasnya sehingga total variabel bebas menjadi 5 buah. Sedangkan nilai residual menggantikan variabel

jumlah penduduk miskin untuk menjadi variabel tak bebas atau bergantungnya. Dari hasil regresi 5 variabel bebas dan 1 variabel tak bebas tersebut, dilihat nilai  $R^2$  sebagai dasar untuk menghitung nilai  $X^2$ .

Tabel 4.8 Uji Autokorelasi LM Test

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
1	.984(a)	.968	.967	381.00360838	.444

a Predictors: (Constant), residual\_2, Pengangguran, Peng.Buah, Peng.Barang\_Tahan\_Lama, PDRB\_Jasa2

b Dependent Variable: residual\_1

$$X^2_{hitung} = (n - 1) \times R^2 = (4 - 1) \times 0,968 = 2,904 \quad (4.4)$$

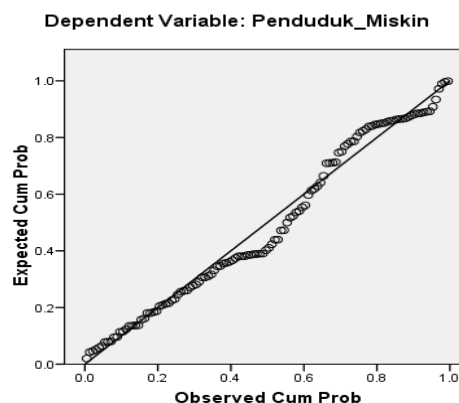
$$X^2_{tabel} = X^2_{((4-1);0,05)} = 7,815 \quad (4.5)$$

Karena  $X^2_{hitung} < X^2_{tabel}$ , maka persamaan regresi tidak mengandung masalah autokorelasi.

#### 4. Uji Normalitas

Uji normalitas data dipergunakan untuk menentukan apakah data terdistribusi secara normal atau tidak. Uji normalitas yang dipergunakan adalah plot grafik di mana asumsi normalitas terpenuhi jika titik-titik pada grafik mendekati sumbu diagonalnya.

Normal P-P Plot of Regression Standardized Residual



Gambar 4.1 Uji normalitas P-P Plot

Gambar menunjukkan bahwa titik-titik pada grafik telah mendekati atau hampir berhimpit dengan sumbu diagonal atau membentuk sudut 45 derajat dengan garis mendatar. Interpretasinya adalah bahwa nilai residual pada model penelitian telah

terdistribusi secara normal. Untuk memperkuat hasil pengujian tersebut dipergunakan uji Kolmogorov-Smirnov yaitu sebagai berikut:

Tabel 4.9 One-Sample Kolmogorov-Smirnov Test

		Unstandardized Residual
N		120
Normal Parameters(a,b)	Mean	.0000000
	Std. Deviation	2096.86459425
Most Extreme Differences	Absolute	.103
	Positive	.103
	Negative	-.071
Kolmogorov-Smirnov Z		1.131
Asymp. Sig. (2-tailed)		.155

a Test distribution is Normal.

b Calculated from data.

karena asymp sig. pada output kolmogorov smirnov sebesar  $0,155 > 0,05$ , maka data tersistribusi normal.

#### 5. Uji Linieritas

Untuk mengetahui apakah model yang dihasilkan bersifat linear atau tidak dapat dideteksi dengan melihat uji ANOVA (overall F test) bila hasilnya signifikan ( $p < \alpha$ ) maka model berbentuk linear. Dari tabel ANOVA, nilai signifikansi model 4 sebesar 0,000 kurang dari  $\alpha = 0,05$ .

Tabel 4.10 Uji linieritas pada ANOVA

Model	Sum of Squares	df	Mean Square	F	Sig.
4 Regression	222223150601.801	4	55555787650.450	12210.668	.000(d)
Residual	523224094.066	115	4549774.731		
Total	222746374695.867	119			

d Predictors: (Constant), PDRB\_Jasa2, Pengangguran, Peng.Barang\_Tahan\_Lama, Peng.Buah

e Dependent Variable: Penduduk\_Miskin

### 4.3 Prediksi Jumlah Penduduk Miskin Menggunakan JST *Time Series* dan Regresi

Pada langkah prediksi pertama ini, penulis melakukan prediksi menggunakan data *time series* dari masing-masing variabel pembentuk persamaan regresi. Hal ini juga dilakukan terhadap data *time series* jumlah penduduk miskin untuk membandingkan keakurasian hasil dari pemilihan variabel yang

berhubungan dengan jumlah penduduk miskin. Dikarenakan data yang sedikit dan prediksi menggunakan data *time series* yang membutuhkan data terbaru, maka proses *training* mendapatkan arsitektur jaringan yang terbaik apabila menggunakan data sebanyak 80%, yaitu 96 data (data bulanan tahun 2002-2009). Dilakukan *trial and error* dalam penentuan *time window* yang digunakan karena pola data yang random pada masing-masing variabel. Begitu juga dengan penentuan jumlah *learning rate* dan jumlah unit *hidden layer* pada dua *hidden layer* tersebut.

Dalam penentuan jumlah data testing, digunakan 20% dari data yaitu sebanyak 24 data untuk melihat ketepatan bobot yang dihasilkan dengan melihat nilai MSE, MAE, dan MAPE untuk pengukuran keakurasian. Selain itu dilihat pula plot antara target dan hasil prediksi.

Dalam penentuan *time window* yang tepat untuk memperoleh hasil prediksi yang akurat, ditetapkan terlebih dahulu untuk konsistensi jumlah unit *hidden layer* pertama sebanyak 10, *hidden layer* kedua sebanyak 1, *error maximum* sebesar 0,00001, jumlah iterasi sebanyak 10000, dan *learning rate* sebesar 0,01. Hasil percobaan perubahan nilai *time window* hingga mendapatkan nilai MSE, MAE, dan MAPE terkecil dari data *testing* menggunakan bobot terbaik yang dihasilkan dari data *training* sebelumnya disajikan pada tabel 4.11.

Tabel 4.11 Perubahan nilai uji coba data pelatihan dengan perubahan nilai *time window*

Variabel	Nilai Error	Time Window				
		1	2	3	4	5
$u_1$	MSE	0,0762	0,0544	0,0510	0,0410	0,0786
	MAE	0,2608	0,2166	0,2100	0,1871	0,2698
	MAPE	0,4466	0,3413	0,3242	0,2757	0,4486
$u_2$	MSE	$7,8684 \times 10^{-4}$	$2,6741 \times 10^{-4}$	$1,9680 \times 10^{-4}$	$1,4765 \times 10^{-4}$	$1,5027 \times 10^{-4}$
	MAE	0,0210	0,0140	0,0125	0,0091	0,0107
	MAPE	0,0313	0,0224	0,0200	0,0152	0,0174
$u_3$	MSE	0,0196	0,0070	0,0082	0,0056	0,0156
	MAE	0,1013	0,0547	0,0516	0,0478	0,0875
	MAPE	0,2850	0,1716	0,1132	0,0905	0,1871

$u_4$	MSE	0,0037	0,0017	0,0017	0,0012	0,0039
	MAE	0,0536	0,0367	0,0367	0,0305	0,0532
	MAPE	0,1316	0,1140	0,1170	0,1077	0,1596
$U$	MSE	0,0421	0,0411	0,0759	0,0369	0,0486
	MAE	0,1799	0,1784	0,2582	0,1687	0,2006
	MAPE	0,2853	0,2795	0,4581	0,2554	0,3162

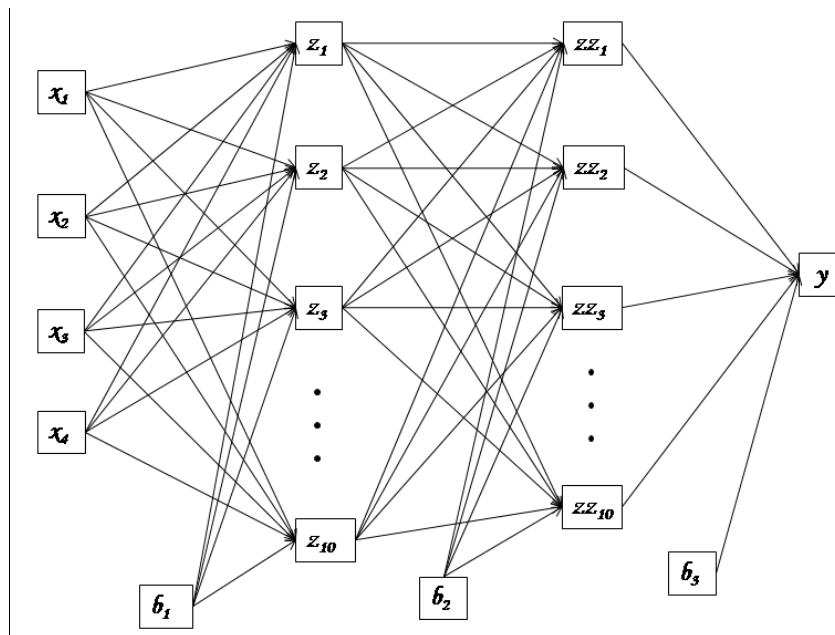
Berdasarkan tabel 4.11, besar *time window* yang paling akurat yaitu sebesar empat buah. Dengan menggunakan *time window* sebanyak empat, pergeseran unit input dan target pada kelima variabel yang ada pada tahapan pelatihan dan pengujian disajikan pada tabel 4.12.

Tabel 4.12 Pergeseran unit input dan target pada tahap pelatihan

Input				Target
$x_1$	$x_2$	$x_3$	$x_4$	$y$
$x_1'$	$x_2'$	$x_3'$	$x_4'$	$x_5'$
$x_2'$	$x_3'$	$x_4'$	$x_5'$	$x_6'$
$x_3'$	$x_4'$	$x_5'$	$x_6'$	$x_7'$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$x_{92}'$	$x_{93}'$	$x_{94}'$	$x_{95}'$	$x_{96}'$

Tabel 4.13 Pergeseran unit input dan target pada tahap pengujian

Input				Target
$x_1$	$x_2$	$x_3$	$x_4$	$y$
$x_{97}'$	$x_{98}'$	$x_{99}'$	$x_{100}'$	$x_{101}'$
$x_{98}'$	$x_{99}'$	$x_{100}'$	$x_{101}'$	$x_{102}'$
$x_{99}'$	$x_{100}'$	$x_{101}'$	$x_{102}'$	$x_{103}'$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$x_{116}'$	$x_{117}'$	$x_{118}'$	$x_{119}'$	$x_{120}'$



Gambar 4.2 Arsitektur JST backpropagation prediksi JST *time series*

Sedangkan dalam penentuan *learning rate*, digunakan parameter tambahan yang tetap yaitu *time window* sebanyak empat, dengan pergantian jumlah *learning rate* dari besar ke kecil yang diwakili oleh *learning rate* sebesar 0,5; 0,1; 0,05; 0,01. Hasil perubahan nilai uji coba data pelatihan tersebut disajikan pada tabel 4.14.

Tabel 4.14 Perubahan nilai uji coba data pelatihan dengan perubahan nilai *learning rate*

Variabel	Nilai Error	<i>Learning Rate</i>			
		0,5	0,1	0,05	0,01
$u_1$	MSE	0,0544	0,0482	0,0477	0,0375
	MAE	0,2166	0,2045	0,2036	0,1776
	MAPE	0,3413	0,3098	0,3079	0,2580
$u_2$	MSE	0,0013	$4,6418 \times 10^{-4}$	$3,1470 \times 10^{-4}$	$2,3232 \times 10^{-4}$
	MAE	0,0312	0,0194	0,0159	0,0133
	MAPE	0,0483	0,0308	0,0271	0,0212
$u_3$	MSE	0,0083	0,0076	0,0072	0,0048
	MAE	0,0574	0,0565	0,0541	0,0413
	MAPE	0,1126	0,1048	0,1050	0,0862
$u_4$	MSE	0,0206	0,0024	0,0012	$8,2601 \times 10^{-4}$

	MAE	0,1197	0,0421	0,0283	0,0231
	MAPE	0,2512	0,1389	0,0955	0,0850
$U$	MSE	0,0389	0,0375	0,0290	0,0256
	MAE	0,1735	0,1695	0,1452	0,1356
	MAPE	0,2650	0,2572	0,2116	0,1943

Berdasarkan tabel 4.14, besar *learning rate* yang menghasilkan nilai MSE, MAE, dan MAPE terkecil sama untuk setiap variabel, yaitu sebesar 0,01. Kemudian, dengan menggunakan *learning rate* sebesar 0,01, dilakukan uji coba data pelatihan tersebut dengan perubahan nilai *hidden layer*, baik pada *hidden layer* pertama maupun *hidden layer* kedua. Hasil uji coba tersebut disajikan pada tabel 4.15.

Tabel 4.15 Perubahan nilai uji coba data pelatihan dengan perubahan nilai *hidden layer*

Variabel	Nilai Error	<i>Hidden Layer</i>				
		10;1	10;10	10;5	15;5	15;15
$u_1$	MSE	0,0375	0,0025	0,0314	0,0453	0,0308
	MAE	0,1776	0,0430	0,1614	0,1974	0,1603
	MAPE	0,2580	0,0498	0,2286	0,2958	0,2265
$u_2$	MSE	$2,3232 \times 10^{-4}$	$2,4196 \times 10^{-5}$	$1,6871 \times 10^{-4}$	$1,7271 \times 10^{-4}$	0,0029
	MAE	0,0133	0,0041	0,0120	0,0124	0,0419
	MAPE	0,0212	0,0068	0,0191	0,0196	0,0633
$u_3$	MSE	0,0048	$3,1419 \times 10^{-4}$	0,0100	0,0106	0,0118
	MAE	0,0413	0,0158	0,0630	0,0785	0,0653
	MAPE	0,0862	0,0428	0,1357	0,2107	0,1788
$u_4$	MSE	$8,2601 \times 10^{-4}$	$6,0686 \times 10^{-4}$	0,0034	$8,6593 \times 10^{-4}$	0,0012
	MAE	0,0231	0,0170	0,0524	0,0238	0,0268
	MAPE	0,0850	0,0789	0,1200	0,0945	0,0948
$U$	MSE	0,0256	0,0021	0,0220	0,0277	0,0340
	MAE	0,1356	0,0426	0,1231	0,1398	0,1512
	MAPE	0,1943	0,0504	0,1729	0,2025	0,2261

Dari sekian banyak tahapan perubahan nilai uji coba data pelatihan, yaitu dengan perubahan nilai *time window*, *learning rate*, dan *hidden layer*, didapatkan arsitektur yang digunakan dalam prediksi sebagai berikut.



1. Jumlah Neuron :
  - a. *Input layer* : 4
  - b. *Hidden layer I* : 10
  - c. *Hidden layer II* : 10
  - d. *Output layer* : 1
2. Fungsi Aktivasi :
  - a. *Hidden layer* : Logsig
  - b. *Output layer* : Purelin
3. Parameter :
  - a. Maks. *Epoch* : 10000
  - b. *Error (Goal)* : 0.00001

Arsitektur yang telah terbentuk disimulasikan menggunakan Matlab R2010a pada masing-masing variabel. Dari tahapan pembaharuan bobot pada data pelatihan, bobot terbaik yang dihasilkan dan digunakan pada saat pengujian memiliki hasil MSE, MAE, dan MAPE yang beragam dan pola yang berbeda-beda pula.

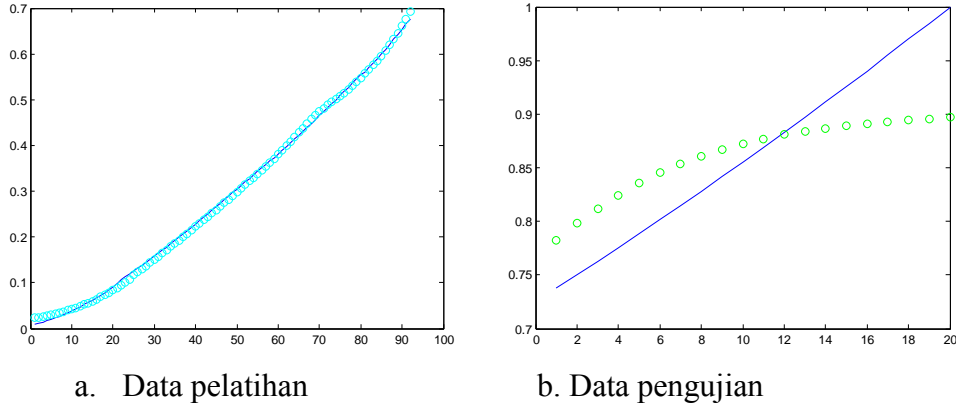
Prediksi yang dilakukan merupakan prediksi jangka panjang untuk mendapatkan data bulanan tahun 2012 dari lima variabel yang ada. Pergeseran input dalam prediksi adalah sebagai berikut

Tabel 4.16 Pergeseran unit input dan target pada tahap prediksi

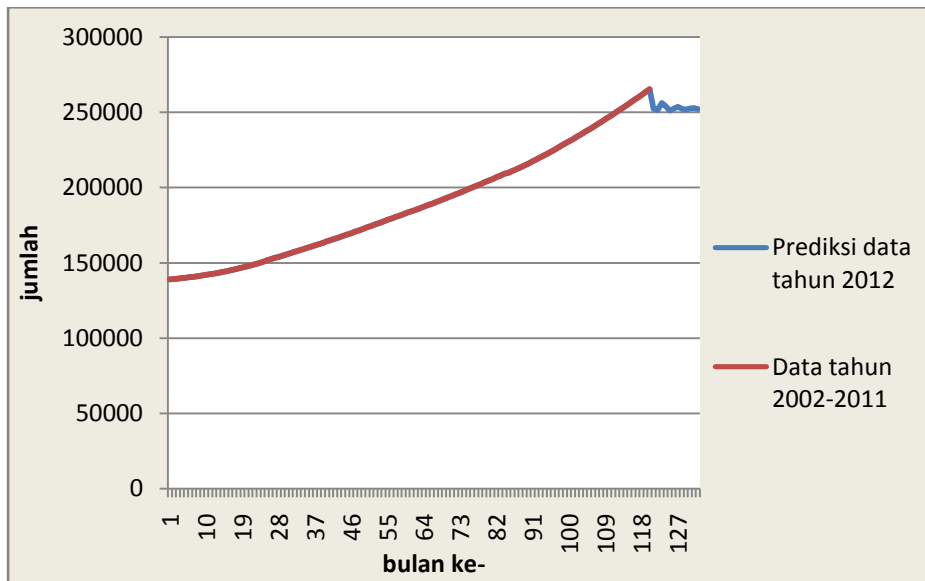
Input				Target
$x_1$	$x_2$	$x_3$	$x_4$	$y$
$x_{117}'$	$x_{118}'$	$x_{119}'$	$x_{120}'$	$\widehat{x}_{121}'$
$x_{118}'$	$x_{119}'$	$x_{120}'$	$\widehat{x}_{121}'$	$\widehat{x}_{122}'$
$x_{119}'$	$x_{120}'$	$\widehat{x}_{121}'$	$\widehat{x}_{122}'$	$\widehat{x}_{123}'$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\widehat{x}_{128}'$	$\widehat{x}_{129}'$	$\widehat{x}_{130}'$	$\widehat{x}_{131}'$	$\widehat{x}_{132}'$

Pada variabel PDRB sektor jasa, apabila dilihat dari gambar 4.3, bobot yang digunakan pada tahap pengujian membuat nilai output kurang mendekati nilai target atau data aktual yang ada. Nilai output pengujian yang pada awalnya naik, mulai mengalami pembelokan arah plot berupa penurunan pada data ke-8.

Nilai error yang terjadi cukup besar, yaitu MSE sebesar 0,0025, MAE sebesar 0,0430, dan MAPE sebesar 0,0498.

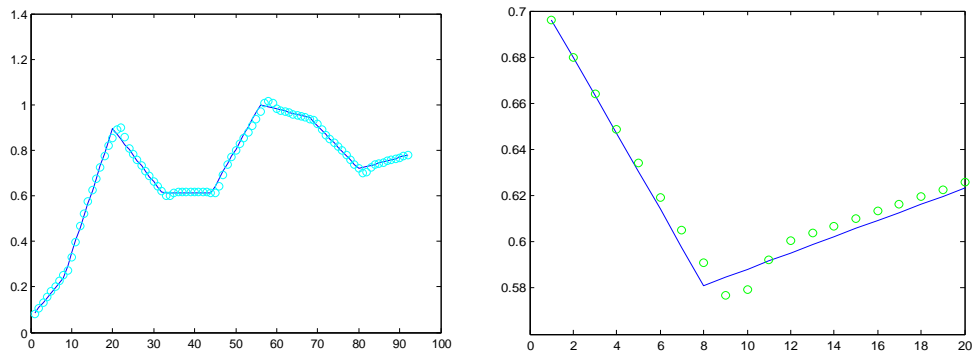


Gambar 4.3 Perbandingan hasil target (o) dan output (-) pada variabel PDRB sektor jasa



Gambar 4.4 Grafik data aktual dan hasil prediksi pada variabel PDRB sektor jasa

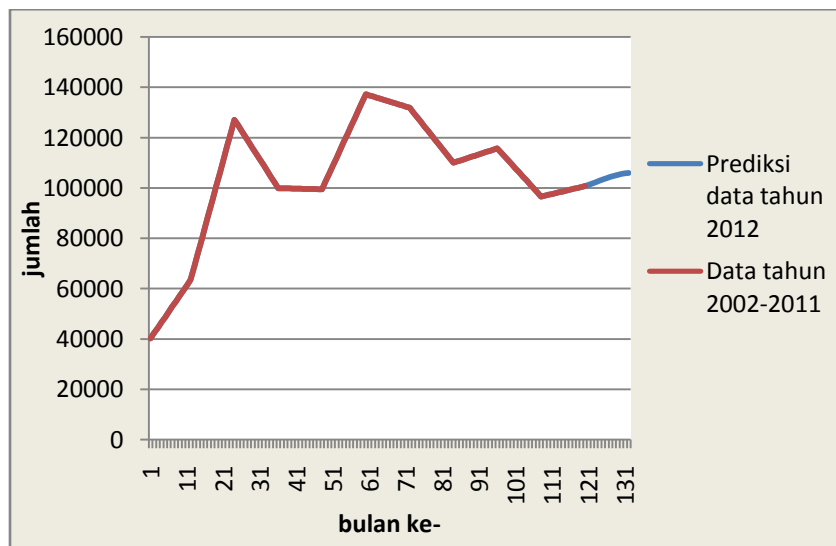
Pada variabel pengangguran, apabila dilihat dari gambar 4.5, bobot yang digunakan pada tahap pengujian membuat nilai output cukup mendekati nilai target atau data aktual yang ada. Hanya saja terjadi penurunan yang cukup signifikan pada data ke-9, namun mengalami kenaikan lagi mengikuti pola yang ada pada data ke-11. Nilai error yang terjadi sangat kecil, yaitu MSE sebesar  $2,4196 \times 10^{-5}$ , MAE sebesar 0,0041, dan MAPE sebesar 0,0068.



a. Data pelatihan

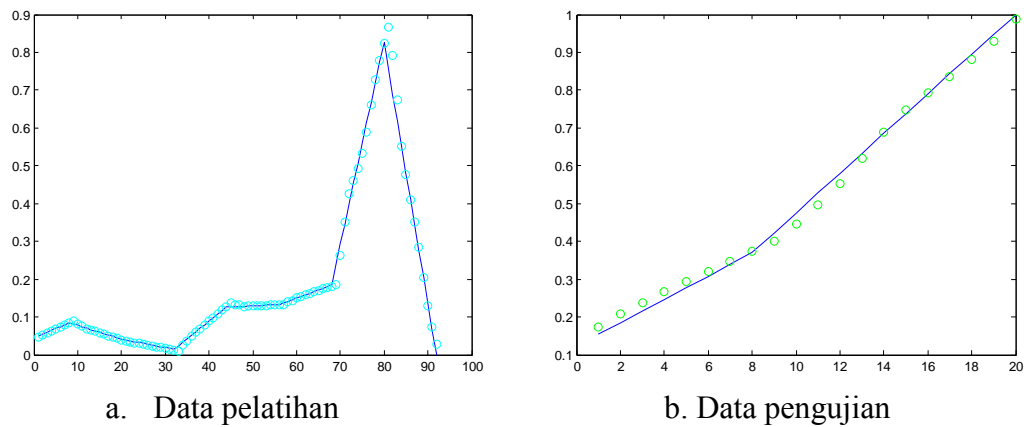
b. Data pengujian

Gambar 4.5 Perbandingan hasil target (o) dan output (-) pada variabel pengangguran

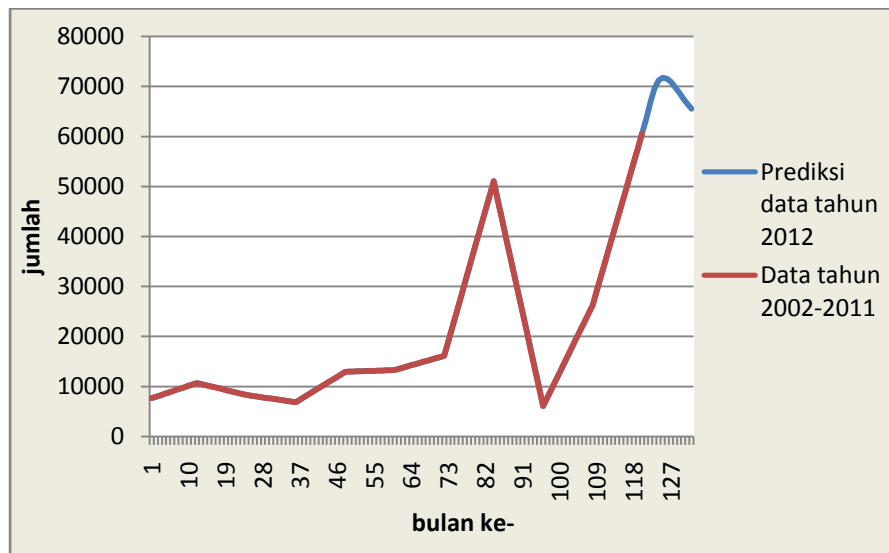


Gambar 4.6 Grafik data aktual dan hasil prediksi pada variabel pengangguran

Pada variabel pengeluaran barang tahan lama, apabila dilihat dari gambar 4.7, bobot yang digunakan pada tahap pengujian membuat nilai output cukup mendekati nilai target atau data aktual yang ada. Nilai output sudah cukup mendekati nilai target dilihat dari bulatan (o) yang hampir tepat berada di atas garis (-) pada setiap titik datanya. Nilai error yang terjadi cukup kecil, yaitu MSE sebesar  $3,1491 \times 10^{-4}$ , MAE sebesar 0,0158, dan MAPE sebesar 0,0428.

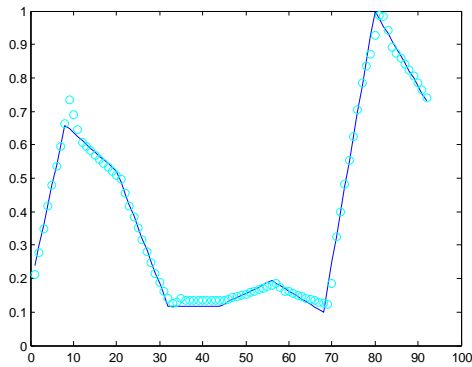


Gambar 4.7 Perbandingan hasil target (o) dan output (-) pada variabel pengeluaran barang tahan lama

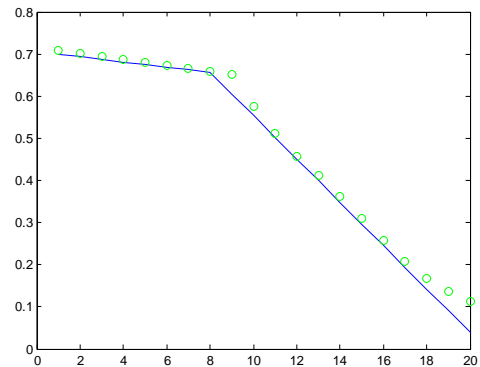


Gambar 4.8 Grafik data aktual dan hasil prediksi pada variabel pengeluaran barang tahan lama

Pada variabel pengeluaran buah, apabila dilihat dari gambar 4.9, bobot yang digunakan pada tahap pengujian membuat nilai output cukup mendekati nilai target atau data aktual yang ada. Hanya terjadi pencilan pada data ke-9. Nilai error yang terjadi cukup besar, yaitu MSE sebesar  $6,0686 \times 10^{-4}$ , MAE sebesar 0,0170, dan MAPE sebesar 0,0789.

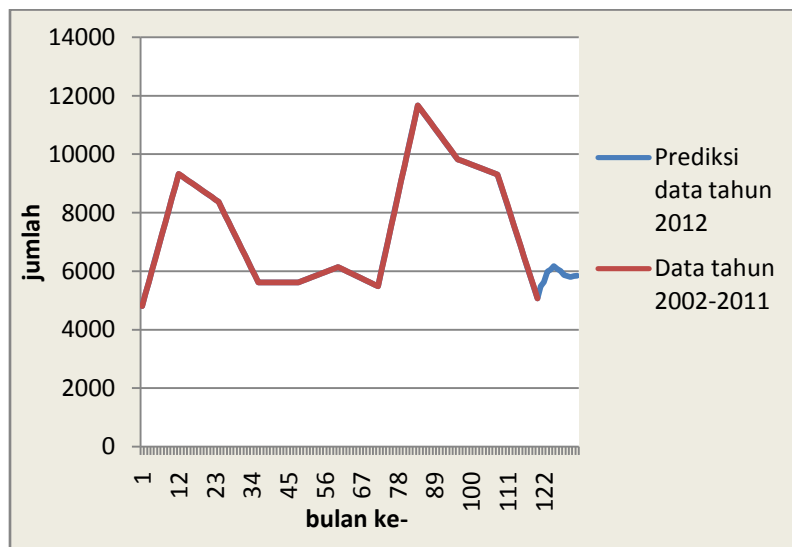


a. Data pelatihan



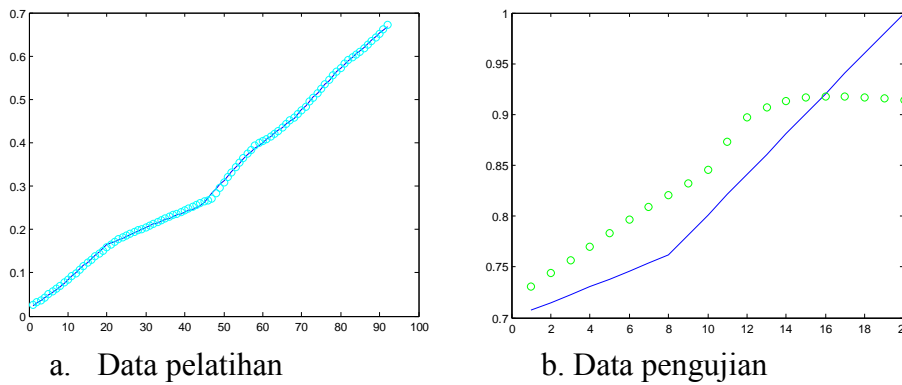
b. Data pengujian

Gambar 4.9 Perbandingan hasil target (o) dan output (-) pada variabel pengeluaran buah

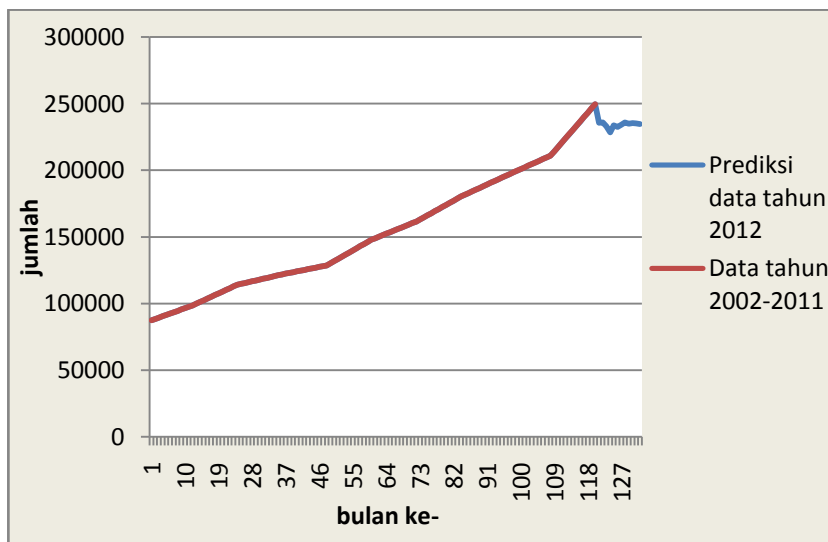


Gambar 4.10 Grafik data aktual dan hasil prediksi pada variabel pengeluaran buah

Pada variabel jumlah penduduk miskin, apabila dilihat dari gambar 4.11, bobot yang digunakan pada tahap pengujian membuat nilai output kurang mendekati nilai target atau data aktual yang ada. Nilai output pengujian terlalu naik dibandingkan dengan nilai target yang dimulai pada data ke-3, hingga terjadi pembelokan pada data ke-11 dan 13. Nilai error yang terjadi cukup besar, yaitu MSE sebesar 0,0021, MAE sebesar 0,0426, dan MAPE sebesar 0,0504.



Gambar 4.11 Perbandingan hasil target (o) dan output (-) pada variabel jumlah penduduk miskin



Gambar 4.12 Grafik data aktual dan hasil prediksi pada variabel jumlah penduduk miskin

Sehingga didapatkan hasil prediksi jumlah penduduk miskin dengan memasukkan hasil prediksi empat variabel bebas ke dalam persamaan regresi (4.1) adalah sebagai berikut

Tabel 4.17 Data hasil prediksi dengan penggunaan variabel hasil PCA dan regresi tahun 2012

Bulan	$u_1$	$u_2$	$u_3$	$u_4$	$U$
1	252374	101306	63455	5474	229322
2	251552	101807	66748	5623	229020
3	256046	102380	69568	5988	234632
4	253955	102944	71206	6054	232645

5	251030	103488	71741	6169	229588
6	252357	104012	71577	6070	231037
7	253539	104489	70899	6001	232257
8	252285	104919	69911	5867	230701
9	251676	105293	68774	5827	229872
10	252553	105607	67598	5796	230681
11	252668	105859	66494	5825	230677
12	252082	106048	65535	5840	229908

Kemudian setelah melakukan prediksi dari variabel jumlah penduduk miskin itu sendiri menggunakan JST *time series* dengan data tahun 2002-2011, didapatkan hasil pada tabel 4.18 sebagai berikut

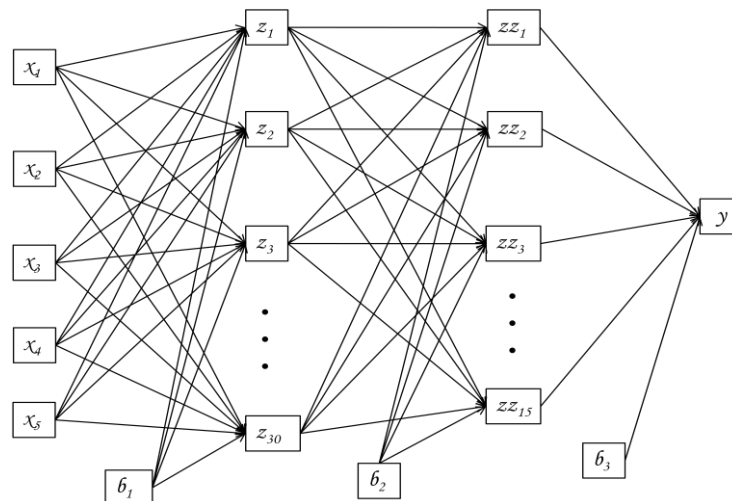
Tabel 4.18 Perbandingan hasil prediksi dari JST *time series* jumlah penduduk miskin tahun 2002 – 2011 dan data sebenarnya tahun 2012

Bulan	Data Prediksi Hasil Regresi ( $U$ )	Data Prediksi hasil JST <i>time series</i> ( $\hat{d}$ )	Data Sebenarnya ( $d$ )	Error <sub>2</sub> = $ U - d $	Error <sub>1</sub> = $ \hat{d} - d $
1	229322	235456	250568	21246	15112
2	229020	235754	251649	22629	15895
3	234632	233050	252730	18098	19680
4	232645	228454	253811	21166	25357
5	229588	233721	254892	25304	21171
6	231037	232466	255973	24936	23507
7	232257	234224	257054	24797	22830
8	230701	235847	258135	27434	22288
9	229872	234848	259216	29344	24368
10	230681	235327	260297	29616	24970
11	230677	235019	261378	30701	26359
12	229908	234525	262459	32551	27934

Dari tabel 4.18, prediksi dengan penginputan hasil prediksi *time series* variabel bebas ke persamaan regresi (4.1) untuk mendapatkan hasil prediksi jumlah penduduk miskin memiliki nilai MAPE yang lebih besar yaitu 9,98 % sehingga lebih kecil keakurasiannya dibandingkan prediksi dengan JST *time series* dari jumlah penduduk miskin yaitu sebesar 8,74%.

#### 4.4 Prediksi Jumlah Penduduk Miskin Menggunakan JST Multivariat

Pada langkah prediksi kedua ini, penulis melakukan prediksi menggunakan algoritma JST multivariat dari masing-masing variabel terpilih. Variabel input yang digunakan sebanyak lima unit yang terdiri dari variabel PDRB sektor jasa – jasa atas dasar harga konstan, jumlah pengangguran, pengeluaran rata-rata perkapita sebulan untuk barang tahan lama, pengeluaran rata-rata perkapita sebulan untuk kelompok buah, dan jumlah penduduk miskin pada waktu ke- $t$ . sedangkan untuk variabel output digunakan data jumlah penduduk miskin pada waktu ke- $t+1$ . Adapun arsitektur jaringannya adalah sebagai berikut



Gambar 4.13 Arsitektur JST Multivariat

Pada proses *training*, data yang digunakan sebanyak 80%, yaitu 576 data atau 96 data dari setiap variabel input dan output (data bulanan tahun 2002-2009). Sedangkan dalam penentuan jumlah data testing, digunakan 20% dari data yaitu sebanyak 138 data atau 23 data dari setiap variabel input dan output (data bulanan tahun 2010-2011). Dilakukan *trial and error* dalam penentuan jumlah *learning rate* dan jumlah unit *hidden layer* pada kedua *hidden layer* tersebut agar mendapatkan arsitektur yang akurat dengan melihat nilai MSE, MAE, dan MAPE terkecil yang menunjukkan ketepatan bobot terpilih pada saat *training*. Selain itu dapat dilihat pula plot antara nilai target dan output.



Dalam penentuan *time window* yang tepat untuk memperoleh hasil prediksi yang akurat, ditetapkan terlebih dahulu untuk konsistensi jumlah unit *hidden layer* pertama sebanyak 10, *hidden layer* kedua sebanyak 1, *error maximum* sebesar 0,00001, dan jumlah iterasi sebanyak 10000. Hasil percobaan perubahan nilai *learning rate* disajikan pada tabel 4.19.

Tabel 4.19 Perubahan nilai uji coba data pelatihan dengan perubahan nilai *learning rate*

Learning Rate	MSE	MAE	MAPE
0,01	0,0159	0,1063	0,1495
0,02	0,0228	0,1378	0,2016
0,03	0,0238	0,1416	0,2083
0,04	0,0217	0,1307	0,1902
0,05	0,0359	0,1675	0,2626

Berdasarkan tabel 4.19, besar *learning rate* yang menghasilkan nilai MSE, MAE, dan MAPE terkecil sama untuk setiap variabel, yaitu sebesar 0,01. Kemudian, dengan menggunakan *learning rate* sebesar 0,01 dengan parameter yang digunakan sama dengan sebelumnya, dilakukan uji coba data pelatihan tersebut dengan perubahan nilai *hidden layer*, baik pada *hidden layer* pertama maupun *hidden layer* kedua. Hasil uji coba tersebut disajikan pada tabel 4.20.

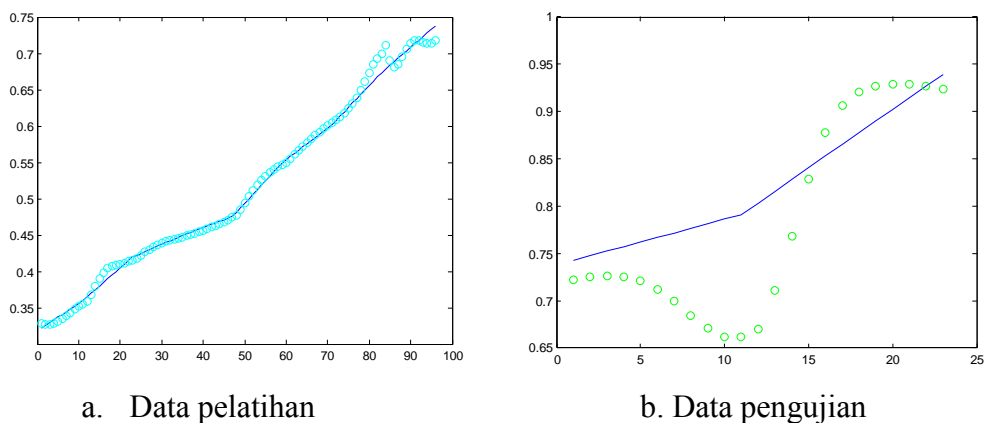
Tabel 4.20 Perubahan nilai uji coba data pelatihan dengan perubahan nilai *hidden layer*

<i>Hidden Layer I</i>	<i>Hidden Layer II</i>	MSE	MAE	MAPE
10	1	0,0159	0,1063	0,1495
10	10	0,0099	0,0656	0,0888
20	10	0,0189	0,1127	0,1620
30	10	0,0157	0,1080	0,1540
30	15	0,0046	0,0539	0,0748

Dari sekian banyak tahapan perubahan nilai uji coba data pelatihan, yaitu dengan perubahan nilai *learning rate*, dan *hidden layer*, didapatkan arsitektur yang digunakan dalam prediksi sebagai berikut.

1. Jumlah Neuron :
  - a. *Input layer* : 5
  - b. *Hidden layer I* : 30
  - c. *Hidden layer II* : 15
  - d. *Output layer* : 1
2. Fungsi Aktivasi :
  - a. *Hidden layer* : Logsig
  - b. *Output layer* : Purelin
3. Parameter :
  - a. Maks. *Epoch* : 10000
  - b. *Error (Goal)* : 0.00001

Gambar 4.14 menunjukkan bahwa bobot yang dihasilkan dari tahapan pelatihan sudah cukup mendekati data aktual, walaupun pada beberapa data akhir terjadi pencilan dan pembelokan plot data. Namun, pada tahapan pengujian data output masih banyak yang cukup melenceng dari data yang sebenarnya. Nilai error yang terjadi cukup besar, yaitu MSE sebesar 0,0046, MAE sebesar 0,0539, dan MAPE sebesar 0,0748.



Gambar 4.14 Perbandingan hasil target (o) dan output (-) pada JST multivariat

Hasil estimasi jumlah penduduk miskin pada bulan ke-1 tahun 2012 adalah sebesar 244.466 orang. Sedangkan pada data sebenarnya berjumlah 250.568 orang. Sehingga nilai MAPE estimasi tersebut sebesar 2,43 %. Untuk prediksi jumlah penduduk miskin pada bulan ke-2 sampai dengan bulan ke-12 tahun 2012 tidak bisa dilakukan karena membutuhkan data aktual bulan ke-1 sampai dengan bulan ke-11 tahun 2011 pada setiap variabel input.



## **BAB 5**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan hasil dan pembahasan pada Bab 4, dapat ditarik kesimpulan sebagai berikut:

1. Dalam penentuan variabel terbaik yang dapat menggambarkan variabel tak bebas dapat dicari dengan kombinasi antara metode PCA dan regresi berganda.
2. Variabel terbaik yang dapat berhubungan erat dengan jumlah penduduk miskin di Kalimantan Selatan adalah PDRB sektor jasa – jasa atas dasar harga konstan, jumlah pengangguran, pengeluaran rata-rata perkapita sebulan untuk barang tahan lama, dan pengeluaran rata-rata perkapita sebulan untuk kelompok buah.
3. Koefisien regresi yang positif pada setiap variabel bebas menunjukkan bahwa jumlah penduduk miskin akan semakin tinggi jika PDRB sektor jasa – jasa atas dasar harga konstan, jumlah pengangguran, pengeluaran rata-rata perkapita sebulan untuk barang tahan lama, dan pengeluaran rata-rata perkapita sebulan untuk kelompok buah semakin tinggi.
4. Prediksi dengan penginputan hasil prediksi *time series* variabel bebas ke persamaan regresi untuk mendapatkan hasil prediksi jumlah penduduk miskin setiap bulannya pada tahun 2012 memiliki nilai MAPE yang lebih besar yaitu 9,98 % sehingga lebih kecil keakurasiannya dibandingkan prediksi dengan JST *time series* dari jumlah penduduk miskin yaitu sebesar 8,74%.
5. Hasil estimasi jumlah penduduk miskin pada bulan ke-1 tahun 2012 adalah sebesar 244.466 orang. Sedangkan pada data sebenarnya berjumlah 250.568 orang. Sehingga nilai MAPE estimasi tersebut sebesar 2,43 %.

#### **5.2 Saran**

Berdasarkan hasil penelitian diharapkan untuk penelitian selanjutnya dapat dipilih data sekunder yang lebih lengkap agar tidak perlu ada interpolasi

data guna keakuratan hasil peramalan. Selain itu, dapat pula dicari metode lain yang lebih baik dalam pemilihan variabel dan model regresi terbaik yang lebih tepat dalam menggambarkan data sesuai dengan kasus yang diambil untuk digunakan dalam peramalan. Algoritma JST yang lain dapat pula dicoba untuk melakukan prediksi.

## DAFTAR PUSTAKA

- Al-Foul, B. M. A. (2012), "Forecasting Energy Demand in Jordan Using Artificial Neural Networks", *Topics in Middle Eastern and African Economies* Vol. 14, hal. 473-478, Sharjah.
- Arsyad, L. (1997), *Ekonomi Pembangunan*, Edisi Ketiga, BP STIE YKPN, Yogyakarta.
- BPS (2012), Berita Resmi Statistik, *Profil Kemiskinan Di Indonesia September 2011*, No. 06/01/Th. XV, 2 Januari 2012, Jakarta.
- Coudouel, A, dkk (2002), *Poverty Measurement Analysis*, PRSP Sourcebook, World Bank, Washington D.C.
- Direktorat Diseminasi Statistik BPS (2002), *Data Strategis BPS*, CV. Nasional Indah, Jakarta.
- Faradisa, Rosiyah (2009), *Algoritma Hybrid Particle Swarm Optimization with Constriction dan Backpropagation (PSOC-BP) untuk Training Feedforward Neural Network (FNN) Pada Permasalahan Forecasting*. Tesis-Magister Jurusan Matematika, ITS, Surabaya.
- Fausett, L. (1994), *Fundamentals Neural Networks, Architectures, Algorithms, and Applications*, Prentice-Hall, USA.
- Gujarati, Damodar N. 2004. *Basic Econometrics*. 4 ed. New York: The McGraw-Hill Companies.
- Juhos, I., dkk (2008), "Forecasting of Traffic Origin NO and NO<sub>2</sub> Concentrations by Support Vector Machines and Neural Networks Using Principal Component Analysis", *Simulation Modelling Practice and Theory* 16, hal. 1488-1502, Hungary.
- Kheirkhah, A., dkk (2012), "Improved Estimation of Electricity Demand Function by Using of Artificial Neural Network, Principal Component Analysis and Data Envelopment Analysis", *Computers and Industrial Engineering* 64, hal. 425-441, Iran.
- Kusumadewi, S. (2004), *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB & Excel Link*, Graha Ilmu, Yogyakarta.
- Maitra, S. dan Yan, J. (2008), "Principle Component Analysis and Partial Least Squares: Two Dimension Reduction Techniques for Regression", Casualty Actuarial Society, Discussion Paper Program.

- Noori, R., dkk (2011), “Assessment of Input Variables Determination on The SVM Model Performance Using PCA, Gamma test, and Forward Selection Techniques for Monthly Stream Flow Prediction”, *Journal of Hydrology* 401, hal. 177–189, Iran.
- Pino, R., dkk (2008), “Forecasting Next-Day Price of Electricity in The Spanish Energy Market Using Artificial Neural Networks”, *Engineering Applications of Artificial Intelligence* 21, hal. 53–62, Spain.
- Sheng-qiang, Y., dkk (2009), “Establishment of Grey-Neural Network Forecasting Model of Coal and Gas Outburst”, *Procedia Earth and Planetary Science* 1, hal. 148-153, China.
- Ulfah, F.M. (2011), *Penerapan Jaringan Syaraf Tiruan RBF dan Metode PCA dalam Memprediksi Risiko Pemberian Kredit*, Tesis-Magister Jurusan Matematika, ITS, Surabaya.
- Utomo, D. B. (2007), *Wavalet-Jaringan Syaraf Tiruan Untuk Prediksi Data Time Series*, Tesis-Magister Jurusan Matematika, ITS, Surabaya.
- Wibisono, D., (2002), *Riset Bisnis (Panduan bagi praktisi dan akedemisi)*, PT Gramedia Pustaka Utama, Jakarta.



## **BIODATA PENULIS**



Penulis dilahirkan di Banjarmasin, 13 Maret 1990, merupakan anak pertama dari dua bersaudara pasangan Ir. H. Rachimi Dharma, Dipl. HE (alm) dan Hj. Mariani. Penulis telah menempuh pendidikan formal yaitu di SDN Sei Miai 7 Banjarmasin (1996-2002), SMP Negeri 2 Banjarmasin (2002-2005), SMA Negeri 1 Banjarmasin (2005-2007), dan melanjutkan pendidikan perguruan tinggi pada jurusan Matematika FMIPA-Universitas Lambung Mangkurat Banjarmasin pada tahun 2007 dan meraih gelar Sarjana Sains (S.Si) pada tahun 2011.

Penulis melanjutkan pendidikan jenjang S2 pada jurusan Matematika FMIPA ITS Surabaya pada semester genap tahun 2011. Pada awal tahun 2014, penulis berhasil menyelesaikan studinya dan meraih gelar Magister Sains (M.Si).

LAMPIRAN 1. *Rotated Component Matrix*

Variabel	Component										
	1	2	3	4	5	6	7	8	9	10	11
PDRB_Jasa2	.996	.024	.002	-.010	-.015	-.025	.049	.028	.013	-.021	-.009
PDRB_Perdagangan_DII2	.996	.037	.017	-.008	-.017	-.031	.026	.043	.011	-.017	-.001
PDRB_Tanpa_Migas1	.996	.027	.058	-.014	-.019	-.031	.004	.031	.013	-.017	-.003
PDRB_Listrik_DII2	.996	-.008	.050	-.023	-.018	-.028	.051	.016	.022	-.020	-.018
PDRB_Perdagangan_DII1	.994	.003	.073	-.016	-.022	-.036	-.017	.048	.011	-.013	1.61E-005
PDRB_Pengangkutan_Komunikasi1	.994	.078	.002	.001	-.012	-.032	.026	.036	.005	-.015	.009
PDRB_Pertanian1	.993	.083	.015	-.003	-.010	-.033	.039	.030	.016	-.020	-.007
PDRB_Tanpa_Migas2	.993	.058	-.015	-.004	-.011	-.024	.077	.029	.016	-.021	-.006
PDRB_Jasa1	.993	.062	.072	-.013	-.017	-.030	-.005	.033	.020	-.025	-.014
PDRB_Pertambangan_Penggalan2	.992	.054	-.025	-.004	-.018	-.016	.070	.023	.006	-.015	.013
PDRB_KeuanganDII1	.992	.071	.046	-.010	-.010	-.034	.050	.026	.027	-.023	-.018
PDRB_Listrik_DII1	.992	.035	.100	-.019	-.018	-.040	-.010	.019	.019	-.016	-.011
PDRB_Bangunan2	.991	.032	-.047	-.005	-.012	-.016	.100	.025	.015	-.024	-.009
PDRB_Pengangkutan_Komunikasi2	.991	.053	-.039	-.002	-.009	-.018	.100	.022	.015	-.022	-.005
PDRB_Pertambangan_Penggalan1	.990	-.036	.108	-.027	-.028	-.029	-.032	.017	.005	-.007	.007
PDRB_Bangunan1	.989	.045	-.048	-.003	-.014	-.007	.076	.023	.009	-.025	-.002
Penduduk	.989	.008	-.002	-.017	-.014	-.032	.133	.011	.022	-.014	-.008
PDRB_Industri1	.988	-.025	.119	-.025	-.024	-.031	-.018	.054	.021	-.016	-.009
Peng.Pendidikan	.985	.102	.064	-.022	-.014	-.058	-.053	.062	.026	-.031	-.041
Peng.Rumah_DII	.983	.129	-.004	-.011	-.017	-.024	.069	.071	.016	-.033	-.020
PDRB_Pertanian2	.981	.122	-.061	.009	.000	-.023	.106	.030	.021	-.032	-.018
Peng.Telur_Susu	.978	.157	.066	-.017	-.014	-.032	.045	.002	.013	-.032	-.027
Peng.Sayur	.978	.161	.055	-.005	-.016	-.050	-.017	.094	.017	-.023	-.008
Peng.Ikan	.971	.110	.042	-.011	.018	-.052	.013	-.168	.038	-.033	-.067
Ekspor_Tambang	.967	-.054	.130	-.021	-.039	-.043	-.099	.161	.023	-.023	-.014
Peng.Konsumsi_Lain	.965	.054	.028	-.052	-.030	-.033	2.34E-005	.104	.035	-.057	-.082
Peng.Pajak	.957	-.042	.145	-.027	-.056	-.001	.079	.102	-.028	.004	.054
PDRB_KeuanganDII2	.954	.039	-.031	-.018	.002	-.023	.242	-.011	.056	-.024	-.040

Peng.Minuman	.934	.246	-.024	-.003	.008	-.015	.130	-.156	.013	-.037	-.038
Peng.Minyak_Lemak	.931	-.066	.046	-.015	-.042	-.041	-.024	.337	.038	-.022	.000
Peng.Kesehatan	.918	.279	-.008	.015	.008	-.077	-.147	.157	.045	-.048	-.054
Industri_Manufaktur	-.917	-.103	-.281	.051	.015	.054	.011	-.054	-.101	.064	.124
PDRB_Industri2	.904	.023	.286	-.025	-.017	-.081	-.160	.065	.021	.002	-.004
Peng.Daging	.886	.146	.219	-.075	-.013	-.122	-.241	-.007	.086	-.045	-.134
Peng.Barang_Jasa	.884	.198	.122	-.060	-.061	-.056	-.138	.317	.014	-.046	-.038
Peng.Pakaian	.871	.363	.028	-.007	.001	-.099	-.231	.135	.054	-.046	-.070
Ekspor_Karet_Alam	.838	-.080	.067	-.018	-.009	.033	.150	-.298	-.096	.083	.177
Peng.Padi	.804	.539	-.153	.045	.041	-.077	.006	.030	.021	-.050	-.040
Peng.Tembakau_Sirih	.748	.504	-.029	.050	-.012	-.029	.190	.360	.004	-.034	.053
Ekspor_Kayu	-.727	-.267	-.262	.036	.069	.082	.248	-.308	-.138	.105	.157
Peng.Barang_Tahan_Lama	.713	.053	.037	.015	-.088	-.004	.090	.637	-.039	.011	.144
Peng.Pesta_Upacara	.643	-.243	-.041	-.076	.016	-.102	-.557	-.124	.130	-.060	-.227
Ekspor_Rotan	.637	-.340	.577	-.122	-.029	-.048	-.169	-.188	.032	.011	-.066
Peng.Minuman_Alkohol	.571	-.565	.133	-.060	-.081	-.042	-.298	.464	.039	.006	-.003
Peng.Buah	.291	.854	.164	-.031	-.007	-.153	-.108	.049	.040	.002	-.006
Peng.Makanan_Jadi	.624	.750	-.008	.051	.024	-.017	.142	-.058	-.015	-.039	.017
Ekspor_Ikan	-.355	-.145	-.813	.069	.061	-.074	.036	-.071	-.123	.021	-.033
LuasPanen_UbiJalar	.039	-.096	-.108	.681	-.343	.023	.050	.001	.044	-.274	.058
LuasPanen_UbiKayu	.012	.098	-.093	.656	.139	.085	-.031	-.017	-.127	.277	.267
LuasPanen_KacangHijau	-.122	.012	-.022	.626	.369	.108	.027	-.042	.273	.105	-.228
LuasPanen_Padi	.070	-.019	.016	.599	-.010	-.053	.038	.060	-.066	.465	-.377
LuasPanen_Kedelai	-.153	.072	.377	.510	.317	-.079	-.160	.011	-.170	-.150	.108
LuasPanen_KacangTana h	-.148	-.031	-.024	.025	.843	.089	.057	-.026	-.007	-.111	.064
LuasPanen_Jagung	.089	.037	-.033	.075	.836	-.047	-.008	-.014	-.142	.023	.028
Inflasi_Makanan_Jadi	-.008	-.013	-.256	-.078	.134	.782	-.072	.024	.041	-.109	.023
Inflasi_Transportasi_Komunikasi	-.108	-.063	.107	.133	.077	.767	-.026	.025	.152	-.129	-.102
Inflasi_Perumahan	-.166	-.094	.221	.021	-.193	.661	.122	-.068	.026	.168	.012
Pengangguran	.305	-.050	-.419	-.014	.057	-.044	.704	-.135	.135	-.012	-.081
Peng.Umbi	.557	.047	.237	-.070	.025	-.029	.612	.151	.137	-.120	-.214

Inflasi_Sandang	.087	-.008	.163	-.101	-.050	.058	.116	.103	.824	.097	.065
Inflasi_Bahan_Makanan	.043	.061	-.155	.154	-.225	.341	-.059	-.192	.628	-.142	.077
Inflasi_Pendidikan_DII	-.177	-.034	-.031	.073	-.081	-.097	-.032	.015	.041	.860	.013
Inflasi_Kesehatan	-.059	-.005	.028	.030	.068	-.061	-.046	.033	.091	-.012	.839

Extraction Method: Principal Component Analysis.

Rotation Method: Varimax with Kaiser Normalization.

a. Rotation converged in 10 iterations.



## LAMPIRAN 2. Listing Program Time Series JST Backpropagation Menggunakan MATLAB R20110a

```
function Backpro_80()
clear all;clc;close all;
nInput=4;
nHidden1=10;
nHidden2=10;
% nOutput=1;
nPrediksi=12;
error=0.00001;
epoch=10000;
lr=0.01;

[FileName,PathName]=uigetfile('*.xls','Buka Data Dari Excel');
ExcelFile=fullfile(PathName,FileName);
Data=xlsread(ExcelFile,-1);
[mData,nData]=size(Data);

% menyusun data dalam satu baris
data1=[];
for i=1:mData
    data1=[data1 Data(i,:)];
end

% normalisasi data
for i=1:length(data1)
    data2(i)=(data1(i)-min(data1))/(max(data1)-min(data1));
end

% data training: input dan target
data80=(mData*nData)*80/100;
data3=data2(1:data80);
idx=1;
idt=nInput+1;
while idt<=length(data3)
    x(idx,1:nInput)=data3((idx-1)+1:(idx-1)+nInput);
    t(idx)=data3(idt);
    idx=idx+1;
    idt=idt+1;
end

% algoritma backpropagation
net=newff(x',t,[nHidden1 nHidden2],{'logsig' 'logsig'},'trainrp');
%Set max epoch, goal, learning rate, show step
net.trainParam.epochs=epoch;
net.trainParam.goal=error;
net.trainParam.lr=lr;
net.trainParam.ShowWindow=200;
net.trainParam.ShowCommandLine=false;

% melakukan training
net=train(net,x',t);

% simulasi training
idx=[1:length(x')];
```

```

y = sim(net,x');
figure (1);
plot (idx,t,idx,y,'co')

% data testing: input dan target
data4=data2(data80+1:mData*nData);
idxtest=1;
idttest=nInput+1;
while idttest<=length(data4)
    xtest(idxtest,1:nInput)=data4((idxtest-1)+1:(idxtest-
1)+nInput);
    ttest(idxtest)=data4(idttest);
    idxtest=idxtest+1;
    idttest=idttest+1;
end

% simulasi testing
idxtest=[1:length(xtest')];
y = sim(net,xtest')
figure (2);
plot (idxtest,ttest,idxtest,y,'go')

% error data aktual dan prediksi pada testing
MSE = sum((ttest-y)*(ttest-y)')/length(ttest)
MAE = sum(abs(ttest-y))/length(ttest)
MAPE = sum(abs(ttest-y)./y)/length(ttest)

% data prediksi
data5=data2((mData*nData)-nInput+1:mData*nData);

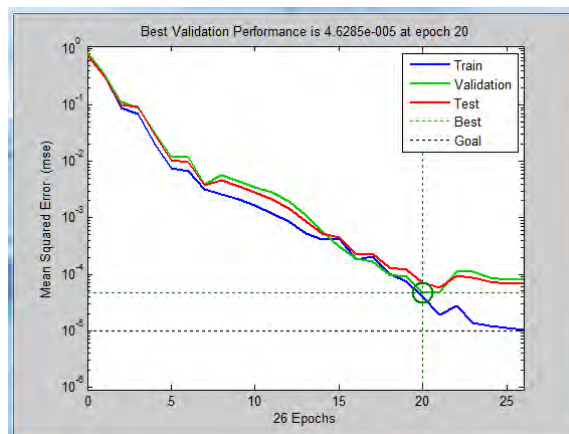
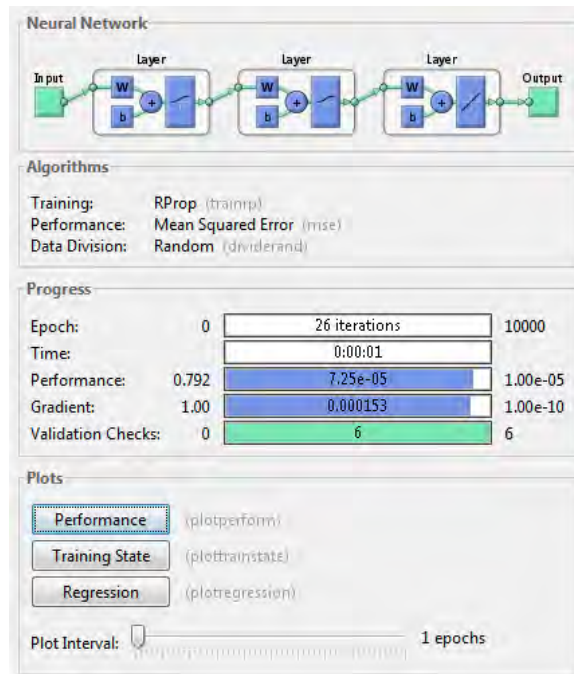
for i=1:nPrediksi
    pred(i)=sim(net,data5');
    data5=[data5(2:nInput) pred(i)];
end

% normalisasi data
for i=1:length(pred)
    pred2(i)=round(pred(i)*(max(data1)-min(data1))+min(data1));
end
disp(pred2)
end

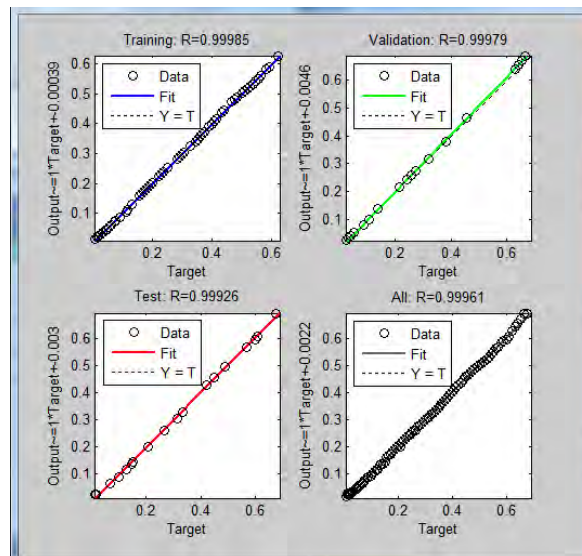
```

## LAMPIRAN 2. Hasil Running Program Time Series JST Backpropagation Menggunakan MATLAB R20110a

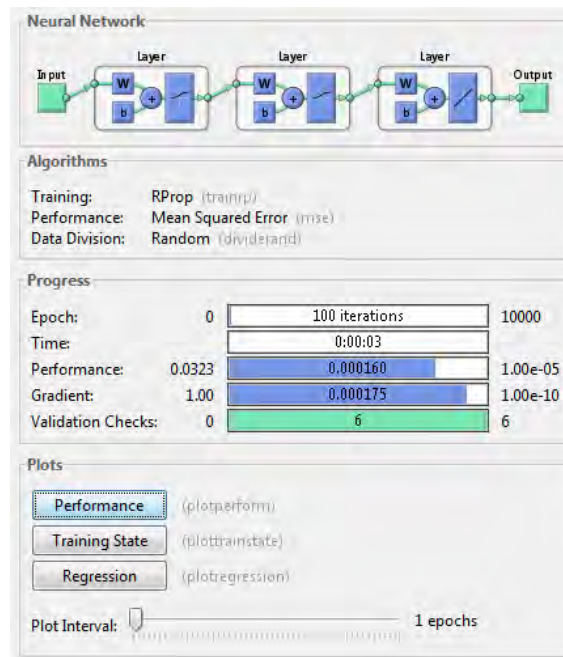
### 1. Variabel PDRB Sektor Jasa

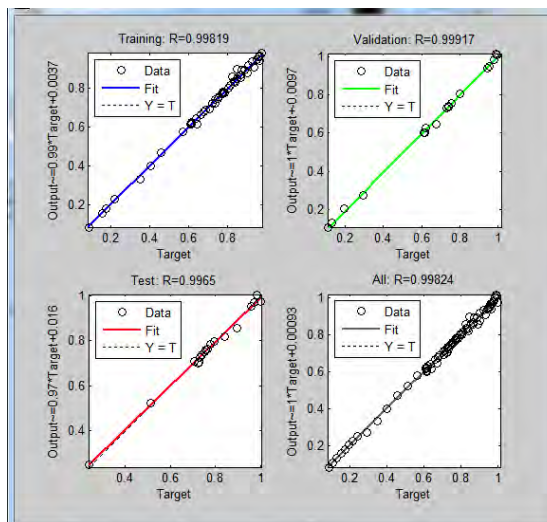
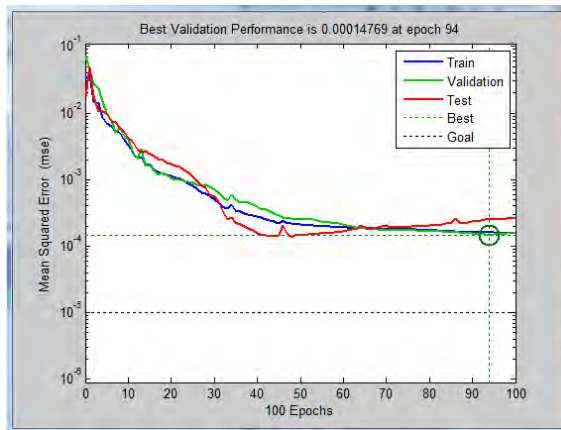






## 2. Variabel Pengukuran





### 3. Variabel Pengeluaran Barang Tahan Lama

**Neural Network**

**Algorithms**

Training: RProp (trainf)

Performance: Mean Squared Error (mse)

Data Division: Random (dividerand)

**Progress**

Epoch:	0	48 iterations	10000
Time:		0:00:04	
Performance:	0.107	9.01e-05	1.00e-05
Gradient:	1.00	0.000142	1.00e-10
Validation Checks:	0	6	6

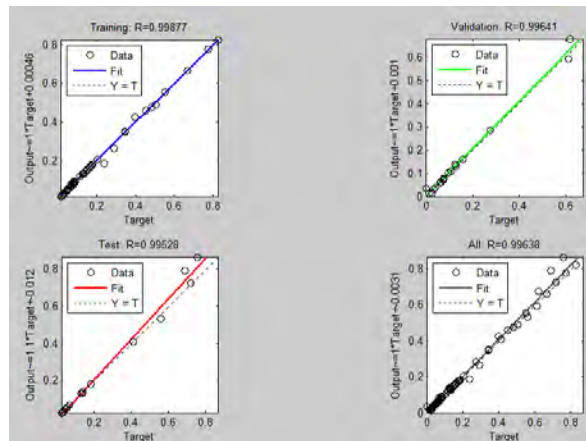
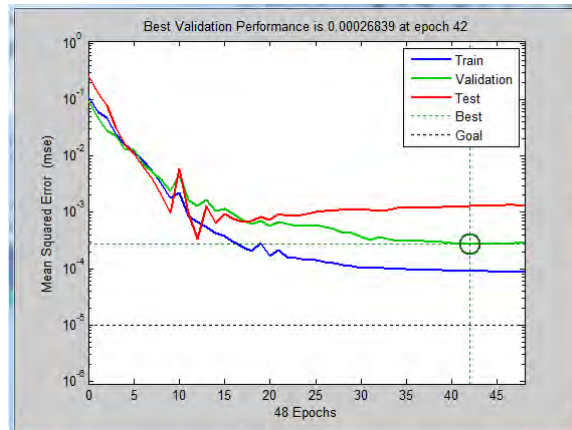
**Plots**

Performance (plotperform)

Training State (plottrainstate)

Regression (plotregression)

Plot Interval: 1 epochs



#### 4. Variabel Pengeluaran Buah

**Neural Network:**

**Algorithms:**

Training: RProp (trainrp)  
 Performance: Mean Squared Error (mse)  
 Data Division: Random (dividerand)

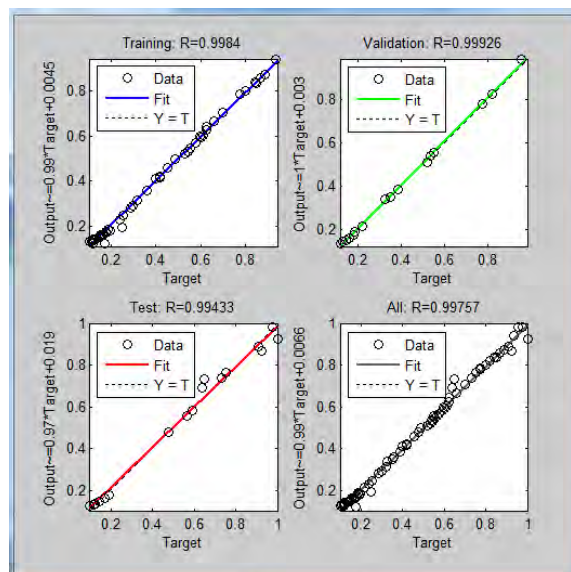
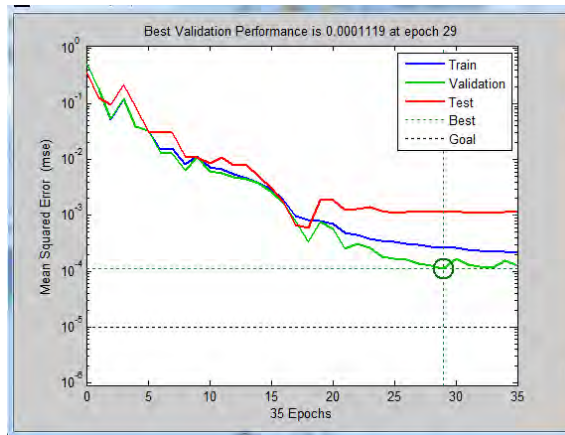
**Progress:**

Epoch:	0	35 iterations	10000
Time:		0:00:01	
Performance:	0.522	0.000277	1.00e-05
Gradient:	1.00	0.00223	1.00e-10
Validation Checks:	0	6	6

**Plots:**

Performance (plotperform)  
 Training State (plottrainstate)  
 Regression (plotregression)

Plot Interval: 1 epochs



## 5. Variabel Jumlah Penduduk Miskin

**Neural Network**

**Algorithms**

Training: RProp (trainrp)  
 Performance: Mean Squared Error (mse)  
 Data Division: Random (dividerand)

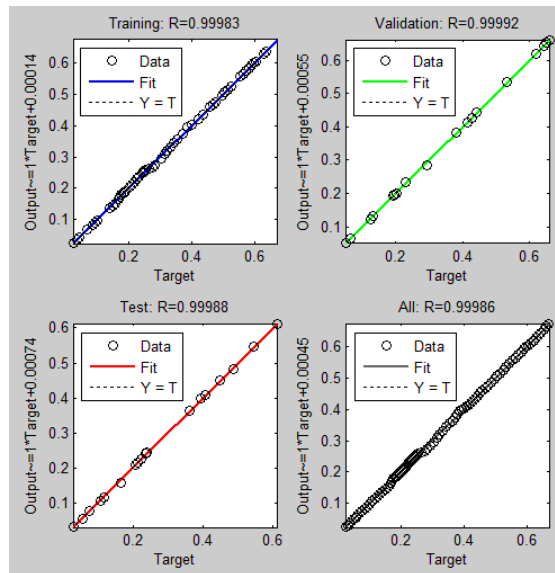
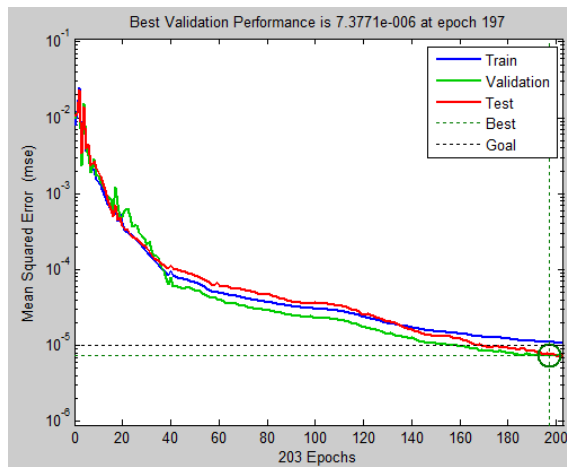
**Progress**

Epoch:	0	203 iterations	10000
Time:		0:00:05	
Performance:	0.00785	1.12e-05	1.00e-05
Gradient:	1.00	3.96e-05	1.00e-10
Validation Checks:	0	6	6

**Plots**

Performance (plotperform)  
 Training State (plottrainstate)  
 Regression (plotregression)

Plot Interval: 1 epochs



#### LAMPIRAN 4. Listing Program Multivariat JST Backpropagation Menggunakan MATLAB R20110a

```
function Backpro_multivariat()
clear all;clc;close all;
nInput=5;
nHidden1=40;
nHidden2=10;
% nOutput=1;
error=0.00001;
epoch=10000;
lr=0.01;

[FileName,PathName]=uigetfile('*.xls','Buka Data Dari Excel');
ExcelFile=fullfile(PathName,FileName);
Data=xlsread(ExcelFile,-1);
[mData,nData]=size(Data);

% menyusun data dalam satu baris
data1=[];
for i=1:mData
    data1=[data1 Data(i,:)];
end

% normalisasi data
for i=1:length(data1)
    data2(i)=(data1(i)-min(data1))/(max(data1)-min(data1));
end

% data training: input dan target
dataTrain=(mData*nData)*80/100;
data3=data2(1:dataTrain);
idx=1;
idt=nInput+1;
while idx<=96
    x(idx,1:nInput)=data3((idx-1)*nInput+idx:idx*nInput+(idx-1));
    t(idx)=data3(idx*idt);
    idx=idx+1;
end

% algoritma backpropagation
net=newff(x',t,[nHidden1 nHidden2],{'logsig' 'logsig'},'trainrp');
%Set max epoch, goal, learning rate, show step
net.trainParam.epochs=epoch;
net.trainParam.goal=error;
net.trainParam.lr=lr;
net.trainParam.ShowWindow=200;
net.trainParam.ShowCommandLine=false;

% melakukan training
net=train(net,x',t);

% simulasi training
idx=[1:length(x')];
y = sim(net,x');
figure (1);
```

```

plot (idx,t,idx,y,'co')

% data testing: input dan target
data4=data2 (dataTrain+1:714);
idxtest=1;
while idxtest<=23
    xtest (idxtest,1:nInput)=data4 ((idxtest-
1)*nInput+idxtest:idxtest*nInput+(idxtest-1));
    ttest (idxtest)=data4 (idxtest*nInput+idxtest);
    idxtest=idxtest+1;
end

% simulasi testing
idxtest=[1:length(xtest')];
y = sim(net,xtest');
figure (2);
plot (idxtest,ttest,idxtest,y,'go')

% error data aktual dan prediksi pada testing
MSE = sum ((ttest-y) * (ttest-y)') / length (ttest)
MAE = sum (abs (ttest-y)) / length (ttest)
MAPE = sum (abs (ttest-y) ./ y) / length (ttest)

% data prediksi
data5=data2 (715:719);
pred=sim (net,data5');

% normalisasi data
pred2=round (pred* (max (data1) -min (data1)) +min (data1))
end

```

