

Lembar Kemajuan Skripsi

Nama	Arini Indah Permatasari
NIM	115060807111096
No HP	081945151595/081555464364
Email	ariniindah.ps@gmail.com
Pembimbing 1	Wayan Firdaus Mahmudy,S.Si.,M.T.,Ph.D.
Pembimbing 2	-
Judul	Pemodelan Regresi Linear dalam Konsumsi kWh Listrik di Kota Batu Menggunakan Algoritma Genetika
Tanggal Mulai	3 November 2014

Tanggal	Pembimbing	Perbaikan
3 Oktober 2014	1 (tatap muka)	Konsultasi perbaikan judul
17 November 2014	1 (email)	Bab 1
22 Januari 2015	1 (email)	Revisi Bab 1(latar belakang dan rumusan masalah) Lampiran Bab 2
24 Februari 2015	1 (email)	Lampiran Bab 3
2 Maret 2015	1 (email)	Lampiran Bab 4
12 Maret 2015	1 (tatap muka)	Konsultasi perbaikan perhitungan representasi kromosom dengan koefisien regresi
13 Maret 2015	1 (email)	Perbaikan Bab3 (hal 24 – 34) perhitungan representasi kromosom sampai proses Algoritma Genetik
19 Maret 2015	1(tatap muka)	Konsultasi formula permasalahan dan representasi kromosom
19 Maret 2015	1 (email)	Perbaikan Bab 3 1. Contoh permasalahan sebelumnya subbab 3.4 direvisi pada subbab 3.3 (hal 22). 2. Perbaikan representasi

02 April 2015	1 (email)	kromosom (hal 26) 1. Perbaiki Penjelasan Bab 3 (hal 22, hal 24 dan hal 27). 2. Lampiran Bab 5
6 April 2015	1 (tatap muka)	Konsultasi dan presentasi program
14 April 2015	1 (email)	1. Perbaiki penjelasan pada subbab 3.4.1 2. Lampiran Bab 6 dan Bab 7
16 April 2015	1 (email)	1. Perbaiki penulisan Tabel dan Gambar 2. Perbaiki penjelasan Analisa Uji Ukuran Populasi dan Uji Generasi
		3. Perbaiki penulisan Tabel dan Gambar 4. Perbaiki penjelasan Analisa Uji Ukuran Populasi dan Uji Generasi 5. Menambahkan total error yang dihasilkan Algoritma Genetika dan analisis regresi pada analisis subab 6.5 6. Perbaiki penulisan daftar pustaka, serta tambahan refrensi berdasarkan saran dosen pembimbing 1 7. Lampiran Jurnal
28 April 2015	1 (email)	1. Perbaiki penulisan daftar pustaka 2. Tambahkan kajian pustaka pada Bab 2 3. Perbaiki program serta perbaikan hasil pengujian

**PEMODELAN REGRESI LINEAR DALAM KONSUMSI KWH LISTRIK
DI KOTA BATU MENGGUNAKAN ALGORITMA GENETIKA**

SKRIPSI

LABORATORIUM KOMPUTASI CERDAS DAN VISUALISASI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun Oleh :

ARINI INDAH PERMATASARI

115060807111096

KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI

PROGRAM STUDI INFORMATIKA/ILMU KOMPUTER

PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

MALANG

2015

LEMBAR PERSETUJUAN
PEMODELAN REGRESI LINEAR DALAM KONSUMSI KWH LISTRIK
DI KOTA BATU MENGGUNAKAN ALGORITMA GENETIKA

SKRIPSI

LABORATORIUM KOMPUTASI CERDAS DAN VISUALISASI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun Oleh :

ARINI INDAH PERMATASARI

NIM. 115060807111096

Telah diperiksa dan disetujui oleh:

Pembimbing

Wayan Firdaus Mahmudy, S.Si., M.T.,Ph.D

NIP. 19771114 200312 2 001

LEMBAR PENGESAHAN
PEMODELAN REGRESI LINEAR DALAM KONSUMSI KWH LISTRIK
DI KOTA BATU MENGGUNAKAN ALGORITMA GENETIKA

SKRIPSI

LABORATORIUM KOMPUTASI DAN SISTEM CERDAS

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer

Disusun Oleh :

ARINI INDAH PERMATASARI

NIM. 115060807111096

Skripsi ini telah diuji dan dinyatakan lulus pada tanggal 15 Mei 2015

Penguji I

Penguji II

Nurul Hidayat, S.Pd., M.Sc
NIP. 19680430 200212 1 001

Candra Dewi, S.Kom., M.Sc
NIP. 19771114 200312 2 001

Penguji III

Barlian Henryranu Prasetyo, S.T., M.T.

NIK. 82102406110254

Mengetahui,

Ketua Program Studi Informatika/Illmu Komputer

Drs. Marji, M.T.

NIP. 19670801 199203 1 001

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata didalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku. (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, Mei 2015

Mahasiswa,

ARINI INDAH PERMATASARI

NIM. 115060807111096

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat, rahmat, ridho dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “**Pemodelan Regresi Linear Dalam Konsumsi kWh Listrik di Kota Batu Menggunakan Algoritma Genetika**” sebagai salah satu persyaratan untuk menyelesaikan studi di Jurusan Informatika, Program Studi Informasi dan Ilmu Komputer Universitas Brawijaya.

Penulis menyadari bahwa tugas akhir ini dapat terselesaikan berkat bantuan, petunjuk, bimbingan dan dukungan dari berbagai pihak yang telah banyak membantu proses penyelesaian tugas akhir ini. Oleh karena itu penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada :

1. Bapak Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D., selaku dosen pembimbing I yang telah banyak sekali memberikan ilmu, membantu dan membimbing serta saran untuk skripsi ini.
2. Bapak Nurul Hidayat, S.Pd., M.Sc., Ibu Candra Dewi, S.Kom., M.Sc., dan Bapak Barlian Henryranu, S.T., M.T. selaku dosen penguji yang telah meluluskan saya serta banyak memberikan kritik dan juga saran untuk skripsi agar menjadi lebih baik lagi.
3. Seluruh dosen Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya Malang atas segala bimbingan serta ilmu yang telah diajarkan kepada penulis
4. Orang tua penulis dan saudara penulis, Bapak Arik Kusuma dan Ibu Kusmarini, S.E., dan Ayu Winda Irene P. yang memberikan dukungan moral dan material.
5. Teman-teman penulis, Wiki, Dwi, Iim, Meidina, Ria, Ihda, Jasicka, Amy, Mas Putra, Mbak Ayu, Mas Dhika, Ganda, Mbak Dhinda, Mbak Dita, Mas Tosi, Mas Gilang dan semua pihak yang tidak bisa penulis sebutkan satu persatu, terima kasih atas segala bantuan yang diberikan.

6. Seluruh pengurus dan staff Badan Eksekutif Mahasiswa Kabinet Bersatu II atas dukungan dan motivasinya.
7. Teman-teman IF-G terimakasih atas kebersamaan serta bantuan yang telah diberikan dan kita lakukan bersama dari semester 1 hingga semester 8 ini
8. Sahabat SMA penulis Nana, Bagus, Bayu, Septian, Viky, Ony dan Mifta terimakasih atas support yang selalu diberikan semoga kelak kita sukses bersama.
9. Teman-teman seperjuangan Program Teknologi Informasi dan Ilmu Komputer angkatan 2011 atas rasa kebersamaan yang begitu erat.
10. Dan semua pihak yang tidak bisa disebutkan satu per satu. Terima kasih atas segala bantuannya.

Penulis sadar bahwa skripsi ini masih banyak kekurangan, oleh karena itu kritik dan saran yang bersifat membangun sangat diharapkan untuk menyempurnakan skripsi ini. Penulis berharap skripsi ini dapat bermanfaat khususnya bagi diri sendiri dan bagi semua pihak.

Malang, Mei 2015

Penulis

ABSTRAK

Arini Indah Permatasari, 2015. Pemodelan Regresi Linear dalam Konsumsi KWh Listrik di Kota Batu Menggunakan Algoritma Genetika.

Dosen Pembimbing : Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D

Kota Batu merupakan salah satu kota kecil yang konsumsi listriknya meningkat sangat pesat setiap tahun karena tumbuhnya sektor pariwisata yang menjadi ikon kota. Maka kebutuhan akan daya listrik di Kota Batu selalu bertambah di tiap tahunnya sehingga menyebabkan konsumsi kWh listrik di Kota Batu semakin besar pula. Perubahan konsumsi kWh listrik tersebut jika tidak diolah dengan baik maka akan menimbulkan beban energi listrik yang tidak terbandung, oleh karena itu dibutuhkan suatu estimasi untuk memperkirakan dengan tepat seberapa besar kWh listrik yang dibutuhkan di tahun-tahun berikutnya. Permasalahan tersebut akan dimodelkan dengan persamaan regresi linear yang terdiri dari variabel bebas (X) dan variabel terikat (Y). Solusi yang dapat diberikan adalah dengan memberikan koefisien regresi terbaik menggunakan konsep Algoritma Genetika. Proses reproduksi menggunakan *extended intermediate* crossover dan *random* mutation, untuk proses seleksinya menggunakan model *elitism* selection. Dari Algoritma Genetika tersebut didapatkan parameter terbaik yaitu ukuran populasi sebanyak 175, generasi terbaik sebanyak 1500 generasi, kombinasi *cr* : *mr* berturut-turut adalah 0,6 : 0,4 dengan nilai *fitness* tertinggi yaitu 5,4098. Hasil akhir berupa koefisien terbaik yang menjadi model regresi.

Kata Kunci : Regresi Linear, Algoritma Genetika, Prediksi, kWh Listrik

ABSTRACT

Arini Indah Permatasari, 2015. Linear Regression Modelling of KWH Electricity Consumption in Batu City Using Genetic Algorithm.

Advisor : Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D

Batu City is one of small town that electricity consumption is rapidly increasing every year due to the growth of the tourism sector become an icon of the city. Accordingly the need for electrical power in Batu always increases in each year, causing of kWh electricity consumption the greater it is. Changes in the electrical kWh consumption if not treated properly, it will cause the electrical energy load unstoppable, therefore it is necessary to estimate to predict exactly how much of electricity need in subsequent years. This problem will be modeled with linear regression tha consists of the independent variable (X) and the dependent variable (Y). Solution that can be given is toprovide the best regression coefisient using the concept of Genetic Algorithm. In the reproduction process using the extended intermediate crossover and random mutation, for the selection process using a elitism selection models. Of the genetic algorithm parameters obtained the best population size is 175, the best generation as much as 1500 generation, combination of cr : mr consecutive is 0,6 : 0,4 with the highest fitness value is 5,4098. The end of result being the best of coefficient regression model.

Keywords : Linear Regression, Genetic Algorithm, Predict, kWh Consumption.

DAFTAR ISI

LEMBAR PERSETUJUAN.....	i
LEMBAR PENGESAHAN	ii
PERNYATAAN ORISINALITAS SKRIPSI	iii
KATA PENGANTAR	iv
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	xii
DAFTAR TABEL.....	xiii
DAFTAR PERSAMAAN	xiv
DAFTAR SOURCE CODE.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	4
1.5 Manfaat.....	4
1.6 Sistematika Penulisan.....	4
BAB II KAJIAN PUSTAKA DAN DASAR TEORI.....	7
2.1 Kajian Pustaka	7
2.2 Regresi Linear	10
2.2.1 Analisis Regresi Linear.....	11
2.2.2 <i>Mean Squared Error (MSE)</i>	15
2.3 Daya Listrik.....	15
2.3.1 KWH Listrik	17

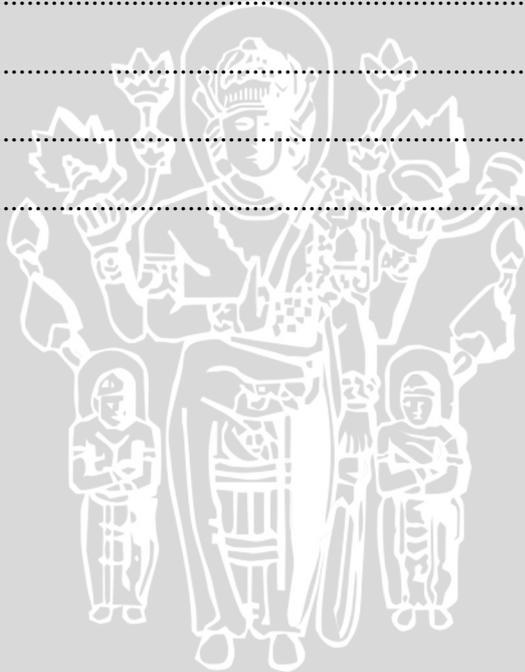


2.3.2	Perhitungan Pemakaian Energi Listrik di Rumah.....	18
2.4	Algoritma Genetika	20
2.4.1	Proses pada Algoritma Genetika.....	22
2.4.2	Representasi Kromosom	25
2.4.3	Reproduksi	26
2.4.4	Fungsi <i>Fitness</i>	26
2.4.5	Seleksi	27
BAB III METODE PENELITIAN.....		28
3.1	Tahapan Penelitian	28
3.2	Analisis Kebutuhan Sistem	30
3.3	Formulasi Permasalahan.....	30
3.3.1	Deskripsi Umum Sistem.....	30
3.3.2	Data yang Digunakan.....	31
3.3.3	Contoh Persoalan	31
3.4	Siklus Penyelesaian Pemodelan Regresi Linear menggunakan Algoritma Genetika.....	33
3.4.1	Representasi Kromosom, Mencari <i>Error</i> dan Perhitungan <i>Fitness</i>	35
3.4.2	Inisialisasi Populasi Awal	37
3.4.3	Proses Reproduksi.....	38
3.4.4	Proses Evaluasi	44
3.4.5	Proses Seleksi.....	45
BAB IV PERANCANGAN		47
4.1	Perancangan <i>User Interface</i>	47
4.1.1	Rancangan Halaman <i>Home</i>	47
4.1.2	Halaman Input Parameter dan Data Konsumsi kWh	47



4.1.3	Rancangan Halaman Inputan Parameter Algoritma Genetika dan Hasil Prediksi konsumsi kWh.....	48
4.2	Perancangan Uji Coba dan Evaluasi.....	49
4.2.1	Uji Coba Range <i>popSize</i>	50
4.2.2	Uji Coba Banyaknya Jumlah Generasi	50
4.2.3	Uji Coba Kombinasi <i>Crossover Rate</i> dan <i>Mutation Rate</i>	51
4.2.4	Uji Coba Banyaknya Periode Konsumsi kWh.....	51
BAB V IMPLEMENTASI.....		53
5.1	Implementasi Program	53
5.1.1	Struktur Data	53
5.1.2	Me-load Data Konsumsi kWh	53
5.1.3	Pembangkitan Representasi Kromosom	55
5.1.4	Perhitungan <i>Error</i> dan <i>Fitness</i>	55
5.1.5	Proses <i>Crossover</i>	56
5.1.6	Proses Mutasi	57
5.1.7	Proses Seleksi dengan Metode <i>Elitism Selection</i>	58
5.1.8	Proses Pemilihan Kromosom Terbaik Sebanyak <i>Popsiz</i> Awal.....	59
5.1.9	Proses Iterasi	60
5.2	Implementasi Antarmuka	60
5.2.1	Halaman Awal.....	60
5.2.2	Halaman Help.....	61
5.2.3	Halaman About	62
5.2.4	Halaman Input Parameter.....	62
5.2.5	Halaman Hasil Koefisien Terbaik dari Proses Algoritma Genetika	63
5.2.6	Halaman Menampilkan Detail Error dan Fitness.....	64
BAB VI PENGUJIAN DAN ANALISA		65

6.1	Hasil dan Analisa Uji Coba Ukuran Populasi (popSize).....	65
6.2	Hasil dan Analisa Uji Coba Banyaknya Generasi.....	66
6.3	Hasil dan Analisa Uji Coba Kombinasi <i>Crossover Rate</i> dan <i>Mutation Rate</i>	68
6.4	Hasil dan Analisa Uji Coba Banyaknya Periode Konsumsi kWh (dalam Bulan).....	70
6.5	Hasil dan Analisa Perbandingan Hasil Regresi dengan Algoritma Genetika.....	72
BAB VII PENUTUP.....		75
7.1	Kesimpulan.....	75
7.2	Saran.....	76
Daftar Pustaka.....		77
Daftar Lampiran.....		79



DAFTAR GAMBAR

Gambar 2.1 KWH Meter Analog	16
Gambar 2.2 <i>Pseudocode</i> Algoritma Genetika.....	20
Gambar 2.3 <i>Flowchart</i> Proses Algoritma Genetika	21
Gambar 2.4 Representasi <i>Real Code</i>	24
Gambar 2.5 <i>Pseudocode</i> <i>Elitism Selection</i>	25
Gambar 3.1 Diagram Alir Metode Penelitian	26
Gambar 3.2 <i>Pseudocode</i> Sistem.....	32
Gambar 3.3 <i>Flowchart</i> Proses <i>Crossover</i>	36
Gambar 3.4 <i>Diagram Alir</i> Proses <i>Extended Intermediate Crossover</i>	37
Gambar 3.5 Hasil <i>Crossover</i>	38
Gambar 3.6 Diagram Alir Proses Mutasi.....	40
Gambar 3.7 Hasil Mutasi	41
Gambar 3.8 <i>Pseudocode</i> Proses <i>Elitism Selection</i>	44
Gambar 4.1 <i>User Interface</i> Halaman <i>Home</i>	45
Gambar 4.2 Rancangan Halaman Input Data Konsumsi kWh.....	46
Gambar 4.3 Rancangan Tampilan Halaman hasil Prediksi.....	47
Gambar 5.1 Implementasi Antarmuka Halaman Awal Aplikasi	58
Gambar 5.2 Implementasi Halaman <i>Help</i>	59
Gambar 5.3 Implementasi Halaman <i>About</i>	60
Gambar 5.4 Implementasi Halaman Inputan Parameter	61
Gambar 5.5 Implementasi Hasil Akhir Algoritma Genetika	61
Gambar 6.1 Grafik Hasil Uji Coba Ukuran Populasi.....	63
Gambar 6.2 Grafik Hasil Uji Coba Ukuran Generasi	65
Gambar 6.3 Grafik Hasil Uji Coba Kombinasi <i>cr</i> dan <i>mr</i>	67
Gambar 6.4 Grafik Hasil Uji Coba Banyaknya Periode	68
Gambar 6.5 Perbandingan Tingkat Error Regresi dengan Algoritma Genetika ...	70

DAFTAR TABEL

Tabel 2.1 Perbandingan Objek dan Metode	8
Tabel 2.2 Perbandingan <i>Input</i> , Proses dan <i>Output</i>	8
Tabel 3.1 Pencapaian Bulanan Per 3 Tahun Terakhir	31
Tabel 3.2 Pemakaian kWh	31
Tabel 3.3 Data Pemakaian kWh Berdasarkan Pemakaian 4 Bulan Kebelakang	32
Tabel 3.4 Pembentukan Kromosom	34
Tabel 3.5 Hasil Perhitungan Konsumsi kWh dan Nilai <i>Error</i>	35
Tabel 3.6 Hasil Perhitungan <i>Fitness</i>	36
Tabel 3.7 Data Populasi Awal	36
Tabel 3.8 Hasil Evaluasi	43
Tabel 3.9 Hasil Seleksi Menggunakan <i>Elitism Selection</i>	44
Tabel 3.10 Detail Kromosom Terbaik	45
Tabel 4.1 Rancangan Uji Coba Range <i>popSize</i>	49
Tabel 4.2 Rancangan Uji Coba Banyaknya Generasi	49
Tabel 4.3 Rancangan Uji Coba <i>Crossover Rate</i> dan <i>Mutation Rate</i>	50
Tabel 4.4 Rancangan Uji Coba Banyaknya Periode Konsumsi kWh	51
Tabel 4.5 Rancangan Tabel Hasil Pengujian Data	Error! Bookmark not defined.
Tabel 6.1 Hasil Uji Coba Ukuran Populasi	62
Tabel 6.2 Hasil Uji Coba Ukuran Banyaknya Generasi	64
Tabel 6.3 Hasil Uji Coba <i>Kombinasi Crossover Rate</i> dan <i>Mutation Rate</i>	66
Tabel 6.4 Hasil Uji Coba Ukuran Populasi	68
Tabel 6.5 Detail <i>Error</i> Regresi dan <i>Error</i> Algoritma Genetika	70

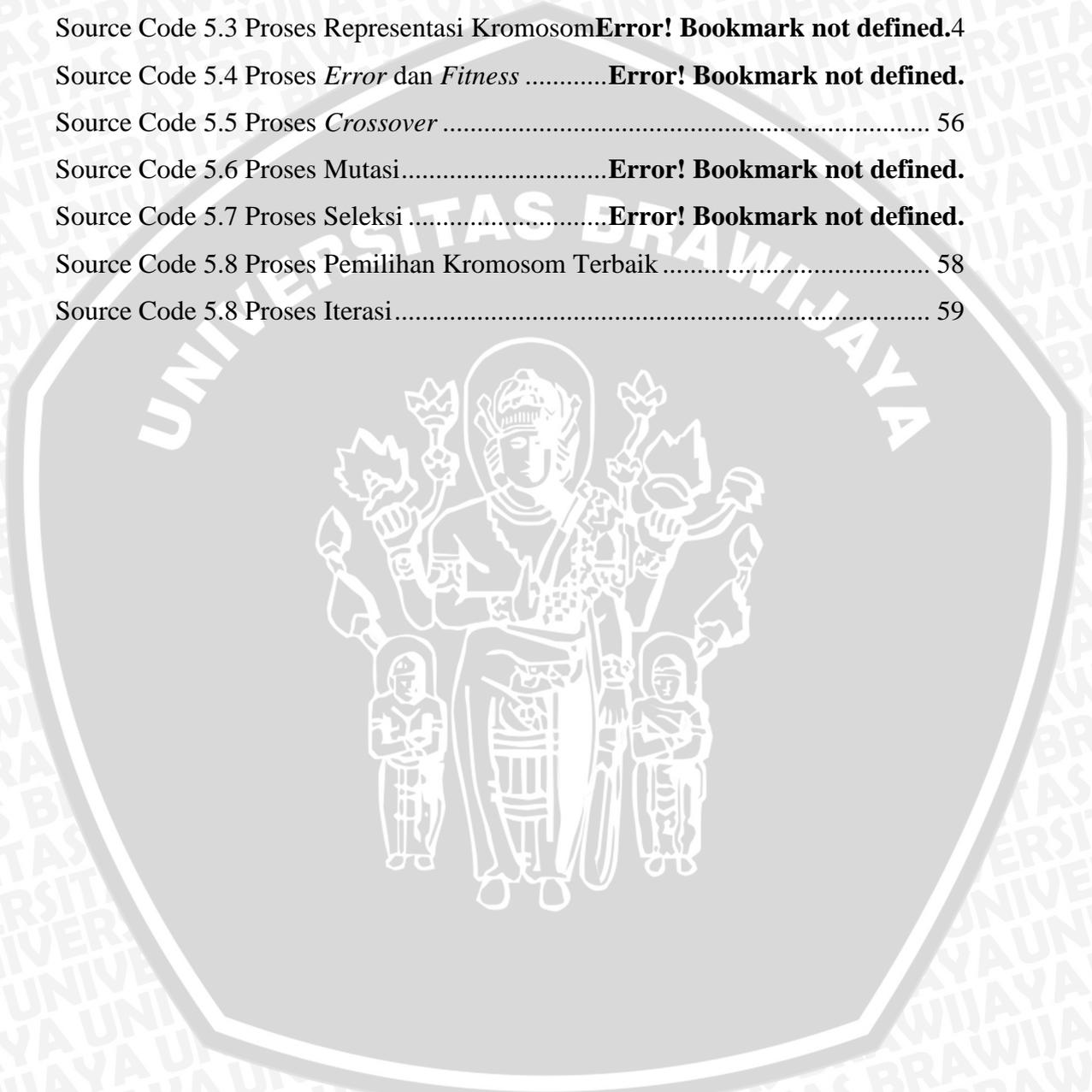
DAFTAR PERSAMAAN

Lampiran 1 Konsumsi kWh Listrik Tahun 2012	75
Lampiran 2 Konsumsi kWh Listrik Tahun 2013 .7Error! Bookmark not defined.	
Lampiran 3 Konsumsi kWh Listrik Tahun 2014	76
Lampiran 4 Data Periode 2 Bulan	77
Lampiran 5 Data Periode 3 Bulan	77
Lampiran 6 Data Periode 4 Bulan	78
Lampiran 7 Data Periode 5 Bulan	79
Lampiran 8 Data Periode 6 Bulan	80
Lampiran 9 Data Periode 7 Bulan	81
Lampiran 10 Data Periode 8 Bulan	82
Lampiran 11 Data Periode 9 Bulan	82
Lampiran 12 Data Periode 10 Bulan	83



DAFTAR SOURCE CODE

Source Code 5.1 Proses Struktur Data	52
Source Code 5.2 Proses <i>Load</i> Data	Error! Bookmark not defined. 3
Source Code 5.3 Proses Representasi Kromosom.....	Error! Bookmark not defined. 4
Source Code 5.4 Proses <i>Error</i> dan <i>Fitness</i>	Error! Bookmark not defined.
Source Code 5.5 Proses <i>Crossover</i>	56
Source Code 5.6 Proses Mutasi.....	Error! Bookmark not defined.
Source Code 5.7 Proses Seleksi	Error! Bookmark not defined.
Source Code 5.8 Proses Pemilihan Kromosom Terbaik	58
Source Code 5.8 Proses Iterasi.....	59



BAB I Pendahuluan

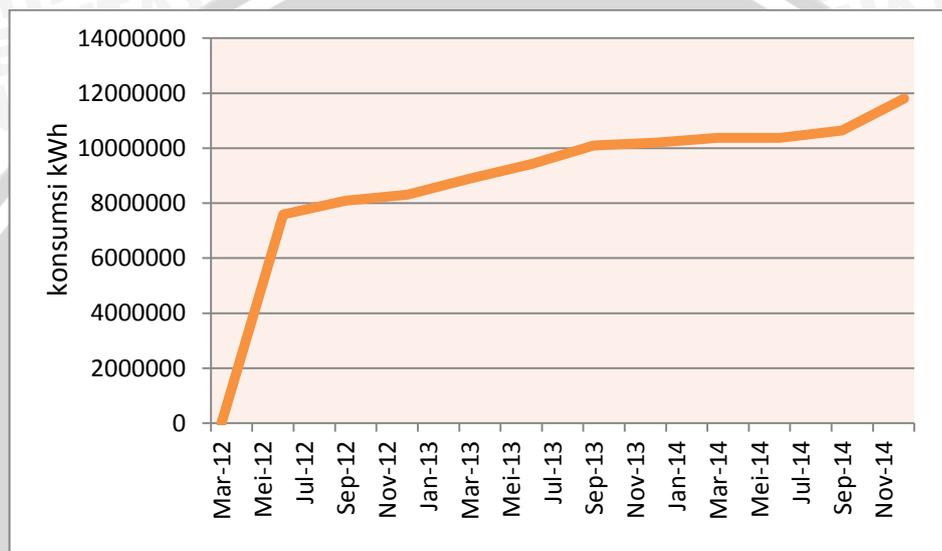
1.1 Latar Belakang

Pertumbuhan penduduk yang semakin pesat serta diiringi dengan pertumbuhan ekonomi yang tinggi menyebabkan kebutuhan akan tenaga listrik semakin meningkat. Penggunaan tenaga listrik saat ini merupakan salah satu kebutuhan penting dalam kehidupan masyarakat dan sering pula dianggap sebagai salah satu tolok ukur taraf kemajuan masyarakat yang sejalan dengan perkembangan teknologi. Besarnya perubahan ini kemudian akan mempengaruhi penyediaan daya energi listrik yang memadai. Untuk memenuhi kebutuhan pelanggan akan energi listrik, maka pihak penyuplai energi listrik dalam hal ini adalah Perusahaan Listrik Negara (PLN) berusaha untuk memenuhi kebutuhan yang semakin meningkat tersebut. Salah satu upaya yang bisa dilakukan adalah menyediakan pembangkit listrik yang baru (Saripurna,2012).

Kota Batu merupakan salah satu kota kecil yang konsumsi listriknya meningkat sangat pesat setiap tahun karena tumbuhnya sektor pariwisata yang menjadi ikon kota (Yonathan, 2013). Beberapa tempat wisata yang tiap tahun bertambah pengunjungnya baik dari domestik maupun mancanegara dapat menunjang perekonomian warga Kota Batu. Dengan meningkatnya laju perekonomian di Kota Batu dalam segi pariwisata, maka kebutuhan akan daya listrik di Kota Batu selalu bertambah di tiap tahunnya sekitar 8% sampai 10% dari tahun 2012 sampai 2014 (PLN, 2014), sehingga menyebabkan konsumsi kWh listrik di Kota Batu semakin besar pula. Peningkatan konsumsi kWh di Kota Batu akan ditampilkan dalam bentuk grafik yang akan ditunjukkan pada Gambar 1.1.

Perubahan konsumsi kWh listrik tersebut jika tidak diolah dengan baik maka akan menimbulkan beban energi listrik yang tidak terbandung jika kWh yang dibeli lebih dari konsumsi penduduk ataupun bisa jadi akan dilakukan pemadaman bergilir jika kWh yang dibeli kurang dari konsumsi, oleh karena itu dibutuhkan suatu estimasi untuk memperkirakan dengan tepat seberapa besar kWh listrik yang dibutuhkan untuk melayani beban serta kebutuhan energi listrik

di tahun tahun berikutnya. Dengan mengadakan prakiraan maka kita dapat menarik kesimpulan dan mengetahui bahwa perkiraan yang tidak tepat nantinya akan menyebabkan kurangnya kapasitas kWh yang disalurkan guna untuk memenuhi kebutuhan beban, sebaliknya jika perkiraan beban terlalu besar maka dapat menyebabkan kelebihan kapasitas pembangkit sehingga menyebabkan kerugian (Sriwati,2013).



Gambar 1.1 Grafik Konsumsi kWh
Sumber : (PLN-Rayon Batu, 2014)

Permasalahan tersebut akan dimodelkan dengan persamaan regresi linear yang terdiri dari variabel bebas (X) dan variabel terikat (Y) (Sriwati, 2013). Dengan model regresi dapat ditentukan apakah ada hubungan antara Y dan X, kemudian bagaimana bentuk hubungan tersebut, dan mengapa terbentuk hubungan tersebut.

Beberapa jurnal telah membahas tentang permasalahan penggunaan daya listrik dengan model regresi linear ataupun non linear, misalnya pada penelitian yang dilakukan oleh Sriwati pada tahun 2013 dengan judul “Prakiraan Kebutuhan Daya Listrik di Kabupaten Moros dengan Pemodelan Linear” dan penelitian yang dilakukan oleh Harifuddin pada tahun 2007 yang membahas permasalahan Estimasi Kebutuhan Daya Listrik Sulawesi Selatan. Dua penelitian tersebut menjadi referensi serta acuan untuk menyelesaikan permasalahan pemodelan regresi linear pada penelitian skripsi ini.

Berdasarkan permasalahan serta solusi yang digunakan untuk menyelesaikan permasalahan tersebut dirasa kurang begitu optimal dengan akurasi akhir bernilai 0,16% pada penelitian Sriwati pada tahun 2013, sedangkan pada penelitian yang dilakukan Harifuddin pada tahun 2007 tingkat akurasi akhir bernilai 3% sampai 5%. Pada saat ini telah muncul berbagai macam metode baru yang ditawarkan bisa jadi menyelesaikan permasalahan tersebut secara optimal. Solusi yang dapat diberikan adalah dengan menyelesaikan permasalahan tersebut menggunakan konsep Algoritma Genetika. Dimulai dari insialiasi representasi kromosom, kemudian proses reproduksi yang dibagi menjadi dua yaitu crossover dan mutasi, lalu proses evaluasi yaitu mengumpulkan semua individu dalam satu populasi, setelah itu proses terakhir adalah seleksi dimana individu yang terbaik akan dipertahankan yang kurang baik akan dibuang (Mahmudy, 2013). Dari beberapa proses tersebut diyakini Algoritma Genetika memiliki beberapa kelebihan dibandingkan algoritma yang lain dalam menghasilkan output yang optimal dan dapat dimanfaatkan untuk menyelesaikan permasalahan diatas.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan di atas maka rumusan masalah yang dibuat adalah :

1. Bagaimana pemodelan regresi linear dalam konsumsi kWh listrik di Kota Batu dengan menggunakan Algoritma Genetika?
2. Bagaimana menentukan parameter Algoritma Genetika yang tepat?
3. Berapa tingkat akurasi prakiraan yang dapat diberikan oleh model regresi yang dibangun menggunakan Algoritma Genetika?

1.3 Batasan Masalah

Agar pembahasan penelitian ini tidak menyimpang dari apa yang telah dirumuskan, maka diperlukan batasan-batasan. Batasan-batasan dalam penelitian ini adalah :

1. Pemodelan regresi yang digunakan dalam permasalahan ini adalah persamaan regresi linear berganda.

2. Metode yang dipakai adalah Algoritma Genetika yang dimulai dari proses inialisasi kromosom yang dibangkitkan secara *random* dalam rentang $[-100,100]$, proses reproduksi yang dilakukan ada dua metode yaitu *crossover* dan mutasi. Metode *crossover* yang digunakan adalah *extended intermediate crossover* sedangkan metode mutasi yang digunakan adalah *random mutation*. Setelah itu dilakukan evaluasi dan seleksi menggunakan model *elitism selection*.
3. Obyek yang digunakan adalah konsumsi kWh listrik Kota Batu
4. Data yang dibuat acuan konsumsi kWh listrik 3 tahun terakhir, 2012 – 2014 yang dibuat dalam bulanan.
5. Output yang dihasilkan oleh sistem adalah pemodelan regresi linear dengan koefisien nilai regresi yang terbaik.

1.4 Tujuan

Tujuan pembuatan skripsi ini adalah :

1. Menerapkan pemodelan regresi linear dengan Algoritma Genetika pada kasus konsumsi kWh listrik di Kota Batu.
2. Mencari parameter Algoritma Genetika yang tepat untuk menyelesaikan permasalahan pemodelan regresi linear.
3. Menentukan tingkat akurasi prakiraan besarnya konsumsi kWh listrik di Kota Batu dengan algoritma yang diusulkan agar menghasilkan data yang lebih efektif.

1.5 Manfaat

Manfaat dari pembuatan skripsi ini adalah untuk membantu serta dapat memberikan solusi bagi warga Kota Batu khususnya kepada pemerintah Kota Batu dalam penggunaan kWh listrik di Kota Batu agar lebih optimal serta dapat memprakirakan pula konsumsi kWh listrik di bulan-bulan berikutnya.

1.6 Sistematika Penulisan

Sistematika penulisan tugas akhir ini dibagi menjadi enam bab dengan masing-masing bab diuraikan sebagai berikut :

BAB I : PENDAHULUAN

Meliputi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, sistematika penulisan laporan serta alokasi penjadwalan penelitian.

BAB II : KAJIAN PUSTAKA DAN DASAR TEORI

Bab ini menjelaskan teori-teori, temuan dan bahan penelitian sebelumnya yang diperoleh dari berbagai referensi yang menunjang penelitian dalam penulisan skripsi. Adapun teori yang tercakup dalam bab ini yaitu mengenai pemodelan regresi linear, definisi dan konsep Algoritma Genetika dan mengenai konsumsi daya listrik.

BAB III : METODE PENELITIAN

Bab ini berisi metode atau langkah-langkah yang akan digunakan dalam penelitian skripsi yang terdiri dari studi literatur, penyusunan dasar teori, metode pengambilan data, analisa dan perancangan sistem, pengujian rancangan aplikasi perangkat lunak yang akan dibuat, pengambilan kesimpulan, hingga penulisan laporan.

BAB IV : PERANCANGAN

Analisis kebutuhan dan perancangan aplikasi yang dibuat, meliputi deskripsi aplikasi, spesifikasi kebutuhan, dan perancangan atau desain sistem Pemodelan Regresi Linear dalam Konsumsi daya Listrik di Kota Batu Menggunakan Algoritma Genetika yang nantinya akan diimplementasikan.

BAB V : IMPLEMENTASI

Pada bagian ini akan berisi penjelasan tentang lingkungan implementasi misalnya bahasa pemrograman, sistem operasi serta

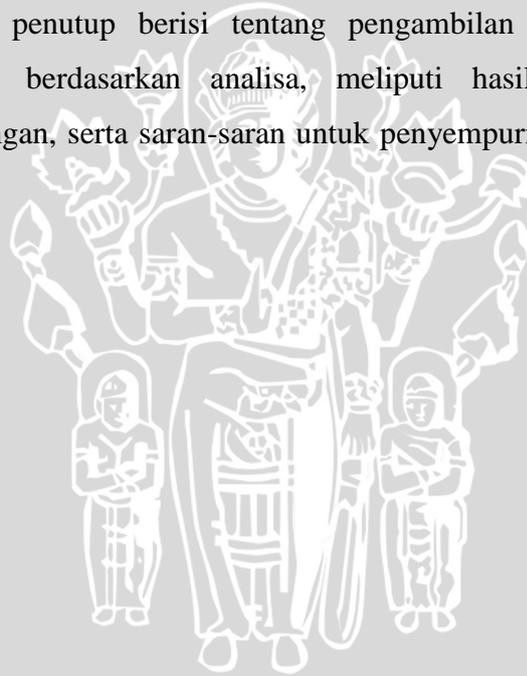
perangkat keras yang digunakan. Akan ditunjukkan pula algoritma operasi yang akan diimplementasikan pada penelitian ini.

BAB VI : PENGUJIAN DAN ANALISIS

Bab ini memuat hasil pengujian dan analisis hasil pengujian penerapan Algoritma Genetika dalam pemodelan regresi linear dalam konsumsi kWh listrik. Pengujian yang dilakukan merupakan pengujian paramater serta perbandingan dengan persamaan regresi oleh excel dengan persamaan regresi oleh sistem.

BAB VII : PENUTUP

Bagian penutup berisi tentang pengambilan kesimpulan yang diambil berdasarkan analisa, meliputi hasil, kelebihan dan kekurangan, serta saran-saran untuk penyempurnaan aplikasi yang dibuat.



BAB II

Kajian Pustaka dan Dasar Teori

2.1 Kajian Pustaka

Untuk obyek prakiraan kebutuhan daya listrik dengan model regresi telah banyak dibahas pada penelitian sebelumnya. Salah satunya adalah penelitian yang berjudul “Prakiraan Kebutuhan Daya Listrik di Kabupaten Maros Tahun 2010 Sampai 2020”. Penelitian tersebut bertujuan untuk mengetahui cara memperkirakan kebutuhan daya maksimum mulai tahun 2010 sampai dengan 2020. Permasalahan tersebut akan ditentukan apakah ada hubungan antara Y dan X, kemudian bagaimana bentuk hubungan tersebut (Sriwati, 2013). Hasil perkiraan menggunakan model regresi linear sederhana untuk memprakirakan daya listrik yang akan dibutuhkan di kabupaten Maros mulai tahun 2009 sampai tahun 2010.

Sedangkan untuk Algoritma Genetika telah banyak diterapkan dalam berbagai penelitian sebagai solusi untuk membantu menyelesaikan permasalahan optimasi, penjadwalan, maupun prediksi. Salah satunya adalah penelitian yang berjudul “Prediksi Data Harga Saham Harian Menggunakan *Feed Forward Neural Networks* (FNN) dengan Pelatihan Algoritma Genetika (Studi Kasus pada Harga Saham Harian PT. XL Axiata Tbk)”. Penelitian tersebut bertujuan untuk mengetahui cara kerja FFNN menggunakan Algoritma Genetika untuk prediksi data harga saham harian PT. XL Axiata Tbk (Sari, 2014). Proses Algoritma yang digunakan dalam penelitian tersebut dimulai dengan proses inialisasi kromosom, lalu untuk proses reproduksi terdapat dua proses genetika yaitu *crossover* dan mutasi. Proses *crossover* dilakukan pada setiap individu dengan menggunakan probabilitas *crossover* (P_c) yang dibuat secara random dalam rentang $[0, 1]$. Sedangkan untuk proses mutasi yaitu dengan mengubah nilai dari satu gen atau lebih dari suatu kromosom, terdapat probabilitas mutasi (P_m) yang ditentukan untuk mengubah nilai satu gen menjadi kebalikannya, misalkan dalam kasus *binary encoding* nilai 0 akan diubah menjadi 1 begitu juga sebaliknya. Proses terakhir dari Algoritma Genetika yaitu proses seleksi, dalam penelitian tersebut menggunakan model *tournament selection*. Hasil dari penelitian tersebut

dibuktikan dengan nilai RSME yang kecil pada metode FFNN dengan menggunakan Algoritma Genetika (Sari, 2014).

Pada penelitian ini, akan digunakan Algoritma Genetika sebagai solusi untuk menyelesaikan permasalahan dari menentukan pemodelan regresi linear dalam konsumsi kWh listrik, model regresi yang dihasilkan nantinya akan dibuat menjadi suatu persamaan untuk mencari prediksi konsumsi kWh di bulan selanjutnya. Perbandingan objek dan metode penelitian yang akan dilakukan dengan penelitian dari masing-masing referensi studi literatur ditunjukkan pada Tabel 2.1.

Tabel 2.1 Perbandingan Objek dan Metode

No	Judul	Objek	Metode	Hasil
1.	Prakiraan Kebutuhan Daya Listrik di Kabupaten Maros Tahun 2010 Sampai 2020 (Sriwati, 2013)	Daya Listrik	Model Regresi Linear	Hasil perkiraan menggunakan model regresi linier sederhana untuk memperkirakan daya listrik yang dibutuhkan mulai tahun 2009 sampai dengan 2020, laju pertumbuhannya sebesar 0,17% tiap tahunnya.
2.	Prediksi Data Harga Saham Harian Menggunakan <i>Feed Forward Neural Networks</i> (FNN) dengan Pelatihan Algoritma Genetika (Studi Kasus pada Harga Saham Harian PT. XL Axiata Tbk)	Data Harga Saham Harian	<i>Feed Forward Neural Networks</i> (FFNN) dengan Algoritma Genetika	Penerapan metode Algoritma mampu mengatasi permasalahan prediksi data harga saham harian, hal tersebut dibuktikan

	(Sari, 2014)			dengan nilai RSME paling kecil hingga mendekati nilai aktualnya.
3.	Usulan Penulis: Pemodelan Regresi Linear Berganda dalam Konsumsi kWh Listrik di Kota Batu Menggunakan Algoritma Genetika	KWh Listrik	Pemodelan Regresi Linear dengan Algoritma Genetika	Koefisien Regresi Linear Berganda dihasilkan oleh Algoritma Genetika yang optimum mampu menyelesaikan prediksi konsumsi kwh listrik.

Tabel perbandingan input, proses dan output dari refrensi-refrensi penelitian sebelumnya dengan penelitian penulis ditunjukkan pada Tabel 2.2.

Tabel 2.2 Perbandingan Input, Proses dan Output

Judul	Prakiraan Kebutuhan Daya Listrik di Kabupaten Maros Tahun 2010 Sampai 2020	Prediksi Data Harga Saham Menggunakan <i>Forward Neural Networks</i> (FNN) dengan Pelatihan Algoritma Genetika (Studi Kasus pada Harga Saham PT. XL Axiata Tbk	Usulan Penulis: Pemodelan Regresi Linear dalam Konsumsi kWh Listrik di Kota Batu Menggunakan Algoritma Genetika
Input	Data daya listrik	- Data time series harga saham dengan model FFNN - Input bobot Algotima Genetika	Algoritma Genetika : - Popsize - <i>Crossoverate</i> - <i>Mutationrate</i> - Generasi Data Konsumsi kWh

Proses	<ul style="list-style-type: none"> - Menentukan model regresi linear sederhana - Menentukan β_0 dan β_1 pada persamaan garis lurus - Substitusi β_0 dan β_1 dalam persamaan model regresi linear 	<ul style="list-style-type: none"> - Memodelkan time series dengan model FFNN yang terdapat <i>hidden layer</i>, <i>input layer</i>, fungsi aktivasi. - Dari arsitektur jaringan FFNN terbentuk jumlah bobot yang dijadikan parameter untuk proses Algoritma Genetika selanjutnya 	<ul style="list-style-type: none"> - Inisialisasi kromosom secara random - Setiap kromosom dicari <i>error</i> serta <i>fitness</i>. - Proses Reproduksi dengan <i>extended intermediate crossover</i> dan <i>random mutation</i> - Proses Evaluasi dan Seleksi dengan model <i>elitism selection</i> - Diulang sebanyak generasi yang dibutuhkan sampai mendapatkan titik kromosom yang stabil dengan nilai <i>fitness</i> terbaik.
Output	Persamaan regresi untuk mencari prakiraan daya listrik	Bobot atau parameter optimum	Model Regresi Linear untuk mencari prediksi konsumsi kWh.

2.2 Regresi Linear

Regresi merupakan suatu istilah yang dikemukakan pertama kali oleh Sir Francis Galton pada tahun 1886. Menurut Galton ada kecenderungan bahwa orang tua yang memiliki tubuh tinggi mempunyai anak yang tinggi pula dan sebaliknya orang tua yang memiliki tubuh pendek mempunyai anak-anak yang pendek pula. Karena demikian, Galton mengamati ada kecenderungan bahwa tinggi anak bergerak menuju rata-rata tinggi populasi secara keseluruhan. Dapat diartikan bahwa ketinggian anak yang mata tinggi atau orang tua yang amat pendek cenderung bergerak ke arah rata-rata tinggi populasi. Hal tersebutlah yang dikenal dengan hukum Galton mengenai regresi universal (Ervin, 1999).

Pada umumnya ada dua macam hubungan antara dua variabel atau lebih, yang biasa disebut bentuk hubungan dan keeratan hubungan. Untuk mengetahui bentuk hubungan maka digunakan analisis regresi. Sedangkan untuk keeratan hubungan dapat diketahui dengan analisis korelasi. Ada pula yang menyebutkan bahwa regresi linear merupakan suatu metode statistika yang digunakan untuk membentuk suatu model hubungan antara variabel terikat (dependen, Y) dengan satu atau lebih variabel bebas (independen, X) (Kurniawan, 2008), dengan tujuan untuk mengestimasi serta memprediksi rata-rata populasi atau nilai rata-rata variabel dependen berdasarkan nilai variabel independen yang diketahui.

Hasil analisis regresi berupa koefisien pada masing-masing variabel X (independen). Koefisien tersebut diperoleh dengan cara memprediksi nilai variabel Y (dependen) dengan suatu persamaan. Koefisien regresi dihitung dengan dua tujuan sekaligus, untuk meminimumkan penyimpangan antara nilai aktual dan nilai estimasi variabel Y (dependen) berdasarkan data yang ada (Syahputra, 2008).

2.2.1 Analisis Regresi Linear

Analisis regresi linear ini umumnya digunakan untuk peramalan, dimana dalam model terpadat variabel independen X dan variabel dependen Y. Regresi linear sendiri mempunyai tujuan untuk menentukan satu model persamaan dan garis yang menunjukkan hubungan antara kedua variabel tersebut, variabel X (independen) dan variabel Y (dependen), yang merupakan persamaan penduga untuk menafsirkan variabel Y (dependen). Analisis regresi ini terdiri dari dua model persamaan, yaitu :

1. Analisis Regresi Linear Sederhana
2. Analisis Regresi Linear Berganda

2.2.1.1 Analisis Regresi Linear Sederhana

Regresi Linear sederhana merupakan satu bentuk untuk mendapatkan hubungan matematis dalam bentuk suatu model persamaan antara variabel Y (dependen) tunggal dengan variabel X (independen) tunggal. Pada regresi linear sederhana ini hanya terdapat satu buah peubah bebas X yang dihubungkan dengan satu buah peubah tak bebas Y. Model umum persamaan regresi sederhana

menunjukkan antara dua variabel X sebagai variabel *independent* dan variabel Y sebagai variabel *dependent* (Syahputra, 2008).

Persamaan (2-1) merupakan persamaan umum persamaan regresi linear sederhana, berikut bentuk umumnya :

$$Y = a + bX \quad (2-1)$$

Dimana :

Y = nilai tafsiran variabel dependen

X = variabel independen

a = konstanta

b = koefisien regresi variabel independen

2.2.1.2 Analisis Regresi Linear Berganda

Regresi linear berganda adalah suatu analisis regresi yang menjelaskan suatu hubungan antara variabel dependen (variabel peubah respon) dengan faktor-faktor yang mempengaruhi lebih dari satu variabel independen (prediktor). Regresi linear berganda ini sebenarnya hampir sama dengan regresi linear sederhana, namun pada regresi linear berganda variabel *independent* (penduga / X) lebih dari satu. Tujuan analisis regresi linear berganda ini adalah untuk mengukur intensitas hubungan antara dua variabel atau lebih dengan memuat perkiraan nilai Y atas nilai X.

Bentuk model persamaan regresi linear berganda yang mencakup dua atau lebih variabel independen akan ditampilkan pada persamaan (2-2) (Syahputra 2008) :

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \dots + \beta_n X_n + \varepsilon \quad (2-2)$$

Dimana :

Y = Variabel dependen

$X_1, X_2, X_3, \dots, X_n$ = Variabel independen

$\beta_1, \beta_2, \beta_3, \dots, \beta_n$ = parameter koefisien regresi variabel bebas

ε = variabel kesalahan / *error*

Bentuk persamaan diatas adalah model regresi untuk populasi, apabila hanya menarik sebagian berupa sampel dari populasi secara acak dan tidak mengetahui regresi populasi sehingga model regresi populasi perlu diduga berdasarkan model regresi sampel.

Bentuk persamaan yang telah terbentuk pada persamaan (2-2) akan diformulasikan menjadi persamaan regresi linear berganda (2-3).

$$Y = a_0 + a_1X_1 + a_2X_2 + a_3X_3 + \dots + a_nX_n \quad (2-3)$$

Dimana :

Y = nilai tafsiran untuk variabel Y

a_0 = tafsiran untuk parameter konstanta β_0

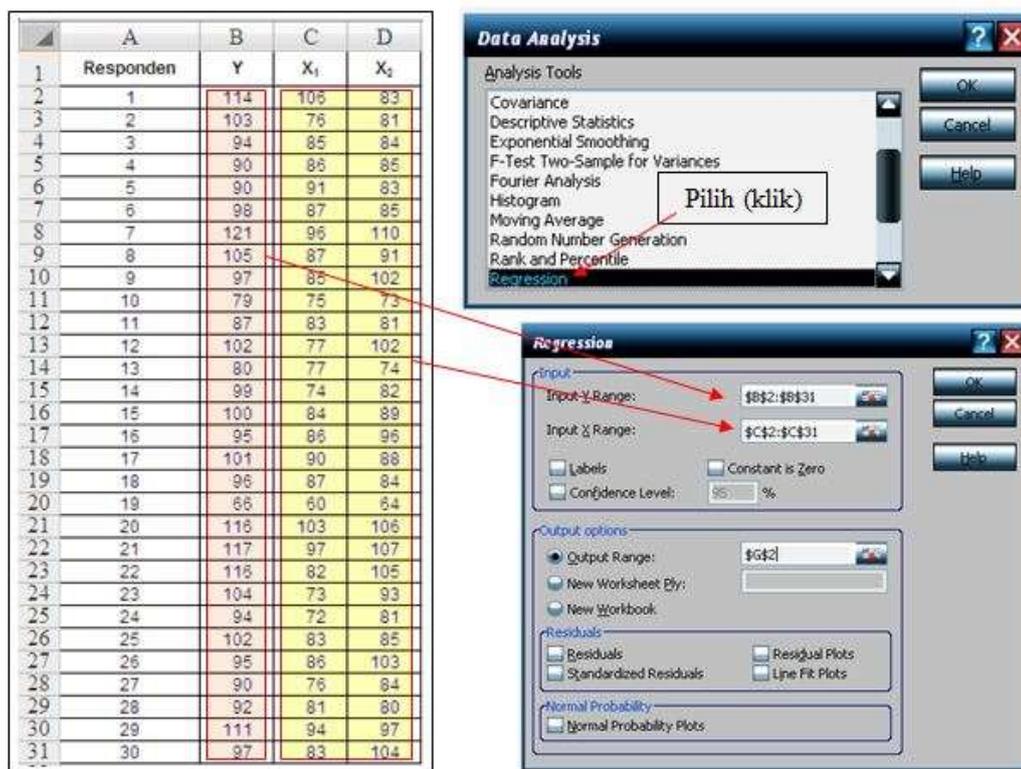
a_1, a_2, \dots, a_n = tafsiran bagi parameter koefisien regresi β_1

X = Variabel bebas (variabel *independent*)

Analisis regresi linear berganda juga dapat dihitung dengan analisis regresi pada Microsoft Excel. Fungsi regresi di microsoft excel dapat di akses dari menu dengan langkah-langkah sebagai berikut :

1. Klik Data pada menu bar
2. Pilih Data Analisis, maka akan muncul jendela *Data Analysis*.
3. Pilih *Regression*
4. Pilih OK, maka akan muncul jendela *Regression*

Untuk lebih jelasnya akan ditunjukkan pada Gambar 2.1 jendela-jendela regresi, seperti jendela *Data Analysis* dan jendela *Regression*.



Gambar 2.1 Proses Data Analisis pada Microsoft Excel

Pada jendela *Regression*: Input Y Range diambil dari data kolom B2 sampai B29 (\$B\$2:\$B\$29), input X range diambil dari kolom C2 sampai D29 (\$C\$2:\$D\$29), *Confidence Level* tetap diatur 95%, output range G2 (\$G\$32). Maka hasil regresinya akan ditunjukkan pada Gambar 2.2.

Hasil output perhitungan regresi yang digunakan selanjutnya adalah pada tabel yang berisi nilai koefisien. Kolom *coefficients* berisi nilai koefisien model hasil perhitungan (menggunakan metode *least square estimate*). Kolom *standard Error* berisi simpangan baku (standard error) perhitungan koefisien model. Kolom *t Stat* adalah t-statistic untuk koefisien model, yang dihitung dengan membagi nilai koefisien oleh standard error-nya, $t\ Stat = coefficient/standard\ error$. Kolom *P-value* berisi p-value yang terkait dengan t statistic untuk masing-masing koefisien. Parameter P-value bergantung pada t Stat dan df, yang dihitung dengan menggunakan rumus two tailed t distribution pada excel.

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0,831
R Square	0,690
Adjusted R Square	0,667
Standard Error	6,965
Observations	30

ANOVA

	df	SS	MS	F	Significance F
Regression	2	2919,097	1459,548	30,085	1,34406E-07
Residual	27	1309,870	48,514		
Total	29	4228,967			

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%
Intercept	4,67198	12,3796	0,377	0,709	-20,729	30,073
X Variable 1	0,50379	0,1603	3,142	0,004	0,175	0,833
X Variable 2	0,57431	0,1335	4,303	0,000	0,300	0,848

Gambar 2.2. Hasil Regresi pada *Data Analysis* Excel

2.2.2 Mean Squared Error (MSE)

Mean Squared Error merupakan metode analisa untuk membandingkan hasil nilai prediksi dengan nilai asli. Semakin kecil nilai MSE maka semakin kecil pula perbedaan nilai prediksi dan nilai asli. Berikut merupakan fungsi MSE akan ditampilkan pada Persamaan (2-4) :

$$MSE = \frac{1}{n} \sqrt{\sum_{i=1}^n (Y_i - Y'_i)^2} \tag{2-4}$$

Dimana :

- MSE = Nilai *error* MSE
- n = Jumlah konsumsi yang diprediksi
- Y = Harga konsumsi sebelumnya/asli
- Y' = Hasil prediksi harga konsumsi
- i = Indeks

2.3 Daya Listrik

Daya adalah nilai dari energi listrik yang dikirimkan dan didistribusikan, dimana besarnya daya listrik tersebut sebanding dengan perkalian besarnya tegangan dan arus listriknya. Sistem suplai daya listrik dapat dikendalikan oleh kualitas dari tegangan, dan tidak dapat dikendalikan oleh arus listrik sebab arus



listrik berada pada sisi beban yang bersifat individual, sehingga pada dasarnya kualitas daya merupakan kualitas dari tegangan itu sendiri.

Daya listrik merupakan suatu besarnya usaha yang dilakukan oleh sumber tegangan dalam 1 sekon. Jika dalam waktu t sekon sumber tegangan telah melakukan usaha sebesar W . Daya listrik bisa juga disebut suatu kekuatan yang dikandung dalam aliran arus dan tegangan listrik melalui hambatan dengan besaran tertentu. Satuan ukuran daya listrik adalah Watt dan mempunyai symbol P . Maka persamaan untuk mencari besarnya daya adalah (Oktyanti, 2011) :

Persamaan (2-5) merupakan rumus mencari daya.

$$P = \frac{W}{t} \quad (2-5)$$

Persamaan (2-6) merupakan rumus mencari gaya/usaha.

$$\text{Karena } W = V \cdot I \cdot t \quad (2-6)$$

Persamaan (2-7) merupakan rumus formulasi untuk mencari daya berdasarkan persamaan (2-6).

$$\text{Maka, } P = \frac{V I t}{t} = V \cdot I \quad (2-7)$$

Menurut hukum Ohm pada persamaan (2-8) :

$$V = I \cdot R, \text{ sehingga} \quad (2-8)$$

Mencari daya berdasarkan persamaan (2-8) maka dihasilkan rumus beda potensial pada persamaan (2-9)

$$P = I^2 R, \text{ dan karena } I = \frac{V^2}{R} \quad (2-9)$$

Dimana :

P = daya (joule/sekon) atau watt

W = usaha (joule)

t = waktu (sekon)

1 joule/sekon = 1 watt

V = beda potensial (volt)

I = kuat arus yang mengalir (ampere)

Misalnya, jika pada suatu peralatan listrik didapati tulisan : 220V 50 W, artinya bahwa alat tersebut dapat berkerja dengan stabil dan baik jika dipasang pada tegangan 220 V dan daya 50 watt sendiri dalam tiap sekon alat tersebut menggunakan listrik sebesar 50 joule (Oktiyanti, 2011).

2.3.1 KWH Listrik

KWh bisa juga disebut kilowatt-jam atau *killojam-hour* merupakan ukuran satuan energi listrik yang dikirim oleh perlatan elektronik yang terhubung dan membutuhkan listrik dan diberi biaya. KWh adalah produk tenaga listrik dalam satuan kallowatt dikalikan dengan waktu dalam jam, bukan kallowatt per hour (Kw per h).

Sedangkan untuk mengukur besarnya kWh pada sebuah alat elektronik bisa menggunakan kWh meter yang mengukur secara langsung hasil kali tegangan, arus faktor kerja, dikalikan waktu yang bekerja selama jangka waktu tersebut (Oktiyanti, 2011). KWh meter adalah kumparan tegangan, kumparan arus, piringan aluminium, magnet tetap yang bertugas menetralkan piringan aluminium dari induksi medan magnet dan gear mekanik yang mencatat jumlah perputaran piringan aluminium.

KWh meter analog bekerja menggunakan metode induksi medan magnet dimana medan magnet tersebut menggerakkan piringan yang terbuat dari aluminium. Putaran piringan tersebut akan menggerakkan counter digit sebagai tampilan jumlah kWhnya. Gambar kWh meter akan ditunjukkan pada Gambar 2.3.



Gambar 2.3 KWH Meter Analog

2.3.2 Perhitungan Pemakaian Energi Listrik di Rumah

Alat ukur yang biasa disebut meteran yang dipakai sebagai pengukur energi listrik yang digunakan di rumah, diukur dengan satuan kilowatt-hour atau kWh. Angka yang ditunjukkan oleh meteran itulah yang dipakai sebagai dasar pembayaran rekening listrik di rumah.

Karena satuan daya P adalah watt dan satuan waktu t adalah sekon, maka satuan usaha atau energi adalah watt-sekon atau WS. Satuan ini dianggap terlalu kecil dalam pengukuran pemakaian listrik sehari-hari. Oleh sebab itu, perlu ditentukan satuan yang lebih besar yaitu watt-hour atau Wh.

$1 \text{ Wh} = 3600\text{s}$, pada umumnya meteran pada rumah penduduk menggunakan satuan kilowatt-hour (kWh) (Bali, 2000).

$$1 \text{ kWh} = 1000 \times 3600 \text{ Ws}$$

$$1 \text{ kWh} = 3,6 \times 10^6 \text{ Ws}$$

$$1 \text{ kWh} = 3,6 \times 10^6 \text{ joule}$$

Sebagai contoh telah diamati posisi meteran bulan menunjukkan 55 341 kWh dan posisi meteran bulan ini 55 439 kWh. Berarti selama sebulan telah memakai energi listrik sebesar $55\,439 - 55\,341 = 98 \text{ kWh}$.

Contoh perhitungan lagi, Pada sebuah rumah terdapat 2 lampu TL 20 watt yang menyala 5 jam tiap hari, 5 buah lampu pijar 25 watt yang menyala 6 jam tiap hari, sebuah setrika listrik 250 watt yang menyala 2 jam tiap hari, sebuah televisi 200 watt yang menyala 10 jam tiap hari dan kipas angin 10 watt yang menyala 4 jam tiap hari. Tarif listrik adalah Rp 250,- tiap kWh, dan bea beban

tetap adalah Rp 3.500,- tiap bulan. Maka berapa rekening yang harus dibayar oleh keluarga tersebut dalam 1 bulan? (Zagy, 2010).

Jawab :

Lampu TL :

$$2 \times 20 \text{ W} \times 5 \text{ jam} \times 30 \text{ Wh} = 6000 \text{ Wh} = 6 \text{ kWh}$$

Lampu pijar :

$$5 \times 25 \text{ W} \times 6 \text{ jam} \times 30 = 22500 \text{ Wh} = 22,5 \text{ kWh}$$

Setrika :

$$1 \times 250 \text{ W} \times 2 \text{ jam} \times 30 = 15000 \text{ Wh} = 15 \text{ kWh}$$

Televisi :

$$1 \times 200 \text{ W} \times 10 \text{ jam} \times 30 = 60000 \text{ Wh} = 60 \text{ kWh}$$

Kipas angin :

$$1 \times 10 \text{ W} \times 4 \text{ jam} \times 30 = 1200 \text{ Wh} = 1,2 \text{ kWh}$$

Tarif listrik per kWh = Rp 250,-

Bea Beban tetap per bulan = Rp 3500,-

Maka jumlah energi yang harus dibayar adalah :

$$= (6 + 22,5 + 15 + 60 + 1,2) \text{ kWh}$$

$$= 104,7 \text{ kWh}$$

Jadi rekening yang harus dibayar adalah :

$$= \text{Rp } 3500,- + (104,7 \times \text{Rp } 250,-)$$

$$= \text{Rp } 3500,- + \text{Rp } 26175,-$$

$$= \text{Rp } 29675,-$$

2.4 Algoritma Genetika

John Holland adalah tokoh yang pertama kali memperkenalkan Algoritma Genetika yang berasal dari Universitas Michigan pada awal 1970 di New York, Amerika Serikat. Ia bersama murid serta rekan kerjanya menghasilkan sebuah buku yang berjudul “*Adaption in Natural and Artificial System*”. Konsep yang digunakan dalam Algoritma Genetika adalah mengikuti apa saja yang dilakukan oleh alam (Eka, 2012).

Algoritma Genetika diterapkan sebagai simulasi komputer dimana sebuah populasi representasi abstrak atau biasa yang disebut sebagai kromosom dari solusi-solusi calon individu pada sebuah masalah optimasi akan berkembang menjadi suatu solusi yang lebih baik. Prinsip kerja Algoritma Genetika meniru proses evolusi biologi, dimulai dari individu yang berperan sebagai *parent* kemudian melakukan proses reproduksi agar menghasilkan keturunan dalam ilmu Algoritma Genetika keturunan dalam proses reproduksi ini biasa disebut dengan *offspring*. *Parent* dan *offspring* tersebut berkumpul dalam suatu populasi akan melakukan sebuah evolusi dan individu yang mampu beradaptasi dengan lingkungannya akan mempunyai peluang lebih besar untuk proses seleksi alam dan bertahan hidup (Mahmudy, 2013).

Keuntungan dalam penggunaan Algoritma Genetika terlihat jelas dari kemudahan implementasi dan kemampuannya untuk menemukan solusi yang dapat diterima secara cepat dan tepat untuk permasalahan yang mempunyai kesulitan maupun kompleksitas tinggi. Algoritma Genetika sangat cocok untuk permasalahan dengan karakteristik sebagai berikut (Eka, 2012) :

1. Ruang masalah yang kompleks dan sulit dipahami.
2. Kurangnya pengetahuan yang memadai untuk merepresentasikan masalah ke dalam ruang pencarian yang lebih sempit.
3. Tidak tersedianya analisis matematika yang memadai.
4. Metode-metode konvensional sudah tidak mampu menyelesaikan masalah yang dihadapi.
5. Solusi yang diharapkan tidak harus paling optimal, namun cukup bagus untuk bisa diterima.

6. Terdapat batasan waktu, misalnya *real time series*.

Beberapa permasalahan yang dapat diselesaikan dengan Algoritma Genetika antara lain (Tanjung, 2010) :

1. Optimasi

Permasalahan optimasi yang dapat diselesaikan dengan Algoritma Genetika yaitu optimasi numerik dan optimasi kombinatorial contohnya *Traveling Salesme Problem*.

2. Pemrograman Otomatis

Pemrograman otomatis dalam Algoritma Genetika telah digunakan untuk melakukan proses evolusi terhadap program komputer untuk merancang sebuah struktur komputasional, misalnya *cellular automata* dan *sorting network*.

3. *Machine Learning*

Algoritma Genetika tidak hanya berhasil dalam mengaplikasikan suatu perancangan jaringan syaraf tiruan atau *neural networks* untuk melakukan proses evolusi terhadap aturan-aturan pada *learning classifier system*, namun Algoritma Genetika juga dapat digunakan untuk mengontrol robot.

4. Model Ekonomi

Algoritma Genetika juga dapat dipakai untuk memodelkan proses-proses inovasi dan pembangunan strategi penawaran atau disebut *bidding strategies*.

5. Model Sistem Imunisasi

Algoritma Genetika berhasil digunakan dalam memodelkan berbagai macam aspek pada sistem imunisasi alamiah, termasuk *somatic mutation* selama kehidupan individu dan menemukan keluarga baru dengan gen ganda atau disebut dengan *multi-gene families* sepanjang waktu evolusi.

6. Model Ekologis

Algoritma Genetika berhasil dalam memodelkan fenomena ekologis seperti *host-parasite co-evolutions*, simbiosis, dan aliran sumber daya dalam ekologi.

2.4.1 Proses pada Algoritma Genetika

Algoritma Genetika merupakan suatu algoritma pencarian yang berdasarkan pada mekanisme sistem natural yaitu genetik dan seleksi alam. Di dalam suatu sistem Algoritma Genetika, variabel solusi dikodekan ke dalam suatu struktur *string* yang merepresentasikan barisan gen, yang akan menjadikan sebuah karakteristik dari solusi sebuah permasalahan.

Algoritma Genetika adalah algoritma untuk pencarian hasil yang terbaik, yang berdasarkan pada proses reproduksi atau perkawinan dan proses seleksi gen secara *natural*, kombinasi individu-individu untuk melakukan proses reproduksi dibangkitkan secara acak (*random*), yang nantinya struktur gen dari hasil proses reproduksi tersebut menghasilkan suatu gen yang inovatif untuk proses selanjutnya yakni proses seleksi.

Setiap generasi dari hasil reproduksi akan diperoleh bagian-bagian gen induk yang terbaik. Diharapkan dengan mengambil gen induk yang terbaik ini nantinya akan diperoleh gen *offspring* atau anak yang lebih baik. Namun, pada kenyataannya dalam proses reproduksi tersebut tidak selalu menghasilkan keturunan yang lebih baik dari induknya. Selalu ada kemungkinan hasil keturunan tersebut lebih baik, sama baiknya, atau bahkan bisa lebih buruk.

Tujuan utama dari algoritma genetika adalah dapat menghasilkan suatu populasi yang terbaik dari populasi sebelumnya. Sedangkan manfaat Algoritma Genetika ini adalah metode pencariannya yang lebih efektif dan optimal, tanpa perlu memperbesar lingkup pencarian.

Berikut *pseudocode* tahapan proses yang diperlukan dalam Algoritma Genetika dimulai dari inialisasi kromosom sampai proses seleksi (Mahmudy, 2013) yang ditunjukkan pada Gambar 2.2.

```

Procedure Algoritma Genetika
begin
  t = 0
  inialisasi P(t)

  while (bukan kondisi berhenti) do
    reproduksi C(t) dari P(t)
    evaluasi P(t) dan C(t)
    seleksi P(t+1) dari P(t) dan C(t)
    t = t + 1
  
```

```
end while  
end
```

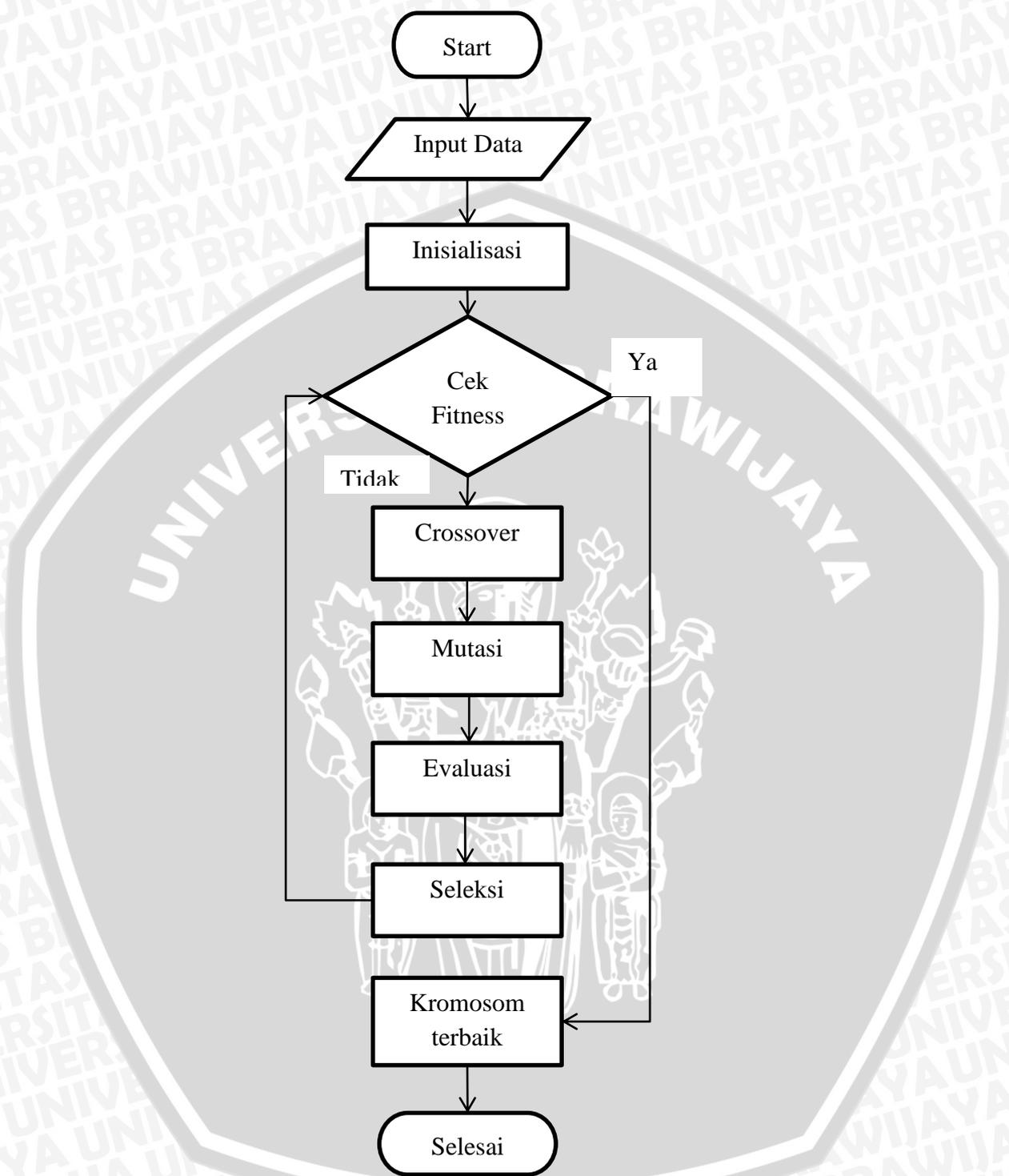
Gambar 2.2 Pseudocode Algoritma Genetika
Sumber : (Mahmudy, 2013)

Pada Algoritma Genetika ini terdapat beberapa definisi umum yang digunakan dan harus dipahami sebelumnya, antara lain (Mahmudy dalam Tanjung, 2013):

1. Gen, merupakan nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan yang disebut sebagai kromosom.
2. Kromosom/individu, merupakan gabungan dari beberapa gen yang membentuk suatu nilai tertentu dan menyatakan solusi yang mungkin dari suatu permasalahan yang ada.
3. Populasi, merupakan sekumpulan individu yang nantinya akan diproses bersama dalam satu satuan siklus evolusi
4. *Fitness*, menyatakan seberapa baik nilai dari suatu individu yang didapatkan.
5. *Seleksi*, merupakan proses untuk mendapatkan calon induk yang baik.
6. *Crossover*, merupakan proses pertukaran atau kawin silang antara gen-gen dari dua induk tertentu.
7. Mutasi, merupakan proses pergantian salah satu gen yang terpilih dengan nilai tertentu.
8. Generasi, merupakan urutan iterasi dimana beberapa kromosom bergabung.

Offspring, merupakan anak dari perkawinan individu membentuk individu baru atau kromosom baru yang dihasilkan.

Berikut *flowchart* Algoritma Genetika yang untuk memperjelas alur proses serta tahapan-tahapan yang harus dilakukan, akan ditunjukkan pada Gambar 2.3.



Gambar 2.3 Flowchart Proses Algoritma Genetika
 Sumber : (Mahmudy, 2013)

Proses Algoritma Genetika diawali dengan insialisasi atau sering disebut sebagai representasi kromosom, yaitu menciptakan individu-individu baru secara *random* yang memiliki susunan gen atau kromosom tertentu, kromosom tersebut nantinya akan menjadi sebuah solusi dari permasalahan yang akan diselesaikan. Setelah representasi kromosom, kemudian proses perkawinan antar gen atau reproduksi, hal tersebut dilakukan untuk menghasilkan keturunan atau *offspring* dari *parent* atau individu yang ada di dalam populasi. Terdapat dua proses reproduksi dalam Algoritma Genetika yaitu perkawinan silang atau *crossover* dan proses mutasi (Mahmudy, 2013).

Setelah diperoleh hasil *offspring* kemudian dilakukan evaluasi, yaitu untuk menghitung nilai *fitness* dari setiap kromosom baru, evaluasi disini tidak hanya *offspring* saja namun kromosom induk juga masuk dalam evaluasi. Semakin besar nilai *fitness* maka semakin baik pula kromosom tersebut untuk dijadikan solusi. Proses terakhir pada Algoritma Genetika adalah Seleksi, yaitu proses untuk memilih individu terbaik dari himpunan populasi beserta *offspring* yang nantinya akan dijadikan induk untuk generasi berikutnya sampai menemukan hasil yang konvergen yang dirasa menjadi solusi terbaik untuk permasalahan yang diciptakan. Individu maupun *offspring* yang mempunyai nilai *fitness* yang lebih besar, maka akan besar pula kemungkinan untuk terpilih (Mahmudy, 2013).

2.4.2 Representasi Kromosom

Kromosom tersusun dari sejumlah gen yang merepresentasikan variabel-variabel solusi. Representasi kromosom pada Algoritma Genetika disertain dengan fungsi *fitness* yang menyatakan kebaikan dari solusi.

Beberapa bentuk representasi kromosom yaitu representasi biner, integer, *real code* dan permutasi (Mahmudy, 2013). Pada pemodelan regresi linear merupakan salah satu permasalahan yang menggunakan *real code*, sebab pada permasalahan ini terdapat optimasi fungsi yang kompleks dan membutuhkan banyak generasi. Contoh representasi kromosom menggunakan *real code* akan ditunjukkan pada Gambar 2.4.

Kromosom				
1	2	3	4	5
-38,3554	26,6219	77,5291	-86,8919	-70,9746

Gambar 2.4 Representasi Real Code

2.4.3 Reproduksi

Proses reproduksi dalam Algoritma Genetika merupakan suatu proses untuk membentuk individu baru. Reproduksi dalam Algoritma Genetika menggunakan proses *crossover* dan mutasi. Proses *crossover* yang dipilih menggunakan metode *extended intermediate crossover* yaitu memilih dua induk secara acak dari populasi yang terbentuk kemudian menghasilkan anak dari kombinasi nilai dua induk. Misalkan P_1 dan P_2 adalah dua kromosom yang terpilih untuk melakukan proses *crossover*, maka *offspring* (anak) bisa dibangkitkan dengan rumus pada Persamaan (2-10) dan (2-11) (Mahmudy,2013).

$$C_1 = P_1 + \alpha(P_2 - P_1) \quad (2-10)$$

$$C_2 = P_2 + \alpha(P_1 - P_2) \quad (2-11)$$

α dipilih secara acak pada interval yang telah ditentukan.

Sedangkan pada proses mutasi menggunakan model *random mutation* yaitu memilih satu induk secara acak dari suatu populasi dengan cara menambah atau mengurangi gen terpilih. Nilai gen *offspring* bisa dibangkitkan dengan rumus pada Persamaan (2-12) (Mahmudy, 2013).

$$x'_i = x_i + r(\max_i - \min_i) \quad (2-12)$$

dengan r dibangkitkan secara acak pada interval yang telah ditentukan.

2.4.4 Fungsi Fitness

Fungsi *fitness* digunakan untuk mengukur kebaikan solusi yang dibawa oleh suatu individu. Individu terbaik dengan nilai *fitness* yang tinggi cenderung menghasilkan solusi yang terbaik. Fungsi *fitness* yang digunakan untuk masalah pemodelan regresi linear pada konsumsi kWh ditunjukkan pada Persamaan (2-13).

$$fitness = \frac{c}{error} \quad (2-13)$$

C adalah *constant* yang ditentukan sebagai pembanding untuk menghindari pembagian oleh nilai nol. *Error* merupakan nilai yang diperoleh dari hasil perhitungan regresi pada sistem dengan nilai aktual, penjelasan *error* sudah dijelaskan pada Subab 2.1.2.

2.4.5 Seleksi

Berdasarkan siklusnya seleksi dalam Algoritma Genetika terdapat banyak model yaitu *elitsm selection*, *roulette wheel*, *binary tournament*, *replacement selection* dan masih banyak lagi. Pada penelitian ini metode seleksi yang digunakan adalah *elitsm selection* yaitu dilakukan dengan menurutkan individu-individu yang memiliki nilai *fitness* tinggi sampai ke rendah. *Elitism selection* menjamin individu yang lolos seleksi adalah individu dengan *fitness* terbaik (Mahmudy, 2013). Individu yang lolos seleksi akan menjai *parent* pada generasi berikutnya. *Pseudocode elitism selection* ditunjukkan pada Gambar 2.5.

```
PROCEDURE ElitismSelection
Input:
    POP: himpunan individu pada populasi
    pop_size: ukuran populasi
    OS: himpunan individu anak (offspring) hasil
        reproduksi menggunakan crossover dan mutasi
Output :
POP: himpunan individu pada populasi setelah proses
seleksi selesai
/* gabungkan individu pada POP dan OS ke dalam TEMP */
TEMP <- Merge (POP,OS)
/* urutkan individu berdasarkan fitness secara ascending */
OrderAscending (Temp)
/* copy pop_size individu terbaik ke POP */
POP <- CopyBest (Temp, pop_size)
END PROCEDURE
```

Gambar 2.5 *Pseudocode Elitism Selection*
Sumber : (Mamudy, 2013)

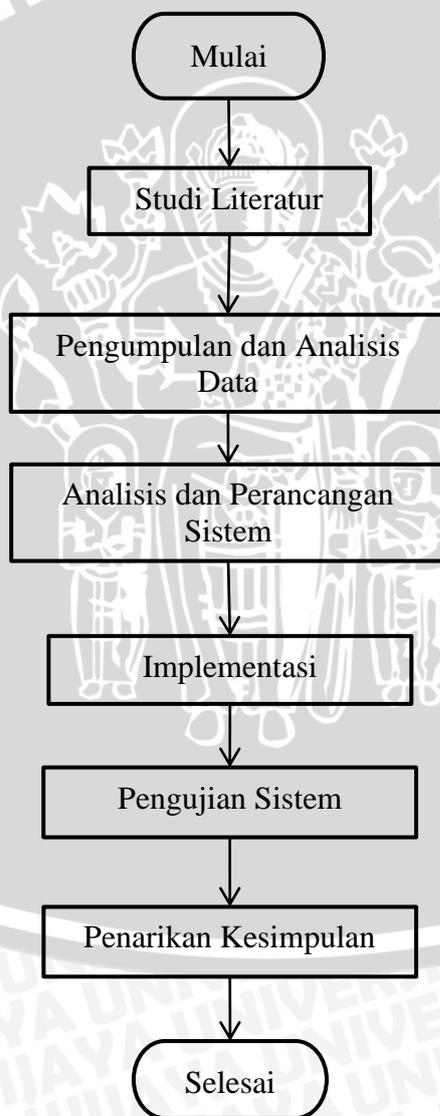
BAB III

METODE PENELITIAN

Dalam bab ini akan dijelaskan tahapan susunan kegiatan penelitian untuk membangun sebuah sistem Algoritma Genetika secara terstruktur serta perancangan sistem yang baik.

3.1 Tahapan Penelitian

Untuk dapat memberikan kemudahan dalam menjelaskan metode penelitian yang digunakan, maka diberikan diagram alir yang ditunjukkan pada Gambar 3.1.



Gambar 3.1 Diagram Alir Metode Penelitian

Berdasarkan Gambar 3.1 akan dijabarkan tiap langkah-langkah pada penelitian ini sebagai berikut :

1. Studi Literatur

Studi literatur merupakan penelusuran literatur yang bertujuan untuk mempelajari tentang penjelasan dasar teori yang digunakan serta beberapa kajian pada penelitian sebelumnya untuk menunjang penulisan skripsi. Adapun literatur yang terkait pada penelitian ini adalah Algoritma Genetika pada pemodelan regresi linear dengan data konsumsi daya listrik sebagai objek penelitian. Penulisan literatur dapat bersumber dari buku, media, artikel, pakar maupun hasil penelitian orang lain. Penyusunan dasar teori dilakukan setelah mendapatkan referensi yang tepat untuk mendukung penulisan penelitian ini.

2. Pengumpulan dan Analisis Data

Mengumpulkan data konsumsi kWh listrik yang ber sumber langsung dari PLN Rayon Batu-Malang. Data yang diperoleh berupa data mentah konsumsi daya kWh tiap bulannya pada tahun 2012.

3. Analisis dan Perancangan Sistem

Menganalisis serta merancang sistem Algoritma Genetika pada pemodelan regresi linear.

4. Implementasi

Mengimplementasikan hasil analisis dan perancangan sistem dengan membangun sebuah perangkat lunak dengan bahasa pemrograman C#.

5. Pengujian sistem

Melakukan Pengujian hasil data konsumsi kWh listrik terhadap sistem yang sudah dibangun. Pengujian yang akan digunakan yaitu pengujian *BlackBox* dan perhitungan manual. Pengujian *BlackBox* adalah pengujian fungsionalitas aplikasi untuk mengetahui apakah aplikasi telah berjalan dengan benar sesuai dengan parameter yang telah di

6. Penarikan Kesimpulan

Melakukan evaluasi dan analisis hasil pengujian terhadap sistem kemudian ditarik dalam sebuah kesimpulan serta saran yang berkenaan dengan hasil

yang telah dicapai guna memperbaiki kesalahan untuk pengembangan lebih lanjut.

3.2 Analisis Kebutuhan Sistem

Analisis kebutuhan bertujuan untuk mendapatkan kebutuhan yang diperlukan untuk sistem pemodelan regresi linear dengan Algoritma Genetika dengan obyek konsumsi daya listrik di Kota Batu. Analisis kebutuhan diterapkan sesuai dengan lokasi penelitian, variabel penelitian dan mempersiapkan kebutuhan penelitian.

Spesifikasi lingkungan implementasi yang digunakan dalam pembuatan sistem regresi linear dengan Algoritma Genetika diantaranya :

1. Spesifikasi Kebutuhan *Hardware*

- Laptop Toshiba C-40 A core i5
- RAM 2 GB
- DDR3
- VGA 1 GB

2. Spesifikasi Kebutuhan *Software*

- Microsoft windows 8.0 sebagai sistem operasi yang digunakan
- Microsoft office 2010 sebagai pengolah dokumentasi dan manual perhitungan
- C# merupakan bahasa pemrograman untuk menjalankan aplikasi developer
- Menggunakan Visual Studio 2013 sebagai software untuk membuat aplikasi
- *Enterprise Architecture* sebagai aplikasi untuk membuat diagram

3.3 Formulasi Permasalahan

Formulasi permasalahan dalam penelitian ini akan dijelaskan kasus yang terkait beserta komputasi perhitungannya.

3.3.1 Deskripsi Umum Sistem

Sistem ini memerlukan data berupa tabel konsumsi daya listrik dalam satuan kWh meter serta beberapa atribut yang mempengaruhi sedikit banyaknya

kWh listrik yang terpakai. Sistem ini memerlukan beberapa *inputan* dari *user* yaitu *popSize*, banyaknya generasi, *crossover rate* dan *mutation rate* serta dapat load data konsumsi kWh. Pada inputan yang telah dimasukkan, sistem akan mengolah data tersebut dalam suatu Algoritma Genetika yang nantinya akan menciptakan persamaan regresi baru untuk menghasilkan prakiraan konsumsi kWh listrik agar nantinya dapat digunakan secara optimal. Dari sisi regresi linear dibutuhkan beberapa variabel x dan variabel y , setelah itu diolah dari sisi Algoritma Genetika, ditentukan terlebih dahulu berapa kromosom atau gen dalam suatu populasi (*popSize*), setelah itu dibangkitkan kromosom dari *popSize* yang dibentuk, kemudian dilakukan proses reproduksi untuk menghasilkan keturunan baru diperlukan *crossover rate* (cr) dan *mutation rate* (mr), serta ditentukan pula jumlah generasinya. Sistem diharapkan dapat memberikan keluaran berupa solusi yaitu persamaan regresi yang efisien terhadap permasalahan pemodelan regresi linear terhadap konsumsi kWh listrik.

3.3.2 Data yang Digunakan

1. Data yang digunakan adalah data pemakaian kWh PLN Rayon Kota Batu pertahun 2012 sampai tahun 2014, yang diperoleh dari data real perusahaan PLN Rayon Kota Batu (terlampir).
2. Paramater harga kWh yang dibedakan menjadi 3 golongan, yaitu golongan rumah tangga, golongan industri dan bisnis, serta golongan pemerintahan (data harga terlampir).
3. Parameter Algoritma Genetika yang diperlukan berupa jumlah populasi (*popSize*), jumlah generasi, *crossover rate* (cr), dan *mutation rate* (mr).

3.3.3 Contoh Persoalan

Dimisalkan seorang petugas PLN ingin memprediksi pemakaian kWh di Kota Batu dengan berdasarkan data 3 tahun terakhir. Maka untuk proses perhitungan regresi adalah sebagai berikut :

1. Penentuan pemakaian kWh listrik berdasarkan periode per-tahun

Sebelum data pemakaian kWh dihitung menggunakan fungsi regresi perlu diketahui bahwa data dengan periode tertentu (pertahun) (x) membutuhkan data sebanyak lebih dari 1 periode. Sehingga, dengan data pemakaian listrik 3 tahun

terakhir (3x) membutuhkan data dengan periode yang sama sebanyak 3 macam.

Data pemakaian konsumsi selama 3 tahun akan ditampilkan pada Tabel 3.1.

Tabel 3.1 Pencapaian Bulanan Per 3 Tahun Terakhir (data dibuat dalam satuan juta kWh)

THN	PENCAPAIAN PERBULAN DALAM 3 TAHUN TERAKHIR											
	JAN	FEB	MAR	APR	MEI	JUNI	JULI	AGS	SEPT	OKT	NOV	DES
2012	8967	9219	8890	8715	9074	9165	9526	9344	9653	9449	9509	9816
2013	9310	9508	9450	10061	9920	10294	10134	10283	10173	10156	10407	10580
2014	10693	10315	9824	10252	10947	10728	11043	10881	10487	10936	10779	11354

Pada Tabel 3.1 merupakan Tabel pemakaian kWh dalam 3 tahun terakhir, untuk menentukan prediksi pemakaian pada bulan berikutnya maka akan diformulasikan ke dalam Tabel 3.2.

Tabel 3.2 Pemakaian kWh

No	Tahun	Bulan	Pemakaian
1	2012	6	9165
2	2012	7	9526
3	2012	8	9344
4	2012	9	9653
5	2012	10	9449
6	2012	11	9509
7	2012	12	9816
8	2013	6	10294
9	2013	7	10134
10	2013	8	10283
11	2013	9	10173
12	2013	10	10156
13	2013	11	10407
14	2013	12	10580
15	2014	6	10728
16	2014	7	11043
17	2014	8	10881
18	2014	9	10487
19	2014	10	10936
20	2014	11	10779
21	2014	12	11354

Berdasarkan Tabel 3.2 selanjutnya diformulasikan pada Tabel 3.3 untuk mendapatkan prediksi pemakaian pada bulan berikutnya berdasarkan pemakaian 4 bulan ke belakang. Permasalahan tersebut diformulasikan pada Tabel 3.3

Tabel 3.3 Data Pemakaian kWh Berdasarkan Pemakaian 4 Bulan Kebelakang

No	Bln/Thn	Y	X1	X2	X3	X4
1	Mei-14	10947	10252	9824	10315	10963
2	Jun-14	10728	10947	10252	9824	10315
3	Jul-14	11043	10728	10947	10252	9824
4	Agu-14	10881	11043	10728	10947	10252
5	Sep-14	10487	10881	11043	10728	10947
6	Okt-14	10936	10487	10881	11043	10728
7	Nov-14	10779	10936	10487	10881	11043
8	Des-14	11357	10779	10936	10487	10881

2. Persamaan Fungsi Regresi

Pada data yang ditunjukkan pada Tabel 3.3 akan dilakukan pembentukan fungsi regresi menggunakan Persamaan (2-2) yang dijabarkan pada bab sebelumnya, maka pembentukan regresi yang terbentuk dari Tabel 3.3 yaitu ditunjukkan pada persamaan (3-1).

$$Y' = 15261 - 0,42 X1 + 0,18 X2 - 0,08 X3 - 0,08 X4 \quad (3-1)$$

Koefisien-koefisien regresi yang terbentuk dapat dicari menggunakan formula data analisis regresi linear pada Microsoft Excel.

Dari Persamaan 3.1 bisa dihitung prediksi pemakaian listrik pada bulan Januari 2015 dengan memasukkan data pemakaian pada bulan Oktober, November, Desember 2014. Contoh perhitungan diolah dengan persamaan 3.1.

Menghitung prediksi harga menggunakan Persamaan 3.1 dengan nilai X menggunakan nilai di Bulan November

$$\begin{aligned} Y' &= 15261 - 0,42 X1 + 0,18 X2 - 0,08 X3 - 0,08 X4 \\ &= 15261 - 0,42 (10936) + 0,18 (10487) - 0,08 (10881) - 0,084 \\ &\quad (11043) \\ &= 10802 \end{aligned}$$

3.4 Siklus Penyelesaian Pemodelan Regresi Linear menggunakan Algoritma Genetika

Pada subab ini akan dibahas proses pemodelan regresi linear pada konsumsi kWh listrik dengan menggunakan Algoritma Genetika. Proses

pemodelan regresi dengan Algoritma Genetika akan dideskripsikan pada *pseudocode* Gambar 3.2.

```

Procedure Pemodelan Regresi Linear dengan Algoritma Genetika
begin
  t = data hystory kWh
  inisialisasi parameter
  pemodelan regresi
  inisialisasi P(t)
  while (bukan kondisi stop) do
    reproduksi C(t) dari P (t)
      crossover
      mutasi
    evaluasi P(t) dan C(t)
    seleksi P(t+1) dari P(t) dan C(t)
    t = t + 1
  end while
  prediksi regresi linear baru
end

```

Gambar 3.2 *Pseudocode* Sistem

Berikut penjabaran *pseudocode* yang telah di buat :

1. Data historis pemakaian kWh listrik berdasarkan 3 tahun terakhir terhitng dari tahun 2012 sampai 2014.
2. Inisialisasi Parameter awal :
 - Jumlah ukuran populasi (*popSize*)
 - Jumlah generasi
 - *Crossover rate* (*cr*)
 - *Mutation rate* (*mr*)
3. Data historis pemakaian kWh diolah dalam bentuk fungsi regresi linear
4. *Generate* populasi awal sebanyak jumlah populasi yang telah ditentukan
5. Membuat populasi baru dengan menggunakan proses berikut sebanyak jumlah generasi yang ditentukan :
 - Melakukan proses reproduksi dengan proses crossover, metode yang digunakan adalah *extended intermediate crossover* berdasarkan *cr* yang telah ditentukan sebelumnya.
 - Proses mutasi dengan metode *random mutation* berdasarkan *mr* yang telah ditentukan pula sebelumnya.
 - Masing-masing kromosom dihitung nilai *fitness*nya
 - Proses evaluasi dari individu awal (*parent*) dan *offspring*

- Melakukan proses seleksi dengan metode *elitism selection* untuk menentukan individu sebanyak ukuran populasi awal pada generasi selanjutnya.
- 6. Hasil akhir adalah kromosom terbaik pada akhir generasi, serta menghasilkan regresi linear yang baru.

Pada setiap proses yang dideskripsikan pada *pseocode* sebelumnya terdapat *predefined process* dimana proses tersebut akan dijabarkan lebih detail dan disertai dengan perhitungan manualnya. Berikut insialisasi parameter awalnya :

- Jumlah Generasi = 1
- Ukuran Populasi (*popSize*) = 10
- *Crossover rate (cr)* = 0,6
- *Mutation rate (mr)* = 0,4

3.4.1 Representasi Kromosom, Mencari *Error* dan Perhitungan *Fitness*

Kromosom tersusun atas bilangan real yang menyatakan koefisien regresi nilai pemakaian kWh listrik bulan-bulan sebelumnya. Panjang kromosom dalam 1 individu sama dengan variabel yang dibuat sebelumnya yaitu 5. Pada Tabel 3.3 index ke-0 hingga ke-4 menyatakan periode pemakaian kWh. Dalam Tabel 3.3 index ke-0 menyatakan variabel a pada persamaan regresi, index ke-1 pemakaian kWh bulan sebelumnya, dan seterusnya sampai index ke-4. Dilanjutkan pada data berikutnya dihitung pemakaian tiap bulannya. Detail pembentukan kromosom akan ditampilkan pada Tabel 3.4.

Tabel 3.4 Pembentukan Kromosom

Kromosom				
1	2	3	4	5
a	b_1	b_2	b_3	b_4
-38,3554	26,6219	77,5291	-86,8919	-70,9746

Sebelum menghitung nilai *fitness*, akan dihitung terlebih dahulu prediksi pemakaian kWh listrik sebelumnya menggunakan fungsi regresi linear yang

dibentuk dari nilai-nilai kromosom pada Tabel 3.4. , lalu menghitung tingkat *error* yang menggunakan Persamaan (2-4).

1. Menghitung Prediksi pemakaian kWh dengan Fungsi Regresi menggunakan nilai-nilai yang ada pada kromosom pada Tabel 3.4

$$Y' = a + b_1 10252 + b_2 9824 + b_3 10315 + b_4 10963$$

$$\begin{aligned} Y' &= (-38,3554) + (26,6219)10779 + (77,5291)10936 + \\ &= (-86,8919)10487 + (-70,9746)10881 \\ &= -639849 \end{aligned}$$

Nilai yang dimasukkan merupakan pemakaian kWh 4 Bulan ke belakang mulai bulan Desember, November, Oktober dan September. Dengan persamaan di atas menghasilkan prediksi untuk Bulan Januari 2015.

2. Menghitung Nilai *Error* mengacu pada persamaan MSE yang telah dijelaskan pada bab sebelumnya yaitu Persamaan (2-4)

$$\begin{aligned} \varepsilon &= \frac{1}{n} \sqrt{\sum_{i=1}^n (Y_i - Y'_i)^2} \\ &= \frac{1}{8} \{ (10974 - (-639849))^2 + (10728 - (-499509))^2 + (11043 - (-453798))^2 \\ &\quad + (10881 - (-553158))^2 + (10487 - (-563347))^2 + (10936 - (-598223))^2 + \\ &\quad (10779 - (-625097))^2 + (11357 - (-548733))^2 \} \\ &= \frac{1}{8} \sqrt{2,63649E + 12} \\ &= 202965,9824 \end{aligned}$$

Tabel 3.5 Hasil Perhitungan Konsumsi kWh dan Nilai Error

No	Bln/Thn	Y	X1	X2	X3	X4	Y'	error
1	Mei-14	10947	10252	9824	10315	10963	-639849	4,2354E+11
2	Jun-14	10728	10947	10252	9824	10315	-499509	2,6034E+11
3	Jul-14	11043	10728	10947	10252	9824	-453798	2,1608E+11
4	Agu-14	10881	11043	10728	10947	10252	-553158	3,1814E+11
5	Sep-14	10487	10881	11043	10728	10947	-563347	3,2929E+11
6	Okt-14	10936	10487	10881	11043	10728	-598223	3,7107E+11
7	Nov-14	10779	10936	10487	10881	11043	-625097	4,0434E+11
8	Des-14	11357	10779	10936	10487	10881	-548733	3,137E+11

$$\varepsilon = \frac{1}{n} \sqrt{\sum_{i=1}^n (Y_i - Y'_i)^2} =$$

202965,982

3. Menghitung Nilai *Fitness*

$$\begin{aligned} \text{fitness} &= \frac{1000000}{\varepsilon} \\ &= \frac{1000000}{202965,982} = 4,926934003 \end{aligned}$$

Hasil perhitungan *fitness* dari pembentukan kromosom sebelumnya dapat dilihat pada Tabel 3.6.

Tabel 3.6 Hasil Perhitungan *Fitness*

Kromosom					Fitness= (1/error)
1	2	3	4	5	
<i>a</i>	<i>b₁</i>	<i>b₂</i>	<i>b₃</i>	<i>b₄</i>	
-38,3554	26,6219	77,5291	-86,8919	-70,9746	4,92693

3.4.2 Inisialisasi Populasi Awal

Inisialisasi populasi awal merupakan pembentukan kromosom-kromosom sebanyak jumlah populasi yang telah ditentukan pada awal pembuatan paramater. Ukuran populasi atau sering disebut dengan *popSize* pada permasalahan kali ini dibuat sebanyak 10 populasi yang dibangkitkan secara *random* dalam rentang [-100,100]. Representasi kromosom awal akan ditunjukkan pada Tabel 3.7.

Tabel 3.7 Data Populasi Awal

	<i>a</i>	<i>b₁</i>	<i>b₂</i>	<i>b₃</i>	<i>b₄</i>
P1	-38,3554	26,6219	77,5291	-86,8919	-70,9746
P2	92,7409	50,8185	37,0315	-76,3115	-48,4821
P3	95,2609	68,5685	35,7244	-28,7652	39,1729
P4	93,8507	98,2261	-55,7109	-82,6776	70,6452
P5	-53,7538	30,9384	42,8046	54,6096	-0,6941
P6	-53,6517	-55,0838	79,0415	-24,9051	-21,4002
P7	-86,9893	30,3916	13,5877	-51,5967	76,7167
P8	38,7366	-24,5091	-33,3793	-99,5021	-13,5475
P9	-76,0795	2,7792	-49,3002	-56,5199	32,7695
P10	66,2390	-66,4502	57,1592	51,9038	47,8381

3.4.3 Proses Reproduksi

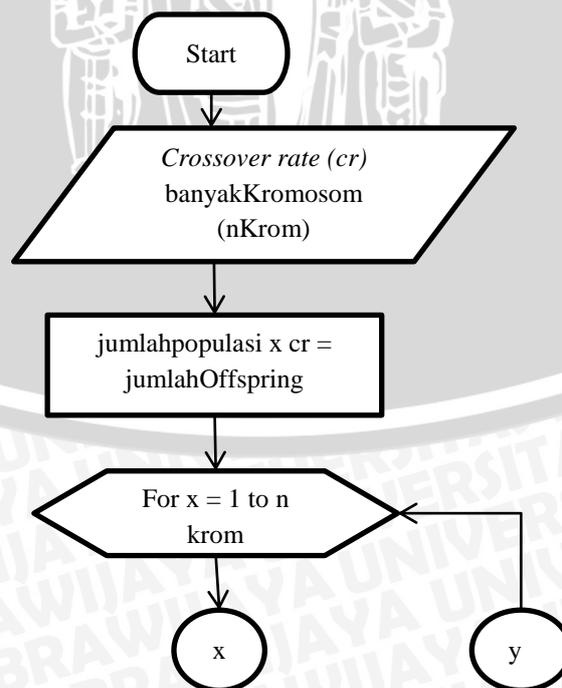
Proses reproduksi merupakan proses individu untuk menghasilkan keturunan baru, pada proses reproduksi dibagi lagi menjadi dua metode yaitu kawin silang atau *crossover* dan mutasi, kedua metode tersebut akan menghasilkan anak atau *offspring*.

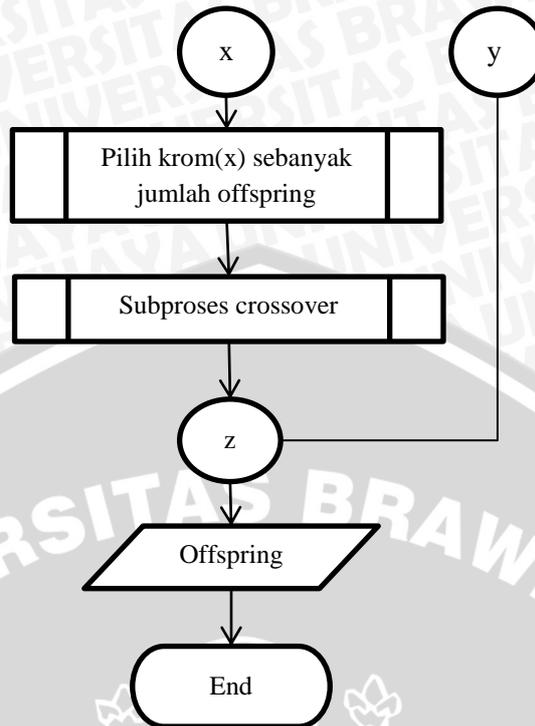
3.4.3.1 Pindah Silang (*Crossover*)

Pada kasus ini metode *crossover* yang dipilih adalah metode *extended intermediate crossover*, yaitu dengan memilih dua induk secara *random* dari populasi yang telah dibentuk dan nantinya akan menghasilkan anak (*offspring*) dari kombinasi kedua induk tersebut (Mahmudy, 2013).

Populasi awal sebanyak 10 individu dengan *crossover rate* (*cr*) sebesar 0,6 maka *offspring* yang dihasilkan adalah $0,6 \times 10 = 6$. Akan dijelaskan proses *crossover* dalam diagram alir *flowchart* pada Gambar 3.3.

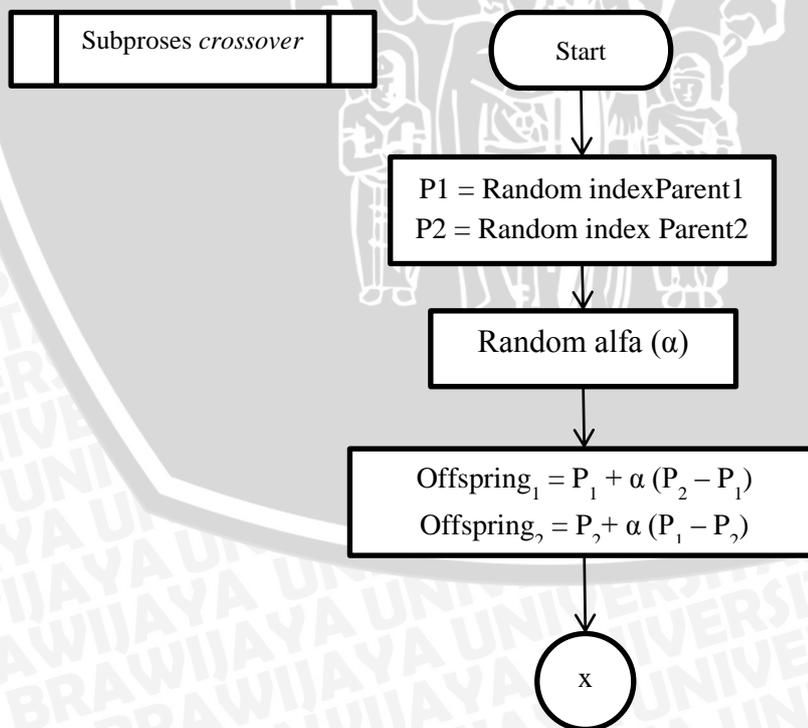
Pada *flowchart* yang ditunjukkan dalam Gambar 3.3 terdapat subproses yaitu subproses *crossover*. Pada diagram alir Gambar 3.4 akan dijelaskan subproses *crossover* yaitu metode *extended intermediate crossover* yang digunakan dalam penelitian ini.

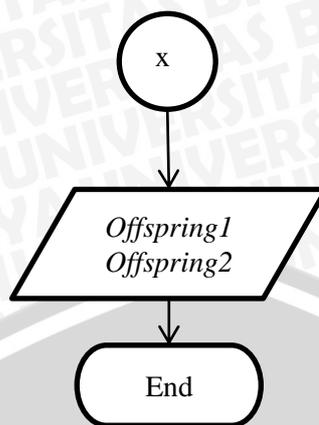




Gambar 3.3 Flowchart Proses Crossover

Untuk subproses *crossover* akan ditampilkan pada *flowchart* yang akan ditunjukkan pada Gambar 3.4.





Gambar 3.4 Diagram Alir Proses *Extended Intermediate Crossover*

Proses *extended intermediate exchange* akan dijabarkan pada perhitungan manual sebagai berikut :

1. α dibangkitkan secara *random* pada interval [0,1]

α	0,7386	0,0054	0,2067	0,1196	0,2991
----------	--------	--------	--------	--------	--------

P1	-38,3554	26,6219	77,5291	-86,8919	-70,9746
P2	92,7409	50,8185	37,0315	-76,3115	-48,4821
C1	58,4787	26,7527	69,1569	-85,6269	-64,2475
C2	-4,0931	50,6877	45,4037	-77,5765	-55,2092

P3	95,2609	68,5685	35,7244	-28,7652	39,1729
P4	93,8507	98,2261	-55,7109	-82,6776	70,6452
C3	94,2193	68,7289	16,8217	-35,2109	48,5857
C4	94,8924	98,0657	-36,8082	-76,2319	61,2324

P5	-53,7538	30,9384	42,8046	54,6096	-0,6941
P6	-53,6517	-55,0838	79,0415	-24,9051	-21,4002
C5	-53,6784	30,4733	50,2959	45,1030	-6,8869
C6	-53,7271	-54,6187	71,5501	-15,3985	-15,2073

Gambar 3.5 Hasil *Crossover*

Nilai α pada index/kolom pertama digunakan untuk menghitung semua anak hanya pada index/kolom pertama. Hasil anak dari proses perhitungan *crossover* didapatkan dari rumus *Extended Intermediate Crossover*.

1. C1 dan C2, Kolom 1 :

$$\alpha = 0,7386$$

$$C_1 = P_1 + \alpha (P_2 - P_1)$$

$$\begin{aligned}
 &= P_2 + \alpha (P_1 - P_2) \\
 &= -38,3554 + 0,7386 (92,7409 - (-38,3554)) \\
 &= 58,4787 \\
 C_2 &= P_2 + \alpha (P_1 - P_2) \\
 &= 92,4787 + 0,7386 (-38,3554 - 92,7409) \\
 &= -4,0931
 \end{aligned}$$

2. C1 dan C2, Kolom 2 :

$$\begin{aligned}
 \alpha &= 0,0054 \\
 C_1 &= 26,6219 + 0,0054 (50,8185 - 26,6219) \\
 &= 16,7527
 \end{aligned}$$

$$\begin{aligned}
 C_2 &= 50,8185 + 0,0054 (26,6219 - 50,8185) \\
 &= 50,6877
 \end{aligned}$$

3. C1 dan C2, Kolom 3 :

$$\begin{aligned}
 \alpha &= 0,2067 \\
 C_1 &= 77,5291 + 0,2067 (37,0315 - 77,5291) \\
 &= 69,1569
 \end{aligned}$$

$$\begin{aligned}
 C_2 &= 37,0315 + 0,2067 (77,5291 - 37,0315) \\
 &= 45,4037
 \end{aligned}$$

4. C1 dan C2, Kolom 4 :

$$\begin{aligned}
 \alpha &= 0,1196 \\
 C_1 &= -86,8919 + 0,1196 (-76,3115 - (-86,8919)) \\
 &= -85,6269
 \end{aligned}$$

$$\begin{aligned}
 C_2 &= -76,3115 + 0,1196 (-86,8919 - (-76,3115)) \\
 &= -77,5765
 \end{aligned}$$

5. C1 dan C2, Kolom 5 :

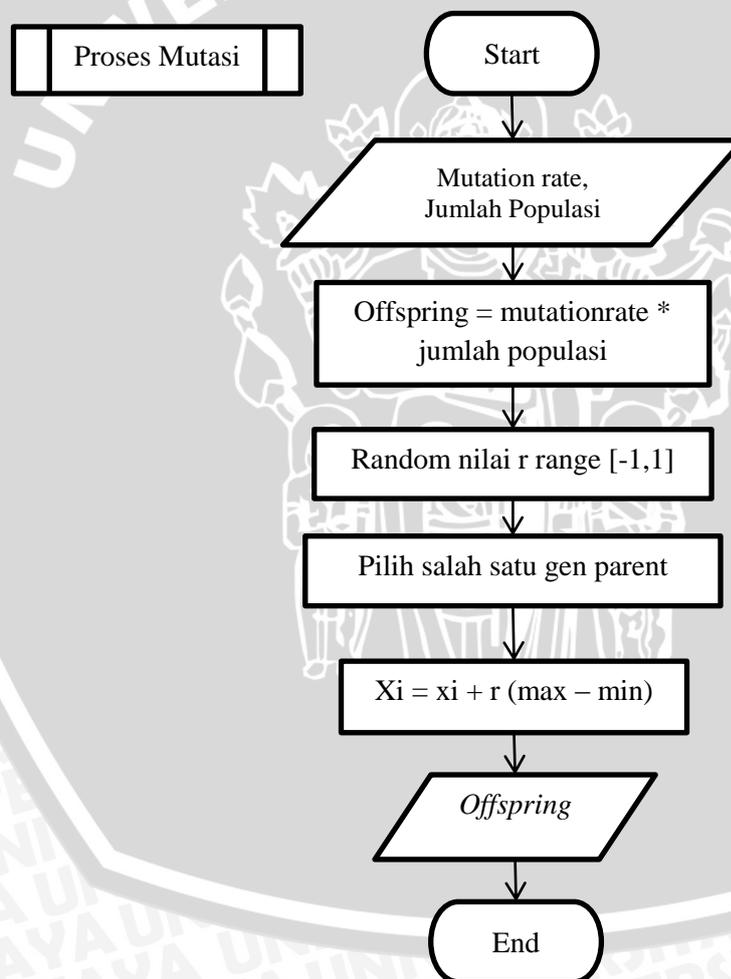
$$\begin{aligned}
 \alpha &= 0,2991 \\
 C_1 &= -70,9746 + 0,2991 (-48,4821 - (-70,9746)) \\
 &= -64,2475
 \end{aligned}$$

$$\begin{aligned}
 C_2 &= -48,4821 + 0,2991 (-70,9746 - (-48,4821)) \\
 &= -55,2092
 \end{aligned}$$

6. Dst.

3.4.3.2 Mutasi

Proses mutasi dilakukan dengan memilih satu induk secara *random* dari populasi. Metode mutasi yang digunakan adalah *random mutation* yang dilakukan dengan menambah atau mengurangi nilai gen terpilih dengan bilangan *random* yang kecil (Mahmudy, 2013). Dengan *mutation rate* sebesar 0,4 maka banyaknya *offspring* yang dihasilkan adalah $0,4 \times 10 = 4$. Parent dipilih secara *random*, pada tiap kromosom yang mengalami mutasi, setiap gen yang terpilih akan mengalami mutasi. Misalkan domain variabel x_j adalah $[\min_j, \max_j]$ dengan *range* r dibangkitkan pada interval $[-0,1, 0,1]$. Berikut flowchart untuk melakukan proses mutasi akan ditunjukkan pada Gambar 3.6.



Gambar 3.6 Diagram Alir Proses Mutasi

Proses *random mutation* akan dijabarkan dalam perhitungan manual sebagai berikut :

Dimana :

$$\max = 100$$

$$\min = -100$$

Nilai maksimum dan minimum ditentukan pada rentang kromosom awal yang telah dibuat, karena representasi kromosom dibangkitkan dalam rentang [-100,100] maka nilai maksimumnya adalah 100 dan nilai minimumnya adalah -100.

Proses mutasi akan ditunjukkan pada Gambar 3.7

R	-0,4133	0,4503	0,6987	-0,7318	0,1583
---	---------	--------	--------	---------	--------

P7	-86,9893	30,3916	13,5877	-51,5967	76,7167
C7	-169,656	120,454	153,338	-197,958	108,373

P8	38,7366	-24,5091	-33,3793	-99,5021	-13,5475
C8	-43,9300	65,5535	106,3705	-245,8630	18,1093

P9	-76,0795	2,7792	-49,3002	-56,5199	32,7695
C9	-158,7461	92,8418	90,4497	-202,8808	64,4263

P10	66,2390	-66,4502	57,1592	51,9038	47,8381
C10	-16,4276	23,6124	196,9091	-94,4571	79,4948

Gambar 3.7 Hasil Mutasi

Perhitungan proses mutasi dimulai dengan memilih salah satu gen pada induk secara random. Pada proses mutasi di atas gen pertama terpilih secara acak dan menghasilkan C4 yang didapatkan dengan cara :

Kolom 1 :

$$\begin{aligned} x'_i &= x_i + r (\max_i - \min_i) \\ &= -86,9893 + -0,4133 (100 - (-100)) \\ &= -169,656 \end{aligned}$$

Kolom 2 :

$$\begin{aligned} x'_i &= x_i + r (\max_i - \min_i) \\ &= 30,3916 + 0,4503(100 - (-100)) \end{aligned}$$

$$= 120,454$$

Kolom 3 :

$$\begin{aligned} x'_i &= x'_i + r(\max_i - \min_i) \\ &= 13,5877 + 0,6987(100 - (-100)) \\ &= 153,338 \end{aligned}$$

Kolom 4 :

$$\begin{aligned} x'_i &= x'_i + r(\max_i - \min_i) \\ &= -51,5967 + -0,7318(100 - (-100)) \\ &= -197,958 \end{aligned}$$

Kolom 5 :

$$\begin{aligned} x'_i &= x'_i + r(\max_i - \min_i) \\ &= 76,7167 + 0,1583(100 - (-100)) \\ &= 108,373 \end{aligned}$$

3.4.4 Proses Evaluasi

Proses evaluasi merupakan proses untuk menghitung nilai *fitness* dari setiap *chromosom* yang terbentuk, proses evaluasi di mencakup individu induk (*parent*) serta *offspring* yang terbentuk dari proses reproduksi sebelumnya. *Chromosome* yang terpilih untuk menjadi solusi berikutnya harus mempunyai nilai *fitness* yang lebih baik, semakin besar nilai *fitness* yang diperoleh maka semakin baik pula untuk dijadikan solusi pada generasi berikutnya (Mahmudy, 2013).

Hasil evaluasi semua individu (*Parent* dan *Offspring*) akan ditampilkan pada tabel 3.8.

Tabel 3.8 Hasil Evaluasi

Parent	Kromosom					Fitness= (1/error)
	1 <i>a</i>	2 <i>b₁</i>	3 <i>b₂</i>	4 <i>b₃</i>	5 <i>b₄</i>	
P1	-38,3554	26,6219	77,5291	-86,8919	-70,9746	4,92693
P2	92,7409	50,8185	37,0315	-76,3115	-48,4821	7,18712
P3	95,2609	68,5685	35,7244	-28,7652	39,1729	2,31985
P4	93,8507	98,2261	-55,7109	-82,6776	70,6452	8,47596

P5	-53,7538	30,9384	42,8046	54,6096	-0,6941	2,09993
P6	-53,6517	-55,0838	79,0415	-24,9051	-21,4002	11,1098
P7	-86,9893	30,3916	13,5877	-51,5967	76,7167	3,87034
P8	38,7366	-24,5091	-33,3793	-99,5021	-13,5475	1,55001
P9	-76,0795	2,7792	-49,3002	-56,5199	32,7695	3,74527
P10	66,2390	-66,4502	57,1592	51,9038	47,8381	3,01054
C1	58,4787	26,7527	69,1569	-85,6269	-64,2475	4,90675
C2	-4,0931	50,6877	45,4037	-77,5765	-55,2092	7,2298
C3	95,2609	68,5685	35,7244	-28,7652	39,1729	2,6892
C4	93,8507	98,2261	-55,7109	-82,6776	70,6452	5,66395
C5	-53,6784	30,4733	50,2959	45,1030	-6,8869	2,25288
C6	-53,7271	-54,6187	71,5501	-15,3985	-15,2073	17,3003
C7	-169,6559	120,4543	153,3375	-197,9576	108,3735	1,4303
C8	-43,9300	65,5535	106,3705	-245,8630	18,1093	4,8383
C9	-158,7461	92,8418	90,4497	-202,8808	64,4263	5,70139
C10	-16,4276	23,6124	196,9091	-94,4571	79,4948	1,29425

3.4.5 Proses Seleksi

Proses seleksi ini bertujuan untuk mempertahankan individu yang mempunyai nilai *fitness* yang tinggi agar dapat hidup untuk generasi selanjutnya. Pada penelitian ini metode seleksi yang akan digunakan adalah metode *elitism selection*, yaitu dengan mengumpulkan semua individu *parent* maupun *offspring* dalam suatu populasi satu penampungan. Individu dengan nilai *fitness* yang besar merupakan *chromosome* terbaik dalam penampungan dalam populasi tersebut akan lolos untuk masuk dalam generasi selanjutnya, individu terbaik diambil sesuai dengan *popSize* yang ditentukan sebelumnya. Metode *elitism selection* ini akan menjamin individu yang terbaik akan selalu lolos (Mahmudy, 2013). Proses *Elitism selection* akan dideskripsikan pada *pseudocode* (Mahmudy, 2013) yang akan ditampilkan pada Gambar 3.8.

```

PROCEDURE ElitismSelecion
Input :
    POP : himpunan individu pada populasi
    Pop_size : ukuran populasi
    OS : himpunan offspring hasil reproduksi menggunakan
        crossover dan mutasi
Output :
    POP : himpunan individu pada populasi setelah proses
        seleksi selese
/* gabungkan individu pada POP dan OS ke dalam TEMP*/
TEMP ← Merge (POP,OS)

```

```

/* urutkan individu berdasarkan fitness secara ascending*/
OrderAscending (Temp)
?* copy pop_size individu terbaik ke POP */
POP ← CopyBest (Temp, pop_size)

END PROCEDURE

```

Gambar 3.8 Pseudocode Proses *Elitism Selection*
Sumber : (Mahmudy,2013)

Hasil akhir berupa hasil seleksi individu-individu terbaik sejumlah populasi awal untuk dilanjutkan ke generasi selanjutnya. Hasil seleksi individu terbaik ditampilkan pada Tabel 3.9.

Tabel 3.9 Hasil Seleksi Menggunakan *Elitism Selection*

P(t-1)	Asal P	Fitness
P1	C6	17,30031
P2	P6	11,10979
P3	P4	8,475963
P4	C2	7,229797
P5	P2	7,187125
P6	C9	5,70139
P7	C4	5,663954
P8	P1	4,926934
P9	C1	4,906751
P10	C8	4,8383

Setelah dilakukan proses seleksi, maka dilanjutkan dengan memilih kromosom terbaik yang dapat ditentukan berdasarkan nilai *fitness* terbesar. Pada Tabel 3.8, kromosom yang memiliki *fitness* terbesar adalah *Offspring* ke-6 dengan nilai 6,6303E-08. Detail *Offspring* ke-6 ditunjukkan pada Tabel 3.10.

Tabel 3.10 Detail Kromosom Terbaik

Parent	Kromosom					Fitness=
	1	2	3	4	5	(1/error)
	<i>a</i>	<i>b₁</i>	<i>b₂</i>	<i>b₃</i>	<i>b₄</i>	
C6	-53,7271	-54,6187	71,5501	-15,3985	-15,2073	17,3003

Dari hasil perhitungan Algoritma Genetika didapatkan koefisien terbaik pada *Offspring* ke-6, sehingga persamaan regresi yang baik untuk mempresiksi harga konsumsi kWh ditunjukkan pada Persamaan (3-2).

$$Y' = -53,7271 - 54,6187 X_1 + 71,5501 X_2 - 15,3985 X_3 - 15,2073 X_4 \quad (3-2)$$

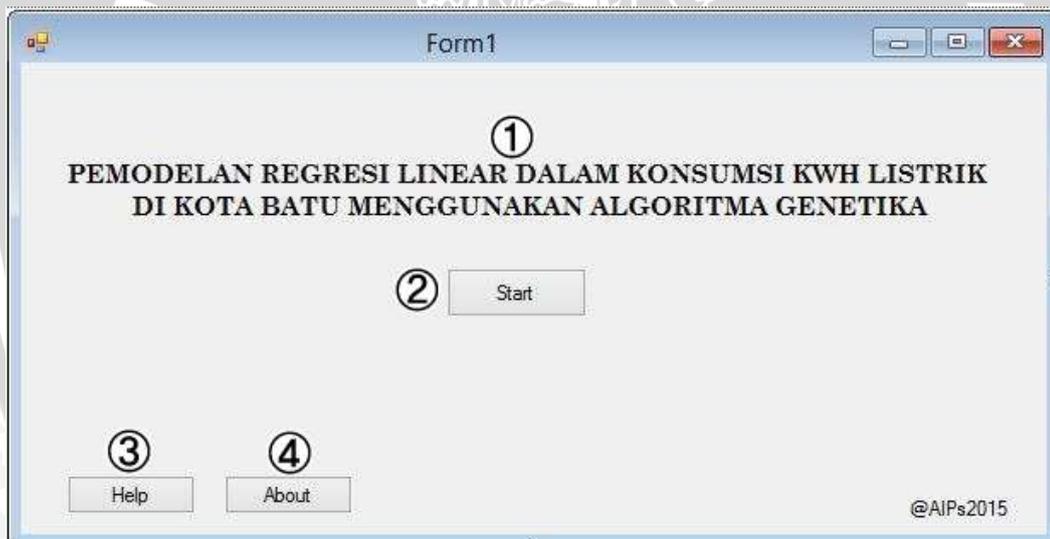
BAB IV PERANCANGAN

4.1 Perancangan *User Interface*

Pada rancangan *user interface* aplikasi ini akan dibuat tiga halaman yaitu halaman pertama adalah halaman *home*, halaman kedua adalah halaman inputan data kWh, dan yang terakhir adalah halaman inputan parameter Algoritma Genetika serta hasil prediksi dari konsumsi kWh listrik.

4.1.1 Rancangan Halaman *Home*

Halaman *home* merupakan tampilan awal ketika memulai aplikasi, halaman ini bersisi nama aplikasi, *button start*, *button help*, dan *button about application*. Rancangan halaman *home* akan ditampilkan pada Gambar 4.1.



Gambar 4.1 *User Interface* Halaman *Home*

Keterangan:

1. Judul aplikasi
2. *Button start* untuk memulai aplikasi
3. *Button help* untuk mendapat petunjuk menjalankan aplikasi
4. *Button about* untuk mendapatkan informasi seputar aplikasi

4.1.2 Halaman Input Parameter dan Data Konsumsi kWh

Halaman kedua merupakan halaman untuk input data kWh. Rancangan halaman input data konsumsi kWh akan ditampilkan pada Gambar 4.2.

Gambar 4.2 Rancangan Halaman Input Data Konsumsi kWh

Keterangan :

1. *TextBox* untuk menampilkan ukuran populasi
2. *TextBox* untuk menampilkan *crossove rate*
3. *TextBox* untuk menampilkan *mutation rate*
4. *TextBox* untuk menampilkan banyaknya generasi
5. *Button* Data kWh untuk meload Data Saham berdasarkan periode
6. *Button* untuk memproses seluruh data menggunakan Algoritma Genetika

4.1.3 Rancangan Halaman Inputan Parameter Algoritma Genetika dan Hasil Prediksi konsumsi kWh

Setelah melakukan input data pada halaman sebelumnya, hasil proses perhitungan regresi dengan Algoritma Genetika akan ditampilkan pada rancangan halaman hasil prediksi yang dapat dilihat di Gambar 4.3

Gambar 4.3 Rancangan Tampilan Halaman Hasil Prediksi

Keterangan :

1. *DataGridView* untuk menampilkan prediksi terbaik konsumsi kWh
2. *DataGridView* untuk menampilkan hasil pemodelan regresi
3. *DataGridView* untuk menampilkan sort hasil generasi
4. *Button Back* untuk kembali ke menu sebelumnya
5. *Button Close* untuk keluar dari aplikasi

4.2 Perancangan Uji Coba dan Evaluasi

Dibuatnya perancangan uji coba dan evaluasi disebabkan karena tidak adanya metode yang pasti untuk menentukan prediksi pemakaian kWh dan menentukan parameter Algoritma Genetika. Maka untuk mengevaluasi sistem yang dibuat akan dilakukan uji coba antara lain :

1. Uji coba untuk menentukan range *popSize* yang optimal
2. Uji coba untuk menentukan ukuran populasi yang optimal
3. Uji coba untuk menentukan banyaknya generasi yang optimal
4. Uji coba untuk mencari nilai *crossover rate* dan *mutation rate* terbaik agar menghasilkan rekomendasi yang optimal.
5. Uji coba untuk mencari banyaknya periode pemakaian kWh yang akan diprediksi.

Jumlah Generasi	Nilai <i>Fitness</i>										Rata-rata nilai <i>fitness</i>	
	Percobaan generasi ke-											
	1	2	3	4	5	6	7	8	9	10		
1750												
2000												

4.2.3 Uji Coba Kombinasi *Crossover Rate* dan *Mutation Rate*

Uji coba kombinasi *crossover rate* dan *mutation rate* dilakukan untuk mengetahui kombinasi *cr* dan *mr* terbaik agar mendapatkan hasil koefisien regresi untuk mendapatkan hasil prediksi konsumsi kWh listrik yang optimal. Uji coba kombinasi *cr* dan *mw* ini menggunakan nilai *cr* dan *mr* yang berbeda dalam range 0 – 1. Rancangan uji coba kombinasi *crossover rate* dan *mutation rate* ditunjukkan pada Tabel 4.3.

Tabel 4.3 Rancangan Uji Coba *Crossover Rate* dan *Mutation Rate*

Kombinasi		Nilai <i>Fitness</i>										Rata-rata Nilai <i>Fitness</i>
		Percobaan generasi ke-										
		1	2	3	4	5	6	7	8	9	10	
0	1											
0,1	0,9											
0,2	0,8											
0,3	0,7											
0,4	0,6											
0,5	0,5											
0,6	0,4											
0,7	0,3											
0,8	0,2											
0,9	0,1											
1	0											

4.2.4 Uji Coba Banyaknya Periode Konsumsi kWh

Uji coba banyaknya periode konsumsi kWh dilakukan untuk mengetahui hasil prediksi mana yang lebih akurat berdasarkan periode konsumsi kWhnya. Banyaknya periode yang digunakan adalah kelipatan perbulan, pertiga bulan, perenam bulan, dan pertahun. Data yang diperoleh adalah tiga tahun terakhir

mulai tahun 2012 sampai 2014. Rancangan uji coba banyaknya periode konsumsi kWh ditunjukkan pada Tabel 4.4.

Tabel 4.4 Rancangan Uji Coba Banyaknya Periode Konsumsi kWh

Banyak	Nilai <i>Fitness</i>										Rata-rata nilai <i>Fitness</i>
	Percobaan generasi ke-										
Periode	1	2	3	4	5	6	7	8	9	10	
2 bulan											
3 bulan											
4 bulan											
5 bulan											
6 bulan											
7 bulan											
8 bulan											
9 bulan											
10 bulan											



BAB V IMPLEMENTASI

5.1 Implementasi Program

5.1.1 Struktur Data

Struktur data awal sebagai data acuan yang dibuat untuk prediksi konsumsi kWh berdasarkan data historis menggunakan Algoritma Genetika ditunjukkan pada Source Code 5.1.

```
1 public class DataTest
2     {
3         //variabel global
4         public float Y { get; set; } // properties
5         //x array
6         public float[] X { get; set; } //properties
7
8         public DataTest(float Y, float[] X) //konstraktor
9         {
10            //variabel lokal
11            this.Y = Y;
12            this.X = X;
13        }
14    }
15
16 //turunan dari data test
17 public class DataPopulasi : DataTest
18     {
19         public float Fitness { get; set; }
20         public float Error { get; set; }
21
22         public DataPopulasi(float Y, float[] X) : base(Y, X)
23         //manggil konstraktor dari data test
24         {
25             Fitness = 0.0f;
26         }
27     }
28 }
```

Source Code 5.1 Proses Struktur Data

5.1.2 Me-load Data Konsumsi kWh

Pengambilan data konsumsi kWh dilakukan dengan membaca file csv (*comma separated values*) yang telah disiapkan sebelumnya. File csv yang dibuat berisi data konsumsi kWh berdasarkan data yang diformulasikan dari pemakaian bulan sebelumnya yang disusun dari perhitungan 2 bulan kebelakang sampai 10 bulan kebelakang. Proses *load* data konsumsi kWh ditunjukkan pada Source Code 5.2.

```
1 private void button1_Click(object sender, EventArgs e)
2     {
3         //OpenFileDialog openFileDialog1 = new
4         OpenFileDialog();
5         //openFileDialog1.InitialDirectory = "c:\\";
6         openFileDialog1.Filter = "Comma Separated
7         Values (*.csv)|*.csv";
8         openFileDialog1.FilterIndex = 2;
9         openFileDialog1.RestoreDirectory = true;
10        data.Clear();
11        if (openFileDialog1.ShowDialog() ==
12        DialogResult.OK)
13        { Console.WriteLine(openFileDialog1.FileName);
14        using (CSVFileReader reader = new
15        CSVFileReader(openFileDialog1.FileName)) //path
16        dari file yang di ambil
17        {
18            DataTable dt = new DataTable("My Table");
19            dt.Columns.Add(new DataColumn("No",
20            typeof(int)));
21            dt.Columns.Add(new DataColumn("Y",
22            typeof(float)));
23
24            CSVRow row = new CSVRow();
25            int col = 0;
26            int No = 1;
27
28            //ketika baris ada 1 maka dimasukkan dlam
29            csv readrow
30            while(reader.ReadRow(row))
31            {
32                float y = 0;
33                List<float> x = new List<float>();
34                col = 0;
35                DataRow dr = dt.NewRow();
36                dr["No"] = No++;
37                foreach (string s in row)
38                {
39                    if (col == 0)
40                    {
41                        dr["Y"] = float.Parse(s); //
42                        nilai awal string harus di parse
43                        menjadi float
44                        y = float.Parse(s);
45                    }
46                    else
47                    {
48                        if(!dt.Columns.Contains(("X" +
49                        col.ToString())))
50                        {
51                            dt.Columns.Add(new
52                            DataColumn(("X" +
53                            col.ToString()),
54                            typeof(float));
55                        }
56                        dr[("X" + col.ToString())] =
57                        float.Parse(s);
58                    }
59                }
60            }
61        }
62    }
63 }
```

```

44         x.Add(float.Parse(s));
45     }
46     col++;
47 }
48 Console.WriteLine(x[0]);
49 data.Add(new DataTest(y, x.ToArray()));
50 dt.Rows.Add(dr);
51 }
52 DataSet ds = new DataSet();
53 ds.Tables.Add(dt);
dataGridView1.DataSource = ds.Tables[0];
}
}
}

```

Source Code 5.2 Proses *Load* Data

5.1.3 Pembangkitan Representasi Kromosom

Pembangkitan populasi awal dilakukan dengan merepresentasikan kromosom berdasarkan hasil pembangkitan *popSize* yang diinputkan oleh *user* serta data konsumsi kWh yang sebelumnya telah di-*load*. Proses representasi kromosom serta pembangkitan populasi awal ditunjukkan pada Source Code 5.3.

```

1 public float RandomFloat(float min, float max)
2     {
3         return ((float)random.NextDouble()) * (max
4             - min) + min;
5     }
6 private void GenerateParent()
7     {
8         parent = new DataPopulasi[jumlahparent];
9         for (int i = 0; i < jumlahparent; i++)
10            {
11                float[] X = new float[jumlahx];
12                for (int j = 0; j < jumlahx; j++)
13                    {
14                        X[j] = random.RandomFloat(-100,
15                            100);
16                    }
17                parent[i] = new
18                    DataPopulasi(random.RandomFloat(-100, 100), X);
19            }
20    }

```

Source Code 5.3 Proses Representasi Kromosom

5.1.4 Perhitungan *Error* dan *Fitness*

Untuk mendapatkan nilai *fitness* sebelumnya harus dihitung nilai *error* terlebih dahulu, nilai *error* didapatkan dari jumlah kuadrat dari selisih kromosom (sebagai koefisien) dengan semua data historis konsumsi kWh yang sudah di-*load*

sebelumnya. Hasil nilai *fitness* sama dengan satu per nilai *error* untuk satu individu. Proses *error* dan *fitness* ditunjukkan pada Source Code 5.4.

```

1 public void HitungFitness(DataTest[] data)
2     {
3         int jumlahdata = data.Length;
4         int jumlahx = X.Length;
5         float sumerror = 0;
6         for (int i = 0; i < jumlahdata; i++)
7         {
8             float Y1 = Y;
9             for (int j = 0; j < jumlahx; j++)
10            {
11                Y1 += (X[j] * data[i].X[j]);
12            }
13            //hitung error
14            float Error = (float)Math.Pow(data[i].Y -
15            Y1, 2);
16            sumerror += Error;
17        }
18        hasil = 1/((Math.Sqrt(sum))/baris);
19        this.Error = sumerror;
20        this.Fitness = 100000000 / sumerror;
21    }

```

Source Code 5.4 Proses *Error* dan *Fitness*

5.1.5 Proses *Crossover*

Proses *crossover* menggunakan metode *extended intermediate crossover*, metode *crossover* tersebut yang pertama dilakukan adalah membangkitkan nilai alfa (α) secara *random* dengan range [0,1] sebanyak panjang gen yang terbentuk dan memilih dua induk secara *random* pula. Dua induk yang terpilih kemudian dilakukan proses *crossover*, setiap dua induk yang terpilih akan menghasilkan dua offspring dalam *crossover* metode ini. Proses *crossover* akan ditunjukkan pada Source Code 5.5.

```

1 //fungsi alfa untuk crossover
2 private void GenerateAlpha()
3     {
4         float[] X = new float[jumlahx];
5         for (int i = 0; i < jumlahx; i++)
6         {
7             X[i] = random.RandomFloat(0, 1);
8         }
9         alpha = new DataTest(random.RandomFloat(0, 1),
10        X);

```

```

10 //proscrossover
11 private void HitungCrossover()
12 {
13     int jumlahcrossover =
14     (int)Math.Ceiling((crossoverrate * jumlahparent)
15     / 2);
16     bool[] crossedparent = new bool[jumlahparent];
17     for (int i = 0; i < jumlahcrossover; i++)
18     {
19         int parent1 = -1;
20         int parent2 = -1;
21         do
22         {
23             parent1 = random.RandomInt(0,
24             jumlahparent);
25             parent2 = random.RandomInt(0,
26             jumlahparent);
27         } while ((crossedparent[parent1] &&
28         crossedparent[parent2] || (parent1 ==
29         parent2));
30         crossedparent[parent1] = true;
31         crossedparent[parent2] = true;
32         //Console.WriteLine("== Crossed Parent ==");
33         //Console.WriteLine("Parent 1 : " + parent1
34         + " && Parent 2 : " + parent2);
35         //child A
36         float Y = parent[parent1].Y + (alpha.Y *
37         (parent[parent2].Y - parent[parent1].Y));
38         float[] X = new float[jumlahx];
39         for (int j = 0; j < jumlahx; j++)
40         {
41             X[j] = parent[parent1].X[j] +
42             (alpha.X[j] * (parent[parent2].X[j] -
43             parent[parent1].X[j]));
44         }
45         AddNewChild(new DataPopulasi(Y, X));
46         //child B
47         Y = parent[parent2].Y + (alpha.Y *
48         (parent[parent1].Y - parent[parent2].Y));
49         X = new float[jumlahx];
50         for (int j = 0; j < jumlahx; j++)
51         {
52             X[j] = parent[parent2].X[j] +
53             (alpha.X[j] * (parent[parent1].X[j] -
54             parent[parent2].X[j]));
55         }
56         AddNewChild(new DataPopulasi(Y, X));
57     }
58 }

```

Source Code 5.5 Proses Crossover

5.1.6 Proses Mutasi

Proses mutasi menggunakan model *random mutation*, dalam model metode tersebut hal pertama yang dilakukan yaitu membangkitkan nilai *random* r dengan range $[-1,0 - 1,0]$, kemudian menetapkan nilai maksimum dan minimum

sebagai batas atas dan batas bawah serta memilih satu induk secara acak. Induk yang terpilih akan diproses menggunakan perhitungan mutasi. Proses mutasi akan ditunjukkan pada Source Code 5.6.

```

1      private void GenerateMutationR()
2      {
3          float[] X = new float[jumlahx];
4          for (int i = 0; i < jumlahx; i++)
5          {
6              X[i] = random.RandomFloat(-1, 1);
7          }
8          mutationr = new DataTest(random.RandomFloat(-1,
9              1), X);
10     }
11     private void HitungMutasi()
12     {
13         int jumlahmutasi = (int)(mutationrate *
14             jumlahparent);
15         //tanda mana yang sudah di mutasi sama yg belum
16         bool[] mutatedparent = new bool[jumlahparent];
17         //mutated parent = status parent awal
18         for (int i = 0; i < jumlahmutasi; i++)
19         {
20             int parent = -1;
21             do
22             {
23                 parent = random.RandomInt(0,
24                     jumlahparent);
25             } while (mutatedparent[parent]);
26             //Console.WriteLine("== Mutated Parent ==");
27             //Console.WriteLine("Parent : " + parent);
28             mutatedparent[parent] = true;
29             float Y = this.parent[parent].Y +
30                 (mutationr.Y * (mutationmax - mutationmin));
31             float[] X = new float[jumlahx];
32             for (int j = 0; j < jumlahx; j++)
33             {
34                 X[j] = this.parent[parent].X[j] +
35                     (mutationr.X[j] * (mutationmax -
36                         mutationmin));
37             }
38             AddNewChild(new DataPopulasi(Y, X));
39         }
40     }

```

Source Code 5.6 Proses Mutasi

5.1.7 Proses Seleksi dengan Metode *Elitism Selection*

Proses seleksi yang digunakan menggunakan model *Elitism Selection*, model tersebut bekerja dengan cara mengumpulkan semua individu dalam populasi *parent* dan *offspring* dalam satu tempat. *popSize* individu terbaik akan lolos untuk proses generasi selanjutnya, individu yang diambil sebanyak *popSize*

seperti awal mula. Model seleksi ini menjamin individu yang terbaik akan selalu lolos (Mahmudy, 2013). Proses seleksi tersebut akan ditunjukkan pada Source Code 5.7.

```

1 private void HitungSeleksi()
2     {
3         int jumlahdata = jumlahparent * 3;
4         //parent+crossover+over
5         DataPopulasi[] semuaadata = new
6         DataPopulasi[jumlahdata];
7         int index = 0;
8         int length = jumlahparent;
9         for (int i = 0; i < length; i++)
10        //masukan semua data parent + child
11        {
12            parent[i].HitungFitness(data);
13            semuaadata[index] = parent[i];
14            index++;
15        }
16        //untuk child
17        length = jumlahparent * 2;
18        for (int i = 0; i < length; i++)
19        {
20            if (child[i] != null)
21            {
22                child[i].HitungFitness(data);
23                semuaadata[index] = child[i];
24                index++;
25            }
26        }
27    }

```

Source Code 5.7 Proses Seleksi

5.1.8 Proses Pemilihan Kromosom Terbaik Sebanyak *Popsize* Awal

Proses pemilihan kromosom terbaik sebanyak *popSize* awal merupakan *sorting descending* (dari paling besar ke paling kecil) dari proses seleksi sebelumnya. Proses sorting tersebut akan ditunjukkan pada Source Code 5.8.

```

1 for (int i = 0; i < index; i++)
2     {
3         for (int j = i; j < index; j++)
4         {
5             if (semuadata[i].Fitness <
6                 semuadata[j].Fitness)
7             {
8                 DataPopulasi temp = semuadata[i];
9                 semuadata[i] = semuadata[j];
10                semuadata[j] = temp;
11            }
12        }
13    }
14    //untuk parent baru setelah disoort dari parent
15    lama + child

```

```

12     for (int i = 0; i < jumlahparent; i++)
13     {
14         parent[i] = semuadata[i];
15     }
16     hasilgenerasi[indexgenerasi] = parent[0];
17     if ((parenthasil == null) ||
18         (parenthasil.Fitness < parent[0].Fitness))
19     {
20         parenthasil = parent[0];
21     }
22     indexgenerasi++;
23 }

```

Source Code 5.8 Proses Pemilihan Kromosom Terbaik

5.1.9 Proses Iterasi

Proses Iterasi merupakan proses perulangan perhitungan Algoritma Genetika sebanyak inputan generasi yang di-inputkan oleh *user*. Proses iterasi generasi ditunjukkan pada Source Code 5.9.

```

1     for (int i = 0; i < jumlahgenerasi; i++)
2     {
3         //semua parent baru
4         for (int j = 0; j < jumlahpopulasi; j++)
5         {
6             allparent[indexparent] = parent[j];
7             indexparent++;
8         }
9         GenerateAlpha();
10        GenerateChild();
11        HitungCrossover();
12        GenerateMutationR();
13        HitungMutasi();
14        HitungSeleksi();
15    }

```

Source Code 5.9 Proses Iterasi

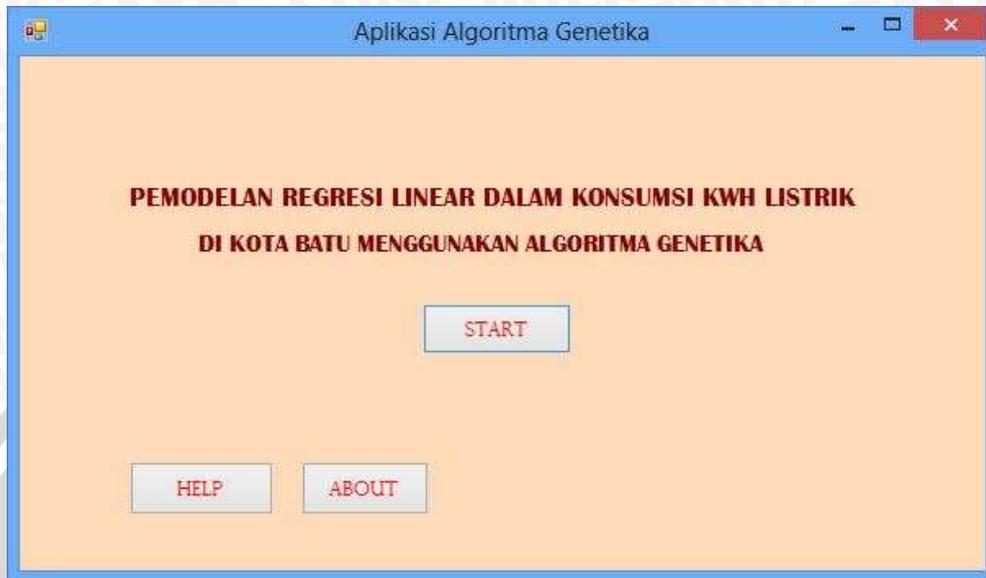
5.2 Implementasi Antarmuka

User interface atau antarmuka sistem terdiri dari dua form utama, yaitu form untuk inputan data dan form hasil perhitungan. Kemudian terdapat tiga form yang lain sebagai pelengkap sistem, yaitu halaman home atau halaman awal aplikasi, halaman help serta halaman about.

5.2.1 Halaman Awal

Halaman awal atau *home page* merupakan halaman *default* ketika aplikasi mulai dijalankan pertama kali. Terdapat *button start* untuk ke form selanjutnya dan melakukan input parameter, *button help* berisi form panduan tentang

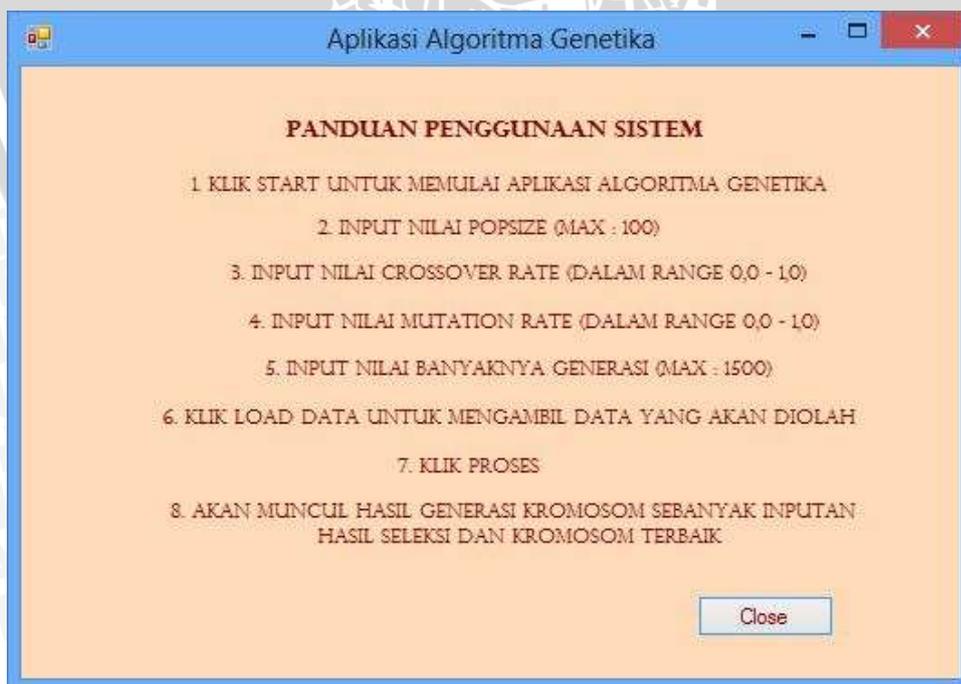
penggunaan sistem, dan *button about* berisi tentang developer aplikasi. Gambar implementasi halaman awal ditunjukkan pada Gambar 5.1.



Gambar 5.1 Implementasi Halaman Awal Aplikasi

5.2.2 Halaman Help

Halaman *help* merupakan halaman panduan untuk menjalankan sistem Algoritma Genetika ini. Gambar implementasi Halaman *Help* akan ditunjukkan pada Gambar 5.2.



Gambar 5.2 Implementasi Halaman Help

5.2.3 Halaman About

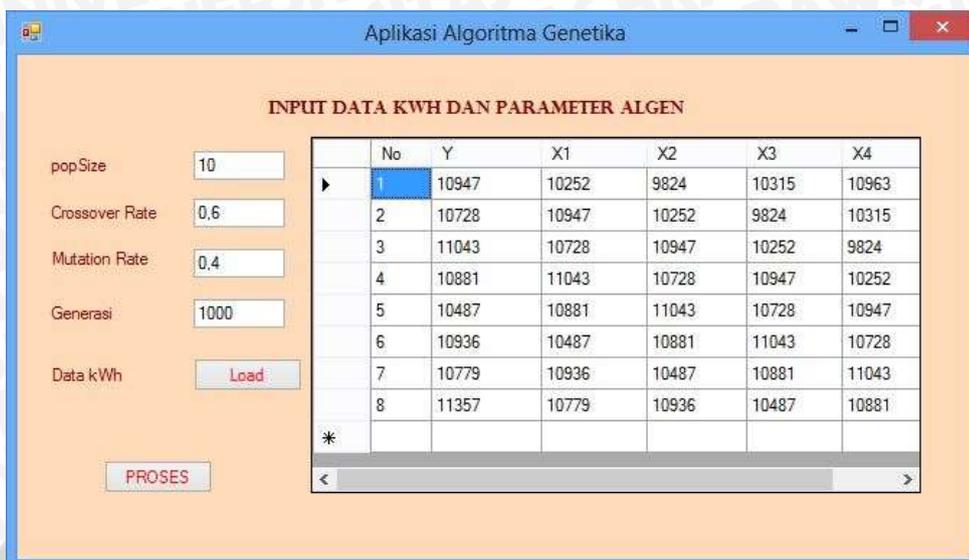
Halaman *About* merupakan halaman yang berisi tentang developer sistem Algoritma Genetika. Gambar implementasi halaman *about* ditunjukkan pada Gambar 5.3.



Gambar 5.3 Implementasi Halaman *About*

5.2.4 Halaman Input Parameter

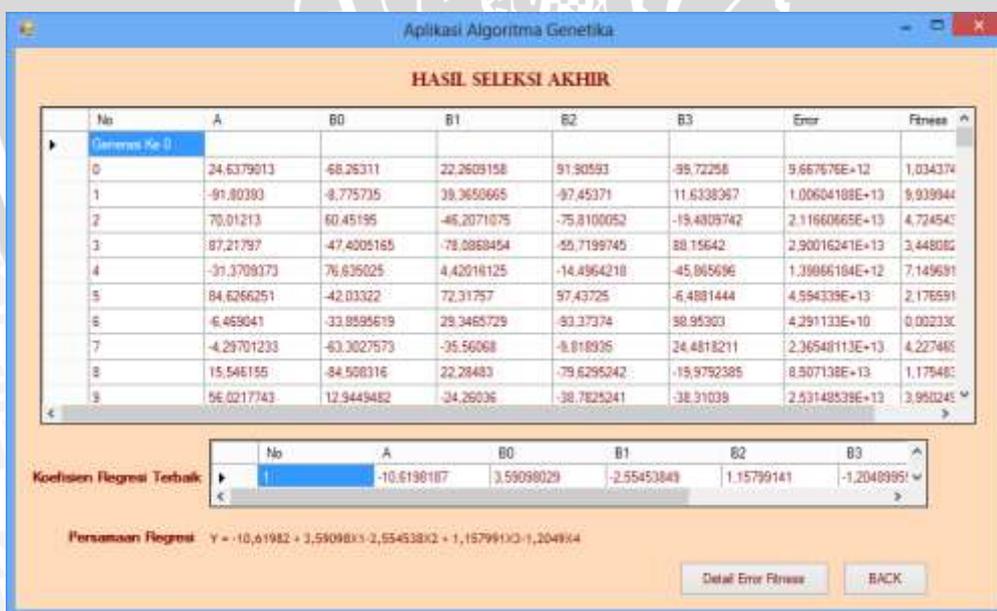
Halaman Input Parameter terdiri dari inputan *popSize*, *crossovertime*, *mutationrate*, jumlah generasi serta button load data dengan format csv. Kemudian data yang di load tadi akan ditampilkan pada space yang telah dibuat. Gambar Halaman Input Parameter akan ditunjukkan pada Gambar 5.4.



Gambar 5.4 Implementasi Halaman Inputan Parameter

5.2.5 Halaman Hasil Koefisien Terbaik dari Proses Algoritma Genetika

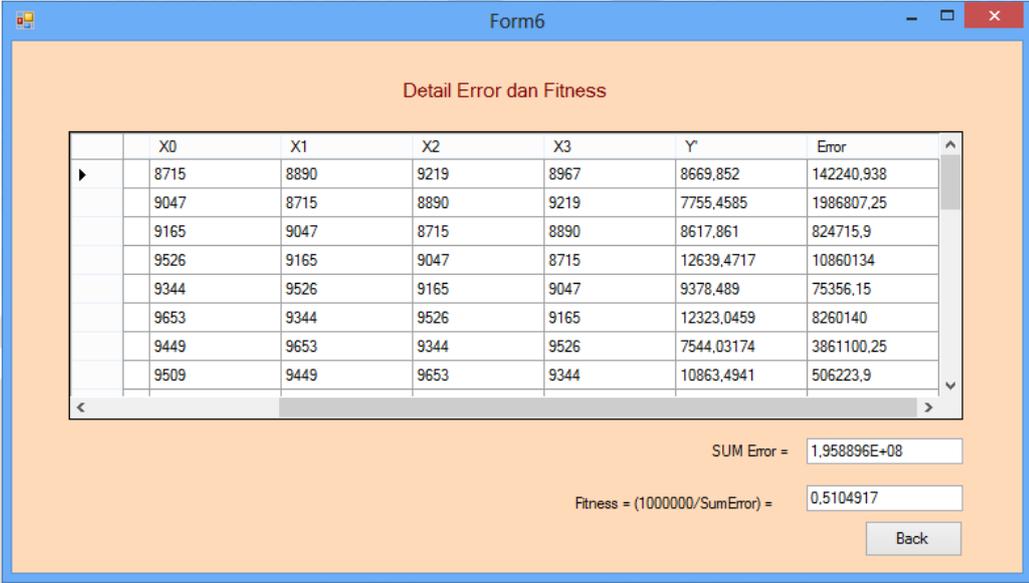
Halaman ini merupakan halaman tampilan hasil proses Algoritma Genetika berdasarkan inputan pada halaman sebelumnya. Terdapat tampilan hasil generasi beserta *fitness*nya dan tampilan koefisien kromosom terbaik. Gambar Halaman Hasil akan ditunjukkan pada Gambar 5.5.



Gambar 5.5 Implementasi Hasil Akhir Algoritma Genetika

5.2.6 Halaman Menampilkan Detail Error dan Fitness

Halaman yang terakhir merupakan halaman untuk menampilkan perhitungan *error* dan *fitness* dari koefisien terbaik yang dihasilkan oleh sistem Algoritma Genetika akan ditunjukkan pada Gambar 5.6.

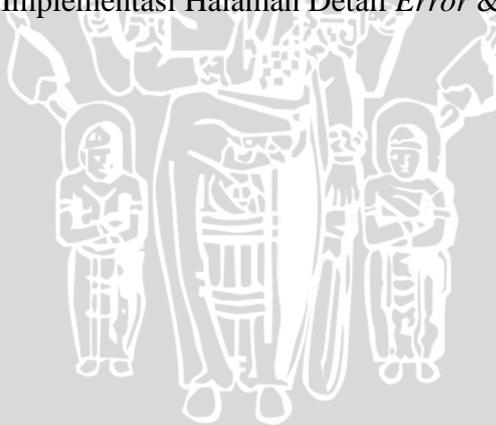


	X0	X1	X2	X3	Y	Error
▶	8715	8890	9219	8967	8669,852	142240,938
	9047	8715	8890	9219	7755,4585	1986807,25
	9165	9047	8715	8890	8617,861	824715,9
	9526	9165	9047	8715	12639,4717	10860134
	9344	9526	9165	9047	9378,489	75356,15
	9653	9344	9526	9165	12323,0459	8260140
	9449	9653	9344	9526	7544,03174	3861100,25
	9509	9449	9653	9344	10863,4941	506223,9

SUM Error =

Fitness = (1000000/SumError) =

Gambar 5.6 Implementasi Halaman Detail Error & Fitness



BAB VI PENGUJIAN DAN ANALISA

Pada bab ini akan dibahas serta dianalisis hasil uji coba yang telah dilakukan dalam pemodelan regresi linear dalam konsumsi kWh listrik menggunakan Algoritma Genetika.

6.1 Hasil dan Analisa Uji Coba Ukuran Populasi (*popSize*)

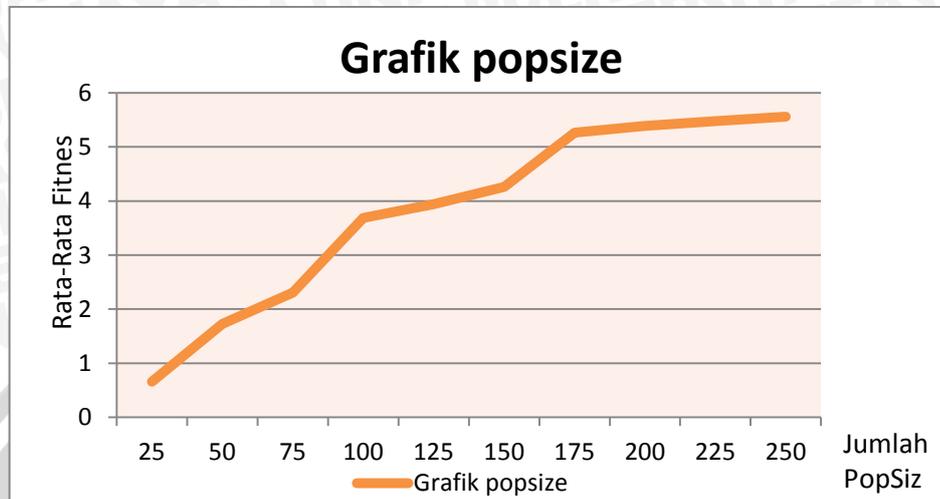
Pada uji coba yang pertama dilakukan adalah uji coba ukuran populasi terhadap perubahan nilai *fitness*. Data yang digunakan dalam pengujian ukuran populasi ini sebanyak 160 data konsumsi kWh untuk prediksi 4 bulan kebelakang mulai tahun 2012 sampai 2014. Jumlah generasi yang dipakai adalah 1000 dengan banyak populasi kelipatan 25, mulai dari 25 *popSize* sampai 250 *popSize*. Nilai *crossover rate* yang digunakan adalah 0,6 dan *mutation rate* yang digunakan adalah 0,4. Setiap ukuran populasi yang diuji akan dilakukan 10 kali percobaan dan akan diperoleh nilai rata-rata nilai *fitness* optimal terhadap percobaan ukuran populasi terbaik. Untuk hasil percobaan ukuran populasi ditunjukkan pada Tabel 6.1.

Tabel 6.1 Hasil Uji Coba Ukuran Populasi

Banyak PopSize	Nilai Fitness										Rata- rata nilai <i>fitness</i>	Rata- rata Waktu (ms)
	Percobaan generasi ke-											
	1	2	3	4	5	6	7	8	9	10		
25	0,752	0,765	0,523	1,211	0,478	0,282	0,961	0,56	0,825	0,233	0,659	3600
50	0,429	1,715	1,683	3,305	0,883	1,740	1,740	0,750	4,293	0,734	1,727	3900
75	2,993	1,200	2,819	7,098	2,042	1,377	0,819	0,774	1,648	2,301	2,306	4255
100	4,200	1,174	8,708	3,98	1,145	2,267	7,064	2,944	1,606	3,770	3,687	4500
125	5,986	1,984	0,957	2,634	2,529	6,596	6,439	2,904	5,911	3,477	3,942	4955
150	8,439	3,508	5,082	3,045	4,586	2,057	1,377	2,041	3,473	9,020	4,263	5500
175	2,306	10,101	4,059	8,035	5,136	7,499	8,001	3,038	2,054	2,441	5,267	6200
200	3,658	9,037	6,364	5,785	1,794	8,410	7,783	2,203	6,830	2,039	5,390	6920
225	5,009	6,408	5,083	6,966	5,064	4,893	5,692	6,911	5,154	3,613	5,479	7555
250	5,449	2,318	4,591	4,206	9,259	5,522	4,027	4,564	11,63	4,016	5,559	8800

Hasil uji coba ukuran populasi (*popSize*) terhadap rata-rata nilai *fitness* pada Tabel 6.1 akan diformulasikan dalam sebuah grafik untuk mempermudah

melihat perbedaan dari hasil pengujian banyaknya ukuran populasi terhadap nilai *fitness*. Grafik hasil uji coba ukuran populasi ditunjukkan pada Gambar 6.1



Gambar 6.1 Grafik Hasil Uji Coba Ukuran Populasi

Pada Gambar 6.1 hasil uji coba untuk ukuran populasi terhadap hasil *fitness* cenderung mengalami kenaikan di titik ketika ukuran popSize sama dengan 175 dengan rata-rata nilai *fitness* 5,267, dengan penambahan popSize sebanyak 25 kali lipat pada percobaan selanjutnya tidak mengalami kenaikan nilai *fitness* secara signifikan dan cenderung stabil. Oleh karena itu didapatkan popSize terbaik pada uji coba ukuran popSize ini adalah 175 popSize.

Semakin besar ukuran populasi maka akan menghasilkan nilai *fitness* semakin besar juga, namun setelah batas tertentu (pada kasus ini sebesar 175) kenaikan yang didapatkan semakin tidak signifikan. Pola ini juga didapatkan pada penelitian (Pratiwi, 2014) yang menerapkan Algoritma Genetika untuk optimasi biaya pemenuhan kebutuhan gizi. Pembatasan ukuran populasi perlu dilakukan karena semakin banyak ukuran populasi maka waktu yang dibutuhkan untuk proses Algoritma Genetika juga semakin besar.

6.2 Hasil dan Analisa Uji Coba Banyaknya Generasi

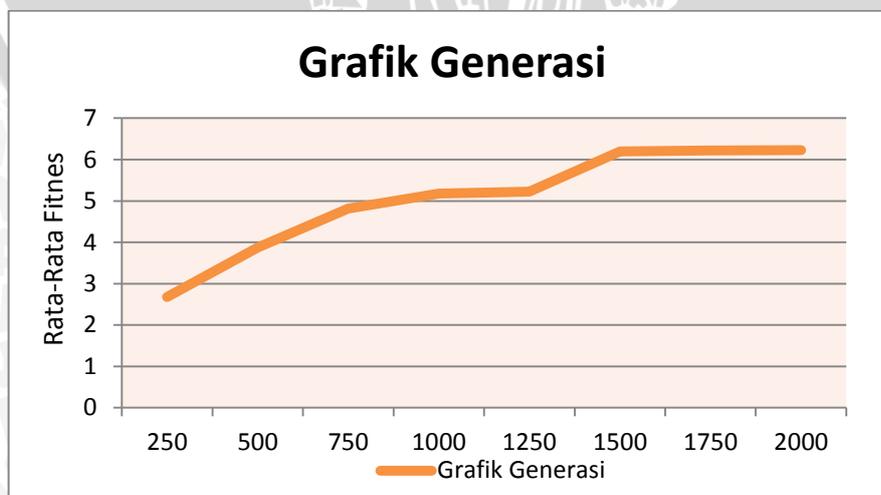
Uji coba ukuran generasi terhadap perubahan nilai *fitness*. Banyak generasi yang akan dilakukan dengan kelipatan 250 mulai dari 250 generasi sampai 2000 generasi. Sebelumnya telah dilakukan uji coba ukuran populasi, populasi terbaik pada uji coba sebelumnya digunakan pada uji coba generasi ini

yaitu 175. Nilai *crossover rate* yang digunakan yaitu 0,6 sedangkan *mutation rate* yang digunakan 0,4. Untuk data yang digunakan dalam pengujian ukuran populasi ini sebanyak 160 data konsumsi kWh untuk prediksi 4 bulan kebelakang mulai tahun 2012 sampai 2014. Setiap ukuran banyaknya generasi yang diuji akan dilakukan 10 kali percobaan dan akan diperoleh nilai rata-rata nilai *fitness* optimal terhadap percobaan ukuran generasi terbaik. Untuk hasil percobaan ukuran generasi ditunjukkan pada Tabel 6.2.

Tabel 6.2 Hasil Uji Coba Ukuran Banyaknya Generasi

Jumlah Generasi	Nilai Fitness										Rata- rata nilai <i>fitness</i>
	Percobaan generasi ke-										
	1	2	3	4	5	6	7	8	9	10	
250	3,142	3,041	2,272	2,074	2,196	2,754	3,036	2,190	3,004	3,06	2,678
500	3,401	3,098	3,374	4,355	4,346	4,50	3,272	4,465	4,385	3,513	3,871
750	4,595	4,548	4,310	4,690	4,763	5,302	4,720	4,430	5,614	5,181	4,815
1000	5,208	5,262	5,462	4,255	6,241	5,505	4,977	4,299	6,23	4,341	5,178
1250	4,864	6,350	5,687	4,899	4,768	5,430	4,190	5,656	4,637	5,719	5,220
1500	5,4	8,172	4,603	8,751	8,320	6,054	5,821	8,014	2,937	3,870	6,194
1750	3,995	8,275	5,110	6,013	6,910	6,869	7,485	6,461	5,573	5,473	6,216
2000	8,057	5,941	4,480	4,913	7,997	6,992	6,381	6,163	4,730	6,645	6,230

Hasil uji coba banyaknya ukuran generasi terhadap nilai *fitness* pada Tabel 6.2 akan diformulasikan dalam sebuah grafik untuk dapat melihat perbedaan dari hasil pengujian banyak generasi terhadap nilai *fitness*. Grafik hasil uji coba ukuran generasi ditunjukkan pada Gambar 6.2.



Gambar 6.2 Grafik Hasil Uji Coba Banyak Generasi

Hasil pengujian pada setiap generasi pada Gambar 6.2 dapat diketahui dari grafik tersebut bahwa jumlah generasi mempengaruhi nilai *fitness* dari proses Algoritma Genetika. Nilai paling rendah diperoleh pada generasi 250 karena Algoritma Genetika belum melakukan proses genetika secara optimal. Akan tetapi pengujian dengan banyak generasi juga belum tentu bisa dikatakan hasilnya akan menjadi lebih baik. Semakin banyaknya generasi nilai *fitness* yang dihasilkan akan cenderung sama sehingga terjadi konvergensi, apabila terus dilakukan penambahan generasi tidak akan menghasilkan solusi yang lebih baik dan cenderung akan membuang waktu (Mahmudy, 2013). Pola seperti ini juga didapatkan oleh Utomo pada tahun 2014 yaitu penyelesaian penjadwalan *Flexible Job Shop Problem* dengan menggunakan *Real Coded Genetic Algorithm*, percobaan uji ukuran generasi yang telah dilakukan dapat dilihat bahwa pada generasi 1500 merupakan titik stabil nilai *fitness* yaitu 6,91465 dan selanjutnya dari generasi 1500 sampai generasi 2000 tidak terjadi peningkatan rata-rata nilai *fitness* yang signifikan.

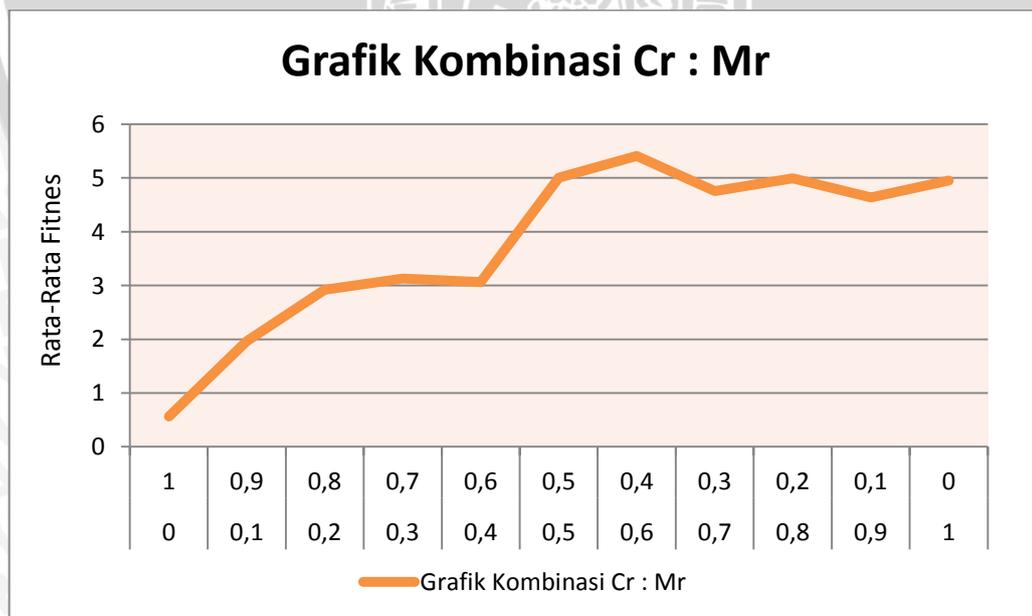
6.3 Hasil dan Analisa Uji Coba Kombinasi *Crossover Rate* dan *Mutation Rate*

Uji coba selanjutnya yaitu dilakukan pengujian kombinasi *crossover rate* dan *mutation rate* terhadap perubahan nilai *fitness*. Ukuran populasi (*popSize*) yang digunakan adalah *popSize* terbaik yang telah didapat pada uji coba ukuran populasi sebelumnya yaitu 175 *popSize*. Jumlah generasi yang digunakan adalah generasi yang menghasilkan nilai *fitness* yang tertinggi sesuai dengan percobaan sebelumnya pada Tabel 6.2 yaitu 1500 generasi. Untuk data yang digunakan dalam pengujian kombinasi *crossover rate* dan *mutation rate* ini sebanyak 160 data konsumsi kWh untuk prediksi 4 bulan kebelakang mulai tahun 2012 sampai 2014. Sedangkan untuk nilai *crossover rate* dan *mutatuion rate* yang digunakan dalam pengujian adalah kombinasi *cr* dan *mr* yang memiliki kombinasi yang nanti hasilnya sama, pada percobaan ini kombinasi *cr* dan *mr* berjumlah 1. Setiap kombinasi *cr* dan *mr* yang diuji akan dilakukan 10 kali percobaan dan akan diperoleh nilai rata-rata nilai *fitness* optimal serta dapat memberikan solusi untuk permasalahan ini. Untuk hasil percobaan ukuran generasi ditunjukkan pada Tabel 6.3.

Tabel 6.3 Hasil Uji Coba Kombinasi *Crossover Rate* dan *Mutation Rate*

Kombinasi		Nilai Fitness										Rata-rata Nilai Fitness
cr	mr	Percobaan Generasi ke-										
		1	2	3	4	5	6	7	8	9	10	
0	1	0,731	0,462	0,776	0,504	0,426	0,615	0,660	0,445	0,510	0,487	0,562
0,1	0,9	1,842	1,307	0,627	1,173	3,334	1,458	3,462	0,594	3,333	2,522	1,965
0,2	0,8	4,981	5,341	2,461	2,970	1,284	1,977	1,264	1,949	0,909	6,096	2,923
0,3	0,7	1,471	3,796	6,795	2,168	2,594	1,381	1,789	3,418	2,712	5,147	3,127
0,4	0,6	2,442	7,702	1,645	4,709	4,724	1,622	4,634	0,651	0,982	1,536	3,065
0,5	0,5	2,658	3,420	10,47	4,523	4,937	3,555	3,527	6,132	3,044	7,782	5,005
0,6	0,4	3,019	6,343	3,307	3,892	9,997	4,021	5,620	12,08	2,937	2,870	5,409
0,7	0,3	4,676	2,544	8,589	9,415	7,240	3,229	4,001	2,097	1,951	3,832	4,757
0,8	0,2	2,632	4,789	9,8085	5,761	5,539	2,076	4,570	2,018	7,734	5,010	4,994
0,9	0,1	6,765	7,473	2,991	4,516	1,879	7,440	4,283	5,675	2,132	3,213	4,637
1	0	3,331	6,979	4,36	5,775	4,366	5,722	6,864	3,261	4,420	4,414	4,949

Hasil uji coba kombinasi *crossover rate* dan *mutation rate* terhadap nilai *fitness* pada Tabel 6.3 akan diformulasikan dalam sebuah grafik untuk dapat melihat perbedaan dari hasil pengujian kombinasi *crossover rate* dan *mutation rate* terhadap nilai *fitness*. Grafik hasil uji coba ukuran generasi ditunjukkan pada Gambar 6.3.

Gambar 6.3 Grafik Uji Coba Kombinasi *Crossover Rate* dan *Mutation Rate*

Pada grafik Gambar 6.3 dapat dilihat bahwa kombinasi *crossoverrate* dan *mutationrate* berpengaruh terhadap perubahan nilai *fitness*. Dari Gambar 6.3 grafik tersebut menunjukkan bahwa rata-rata *fitness* terendah ada pada kombinasi *crossover rate* sama dengan 0 dan *mutation rate* sama dengan 1 yaitu 0,562. Sedangkan rata-rata nilai *fitness* tertinggi ada pada kombinasi *crossover rate* sama dengan 0,6 dan *mutation rate* sama dengan 0,4 yaitu 5,4098.

6.4 Hasil dan Analisa Uji Coba Banyaknya Periode Konsumsi kWh (dalam Bulan)

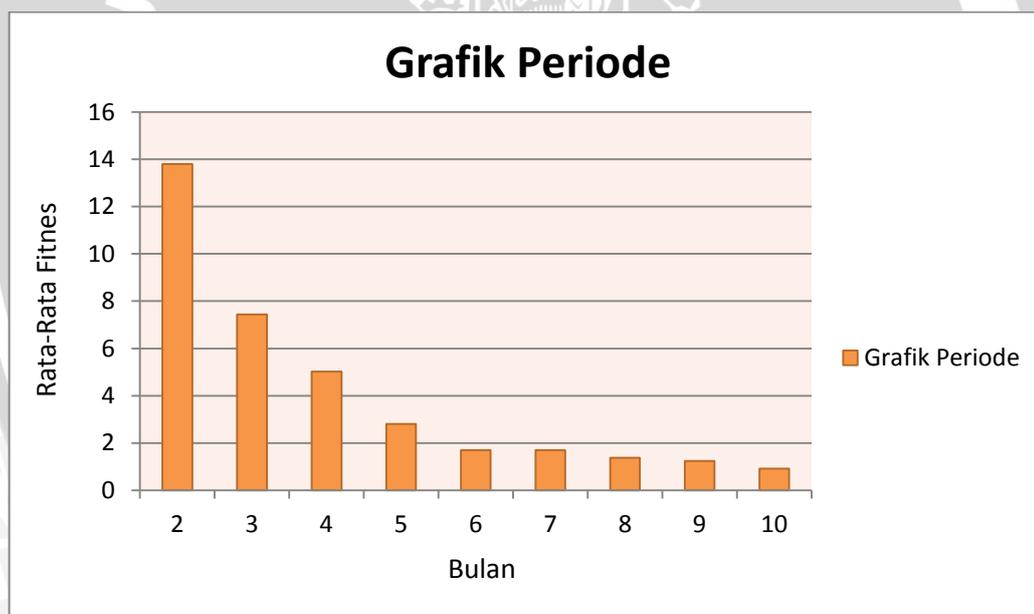
Pada uji coba yang terakhir adalah uji coba banyaknya periode konsumsi kWh (dalam Bulan) terhadap perubahan nilai *fitness*. Ukuran populasi (*popSize*) yang digunakan adalah *popSize* terbaik yang telah didapat pada uji coba ukuran populasi sebelumnya yaitu 175 *popSize*. Jumlah generasi yang digunakan adalah generasi yang menghasilkan nilai *fitness* yang tertinggi sesuai dengan percobaan sebelumnya pada Tabel 6.2 yaitu 1500 generasi. Kombinasi nilai *crossover rate* dan *mutation rate* yang digunakan merupakan kombinasi *cr* dan *mr* terbaik terhadap nilai *fitness* yang optimal pada pengujian sebelumnya yaitu *cr* = 0,6 dan *mr* = 0,4. Untuk data yang di uji menggunakan data konsumsi kWh dari tahun 2012 sampai 2014 dan diformulasikan menjadi data periode yang akan diuji. Bentuk data yang digunakan dimulai dari prediksi dari 2 bulan ke belakang sampai 10 bulan ke belakang. Setiap periode yang diuji akan dilakukan 10 kali percobaan dan akan diperoleh nilai rata-rata nilai *fitness* optimal terhadap percobaan periode konsumsi kWh. Untuk hasil percobaan banyaknya periode konsumsi ditunjukkan pada Tabel 6.4.

Tabel 6.4 Hasil Uji Banyak Periode

Banyak Periode	Nilai Fitness										Rata-rata	Rata-rata
	Percobaan generasi ke-										nilai	Waktu
	1	2	3	4	5	6	7	8	9	10	<i>fitness</i>	(ms)
2	16,710	16,741	3,677	16,652	16,711	16,739	16,385	3,631	16,428	14,334	13,79	8200
3	5,881	2,162	9,529	12,304	16,074	2,319	5,257	5,583	6,321	8,988	7,44	11500
4	4,012	4,914	11,06	1,604	0,803	4,131	1,068	2,634	17,565	2,406	5,01	13200
5	1,633	4,591	1,476	1,005	4,212	0,951	1,636	2,186	7,809	2,504	2,800	14300
6	3,673	0,963	2,636	0,729	1,243	2,423	1,374	2,151	1,077	0,738	1,700	16700

	Nilai Fitness										Rata-rata	Rata-rata
Banyak	Percobaan generasi ke-										nilai	Waktu
Periode	1	2	3	4	5	6	7	8	9	10	<i>fitness</i>	(ms)
7	1,165	1,241	2,579	0,990	2,808	1,331	1,417	1,158	2,090	2,258	1,702	18200
8	1,350	1,170	2,335	0,801	1,316	1,129	1,352	0,760	2,216	1,335	1,37	22200
9	1,320	0,868	1,897	0,720	2,112	0,845	0,998	1,698	0,624	1,347	1,24	26500
10	0,813	1,060	1,136	0,906	0,538	0,919	0,854	0,929	1,055	0,959	0,916	36800

Hasil uji coba banyaknya periode (dalam Bulan) terhadap rata-rata nilai *fitness* pada Tabel 6.4 akan diformulasikan dalam sebuah grafik untuk mempermudah melihat perbedaan dari hasil pengujian banyaknya periode konsumsi kWh terhadap nilai *fitness*. Grafik hasil uji coba banyaknya periode konsumsi kWh ditunjukkan pada Gambar 6.4



Gambar 6.4 Grafik Hasil Uji Coba Banyaknya Periode

Pada Gambar 6.4 hasil uji coba untuk ukuran populasi terhadap hasil *fitness* cenderung mengalami penurunan dimulai dari periode 2 bulan kebelakang hingga 10 bulan kebelakang, nilai rata-rata *fitness* paling rendah ada pada periode 10 bulan yaitu 0,916 Sedangkan nilai rata-rata *fitness* paling tinggi ada pada periode 2 bulan yaitu 13,79.

Semakin banyak periode yang akan diuji dapat mempengaruhi nilai *fitness* yang dihasilkan dan semakin rendah pula nilai *fitness*-nya. Akan tetapi semakin banyak periode yang akan diuji maka waktu yang dibutuhkan untuk proses Algoritma Genetika juga semakin besar.

6.5 Hasil dan Analisa Perbandingan Hasil Regresi dengan Algoritma Genetika

Setelah mendapatkan parameter-parameter terbaik pada Algoritma Genetika pada permasalahan prediksi konsumsi kWh ini selanjutnya akan dianalisa lebih lanjut lagi dan dibandingkan terhadap perhitungan manual regresi. Solusi terbaik yang diberikan pada sistem ini yaitu pada ukuran *popSize* sebesar 175, generasi terbaik sebanyak 1500 generasi, kombinasi nilai *crossover rate* dan *mutation rate* terbaik yaitu 0,6 pada *crossover rate* dan 0,4 pada *mutation rate*.

Lalu hasil dari sistem tersebut akan dibandingkan dengan rumus regresi dengan perhitungan excel manual pada periode 2 bulan. Data sampel yang diambil untuk testing perbandingan juga menggunakan pemakaian pada periode 2 bulan.

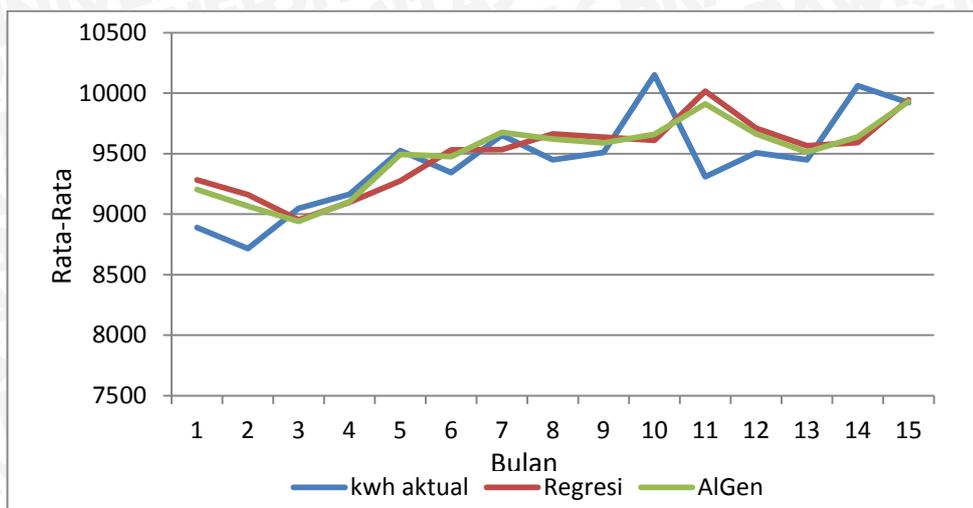
Pada rumus regresi yang dihasilkan oleh data analisis regresi linear Microsoft Excel menghasilkan Persamaan (6-1).

$$Y' = 878,483 + 0,605X_1 + 0,315X_2 \quad (6-1)$$

Sedangkan pada sistem Algoritma Genetika menghasilkan Persamaan (6-2).

$$Y' = 4,27 + 0,672X_1 + 0,335X_2 \quad (6-2)$$

Perbandingan tersebut akan ditampilkan pada grafik yang ditunjukkan pada Gambar 6.5.



Gambar 6.5 Perbandingan Tingkat Error Regresi dengan Algoritma Genetika

Dari grafik tersebut dapat disimpulkan tingkat *error* regresi dengan total *error* sebanyak 1.610.896 dengan Algoritma Genetika dengan total *error* sebanyak 1.101.826 menghasilkan selisih yang tidak terlalu jauh dengan konsumsi kWh aktual, dan Algoritma Genetika bisa dikatakan dapat memberikan solusi lebih baik pada perhitungan prediksi konsumsi kWh ini. Detail *error* pada regresi dan Algoritma Genetika akan ditunjukkan pada Tabel 6.5

Tabel 6.5 Detail *Error* Regresi dan *Error* Algoritma Genetika

No	kWh Aktual	kWh Regresi	kWh GA	Error	
				Regresi	GA
				$(Y_i - Y'_i)^2$	
1	8890	9281,965	9203,383	153636,413	98208,905
2	8715	9162,050	9066,715	199853,986	123703,441
3	9047	8952,570	8938,9	8917,091	11685,610
4	9165	9098,529	9103,379	4418,438	3797,148
5	9526	9274,440	9493,895	63282,263	1030,731
6	9344	9530,151	9476,017	34652,210	17428,488
7	9653	9533,540	9674,648	14270,578	468,636
8	9449	9663,371	9621,326	45954,783	29696,250
9	9509	9637,078	9587,753	16403,849	6202,035
10	10152	9609,218	9659,733	294611,952	242326,799
11	9310	10017,428	9911,929	500453,782	362318,521
12	9508	9709,925	9661,51	40773,599	23565,320
13	9450	9564,876	9512,496	13196,604	3905,750

No	kWh Aktual	kWh Regresi	kWh GA	Error Regresi	Error GA
14	10061	9592,059	9639,85	$(Y_i - Y'_i)^2$	177367,323
15	9920	9943,764	9931,012	564,709	121,264
$sumerror = (Y_i - Y'_i) =$				1610896,158	1101826,221
$\epsilon = \frac{1}{n} \sqrt{sumerror} =$				84,614	69,979

Error Regresi = $(1610896/141699) * 100\% = 11,36\%$

Error Regresi Algoritma Genetika = $(1101826/141699) * 100\% = 7,7\%$

Berdasarkan hasil pengujian perbandingan tingkat *error* regresi dengan *error* regresi yang dibangun oleh sistem prosentase *error* yang dihasilkan yaitu 11,36% pada regresi dan 7,7% pada regresi yang dihasilkan oleh Algoritma Genetika.



BAB VII PENUTUP

7.1 Kesimpulan

Kesimpulan yang dihasilkan dari hasil uji coba yang telah dilakukan dalam penerapan Algoritma Genetika untuk pemodelan regresi linear dalam konsumsi kWh listrik adalah sebagai berikut :

1. Algoritma Genetika dapat diterapkan untuk pemodelan regresi linear dengan menemukan koefisien-koefisien terbaik untuk tiap variabel regresi. Dalam penelitian ini representasi kromosom yang digunakan untuk merepresentasikan solusi adalah representasi *Real Code Genetic Algorithm* (RCGA) dengan rentang nilai yang dibangkitkan antara $[-100, 100]$. Panjang kromosom yang digunakan untuk penyelesaian masalah ini adalah kromosom dengan panjang antara 2 – 10 gen yang berisi data konsumsi kWh listrik dalam rentang waktu perbulan. Nilai *fitness* dihitung dengan konstanta sebesar 1000000 dibagi dengan nilai *error*.
2. Parameter Algoritma Genetika juga berpengaruh terhadap hasil solusi yang diberikan. Nilai parameter yang besar belum tentu menghasilkan solusi yang terbaik. Dalam sistem pemodelan regresi untuk konsumsi kWh listrik dengan menggunakan Algoritma Genetika, ukuran populasi terbaik adalah sebanyak 175 *popSize* dengan rata-rata nilai *fitness* 3,86183, generasi yang paling mendekati solusi terbaik adalah sebanyak 1500 generasi dengan rata-rata nilai *fitness* 3,75296, serta nilai kombinasi tertinggi adalah *crossover rate* sama dengan 0,6 dan *mutation rate* sama dengan 0,4 dengan rata-rata nilai *fitness* 3,7711.
3. Koefisien yang dihasilkan oleh sistem menghasilkan nilai *error* yang lebih rendah dibandingkan dengan persamaan regresi, dan pada sistem Algoritma Genetika ini dapat memberikan solusi yang lebih baik dengan tingkat akurasi nilai *error* koefisien regresi yang dihasilkan oleh sistem Algoritma Genetika adalah sebesar 7,7%.

7.2 Saran

Berdasarkan kesimpulan diatas, ada beberapa point yang perlu diperhatikan jika pembaca ingin melakukan penelitian lebih lanjut.

1. Algoritma Genetika merupakan algoritma pencarian heuristik yang cukup *powerfull*, tidak hanya Algoritma Genetika saja terdapat beberapa Algoritma Evolusi yang sejenis namun dengan model konsep yang berbeda tetapi tetap dalam satu kesatuan algoritma pencarian heuristik, misalnya Algoritma Evolusi Strategi, *Genetic Programming*, dan *Evolution Programming*. Pada permasalahan yang telah ditelaah oleh peneliti algoritma-algoritma evolusi tersebut bisa menyelesaikan permasalahan peneliti.
2. Titik optimal pada uji coba populasi, uji coba generasi, dan uji coba kombinasi *cr* dan *mr* terjebak dalam kisaran yang sama. Untuk memperluas area pencarian algoritma genetika dalam kasus prediksi konsumsi kWh listrik dengan pemodelan regresi linear dapat diterapkan metode *crossover*, mutasi, dan seleksi yang lain.
3. Semakin besar periode yang digunakan, prediksi konsumsi kWh listrik di akan menjauhi harga aktual. Agar mendekati harga aktual, bisa dilakukan pengujian periode menggunakan data campuran harian ataupun mingguan.
4. Saran selanjutnya agar nilai *error* mendekati minimum bisa dilakukan dengan mengubah rentang kromosom yang dibangkitkan secara acak, *range* kromosom diuji terlebih dahulu nilai yang dihasilkan akan mengarah pada bilangan real positif ataukah pada bilangan real negatif, maka akan mendapatkan *range* kromosom yang optimal untuk koefisien regresi.

Daftar Pustaka

- Bali, P. P. (2000). *Teori Dasar Listrik*. Jawa Bali: PT. PLN (Persero) UBS P3B.
- Eka Risky Formansyah, S. S. (2012). *Algoritma Genetika*. Jakarta: Universitas Islam Negeri Syarif Hidayatullah.
- Khair, Aulia. (2011). Peramalan Beban Listrik Jangka Pendek Menggunakan Kombinasi *Metode Autoregressive Integrated Moving Average (ARIMA)* Dengan Regresi Linear Antara Suhu dan Daya Listrik. *FT-UI*, 47-86.
- Sari, Ira Puspita T. W. (2014). Prediksi Data Harga Saham Harian Menggunakan Feed Forward Neural Networks (FFNN) Dengan Pelatihan Algoritma Genetika (Studi Kasus pada Harga Saham Harian PT. XL Axiata Tbk). *Jurnal Gaussian, Volume 3, Nomer 3*, 441-450.
- Ervin, J. S. (1999). Using Heteroscedasticity Consistent Standard Errors in the Linear Regression Model. 1-21.
- Harifuddin. (2007). Estimasi Kebutuhan Daya Listrik Sulawesi Selatan Sampai Tahun 2007. *Media Elektrik*, 14-23.
- Kurniawan, D. (2008). *Regresi Linear (Linear Regression)*. Vienna, Austria: R. Development Core Team.
- Mahmudy, W. F. (2013). *Algoritma Evolusi*. Malang: Program Teknologi Informasi dan Ilmu Komputer UB. (Di akses pada Tanggal 17 Januari 2015)
- Oktiyanti. (2011). *Materi Dasar Teori Listrik Untuk SMK*. WordPress.com.
- Darjat Saripurna, J. H. (2012). Optimasi Pengolahan KWH Meter pada PT. PLN (PERSERO) Menggunakan Genetic Algoritm. *SNASTIKOM 2012*, 85-88.
- Sriwati. (2013). Prakiraan Kebutuhan Daya Listrik di Kabupaten Maros Tahun 2010 Sampai 2020. *ILTEK Volume 8*, 1025-1030.

(Syahputra, 2008) Syahputra, L. (2008). *Menentukan Regresi Linear Berganda Untuk Pendapatan Domestik Regional Bruto (PDRB) Di Deli Serdang*. Medan: USU Repository.

Tanjung, W. S. (2010). *Kajian Algoritma Genetika Pada Travelling Salesman Problem*. Medan: Universitas Sumatera Utara.

Zagy. (2010). *Energi dan Daya Listrik*. Wordpress.com. (Di akses pada Tanggal 10 Februari 2015).

Utomo, M.C.C., Mahmudy, W. F. (2014). Penyelesaian Penjadwalan Flexible Job Shop Problem dengan menggunakan Real Code Genetic Algorithm. *DORO : Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya*, vol. 3, no. 13, 1-5.

Pratiwi, Monica Intan. Mahmudy, W. F. Dewi, Candra (2014). Implementasi Algoritma Genetika Pada Optimasi Biaya Pemenuhan Kebutuhan Gizi. *DORO : Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya*, vol. 4, no. 6, 1-8.

Yonathan, M. (2013). Peranan Dinas Pariwisata Dan Kebudayaan Kota Batu Dalam Kegiatan Promosi Pariwisata Kota Batu. *DORO : Repository Jurnal Mahasiswa Fakultas Ekonomi Jurusan Manajemen Universitas Brawijaya* vol 2, no. 6, 1-17.

Daftar Lampiran

