



TESIS - KI 092361

**PREDIKSI HARGA EMAS MENGGUNAKAN
METODE *GENERALIZED REGRESSION NEURAL
NETWORK* DAN ALGORITMA GENETIKA**

GITA INDAH MARTHASARI
5111 202 002

DOSEN PEMBIMBING
Prof. Ir. Arif Djunaidy, M.Sc., Ph.D.

PROGRAM MAGISTER
BIDANG KEAHLIAN SISTEM INFORMASI
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2014



THESIS - KI 092361

GOLD PRICE PREDICTION USING GENERALIZED REGRESSION NEURAL NETWORK AND GENETIC ALGORITHM

**GITA INDAH MARTHASARI
5111 202 002**

**SUPERVISOR
Prof. Ir. Arif Djunaidy, M.Sc, Ph. D.**

**MASTER PROGRAM
MAJORING IN INFORMATION SYSTEMS
DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2014**

LEMBAR PENGESAHAN TESIS

**PERAMALAN HARGA EMAS MENGGUNAKAN METODE
GENERALIZED REGRESSION NEURAL NETWORK DAN
ALGORITMA GENETIKA**

**Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom)**

Di

Institut Teknologi Sepuluh Nopember

Oleh :

Gita Indah Marthasari

NRP : 5111202002

Tanggal ujian : 5 Februari 2014

Periode wisuda : Agustus 2014

Disetujui oleh :

1. Prof. Ir. Arif Djunaidy, M.Sc., Ph.D.
NIP. 19581005 198603 1 003

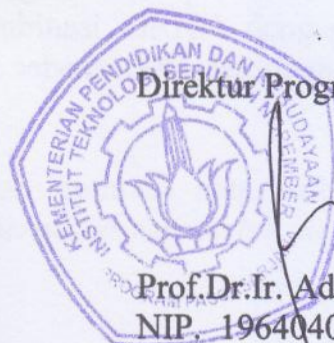
(Pembimbing)

2. Dr. Eng. Febriliyan Samopa, M.Kom.
NIP. 19730219 1998021 001

(Penguji I)

3. Mahendrawathi E.R, S.T, M.Sc., PhD.
NIP. 19761011 2006042 001

(Penguji II)



Direktur Program Pasca Sarjana,

Prof.Dr.Ir. Adi Soeprijanto, MT
NIP. 19640405 199002 1 001

PREDIKSI HARGA EMAS MENGGUNAKAN METODE *GENERALIZED REGRESSION NEURAL NETWORK* DAN ALGORITMA GENETIKA

Nama mahasiswa : Gita Indah Marthasari
NRP : 51 11 202 002
Pembimbing : Prof. Ir. Arif Djunaidy, M.Sc., Ph.D

ABSTRAK

Prediksi harga emas merupakan aktivitas penting bagi banyak pihak. Salah satu metode prediksi harga emas yang dapat digunakan adalah jaringan syaraf tiruan berbasis *generalized regression neural network* (GRNN). Dalam penelitian sebelumnya, GRNN digabungkan dengan teknik dekomposisi *Seasonal Trend Decomposition based on Locally Weighted Regression* (STL) dan metode theta. Kinerja GRNN dipengaruhi oleh data latih yang digunakan karena ukuran jaringan yang terbentuk akan berbanding lurus dengan jumlah data latih. Untuk mengatasi meningkatnya ukuran jaringan seiring dengan bertambahnya data latih, proses reduksi data latih tanpa mengurangi akurasi prediksi perlu dilakukan.

Dalam penelitian ini, metode peramalan GRNN diintegrasikan dengan algoritma genetika untuk mereduksi data latih guna menghasilkan model peramalan yang lebih efisien. Sebelum diramalkan, data harga emas didekomposisi menggunakan STL menjadi komponen data musiman, tren, dan residual. Ketiga komponen data tersebut diramalkan menggunakan dua metode yang berbeda, yaitu GRNN untuk meramalkan komponen data musiman dan residual, dan metode theta untuk meramalkan komponen data tren. Hasil peramalan dari ketiga komponen tersebut selanjutnya digabungkan menggunakan jaringan syaraf tiruan propagasi balik untuk memperoleh hasil peramalan akhir.

Hasil pengujian menunjukkan bahwa GRNN yang diintegrasikan dengan algoritma genetika, selain mampu menghasilkan peramalan dengan akurasi yang setara dengan GRNN tanpa algoritma genetika, juga mampu memberikan akurasi yang lebih baik dibandingkan dengan hasil peramalan menggunakan model peramalan Arima. Selain itu, kombinasi GRNN dengan algoritma genetika mampu mereduksi jumlah data latih sebesar 50% dan mampu mengurangi waktu proses peramalan sebesar 34%.

Kata kunci: peramalan harga emas, optimasi data latih, algoritma genetika, dekomposisi data runut waktu, *generalized regression neural network*, metode theta

PREDICTION OF GOLD PRICE BY USING THE GENERALIZED REGRESSION NEURAL NETWORK METHOD AND GENETIC ALGORITHM

By : Gita Indah Marthasari
Student Number : 51 11 202 002
Supervisor : Prof. Ir. Arif Djunaidy, M.Sc, Ph.D

ABSTRACT

The prediction of gold price is an important activity for all parties. One of the gold prediction methods that can be used is the artificial neural network based on the generalized regression neural network (GRNN). In the previous research, GRNN was combined with the decomposition technique of Seasonal Trend Decomposition based on locally weighted regression (STL) and the Theta method. The GRNN performance was influenced by the training data size used, since the network size formed is proportionally dependent on the training data size. To cope with the increasing of the network size along with the increasing of the training data size, the process of training data reduction that is capable of maintaining the accuracy of the prediction is interesting to investigate.

In this research, the GRNN prediction method is integrated with the genetic algorithm in order to reduce the training data size that will in turn produce the more efficient prediction model. Initially, the gold price data is decomposed into three components; i.e., seasonal, trend, and residual data using STL decomposition technique. Those three components are then predicted using two different methods, namely GRNN to predict the component of the seasonal and residual data, and the theta method to predict the trend data component. The prediction results of those three components are combined together using the back propagation neural network algorithm to obtain final results of the prediction.

Experimental results showed that the GRNN method that was integrated with the genetic algorithm was not only capable of producing prediction results with the accuracy similar to those produced using the original GRNN method, but also capable of giving the better prediction accuracy in compared to that produced using ARIMA prediction model. In addition to that, the combination of GRNN with the genetic algorithm is also capable of reducing the amount of the training data as much as 50% and reducing the computing time consumed by the prediction process as much as 34%.

Keywords : gold price forecasting, training data optimization, genetic algorithm, time series data decomposition, generalized regression neural network, theta method

KATA PENGANTAR

Alhamdulillahirobbil'alaamin penulis panjatkan ke hadirat Allah SWT karena atas limpahan hidayah dan petunjuk-Nya laporan tesis yang berjudul **“Prediksi Harga Emas menggunakan Metode Generalized Regression Neural Network dan Algoritma Genetika”** yang merupakan salah satu syarat kelulusan pada Magister Teknik Informatika, Fakultas Teknologi Informatika, Institut Teknologi Sepuluh Nopember Surabaya dapat diselesaikan. Tak lupa shalawat dan salam penulis haturkan kepada Sang Pembawa Risalah Nabi Muhammad SAW.

Selama proses pengerjaan, penulis mendapatkan banyak bantuan dan dukungan baik moril maupun materiil hingga penelitian ini dapat selesai dengan baik. Atas berbagai bantuan dan dukungan tersebut penulis menghaturkan terima kasih sebesar-besarnya kepada :

- Abi Dian Ferianto dan anak-anakku tercinta, Hana Syahidah Ferianto dan Fahmida Sabrina Ferianto yang telah memberikan doa, dukungan, dan pengertian yang luar biasa selama penulis menyelesaikan studi di ITS
- Bapak Prof. Ir. Arif Djunaidy, M.Sc, Ph.D selaku pembimbing penelitian, Bapak Dr. Eng. Febriliyan Samopa, S.Kom, M.Kom dan Ibu Mahendrawathi ER., S.T, M.Sc., Ph.D selaku penguji sidang penelitian
- Kedua orang tua serta keluarga besar penulis
- Seluruh rekan-rekan pasca sarjana Sistem Informasi angkatan 2011 serta rekan-rekan di tempat kerja penulis
- Seluruh dosen dan staf di Jurusan Sistem Informasi ITS

Dan semua pihak yang tak dapat disebutkan satu per satu di sini. Semoga Allah memberikan balasan berlipat ganda atas segala yang telah diberikan kepada penulis. Penulis menyadari bahwa masih banyak yang dapat dikembangkan dari penelitian ini. Oleh karena itu, penulis menerima segala bentuk kritik dan saran yang disampaikan. Semoga penelitian ini dapat memberikan manfaat bagi bangsa dan agama.

Surabaya, Februari 2014

Penulis

DAFTAR ISI

ABSTRAK.....	vii
ABSTRACT.....	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
BAB 1 PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Perumusan Masalah.....	5
1.3. Tujuan dan Manfaat.....	6
1.4. Batasan dan Asumsi.....	7
1.5. Kontribusi Penelitian.....	7
BAB 2 LANDASAN TEORI DAN KAJIAN PUSTAKA.....	9
2.1. Peran Emas di Sektor Perekonomian.....	9
2.2. Peramalan Runut Waktu.....	11
2.2.1. Jenis-jenis Data Runut Waktu.....	14
2.2.2. Beberapa Metode Peramalan.....	14
2.3. Jaringan Syaraf Tiruan dalam Peramalan.....	17
2.4. <i>Generalized Regression Neural Network</i> (GRNN).....	19
2.5. Dekomposisi Data Runut Waktu.....	22
2.5.1. <i>Seasonal Trend Decomposition based on Loess</i>	23
2.5.2. <i>Locally Weighted Regression</i> (LOESS).....	25
2.6. Metode Theta.....	26
2.7. Algoritma Genetika.....	28
2.8. Penelitian-penelitian Terkait.....	31
BAB 3 METODOLOGI PENELITIAN.....	35
3.1. Perancangan Model Peramalan.....	36
3.1.1. Dekomposisi STL.....	38
3.1.2. Peramalan Komponen Data Musiman dan Residual.....	39
3.1.3. Peramalan Komponen Tren.....	44
3.1.4. Penggabungan menggunakan Algoritma JST Propagasi Balik.....	45

3.2.	Pengujian dan Analisis Hasil.....	46
3.2.1.	Data Uji Coba.....	46
3.2.2.	Lingkungan Uji Coba.....	47
3.2.3.	Jenis Uji Coba	47
3.2.4.	Analisis Hasil Uji Coba.....	48
BAB 4	UJI COBA DAN ANALISIS HASIL	51
4. 1.	Lingkungan Uji Coba.....	51
4. 2.	Data Uji Coba.....	51
4. 3.	Jenis Uji Coba	52
4.3.1.	Uji Coba Penentuan Parameter Model Peramalan	52
4.3.2.	Uji Coba Perbandingan Metode Peramalan	52
4. 4.	Pelaksanaan dan Hasil Uji Coba	53
4.4.1.	Pelaksanaan Uji Coba Penentuan Parameter Model Peramalan.....	53
4.4.2.	Pelaksanaan Uji Coba Beberapa Metode Peramalan.....	63
4. 5.	Analisis Hasil Uji Coba.....	65
4.5.1.	Analisis Uji Coba Penentuan Parameter Model Peramalan	65
4.5.2.	Analisis Uji Coba Perbandingan Metode Peramalan	72
BAB 5	PENUTUP.....	77
5.1.	Kesimpulan	77
5.2.	Saran.....	79
DAFTAR PUSTAKA		80
BIOGRAFI PENULIS		87

DAFTAR GAMBAR

Gambar 2.1 Grafik harga emas dalam USD (Sumber : www.gold.org)	12
Gambar 2.2 Arsitektur jaringan syaraf runut maju (Suhartono, 2005)	16
Gambar 2.3 Arsitektur GRNN (Cigizoglu, 2005).....	20
Gambar 2.4 Hasil pendataran dengan model Theta (Theodosiou, 2011b).....	27
Gambar 2.5 Hasil perbesaran dengan model Theta (Theodosiou, 2011b).....	27
Gambar 3.1 Metodologi Penelitian	35
Gambar 3.2 Perancangan model peramalan.....	36
Gambar 3.3 Model peramalan GRNN-AG (langkah II)	37
Gambar 3.4 Kode Program Dekomposisi STL	38
Gambar 3.5 Hasil dekomposisi harga emas harian menggunakan metode STL.....	39
Gambar 3.6 Pembentukan matriks dari data runut waktu	40
Gambar 3.7 Kode program untuk mengubah data seri waktu menjadi matriks.....	41
Gambar 3.8 Kode program pembangkitan kromosom.....	42
Gambar 3.9 Kode program evaluasi kromosom	43
Gambar 3.10 Kode program proses seleksi	43
Gambar 3.11 Kode program proses rekombinasi.....	44
Gambar 3.12 Kode program proses mutasi.....	44
Gambar 3.13 Implementasi metode theta	45
Gambar 3.14 Kode program rekombinasi.....	46
Gambar 4.1 Hasil uji coba penentuan nilai penyebaran untuk pemodelan komponen residual	58
Gambar 4.2 Hasil uji penentuan jumlah simpul masukan GRNN untuk komponen data musiman	66
Gambar 4.3 Hasil uji coba penentuan populasi GRNN-AG untuk komponen data musiman	67
Gambar 4.4 Hasil uji coba probabilitas mutasi pada pemodelan komponen data musiman	68
Gambar 4.5 Hasil uji coba generasi untuk komponen data musiman	68
Gambar 4.6 Hasil uji coba jumlah simpul masukan dan nilai penyebaran pemodelan komponen data residual menggunakan GRNN	70
Gambar 4.7 Hasil uji populasi GRNN-AG untuk komponen data residual	71

Gambar 4.8 Hasil uji probabilitas mutasi untuk komponen data residual.....	71
Gambar 4.9 Hasil uji coba generasi GRNN-AG untuk komponen data residual.....	72
Gambar 4.10 Perbandingan hasil pengujian peramalan harga emas harian	73
Gambar 4.11 Hasil validasi peramalan harga emas Jan – Des 2013	74
Gambar 4.12 Hasil validasi menggunakan data Jan - Des 2012	75

DAFTAR TABEL

Tabel 4.1 Hasil pengujian penentuan jumlah simpul masukan GRNN untuk komponen data musiman	54
Tabel 4.2 Hasil pengujian penentuan jumlah populasi untuk komponen data musiman ..	55
Tabel 4.3 Hasil uji coba penentuan probabilitas mutasi untuk komponen data musiman	56
Tabel 4.4 Hasil uji coba penentuan jumlah generasi untuk komponen data musiman	59
Tabel 4.5 Hasil uji coba penentuan jumlah masukan GRNN untuk komponen residual..	59
Tabel 4.6 Hasil uji coba penentuan populasi untuk komponen residual	60
Tabel 4.7 Hasil uji coba penentuan probabilitas mutasi untuk komponen residual	61
Tabel 4.8 Hasil uji coba penentuan generasi untuk komponen residual	62
Tabel 4.9 Uji coba STL-GRNN-AG untuk komponen data musiman	64
Tabel 4.10 Uji coba STL-GRNN-AG untuk komponen data residual	64
Tabel 4.11 Hasil uji coba beberapa metode peramalan	65

BAB 1 PENDAHULUAN

1.1. Latar Belakang

Emas merupakan sebuah aset dasar dari investasi atau portofolio. Sepanjang sejarah, emas telah dikenal sebagai mata uang yang dapat diperdagangkan setiap saat dan dalam kondisi apapun (Parisi, dkk, 2008). Emas telah terbukti sebagai simpanan tunai yang efektif selama krisis saham pada tahun 1987, dan krisis moneter Asia pada tahun 1997 dan 1998. Selama beberapa tahun terakhir, makin banyak investor yang tertarik berinvestasi pada komoditas ini. Fenomena ini antara lain disebabkan oleh adanya ketidak-stabilan nilai mata uang resmi dan tren nilai emas yang meningkat (Hussein, dkk, 2011). Para investor tertarik untuk berinvestasi pada komoditas yang nilainya relatif aman dan stabil. Sebagai salah satu aset keuangan, emas menawarkan jaminan berkaitan dengan pergerakan nilai yang ekstrim dibandingkan dengan jenis aset lainnya. Beberapa alasan yang mendukung hal ini adalah untuk keanekaragaman portofolio, ketahanan terhadap inflasi, dan kestabilan nilai tukar (www.gold.org).

Harga emas dari tahun ke tahun menunjukkan tren peningkatan. Hal ini dapat diamati dari grafik harga emas dimana terdapat tren peningkatan harga dari tahun ke tahun. Selain itu, harga emas juga memiliki komponen musiman (*seasonality*) yaitu periode waktu dimana harga emas cenderung tinggi. Menurut Linde (2011), periode kenaikan harga emas terjadi pada kuartal keempat setiap tahunnya yaitu mulai bulan September hingga Desember. Menurutnya, hal ini didorong oleh peningkatan permintaan akan emas oleh masyarakat di India yang turut mempengaruhi naiknya harga emas.

Pengetahuan terhadap harga emas di masa mendatang merupakan sebuah hal penting bagi banyak pihak. Witjaksono (2010) melakukan penelitian untuk mengetahui faktor-faktor yang mempengaruhi indeks harga saham gabungan (IHSG) di Bursa Efek Indonesia. Salah satu variabel yang diteliti adalah harga emas. Berdasarkan pengujian, variabel harga emas merupakan faktor dominan dalam perubahan IHSG. Wang, dkk (2012) yang meneliti mengenai hubungan

antara harga emas dan harga minyak mentah menemukan bahwa keduanya saling mempengaruhi. Hussein, dkk (2011) dan Sopipan, dkk (2012) menunjukkan potensi emas sebagai bentuk investasi yang menguntungkan. Shafiee, dkk (2010) menyatakan bahwa prediksi harga emas penting bagi perusahaan pertambangan sebagai acuan dalam pengambilan keputusan kelayakan sebuah proyek.

Peramalan runut waktu (*time series*) merupakan salah satu topik penting pada bidang peramalan dimana observasi masa lalu dari sebuah variabel dikumpulkan dan dianalisis untuk membangun sebuah model yang mendeskripsikan relasi dasar yang ada (Zou, dkk, 2007). Teknik pemodelan ini sangat bermanfaat ketika hanya sedikit pengetahuan yang dapat diperoleh dari proses pembangkitan data atau ketika tidak ada model yang mampu dengan tepat menjelaskan keterkaitan antara variabel prediktor dan variabel lainnya. Beberapa metode tradisional yang umum digunakan pada peramalan runut waktu adalah pemodelan linier dengan *autoregressive integrated moving average* (ARIMA) dan pemodelan non-linier antara lain dengan model linier ganda, model *threshold autoregressive* (TAR), model *smoothing transition autoregressive* (STAR), model *autoregressive conditional heteroscedastic* (ARCH), dan *generalized autoregressive conditional heteroscedastic* (GARCH).

Selain metode pemodelan tradisional, para peneliti telah banyak menggunakan jaringan syaraf tiruan (JST) dalam peramalan. Sebagai contoh, Hussein, dkk. (2011) yang membandingkan antara *radial basis function neural network* (RBFNN) dengan metode model *autoregressive*, Parisi, dkk. (Parisi, dkk, 2008) yang menggunakan jaringan syaraf *rolling* dan *recursive*, Zou, dkk. (2007) dan Hansen, dkk. (2003) yang membandingkan antara JST dengan ARIMA, Theodosiou (2011) yang mengkombinasikan model JST *generalized regression* (GRNN) dan metode theta, dan Kaastra, dkk. (1996) yang menggunakan JST untuk melakukan peramalan di bidang ekonomi dan keuangan. Beberapa peneliti tersebut memperoleh kesimpulan berbeda berkaitan dengan hasil peramalan menggunakan JST. Masing-masing pendekatan baik tradisional maupun JST memiliki keunggulan dan kelemahan.

Penelitian di bidang peramalan harga emas dengan menggunakan JST bukanlah topik baru. McCann, dkk (1994) menggunakan jaringan syaraf rekursif

untuk mengenali titik-titik perubahan pada pasar emas berdasarkan sepuluh indikasi dari data-data historis yang hasilnya membawa keuntungan pada periode tertentu. Parisi, dkk. (2008) menggunakan jaringan syaraf rekursif dan berputar untuk meramalkan harga emas. Hasil dari penelitian ini disimpulkan bahwa pendekatan ini dapat mengurangi resiko pada transaksi emas dan dapat menjadi alternatif pengganti pemodelan ARIMA. Hussein, dkk. (2011) melakukan peramalan harga emas menggunakan *radial basis function neural network* (RBFNN) yang kemudian dibandingkan akurasi dengan pemodelan *autoregressive*. Tipe RBFNN yang digunakan adalah tunggal dan jamak dimana berdasarkan pengujian tingkat akurasi RBFNN jamak lebih tinggi dibandingkan RBFNN tunggal namun akurasi hasil dari kedua tipe RBFNN ini masih kalah dibandingkan ARIMA. Menurut Theodosiu (2011a), akurasi peramalan dengan jaringan syaraf dapat ditingkatkan dengan terlebih dahulu mendekomposisi data runtut waktu.

Data yang digunakan dalam peramalan dengan JST minimal dibagi menjadi dua kelompok yaitu data latih untuk membentuk model dan data uji untuk menguji model. Bowden, dkk (2002) menggunakan JST untuk meramalkan kadar garam di Sungai Murray, Australia. Untuk menghasilkan pengelompokan data yang memenuhi properti statistik yaitu nilai rata-rata data pada tiap kelompok, standar deviasi, dan jangkauan data, Bowden menggunakan algoritma genetika untuk membagi keseluruhan data menjadi tiga kelompok data yaitu data latih, data uji, dan data validasi. Berdasarkan pengujian yang dilakukan oleh Bowden, data yang dikelompokkan dengan algoritma genetika menghasilkan peramalan oleh JST dengan nilai galat RMS (*root mean square*) lebih rendah dibandingkan pengelompokan data secara konvensional yaitu dengan menggunakan pembagian berdasarkan urutan waktu. Selain itu, metode ini menghasilkan tiga kelompok data yang representatif terhadap populasi yang sama, ditunjukkan dengan kedekatan nilai rata-rata, standar deviasi, dan jangkauan. Kamp, dkk (2006) juga mengintegrasikan algoritma genetika dengan JST untuk memodelkan limpasan curah hujan (*rainfall-runoff*) di Belanda. Oleh karena tidak tersedia cukup data untuk pembentukan model, Kamp menggunakan data latih buatan untuk membentuk model JST. Teknik ini menyebabkan terbentuknya sejumlah besar

data latih dan munculnya data yang redundan. Untuk menyeleksi data buatan tersebut, Kamp menggunakan algoritma genetika. Berdasarkan pengujian oleh Kamp, model JST yang data latihnya diseleksi dengan algoritma genetika memberikan hasil yang lebih akurat dan waktu pelatihan yang lebih singkat dibandingkan ketika menggunakan seluruh data latih.

Salah satu model JST yang banyak digunakan dalam peramalan adalah *generalized regression neural network* (GRNN). GRNN merupakan jaringan berbasis memori yang mampu menghasilkan nilai perkiraan dari variabel-variabel kontinu dan konvergen terhadap dasar dari permukaan regresi (Specht, 1991). Proses pembelajaran dari GRNN dapat dianalogikan dengan proses untuk menemukan sebuah permukaan dalam sebuah ruang multidimensi yang paling sesuai dengan data latih. Salah satu kelebihan GRNN adalah bahwa jaringan syaraf ini merupakan pembelajar yang cepat dan selalu menghasilkan permukaan regresi optimal yang konvergen sehingga mampu mengatasi persoalan yang kerap dialami oleh model JST lainnya yaitu solusi dengan nilai galat yang bersifat lokal minimum. Berdasarkan pengujian oleh Specht, dengan menggunakan 1000 data latih, GRNN yang hanya memerlukan satu tahap pelatihan untuk menghasilkan prediksi yang akurasinya sama dengan model JST propagasi-balik dengan 1000 data dan 100 kali pelatihan.

Dalam penelitiannya, Theodosiou (2011a) menggunakan GRNN untuk peramalan runut waktu yang dikombinasikan dengan metode theta. Sebelum diramalkan, data runut waktu didekomposisi terlebih dahulu menggunakan metode STL (*Seasonal Trend Decomposition based on Loess*) untuk memisahkan komponen data tren (*trend*) dan komponen musiman. Dekomposisi ini bertujuan untuk meningkatkan kinerja dan akurasi peramalan dengan GRNN. Metode GRNN digunakan untuk meramalkan komponen data musiman dan residual, sedangkan peramalan komponen data tren dilakukan dengan metode Theta karena GRNN tidak dapat digunakan untuk meramalkan komponen ini.

Pada penelitian Theodosiou, digunakan 60 macam data runut waktu dari kompetisi NN3 sebagai data eksperimen. Pelatihan dan pengujian GRNN menggunakan teknik *pseudo out-of sample* dimana pelatihan dan pengujian model diterapkan terhadap seluruh data kecuali jendela terakhir. Setelah pelatihan,

GRNN digunakan untuk meramalkan 18 observasi terakhir. Seluruh data latih digunakan untuk membangun model GRNN. Kelemahan dari ketentuan ini adalah ukuran jaringan akan bertambah seiring meningkatnya jumlah data latih. Specht (1991) menyarankan penggunaan teknik klasterisasi untuk mereduksi jumlah vektor latih sehingga menghasilkan model jaringan yang lebih efisien. Di pihak lain, Popescu, dkk (2004) menyatakan bahwa perlu dipilih data latih yang tepat untuk meningkatkan kinerja GRNN. Alternatif dari teknik klasterisasi adalah penggunaan algoritma genetika seperti yang telah dilakukan oleh Kamp, dkk (2006) dan Yazici, dkk (2006). Berdasarkan pengujian yang telah dilakukan oleh Kamp, ukuran data latih yang dihasilkan oleh algoritma genetika berkurang hampir sebesar 50% dari seluruh data latih dan akurasi yang dihasilkan meningkat sebesar 39% dibandingkan dengan menggunakan seluruh data latih. Oleh karena itu, pada penelitian ini GRNN akan dikombinasikan dengan algoritma genetika untuk seleksi data latih sehingga menghasilkan struktur jaringan yang lebih efisien dan peramalan harga emas yang lebih akurat.

1.2. Perumusan Masalah

Berdasarkan latar belakang yang telah diuraikan pada sub-bab sebelumnya, Theodosiu (2011a) mengusulkan sebuah metode peramalan yang menggabungkan teknik dekomposisi dan jaringan syaraf. Metode dekomposisi yang digunakan adalah STL (*Seasonal Trend Decomposition based on Loess*) dimana data runtut waktu akan dipisahkan menjadi komponen data musiman, tren, dan residual. Selanjutnya, data hasil dekomposisi akan diramalkan menggunakan dua metode yang berbeda, yaitu *generalized regression neural network* (GRNN) dan metode theta. Komponen data musiman dan residual akan diramalkan menggunakan GRNN sedangkan komponen data tren diprediksi menggunakan metode theta.

Dalam penelitian Theodosiou digunakan 60 data runtut waktu dari kompetisi NN3 sebagai data latih. Pelatihan dan pengujian GRNN menggunakan teknik *pseudo out-of sample*, dimana pelatihan dan pengujian model diterapkan terhadap seluruh data kecuali jendela terakhir. Setelah pelatihan, GRNN

digunakan untuk meramalkan 18 observasi terakhir. Seluruh data latih digunakan untuk membangun model GRNN. Di lain pihak, Popescu, dkk (2004) menyatakan bahwa perlu dipilih data latih yang tepat untuk menghasilkan GRNN dengan kinerja baik. Penentuan data latih dan data pengujian dari keseluruhan data yang ada mempengaruhi kinerja jaringan syaraf (Kamp, dkk, 2006). Kamp menggunakan algoritma genetika untuk mereduksi jumlah data latih dengan tujuan menghasilkan peramalan yang lebih akurat. Berdasarkan pengujian yang dilakukan oleh Kamp, jaringan syaraf yang data latihnya diseleksi dengan algoritma genetika memiliki akurasi hasil 39% lebih baik dibandingkan dengan menggunakan seluruh data latih.

Persoalan seleksi data latih berkaitan dengan menentukan bagian data yang mana yang mempengaruhi akurasi hasil peramalan dan berapa banyak data yang dibutuhkan. Hal ini merupakan sebuah persoalan optimasi yang bersifat kombinatorial dimana ada banyak kemungkinan solusi yang dapat dicari. Algoritma genetika merupakan sebuah metode dengan kemampuan optimasi global yang memiliki beberapa kelebihan antara lain dapat menyelesaikan persoalan optimasi dengan berbagai jenis domain baik domain variabel kontinyu maupun diskrit. Dengan demikian, rumusan masalah dalam penelitian ini adalah **bagaimana mengintegrasikan algoritma genetika untuk seleksi data latih GRNN sehingga dihasilkan peramalan harga emas yang akurasinya setara atau lebih baik dibandingkan dengan menggunakan seluruh data latih?**

1.3. Tujuan dan Manfaat

Adapun tujuan dari penelitian ini adalah menghasilkan sebuah model peramalan yang akan mengkombinasikan metode theta, GRNN, dan algoritma genetika dimana GRNN akan diintegrasikan dengan algoritma genetika. Integrasi GRNN dan algoritma genetika digunakan untuk menyeleksi data latih sehingga menghasilkan struktur jaringan yang lebih efisien dan hasil peramalan yang akurat. Adapun manfaat yang diharapkan dari penelitian ini adalah memberikan informasi harga emas yang lebih akurat bagi para pelaku bisnis.

1.4. Batasan dan Asumsi

Adapun batasan pada penelitian ini adalah :

- a. Data yang digunakan adalah harga emas harian yang digunakan bersumber dari *World Gold Council* selama 10 tahun mulai dari tahun 2003 hingga tahun 2012.
- b. Dekomposisi data dalam penelitian ini menggunakan dekomposisi penjumlahan sesuai dengan metode yang diterapkan pada penelitian acuan.
- c. Nilai sigma yang merupakan salah satu parameter bebas pada GRNN akan menggunakan sigma global dimana nilai sigma adalah sama untuk seluruh simpul pada layer pola.

1.5. Kontribusi Penelitian

Adapun kontribusi dari penelitian ini adalah mengusulkan sebuah model peramalan harga emas yang mengembangkan model penelitian Theodosiou (2011a). Pengembangan yang dimaksud adalah integrasi algoritma genetika dengan GRNN untuk seleksi data latih dengan tetap mempertahankan karakteristik dan kualitas dari peramalan data runut waktu.

BAB 2

LANDASAN TEORI DAN KAJIAN PUSTAKA

Pada bab ini akan diuraikan mengenai teori-teori yang mendasari penelitian dan kajian mengenai penelitian-penelitian terkait. Teori-teori yang akan dipaparkan antara lain mengenai peran emas di sektor perekonomian, metode dekomposisi untuk peramalan runut waktu, jaringan syaraf tiruan dalam bidang peramalan, *generalized regression neural network* (GRNN), dan algoritma genetika. Adapun penelitian-penelitian terkait yang dikaji antara lain penelitian seputar peramalan harga emas dan penelitian tentang peramalan menggunakan GRNN yang diintegrasikan dengan algoritma genetika.

2.1. Peran Emas di Sektor Perekonomian

Emas merupakan sejenis logam mulia yang dapat dikategorikan sebagai sebuah komoditas dan aset moneter. Selama berabad-abad, emas dipandang sebagai logam yang multifungsi, sebagai mata uang yang dapat digunakan untuk menyimpan kekayaan, sebagai alat tukar, dan sebuah unit nilai (Sopipan, dkk, 2012). Emas memiliki peranan penting antara lain di bidang industri, sebagai perhiasan, dan sebagai aset investasi. Kuantitas persediaan emas ditentukan oleh kuantitas yang diminta oleh industri investasi dan perhiasan. Oleh karena itu, naiknya permintaan dari industri menjadi salah satu faktor yang mempengaruhi naiknya harga emas.

Sejak tahun 1968, harga emas yang dijadikan patokan seluruh dunia adalah harga emas berdasarkan standar pasar emas London (Witjaksono, 2010). Sistem ini dinamakan London Gold Fixing. London Gold Fixing adalah prosedur dimana harga emas ditentukan dua kali sehari setiap hari kerja di pasar London oleh lima anggota Pasar London Gold Fixing Ltd (www.goldfixing.com). Kelima anggota tersebut adalah : Bank of Nova Scotia, Barclays Capital, Deutsche Bank, HSBC, dan Societe Generale.

Proses penentuan harga adalah melalui lelang diantara kelima anggota tersebut. Pada setiap awal tiap periode perdagangan, Presiden London Gold Fixing Ltd akan mengumumkan suatu harga tertentu. Kemudian kelima anggota tersebut akan mengabarkan harga tersebut 53 kepada penjual. Penjual inilah yang berhubungan langsung dengan para pembeli sebenarnya dari emas yang diperdagangkan tersebut. Posisi akhir harga yang ditawarkan oleh setiap penjual kepada anggota Gold London Fixing merupakan posisi bersih dari hasil akumulasi permintaan dan penawaran klien mereka. Dari sinilah harga emas akan terbentuk. Apabila permintaan lebih banyak dari penawaran, secara otomatis harga akan naik, demikian pula sebaliknya. Penentuan harga yang pasti menunggu hingga tercapainya titik keseimbangan. Apabila harga sudah pasti, maka presiden akan mengakhiri rapat dan mengatakan “There are no flags, and we’re fixed”.

Proses penentuan harga emas dilakukan dua kali sehari, yaitu pada pukul 10.30 (harga emas Gold A.M) dan pukul 15.00 (harga emas Gold P.M). Harga emas ditentukan dalam mata uang Dollar Amerika Serikat, Poundsterling Inggris, dan Euro. Pada umumnya Gold P.M dianggap sebagai harga penutupan pada hari perdagangan dan sering digunakan sebagai patokan nilai kontrak emas di seluruh dunia.

Emas merupakan salah satu bentuk investasi yang relatif bebas resiko karena nilainya cenderung stabil dan naik. Investasi emas menjadi salah satu alternatif investasi yang berpedoman pada harga emas (Sopipan, dkk, 2012). Para investor emas dapat mengambil keuntungan dari dua arah pergerakan harga emas, naik atau turun, sebagaimana halnya saham. emas adalah alat yang dapat digunakan untuk menangkal inflasi yang kerap terjadi setiap tahunnya. Ketika akan berinvestasi, investor akan memilih investasi yang memiliki tingkat imbal balik tinggi dengan resiko tertentu atau tingkat imbal balik tertentu dengan resiko yang rendah (Witjaksono, 2010). Investasi di pasar saham tentunya lebih berisiko daripada berinvestasi di emas, karena tingkat pengembaliannya yang secara umum relatif lebih tinggi dari emas.

Selain untuk investasi, pengetahuan tentang harga emas di masa mendatang memiliki peran penting dalam berbagai bidang. Witjaksono (2010) meneliti tentang pengaruh harga emas terhadap nilai indeks harga saham gabungan (IHSG) di Bursa Efek Indonesia. Hasil dari penelitian ini menunjukkan bahwa variabel harga emas dunia berpengaruh positif terhadap IHSG. Hal ini disebabkan bahwa selama periode pengamatan dari tahun 2000-2009, tingkat pendapatan masyarakat Indonesia secara umum meningkat (www.bps.go.id). Peningkatan pendapatan masyarakat ini tentunya mengakibatkan investor dapat melakukan diversifikasi investasi untuk mengurangi resiko. Salah satu investasi yang cenderung bebas dari resiko adalah emas. Hal ini didasari karena sifat dari emas yang nilainya relatif bebas dari tekanan inflasi, selain itu nilainya yang cenderung naik dari tahun ke tahun.

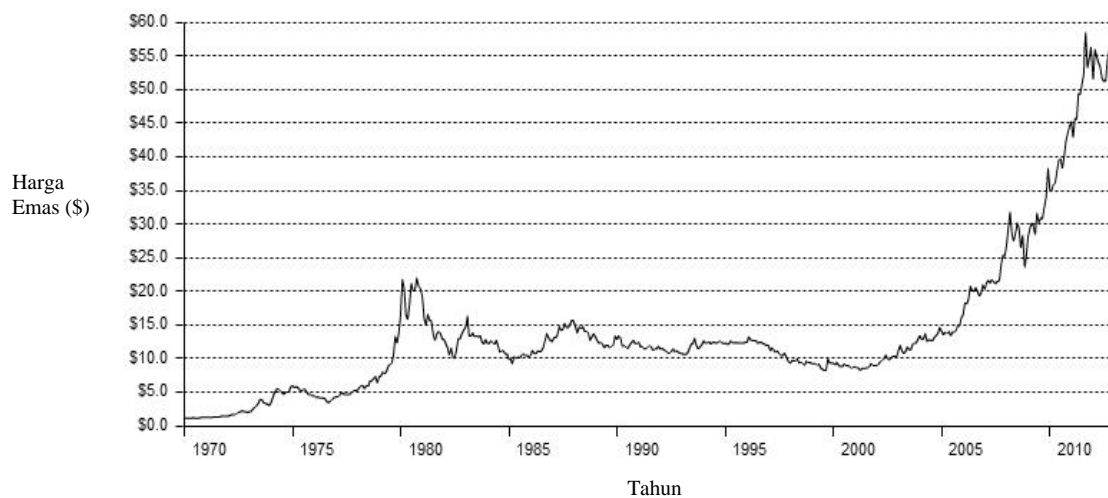
Penelitian dari Wang, dkk (2012) juga menunjukkan relevansi prediksi harga emas. Dalam jangka pendek, harga emas memiliki efek saling mempengaruhi terhadap harga minyak mentah. Hal ini kemungkinan disebabkan oleh karena kedua harga tersebut bergerak sejalan dengan inflasi, rasio bunga, dan produksi industri. Relevansi lainnya ditunjukkan oleh penelitian yang dilakukan oleh Shafiee, dkk (2010). Penelitian ini menyampaikan pentingnya prediksi harga emas bagi perusahaan pertambangan dalam memutuskan kelayakan investasi untuk sebuah proyek. Selain itu, juga terdapat kesimpulan yang mendukung penelitian Wang, dkk (2012) bahwa harga emas dan harga minyak mentah saling mempengaruhi.

2.2. Peramalan Runut Waktu

Sebuah data runut waktu (*time-series*) adalah sebuah koleksi hasil observasi yang dilakukan secara urut waktu (Chatfield, 2000). Beberapa contoh dari data runut waktu antara lain : penjualan produk tertentu per bulan selama setahun, suhu tertinggi di sebuah daerah pada waktu sore hari, jumlah kelulusan mahasiswa per tahun selama sepuluh tahun, harga emas per tahun selama sepuluh tahun, dan sebagainya. Pada Gambar 2.1 ditunjukkan sebuah contoh grafik data

runut waktu untuk harga emas per gram dalam USD sejak tahun 1970 hingga 2012.

Peramalan atau prediksi merupakan salah satu aktivitas penting di bidang ilmiah, industri, komersial, dan ekonomi. Terdapat tiga kategori prediksi yaitu 1) metode peramalan obyektif/kuantitatif yang mengacu pada perhitungan matematis dan statistik, 2) metode peramalan subyektif/kualitatif yang didasarkan pada opini para pakar, dan 3) ramalan/tebakan (Crone, 2005). Jika diketahui sebuah data runut waktu $x_1, x_2, x_3, \dots, x_N$ maka data ini dapat digunakan untuk meramalkan nilai di masa mendatang x_{N+h} . Bilangan bulat h disebut dengan waktu tempuh (*lead time*) atau horison peramalan (h untuk horison) dan ramalan untuk x_{N+h} untuk h langkah dari N dinotasikan dengan $\hat{x}_N(h)$. Salah satu hal penting dalam peramalan runut waktu adalah menentukan waktu peramalan dan waktu tempuhnya.



Gambar 2.1 Grafik harga emas dalam USD (Sumber : www.gold.org)

Sebuah metode peramalan merupakan teknik untuk memprediksi nilai di masa depan berdasarkan nilai di masa lampau. Terdapat tiga kategori metode peramalan yaitu :

- a. peramalan dengan asumsi yang didasarkan pada penilaian subyektif, intuisi, dan informasi lain yang relevan
- b. metode variasi tunggal (*univariate*) dimana peramalan hanya didasarkan pada nilai saat ini dan masa lampau dari sebuah data runut waktu tunggal yang akan diramal
- c. metode variasi majemuk (*multivariate*) dimana peramalan untuk sebuah peubah bergantung pada satu atau lebih peubah runut waktu lainnya

Beberapa alternatif lainnya dalam mengelompokkan metode-metode peramalan adalah peramalan secara otomatis yang berarti tidak memerlukan intervensi manusia dan sebaliknya, peramalan non-otomatis, atau dikelompokkan berdasarkan tingkat kerumitannya yaitu peramalan sederhana atau rumit.

Dalam peramalan di dunia nyata, analis umumnya memiliki data hingga waktu N , kemudian melakukan peramalan untuk masa mendatang dengan cara menyesuaikan model terhadap data dan selanjutnya menggunakan model tersebut untuk membuat proyeksi/peramalan. Jika $\hat{x}_N(h)$ hanya menggunakan data hingga waktu N , maka hasil peramalan dapat dikatakan sebagai *out-of-sample*. Para ekonom menyebutnya dengan peramalan *ex-ante*. Sedangkan apabila analis menggunakan model yang telah disesuaikan dengan data untuk meramalkan data yang telah digunakan pada proses penyesuaian, maka hasil peramalan ini disebut dengan *in-sample*. Alternatifnya, analis membagi data menjadi dua bagian yaitu untuk penyesuaian model, yang sering disebut dengan data latih, dan untuk pengujian model, yang umum disebut dengan data uji. Beberapa tujuan utama dari analisis data runut waktu adalah :

- a. Deskripsi – yaitu mendeskripsikan data menggunakan ringkasan statistik dan/atau secara grafis.
- b. Pemodelan – untuk mencari model statistik yang paling menggambarkan proses pembangkitan data.
- c. Peramalan – untuk memperkirakan nilai masa mendatang dari data.
- d. Pengendalian – menggunakan teknik peramalan untuk memberikan acuan dalam proses pengendalian sebuah proses.

2.2.1. Jenis-jenis Data Runut Waktu

Sebuah data runut waktu merupakan data yang diperoleh dari pengamatan secara terurut dalam rentang waktu tertentu. Pengukuran ini dapat dihasilkan dari pengamat secara kontinu sepanjang waktu atau diambil pada waktu-waktu diskrit (Chatfield, 2000). Data runut waktu yang diperoleh melalui cara pertama umum disebut dengan data kontinu dan yang kedua disebut dengan data diskrit, terlepas dari nilai peubah yang diukur dalam format kontinu ataupun diskrit.

Untuk data runut waktu kontinu, peubah yang diamati umumnya merupakan sebuah peubah kontinu yang diambil secara berkelanjutan mengikuti sebuah jejak, misalnya aktivitas otak yang direkam oleh mesin EEG. Umumnya, analisis terhadap data runut waktu kontinu dilakukan dengan membagi data tersebut menjadi beberapa rentang waktu sehingga menghasilkan sebuah data runut waktu diskrit. Untuk menghindari hilangnya informasi penting akibat pembagian, maka rentang pembagian harus dibuat cukup kecil.

Adapun data runut waktu diskrit merupakan data yang dapat diperoleh dari salah satu proses berikut yaitu (a) dengan mengambil sampel dari data kontinu (misalnya suhu yang diukur dalam selang waktu satu jam, data tersebut dapat diperoleh dengan mengambil sampel dari data kontinu atau pengukuran yang diambil dengan rentang waktu satu jam) (b) dengan melakukan agregasi untuk periode waktu tertentu (misalnya total penjualan dalam beberapa bulan berturut-turut), dan (c) data yang diambil secara diskrit (misalnya sisa hasil usaha (SHU) yang dibagikan oleh sebuah koperasi selama beberapa tahun berturut-turut). Pada ketiga jenis data diskrit di atas, data runut waktu umumnya memiliki rentang waktu yang sama.

2.2.2. Beberapa Metode Peramalan

Beberapa metode peramalan yang umum digunakan antara lain Winter's *Exponential Smoothing*, metode dekomposisi, regresi runut waktu, model Arima, dan metode jaringan syaraf (Suhartono, 2005). Adapun gambaran ringkas mengenai setiap metode tersebut adalah sebagai berikut :

a. *Winter's Exponential Smoothing*

Exponential Smoothing adalah metode yang secara berkelanjutan memperbaiki hasil perkiraan berdasarkan data-data terbaru. Model Winter merupakan sebuah *exponential smoothing* yang umumnya digunakan untuk meramalkan komponen data tren dan musiman.

b. Metode Dekomposisi

Terdapat dua pendekatan dekomposisi yaitu penjumlahan (*additive*) dan perkalian (*multiplicative*). Model dekomposisi perkalian sesuai jika digunakan pada data runtut waktu yang variasi musimannya menunjukkan kenaikan atau penurunan konstan. Asumsi utama dari metode ini adalah bahwa komponen data musiman dapat dipisahkan dari komponen lain pada data runtut waktu. Model dekomposisi perkalian memiliki formula seperti pada persamaan 2.1 dimana y_t = nilai pengamatan pada data runtut waktu pada periode waktu t , T_t = komponen tren pada periode waktu t , S_t = komponen data musiman pada periode waktu t , C_t = komponen data siklus pada periode waktu t , dan I_t = komponen data residual pada periode waktu t .

$$y_t = T_t \times S_t \times C_t \times I_t \quad , \quad (2.1)$$

c. Regresi Runtut Waktu

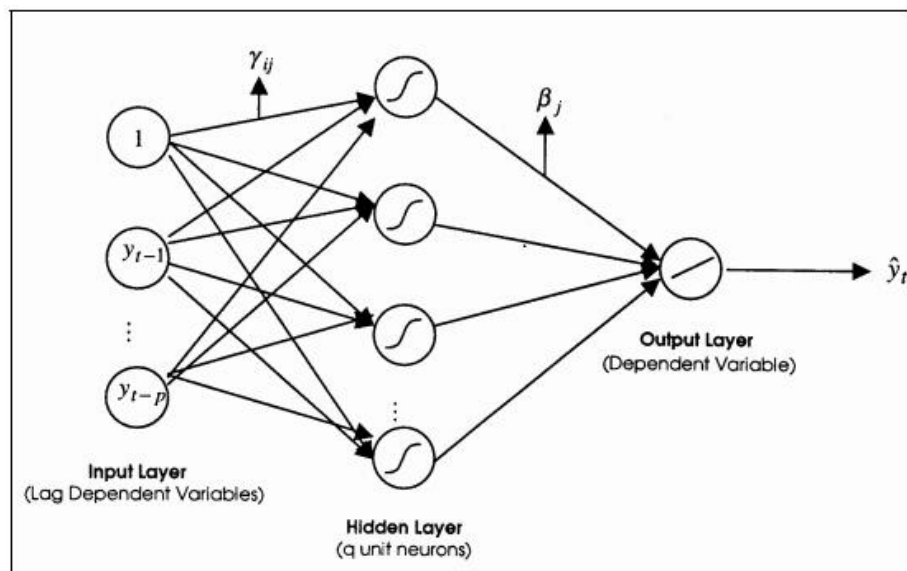
Model regresi runtut waktu mencari hubungan antara y_t dengan fungsi terhadap waktu. Model ini paling sesuai diterapkan jika parameter yang mempengaruhi data runtut waktu nilainya konstan sepanjang waktu. Misalnya, ketika sebuah data runtut waktu menunjukkan tren linier, maka nilai kemiringan (*slope*) komponen trennya nilainya konstan. Contoh lain, jika sebuah data runtut waktu dapat dideskripsikan menggunakan parameter musiman bulanan, maka parameter musiman untuk setiap bulan selama setahun akan sama untuk tahun-tahun berikutnya.

d. Model musiman ARIMA

Model ARIMA termasuk dalam kelompok pemodelan waktu linier yang fleksibel. Model ini dapat digunakan untuk memodelkan berbagai tipe data runtut waktu yang sifatnya musiman maupun yang non-musiman.

e. Model Jaringan Syaraf

Jaringan syaraf (JS) merupakan sebuah model non-linier yang fleksibel dan dapat secara adaptif menemukan pola dari data. JS dapat belajar dari pengalaman dan memperkirakan relasi fungsi kompleks apapun dengan tingkat akurasi yang tinggi. Banyak aplikasi di bidang pengenalan pola dan peramalan yang sukses mengimplementasikan JS. Jaringan syaraf runut maju (JSRM) merupakan salah satu jenis JS yang populer digunakan dalam bidang peramalan runut waktu. Gambar 2.2 menunjukkan arsitektur umum dari JSRM tiga lapisan yang diterapkan dalam peramalan. Simpul masukan mewakili data observasi beberapa jeda (*lag*) sebelumnya, sedangkan simpul keluaran menyatakan nilai ramalan untuk waktu mendatang. Simpul-simpul tersembunyi dengan fungsi transfer non-linier tertentu digunakan untuk memproses informasi dari simpul masukan.



Gambar 2.2 Arsitektur jaringan syaraf runut maju (Suhartono, 2005)

Model JSRM pada gambar 2.2 dapat diformulasikan seperti pada persamaan 2.2. Pada persamaan 2.2., p menyatakan jumlah simpul masukan, q menyatakan jumlah simpul tersembunyi, f merupakan fungsi transfer sigmoid, $\{\beta_j, j = 0, 1, \dots, q\}$ adalah sebuah vektor bobot dari simpul-simpul tersembunyi ke simpul keluaran dan $\{\gamma_{ij}, i = 0, 1, \dots, p; j = 1, 2, \dots, q\}$ menunjukkan bobot dari simpul-simpul masukan ke simpul-simpul tersembunyi.

$$y_t = \theta_0 + \sum_{j=1}^q \beta_j f\left(\sum_{i=1}^p \gamma_{ij} y_{t-1}\right) + \varepsilon_t, \quad (2.2)$$

2.3. Jaringan Syaraf Tiruan dalam Peramalan

Jaringan syaraf (JS) dibangun dari adanya kesadaran bahwa otak manusia, yang bekerja seperti halnya komputer paralel yang rumit dan non-linier, berfungsi dalam cara-cara yang berbeda dibandingkan dengan pemroses komputer. Sejak awal tahun 1950, para peneliti telah membuat banyak penelitian tentang struktur dan kemampuan otak dalam memproses informasi dan telah berusaha meletakkan banyak karakteristik otak ke dalam JS (Theodosiou, 2011). Pendekatan ini telah membantu banyak penelitian seperti bidang rekayasa, ilmu komputer, biokimia, dan fisika.

Adanya keberhasilan implementasi JST di bidang-bidang tersebut menjadi dorongan untuk menerapkannya di bidang keuangan, ekonomi, dan industri. Karakteristik-karakteristik yang dimiliki JST menjadikannya menarik untuk digunakan dalam pemodelan berbagai persoalan ekonomi dan keuangan. Diantaranya, JS memiliki struktur yang non-linier, kemampuannya untuk beradaptasi terhadap informasi baru, dan keseragaman rancangan di berbagai ranah persoalan. Kelebihan lainnya adalah tidak diperlukan adanya asumsi eksplisit mengenai relasi fungsional antara keluaran yang diinginkan dan variabel-variabel bebas dari model. JS mempunyai kelebihan dibandingkan teknik statistik terkini dalam hal pemodelan dan prediksi data keuangan dan ekonomi (Theodosiou, 2011a; Zhang, dkk, 1998).

Di samping mulai maraknya penggunaan JS dalam bidang peramalan, para peneliti di bidang statistik dan ekonometrik belum mempercayai kinerja dan akurasi JS dalam melakukan prediksi dibandingkan dengan teknik statistik. Hal ini diperoleh dari berbagai kesimpulan penelitian-penelitian yang membandingkan kinerja JS dengan metode peramalan statistik tradisional (Theodosiou, 2011a). Hal lain yang mendukung pendapat ini adalah hasil yang diperoleh dari beragam kompetisi peramalan berskala besar, seperti halnya kompetisi M3 dan NN3, telah

menunjukkan bahwa JS memiliki kinerja, dalam hal ini akurasi peramalan, di bawah teknik statistik sederhana. Zhang, dkk (1998) merangkum penelitian-penelitian di bidang ini dan menunjukkan kelebihan serta keterbatasan JS dalam bidang peramalan dibandingkan metode tradisional.

Para pakar di bidang ini menyimpulkan bahwa perancangan JS dalam berbagai persoalan lebih cenderung ke arah seni daripada ilmiah. Pernyataan ini berakar dari kenyataan bahwa aplikasi JS melibatkan sejumlah besar derajat kebebasan. Konsekuensinya, diperlukan banyak percobaan dan investigasi, melalui uji coba, untuk menentukan nilai parameter yang sesuai sehingga dihasilkan arsitektur jaringan terbaik (Theodosiou, 2011a). Derajat-derajat kebebasan ini meliputi, tidak terbatas pada, jenis fungsi aktivasi, inisialisasi bobot sinaptik, konektivitas antar simpul, proses pembelajaran jaringan, dan penentuan simpul-simpul masukan dan tersembunyi. Penentuan nilai dari beberapa parameter ini mempengaruhi akurasi dan kinerja jaringan sekaligus kemampuan penyamarataannya. Oleh karena itu, aplikasi JS pada peramalan melibatkan proses untuk memodelkan kompleksitas, sehingga membutuhkan waktu yang cukup lama.

Jika dibandingkan dengan metode tradisional, JST memiliki beberapa keunggulan yang membuatnya menarik untuk digunakan dalam peramalan. Pertama, berlawanan dengan pendekatan tradisional, JST adalah sebuah metode yang mampu secara mandiri menyesuaikan dengan data (*data-driven*) dimana hanya terdapat sedikit pengetahuan berkaitan dengan persoalan yang ingin diselesaikan (Zhang, dkk, 1998). JST mampu belajar dari contoh-contoh yang diberikan dan menangkap relasi fungsional samar di antara data bahkan jika relasi tersebut belum diketahui atau sulit dideskripsikan. Kedua, JST dapat menyamaratakan. Setelah belajar dari contoh-contoh yang diberikan, JST seringkali dapat menebak dengan tepat bagian yang tak terlihat dari populasi meskipun data-data untuk proses pembelajaran mengandung derau. Ketiga, JST merupakan fungsi penaksir umum dimana telah ditunjukkan bahwa sebuah jaringan dapat menaksir fungsi kontinu apapun dengan berapapun derajat akurasi yang diinginkan. Terakhir, JST bersifat nonlinier yang jika dibandingkan dengan

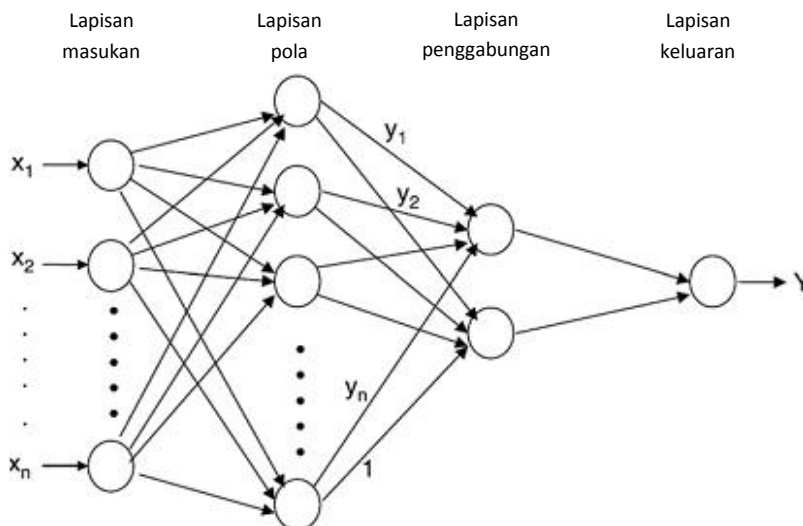
dengan pendekatan umum pada peramalan yaitu linier, JST dianggap lebih sesuai dengan kondisi di dunia nyata yang sebagian besar bersifat nonlinier.

2.4. *Generalized Regression Neural Network (GRNN)*

GRNN merupakan bentuk modifikasi dari *radial basis function neural network* (RBFNN) (Kim, dkk, 2008). GRNN adalah sebuah jaringan syaraf yang didasarkan pada teori regresi nonlinier. GRNN adalah sebuah aproksimator *universal* untuk fungsi perataan sehingga dapat menyelesaikan segala bentuk persoalan aproksimasi perataan. Menurut Theodosiou (2011a), penggunaan GRNN dapat mengurangi kesulitan dalam merancang dan menguji-coba jaringan syaraf akibat adanya parameter-parameter bebas. Jenis jaringan ini, yang dikembangkan oleh Specht (1991), menunjukkan kelebihan-kelebihan penting dibandingkan arsitektur jaringan syaraf standar seperti jaringan syaraf runut maju propagasi balik (RMPB).

Topologi GRNN terdiri atas empat lapisan, yaitu lapisan masukan, lapisan basis radial atau lapisan pola (*pattern*) atau lapisan tersembunyi (*hidden*), lapisan penggabungan (*summation*), dan lapisan keluaran (Duzgun, 2010). Lapisan pertama, lapisan pola, dan lapisan penggabungan memiliki koneksi penuh, sedangkan lapisan keluaran hanya terhubung ke beberapa neuron pada lapisan penggabungan. Lapisan penggabungan terdiri atas dua tipe neuron yaitu satu atau lebih simpul penggabungan (*summation*) dan sebuah simpul pembagian (*division*). Banyaknya neuron penggabungan-S sesuai dengan banyaknya neuron pada layer keluaran (Kim, dkk, 2008). Simpul pembagian tidak menggunakan fungsi transfer, tetapi menggunakan fungsi penjumlahan nilai keluaran terbobot dari lapisan pola. Setiap simpul pada lapisan keluaran terkoneksi ke simpul penggabungan dan pembagian di lapisan penggabungan, dan tidak ada bobot pada koneksi antara lapisan penggabungan dan lapisan keluaran. Nilai keluaran dari lapisan keluaran dihitung dengan membagi perhitungan nilai keluaran dari simpul penggabungan dengan perhitungan nilai keluaran dari simpul pembagian pada lapisan penggabungan. Gambar 2.3 menunjukkan topologi dari GRNN.

Kinerja pelatihan GRNN berbeda dengan jaringan syaraf *multi layer perceptron* (MLP). Proses pelatihan antara lapisan masukan dan lapisan pola dilakukan secara tak-terbimbing (*unsupervised*). Di sisi lain, pelatihan yang dilakukan antara lapisan pola dan lapisan penggabungan menggunakan metode terbimbing (*supervised*) dengan tujuan menghasilkan keluaran dari lapisan pola dengan nilai galat seminimal mungkin. Pada lapisan pola di GRNN, fungsi transfer yang umum digunakan adalah fungsi *radial basis* (FRB). Menentukan parameter dari FRB antara lapisan masukan dan lapisan pola berkaitan erat dengan penentuan sigma. Bobot koneksi antara lapisan pola dan lapisan penggabungan ditentukan dengan pelatihan terbimbing (*supervised*).



Gambar 2.3 Arsitektur GRNN (Cigizoglu, 2005)

Fungsi koneksi antara lapisan masukan dan lapisan pola dari GRNN dinyatakan sebagai jumlah kuadrat dari vektor masukan x_i dikurangi dengan pusat u_{ij} seperti ditunjukkan pada persamaan 2.3 dimana i adalah lapisan masukan dan j adalah lapisan pola, dan $R_j =$ jarak radius dari vektor masukan. Sedangkan vektor masukan x_i dinyatakan melalui persamaan 2.4 dan pusat radius $U^{(j)}$ dinyatakan dengan persamaan 2.5. Berdasarkan persamaan tersebut, maka persamaan 2.3 dapat dinyatakan sebagai persamaan 2.6 dimana $\|\cdot\|$ menunjukkan jarak Euclidian.

R_j selanjutnya dapat dimasukkan ke fungsi transfer dari lapisan pola $\Phi_1(\cdot)$ dan menghasilkan sebuah persamaan S_j seperti ditunjukkan pada persamaan 2.7.

$$R_j = \sqrt{\sum_{i=1}^m (x_i - u_{ji})^2} \quad (2.3)$$

$$X = [x_1, x_2, \dots, x_m]^T \quad (2.4)$$

$$U^{(j)} = [u_{j1}, u_{j2}, u_{j3}, \dots, u_{jm}]^T \quad (2.5)$$

$$R_j = \|X - U^{(j)}\| \quad (2.6)$$

$$S_j = \Phi_1(R_j) = \Phi_1(\|X - U^{(j)}\|) \quad (2.7)$$

Umumnya, fungsi transfer yang digunakan pada lapisan pola adalah fungsi *radial basis* antara lain fungsi kernel Gaussian (FKG). Semakin besar jarak antara vektor masukan dan pusat, semakin besar juga nilai vektor keluaran. FKG dapat dinyatakan dengan persamaan 2.8 dimana $B_1 = \frac{1}{2\sigma^2}$, σ = sigma/lebar/penyebaran dari fungsi *radial basis*. Lapisan penggabungan dapat menerima S_j dan hasil perhitungan dari vektor bobot koneksi dari lapisan pola ke lapisan penggabungan seperti ditunjukkan pada persamaan 2.9 dimana k = lapisan penggabungan, W_{kj} = bobot koneksi antara lapisan pola dengan lapisan penggabungan. Sedangkan simpul penggabungan dan simpul pembagian pada lapisan penggabungan dapat ditunjukkan pada persamaan 2.10 dan 2.11 dimana $\Phi_2(\cdot)$ = fungsi transfer untuk lapisan penggabungan, S_k = perhitungan keluaran simpul penggabungan, dan D_k = perhitungan keluaran simpul pembagian. Akhirnya, fungsi dari lapisan keluaran dinyatakan dengan pembagian total simpul penggabungan dan simpul pembagian dari lapisan penggabungan.

$$\Phi_1 = \exp(-B_1 R_j^2) = \exp\left[-\frac{\sum_{i=1}^m (x_i - u_{ji})^2}{2\sigma^2}\right] \quad (2.8)$$

$$T_k = \sum_{j=1}^n W_{kj} \cdot S_j = \sum_{j=1}^n W_{kj} \cdot \Phi_1(\|X - U^{(j)}\|) \quad (2.9)$$

$$S_k = \Phi_2\left(\frac{\sum_{j=1}^n W_{kj} \cdot \Phi_1(\|X - U^{(j)}\|)}{\sum_{j=1}^n W_{kj}}\right) \quad (2.10)$$

$$D_k = \sum_{j=1}^n W_{kj} \cdot \sigma(\|X - U^{(j)}\|) \quad (2.11)$$

Dalam GRNN terdapat sebuah parameter bebas yaitu parameter penghalusan (*smoothing parameter*) atau sigma (σ). Parameter ini terpisah dari lapisan masukan dan keluaran dan turut diperhitungkan dalam perancangan jaringan. Nilai sigma akan menentukan derajat generalisasi dari GRNN. Nilai sigma yang besar membuat lurus jalur dari garis prediksi, sedangkan nilai sigma yang rendah (mendekati nol) akan membuat sebuah jalur yang menyerupai peta titik-ke-titik. Dengan kata lain, semakin tinggi nilai sigma maka akan meningkatkan kemampuan jaringan dalam memprediksi dan mengurangi kesalahan prediksi, sedangkan nilai sigma yang terlalu rendah akan menghilangkan kemampuan jaringan dalam memprediksi sehingga perlu dicari nilai sigma yang optimal.

2.5. Dekomposisi Data Runut Waktu

Teknik dekomposisi awalnya dikembangkan oleh Persons (1919) untuk mengidentifikasi dan mengisolasi komponen penting dari sebuah data runut waktu seperti pola tren, musiman, dan siklus (Theodosiou, 2007). Beberapa metode yang umum digunakan antara lain teknik berbasis rata-rata bergerak (*moving average*) seperti X-11 ARIMA, SABL (*Seasonal Adjustment at Bell Laboratories*), dan STL (*Seasonal-Trend decomposition based on Loess smoothing*), teknik berbasis regresi jamak (*multiple regression*), dan metode berbasis pemodelan ARIMA.

Meskipun pengembangan teknik dekomposisi tidak khusus bertujuan untuk melakukan prediksi, namun ide di balik teknik tersebut sangat menarik diterapkan dalam peramalan. Pemisahan berbagai komponen dari data dan prediksi untuk tiap komponen hasil pemisahan dapat dipandang sebagai sebuah proses pengisolasian bagian kecil dari keseluruhan proses, dan memisahkan komponen tersebut dari segala bentuk derau dan variabilitas yang tidak konsisten. Komponen-komponen yang telah dipisahkan ini akan lebih mudah diekstrapolasi karena kecenderungannya yang deterministik. Oleh karena itu, peramalan dengan pemisahan komponen ini dipandang dapat memberikan hasil dengan akurasi

tinggi dibandingkan peramalan terhadap keseluruhan data. Sebuah data runut waktu dapat didekomposisi menjadi :

- a. Variasi musiman (*seasonal*) – variasi ini umumnya terlihat dalam periode tahunan dan muncul di banyak runut, meskipun data diperoleh secara harian, mingguan, atau bulanan, saat pola perilaku tertentu muncul pada waktu-waktu tertentu sepanjang tahun.
- b. Tren (*trend*) – variasi ini ditunjukkan dengan adanya tren naik atau menurun yang konsisten dari data dalam kurun waktu tertentu.
- c. Siklus (*cycle*) – meliputi siklus rutin yang kemunculannya selain tahunan.
- d. Residual (*irregular*) – menunjukkan komponen data yang tersisa setelah sebuah data runut waktu diambil komponen data musiman, tren, dan siklusnya.

2.5.1. Seasonal Trend Decomposition based on Loess

Dekomposisi komponen data musiman dan tren menggunakan prosedur LOESS (STL – *Seasonal Trend Decomposition based on Loess*) (Cleveland, dkk, 1990) diterapkan untuk dekomposisi penjumlahan dari data runut waktu global (Theodosiou, 2011b). STL merupakan sebuah prosedur penyaringan yang akan mendekomposisi data melalui serangkaian aplikasi dari operasi penghalusan menggunakan *locally weighted regression* (LOESS). Penghalusan dengan Loess menerapkan regresi polinomial terbobot lokal (*locally weighted polynomial regression*) untuk tiap titik pada data, dengan peubah penjelasnya adalah nilai yang dekat dengan titik yang responnya ingin diukur. Parameter-parameter dari prosedur STL diperoleh dari nilai eigen dan analisis respon frekuensi dari sebuah data runut waktu (Theodosiou, 2011a).

Beberapa fitur yang menjadi daya tarik dari STL untuk dekomposisi data runut waktu antara lain kemampuannya untuk menangani beragam variasi dari komponen data tren dan musiman, serta ketahanannya terhadap penyimpangan yang terjadi sebagai akibat ketidak-stabilan data runut waktu. Selain itu, prosedur ini lebih terkendali secara komputasi dan lebih mudah diimplementasikan

menggunakan perangkat lunak statistik. Kelebihan lainnya adalah, tidak seperti metode dekomposisi lainnya, STL dapat mengatasi data musiman dengan frekuensi musiman lebih dari satu, sehingga tidak terbatas pada data satu atau empat bulanan.

Prosedur dari dekomposisi STL adalah dengan secara berulang melakukan pengambilan komponen tren (*detrending*) dan pembaruan komponen data musiman pada data yang tersisa. Pada tiap iterasi, bobot keandalan ditentukan berdasarkan perkiraan komponen residual (*irregular*); nilai sebelumnya digunakan untuk menurunkan bobot dari pengamatan lain pada perhitungan selanjutnya. Proses iterasi melibatkan dua prosedur rekursif yaitu putaran dalam (*inner loop*) dan luar (*outer loop*). Untuk putaran dalam langkah-langkahnya adalah sebagai berikut :

- a. Pengeluaran komponen tren (*detrending*) : jika $s_t^{(k)}$ dan $m_t^{(k)}$ adalah komponen data musiman dan tren yang diperoleh di akhir putaran ke- k . Pada iterasi $k+1$, data global x_t diambil komponen trennya dengan cara mengurangi komponen data tren $m_t^{(k)}$, misalnya $x_t - m_t^{(k)}$. Nilai $m_t^{(0)}$ diinisialisasi nol.
- b. Penghalusan musiman (*seasonal smoothing*) : Prosedur penghalusan Loess diaplikasikan ke sub-seri yang diperoleh dari proses (a), $(x_t - m_t^{(k)})$, untuk mendapatkan komponen data musiman pendahuluan, $s_t^{(k+1)}$.
- c. Penyaringan komponen musiman yang telah diperhalus : penerapan *simple moving average* sederhana dari komponen data musiman pendahuluan yang diperoleh dari langkah (b), $s_t^{(k+1)}$, diikuti dengan penerapan Loess untuk mengidentifikasi komponen tren yang tersisa, $m_t^{(k+1)}$.
- d. Pengambilan data tren dari komponen data yang sudah diperhalus : komponen data musiman penjumlahan kemudian dihitung sebagai selisih antara komponen data musiman pendahuluan pada langkah (b), $\widetilde{s_t^{(k+1)}}$, dan komponen tren pendahuluan pada langkah (c), $\widetilde{m_t^{(k+1)}}$, misalnya $s_t^{(k+1)} = \widetilde{s_t^{(k+1)}} - \widetilde{m_t^{(k+1)}}$.

- e. *Deseasonalizing* : data yang telah disesuaikan komponen musimannya diperoleh dengan cara mengurangi data awal dengan hasil pada langkah (d), $x_t - s_t^{(k+1)}$.
- f. Penghalusan tren : hasil dari langkah (e) diperhalus lagi dengan Loess untuk memperoleh perkiraan dari komponen tren, $m_t^{(k+1)}$.

Tiap iterasi dari putaran dalam menerapkan penghalusan musiman yang akan memperbarui komponen data musiman, kemudian diikuti oleh penghalusan tren yang akan memperbarui komponen tren. Iterasi pada putaran luar terdiri atas satu iterasi dari putaran dalam yang akan menghasilkan perkiraan dari komponen data musiman dan tren yang akan digunakan untuk memperoleh komponen data residual ($e_t^{(k+1)} = x_t - m_t^{(k+1)} - s_t^{(k+1)}$). Setiap nilai besar pada e_t dianggap sebagai nilai ekstrim dan memicu perhitungan sebuah bobot. Proses ini akan mengakhiri putaran luar. Iterasi untuk putaran selanjutnya dalam menggunakan bobot ini untuk menurunkan efek dari nilai ekstrim, yang diperoleh pada putaran luar sebelumnya. Dengan demikian, untuk setiap runut waktu x_t , STL akan menghasilkan: m_t , s_t , dan e_t dimana m_t adalah komponen tren dari data runut waktu, s_t merupakan komponen data musimannya, dan e_t adalah komponen galat sisa sesuai dengan persamaan 2.13.

$$x_t = m_t + s_t + e_t \tag{2.13}$$

2.5.2. *Locally Weighted Regression (LOESS)*

Jika x_i adalah pengukuran dari sebuah peubah bebas dan y_i adalah pengukuran dari sebuah peubah tak-bebas untuk $i = 1, 2, \dots, n$ (Cleveland, dkk, 1990). Kurva regresi loess, $\hat{g}(x)$, adalah penghalusan untuk y berdasarkan x yang dapat dihitung untuk setiap nilai x yang merupakan sebuah peubah bebas. $\hat{g}(x)$ dihitung dengan cara sebagai berikut : dipilih q , sebuah bilangan bulat positif, misalnya $q < n$, kemudian ditentukan sejumlah q nilai dari x_i yang terdekat dengan x dan tiap q diberikan sebuah bobot ketetanggaan berdasarkan jaraknya dari x . Jika $d_q(x)$ merupakan jarak terjauh ke- q x_i dari x dan W adalah sebuah fungsi

bobot *tricube* maka W dapat dinyatakan dengan persamaan 2.14 dimana bobot keanggotaan untuk tiap x_i dinyatakan melalui persamaan 2.15.

$$W(u) = \begin{cases} (1-u^3)^3 & \text{untuk } 0 \leq u < 1 \\ 0 & \text{untuk } u \geq 1 \end{cases} \quad (2.14)$$

$$v_i(x) = W\left(\frac{|x_i - x|}{\lambda_q(x)}\right) \quad (2.15)$$

Berdasarkan persamaan 2.14 dan 2.15, nilai x_i yang dekat dengan x memiliki bobot terbesar dan bobot makin menurun jika jarak x_i terhadap x makin jauh dan menjadi nol pada titik terjauh ke- q . Langkah selanjutnya adalah menentukan sebuah polinomial berderajat d terhadap data dengan bobot $v_i(x)$ pada (x_i, y_i) . Jika $d=1$ berarti polinomial bersifat linier-lokal dan jika $d=2$ polinomial bersifat kuadrat-lokal. Jika $q > n$ dan $n(x)$ adalah jarak dari x ke titik terjauh x_i . Untuk $q > n$ definisi dari $\lambda_q(x)$ adalah sebagaimana persamaan 2.16.

$$\lambda_q(x) = n(x) \frac{q}{n} \quad (2.16)$$

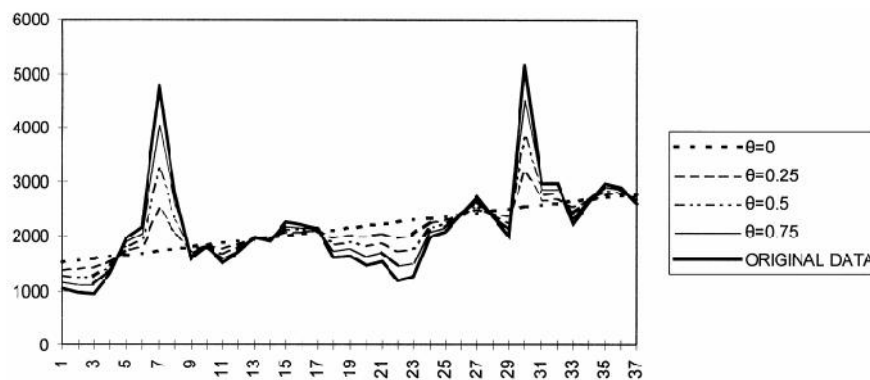
Untuk menggunakan Loess, nilai d dan q harus ditentukan terlebih dahulu. Semakin tinggi nilai q maka $\hat{g}(x)$ menjadi semakin halus. Jika nilai q mendekati tak terbatas, maka $v_i(x)$ mendekati satu dan $\hat{g}(x)$ akan mendekati sebuah polinomial *least-square* berderajat d .

2.6. Metode Theta

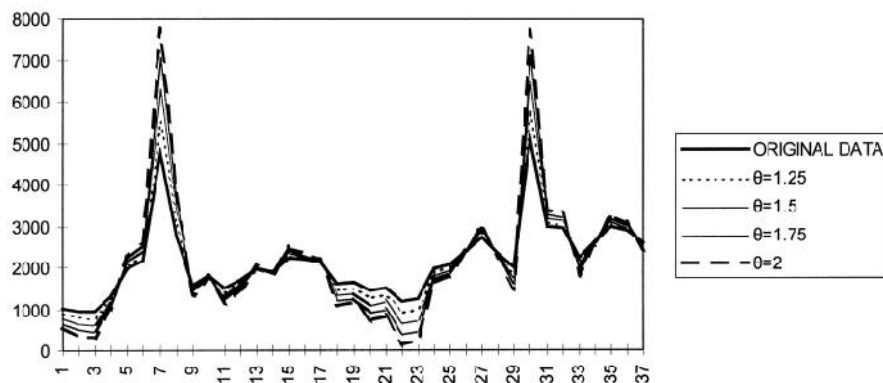
Model theta merupakan sebuah metode dekomposisi yang dapat mendekomposisi komponen data musiman dari sebuah data runtut waktu dan menghasilkan komponen jangka pendek dan jangka panjang (Assimakopoulos, dkk, 2000). Model ini didasarkan pada konsep untuk memodifikasi lengkungan lokal dari data runtut waktu. Modifikasi ini ditentukan oleh sebuah koefisien, yang disebut dengan koefisien Theta (θ) dan diterapkan pada perbedaan kedua dari data seperti pada persamaan 2.17 dimana $x''_{data} = X_t - 2X_{t-1} + X_{t-2}$ pada waktu t .

$$x''_{new}(\theta) = \theta \cdot x''_{data} \quad (2.17)$$

Jika lengkung lokal secara terus-menerus diturunkan maka grafik runut waktu akan menjadi datar seperti yang dapat dilihat pada Gambar 2.4. Semakin kecil nilai koefisien θ maka semakin hilang lengkungannya dan ketika nilainya nol maka grafik akan menjadi linier. Sebaliknya, jika nilai $\theta > 1$ maka grafik data menunjukkan adanya perbesaran seperti pada Gambar 2.5. Semakin besar nilai θ maka akan semakin memperbesar perilaku jangka pendek.



Gambar 2.4 Hasil pendataran dengan model Theta (Theodosiou, 2011b)



Gambar 2.5 Hasil perbesaran dengan model Theta (Theodosiou, 2011b)

Terdapat beberapa tahap peramalan menggunakan metode Theta. Langkah-langkah untuk melakukan peramalan dengan metode Theta adalah :

- a. Pengujian musiman : setiap data runut waktu diuji sifat musimannya secara statistik. Kriterianya adalah nilai t-test untuk nilai fungsi *auto-correlation* dalam periode satu tahun (jika data bulanan maka ada 12 pengamatan dan 4

jika data bersifat empat-bulanan) dibandingkan terhadap 1.645 yang merupakan nilai t-statistik untuk peluang 0.1

- b. *Deseasonalization* : pengambilan komponen data musiman dengan metode dekomposisi klasik
- c. Dekomposisi : setiap data runtut waktu didekomposisi menjadi 2 garis theta, garis regresi linear ($\theta = 0$) dan garis theta untuk $\theta = 2$.
- d. Ekstrapolasi : garis regresi linear diekstrapolasi dengan cara biasa sedangkan garis kedua diekstrapolasi menggunakan *exponential smoothing* sederhana.
- e. Kombinasi : hasil peramalan dari langkah (d) dikombinasikan dengan bobot setara
- f. *Reseasonalization* : hasil peramalan dikembalikan komponen data musimannya

2.7. Algoritma Genetika

Algoritma genetika yang pertama kali diperkenalkan secara terpisah oleh Holland dan De Jong pada tahun 1975 merupakan teknik pencarian nilai optimum secara stokastik berdasarkan prinsip dasar dari teori evolusi. Algoritma genetika berbeda dengan teknik konvergensi konvensional yang lebih bersifat deterministik (Gen, 1997). Algoritma genetika melakukan proses pencarian nilai optimum pada beberapa titik secara bersamaan (satu generasi) (Kosasih, dkk). Proses iterasi kemudian dilakukan dengan pendekatan generasi ke generasi yang mengalami proses evolusi, tetapi jumlah anggota (*chromosome*) pada setiap generasi, yang merupakan kumpulan solusi, umumnya dipertahankan tetap. Kromosom yang dianalisis dapat merupakan kode biner, integer atau desimal. Dalam proses evolusi, sejumlah gen yang membentuk kromosom melewati proses *crossover* (kawin silang) dan/atau mutasi (perubahan gen secara alami). Kromosom yang baik akan terus hidup, sedangkan kromosom yang buruk akan mati dengan sendirinya. Algoritma genetika menggunakan hukum transisi probabilistik untuk memilih solusi (kromosom) yang terus dipertahankan hidup sesuai dengan

ketentuan yang diinginkan (*fitness function*), sehingga proses pencarian solusi optimum dapat diarahkan ke arah yang diperkirakan akan lebih baik.

Umumnya algoritma genetika memiliki empat tahapan yaitu proses pengkodean, proses seleksi, rekombinasi, dan mutasi (Mutakhiroh, dkk, 2007). Penjelasan ringkas setiap tahap adalah sebagai berikut:

- a. Proses pengkodean merupakan sebuah proses yang penting dalam algoritma genetika karena proses pengkodean untuk setiap permasalahan berbeda-beda dan tidak semua teknik pengkodean cocok untuk setiap permasalahan. Proses pengkodean ini menghasilkan suatu deretan yang kemudian disebut kromosom. Kromosom terdiri dari sekumpulan bit yang dikenal sebagai gen. Ada beberapa macam teknik pengkodean yang dapat dilakukan dalam algoritma genetika (Lukas, 2005), diantaranya pengkodean biner (*binary encoding*), pengkodean permutasi (*permutation encoding*), pengkodean nilai (*value encoding*) dan pengkodean pohon (*tree encoding*).
- b. Seleksi adalah proses untuk menentukan individu mana saja yang akan dipilih untuk dilakukan rekombinasi dan bagaimana keturunan terbentuk dari individu-individu terpilih tersebut. Langkah pertama yang dilakukan dalam seleksi adalah pencarian nilai fitness. Masing-masing individu dalam suatu wadah seleksi akan menerima probabilitas reproduksi yang tergantung pada nilai obyektif dirinya sendiri terhadap nilai obyektif dari semua individu dalam wadah seleksi tersebut. Nilai fitness kemudian akan digunakan pada tahap seleksi berikutnya. Ada beberapa macam proses seleksi yang ada pada algoritma genetika, diantaranya :
 - Seleksi dengan Roda Roulette (*Roulette Wheel Selection*), dengan memetakan individu-individu dalam suatu segmen garis secara berurutan sedemikian hingga tiap-tiap segmen individu memiliki ukuran yang sama dengan ukuran fitness-nya.
 - Seleksi berdasarkan *Ranking Fitness* (*Rank-based Fitness*), yaitu dengan cara mengurutkan populasi menurut nilai obyektifnya.
 - Seleksi Pengambilan *Sampling Stochastic* (*Stochastic Universal Sampling*), dengan memetakan individu-individu seperti halnya roda

roulette, kemudian memberikan sejumlah pointer sebanyak individu yang ingin diseleksi pada garis tersebut.

- Seleksi Lokal (*local selection*), seleksi yang dilakukan hanya pada konstrain tertentu.
- Seleksi dengan Pemotongan (*truncation selection*), seleksi buatan yang biasanya digunakan oleh populasi yang jumlahnya sangat besar.
- Seleksi dengan turnamen (*tournament selection*), menetapkan suatu nilai turnamen untuk individu-individu yang dipilih secara acak dari suatu populasi.

c. Rekombinasi adalah proses untuk menyilangkan dua kromosom sehingga membentuk kromosom baru yang harapannya lebih baik dari pada induknya. Rekombinasi dikenal juga dengan nama pindah-silang (*crossover*). Tidak semua kromosom pada suatu populasi akan mengalami proses rekombinasi. Kemungkinan suatu kromosom mengalami proses rekombinasi didasarkan pada probabilitas pindah-silang yang telah ditentukan terlebih dahulu. Probabilitas pindah-silang menyatakan peluang suatu kromosom akan mengalami pindah-silang. Ada beberapa macam proses rekombinasi yang ada pada algoritma genetika, diantaranya :

- Rekombinasi diskrit, dengan menukar nilai variabel antar kromosom induk
- Rekombinasi menengah, merupakan metode rekombinasi yang hanya digunakan untuk variabel real dan variabel yang bukan biner.
- Rekombinasi garis, memiliki prinsip yang sama dengan rekombinasi menengah, dengan nilai alpha sama untuk semua variabel.
- Penyilangan satu titik, dengan menukar variabel-variabel antar kromosom pada satu titik untuk menghasilkan anak.
- Penyilangan banyak titik, dengan menukar variabel-variabel antar kromosom pada banyak titik untuk menghasilkan anak.
- Penyilangan seragam, dengan membuat sebuah *mask* penyilangan sepanjang panjang kromosom secara acak.

- Penyilangan dengan permutasi, dengan cara memilih sub-barisan suatu turnamen dari satu induk dengan tetap menjaga urutan dan posisi solusi yang mungkin terhadap induk lainnya.
- d. Mutasi adalah proses penambahan nilai acak yang sangat kecil dengan probabilitas rendah pada variabel keturunan. Peluang mutasi didefinisikan sebagai persentasi dari jumlah total gen pada populasi yang mengalami mutasi. Peluang mutasi mengendalikan banyaknya gen baru yang akan dimunculkan untuk dievaluasi. Jika peluang mutasi terlalu kecil, banyak gen yang mungkin berguna tidak dievaluasi, tetapi bila peluang mutasi ini terlalu besar maka akan terlalu banyak gangguan acak, sehingga anak akan kehilangan kemiripan dari induknya dan algoritma juga akan kehilangan kemampuan untuk belajar dari sejarah pencarian. Ada beberapa macam proses mutasi yang ada pada algoritma genetika, diantaranya 1) mutasi bilangan real, dengan mendefinisikan ukuran langkah mutasi, kecil atau besar ; dan 2) mutasi biner, dengan mengganti satu atau beberapa nilai gen dari kromosom.

2.8. Penelitian-penelitian Terkait

Beberapa peneliti telah menggabungkan kemampuan dari algoritma genetika dengan jaringan syaraf. Tujuan dari penggunaan algoritma genetika ke dalam jaringan syaraf antara lain untuk menentukan pembagian keseluruhan data menjadi data latih, data uji, dan data validasi. Bowden, dkk (2002) meneliti pengaruh pembagian data terhadap kinerja jaringan syaraf untuk meramalkan kebutuhan air di Australia Selatan. Metode yang digunakan adalah algoritma genetika dan *self-organizing map* (SOM). Berdasarkan pengujian yang dilakukan oleh Bowden, jaringan syaraf yang datanya dibagi menggunakan algoritma genetika dan SOM menghasilkan peramalan dengan pengurangan galat (RMSE) sebanyak 24.2 % dan 9.9% dibandingkan pembagian data secara konvensional. Kamp, dkk (2006) melakukan optimasi data latih menggunakan algoritma genetika untuk meningkatkan kinerja jaringan syaraf dalam memodelkan curah

hujan di Belanda. Berdasarkan pengujian yang dilakukan oleh Kamp, pemodelan dengan jaringan syaraf yang data latihnya dioptimasi dengan algoritma genetika memiliki akurasi lebih tinggi dibandingkan tanpa optimasi.

Beberapa kelebihan GRNN telah menarik minat banyak peneliti untuk mengaplikasikannya dalam persoalan prediksi atau peramalan. Cigizoglu (2005) menggunakan GRNN untuk memperkirakan aliran air bulanan di Turki. Duzgun (2010) mengimplementasikan GRNN untuk memprediksi rasio inflasi di Turki dan menyimpulkan bahwa hasil GRNN lebih baik dibandingkan dengan model ARIMA. Zhou, dkk (2010) mengaplikasikan GRNN yang dikombinasikan dengan *particle swarm optimization* (PSO) untuk meramalkan penggunaan air di China. Ladlani, dkk (2012) menerapkan GRNN untuk memprediksi evapotranspirasi dari permukaan air yang digunakan dalam irigasi di Algeria dan menunjukkan kelebihan kinerja GRNN dibandingkan dengan RBFNN. Hannan, dkk (2010) membandingkan kinerja RBF dan GRNN dalam mendiagnosa penyakit hati. Popescu, dkk (2004) menggunakan GRNN untuk memprediksi hilangnya jalur perambatan pada lingkungan dalam ruang (*indoor*).

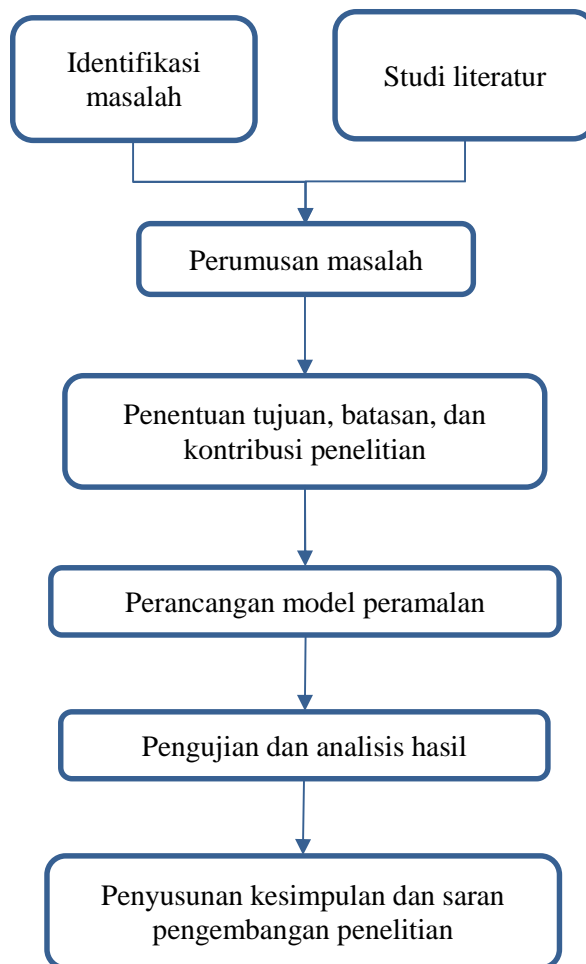
Kombinasi antara algoritma genetika dengan GRNN juga pernah diteliti sebelumnya. Umumnya, algoritma genetika digunakan untuk menemukan nilai sigma yang merupakan salah satu parameter bebas pada GRNN. Casey, dkk (2006) menggunakan GRNN yang digabungkan dengan algoritma genetika untuk memprediksi tsunami. Parameter yang dioptimasi adalah nilai sigma dan *threshnold*. Peneliti menguji kinerja GRNN dengan menggunakan sigma tunggal yang berlaku untuk seluruh neuron pada lapisan pola dan sigma jamak yang berbeda nilainya pada tiap neuron di lapisan pola. Berdasarkan pengujian, GRNN dengan sigma tunggal memiliki kinerja akurasi yang lebih baik dibandingkan sigma jamak. Kim, dkk (2008) menggunakan GRNN dan algoritma genetika untuk meramalkan evapotranspirasi di Korea. Algoritma genetika digunakan untuk mengoptimasi nilai sigma dan untuk pelatihan jaringan (menggantikan algoritma propagasi balik). Yazici, dkk (2006) menguji kinerja GRNN dengan algoritma genetika pada persoalan klasifikasi. Algoritma genetika digunakan untuk menyeleksi vektor latih. Berdasarkan pengujian oleh Yazici, akurasi

klasifikasi oleh GRNN dengan algoritma genetika lebih baik dibandingkan tanpa algoritma genetika. Selain itu, jaringan yang dihasilkan juga lebih efisien karena tidak seluruh data latih digunakan dalam proses pelatihan.

BAB 3

METODOLOGI PENELITIAN

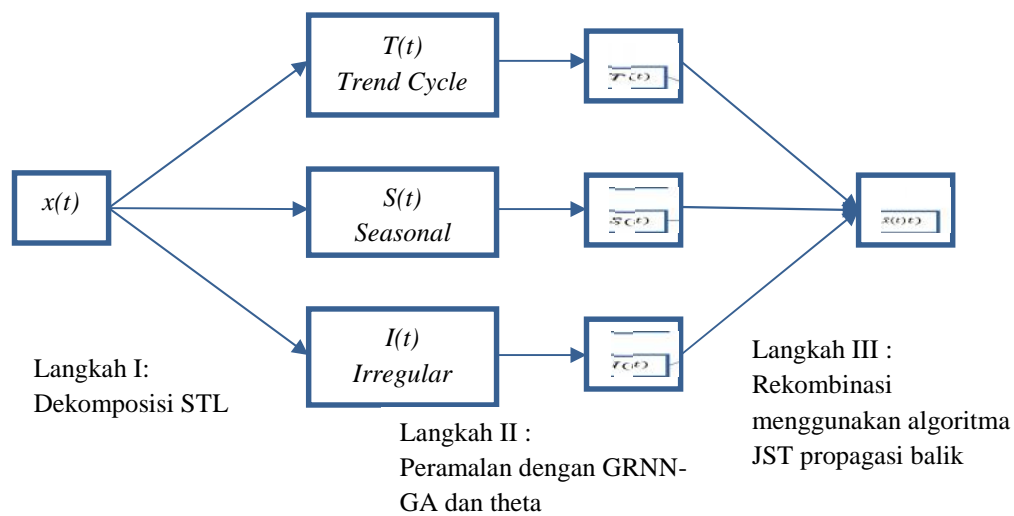
Dalam bab ini diuraikan tahap-tahap yang dilakukan pada penelitian ini. Secara keseluruhan, tahap-tahap penelitian yang dilakukan dapat dilihat pada Gambar 3.1. Tahapan identifikasi masalah, perumusan masalah, dan penentuan tujuan, batasan, dan kontribusi penelitian telah dijelaskan pada bab 1, sedangkan studi literatur telah diuraikan pada bab 2, sehingga pada bab 3 akan dijelaskan mengenai model peramalan, rencana pengujian, dan analisis hasil pengujian.



Gambar 3.1 Metodologi Penelitian

3.1. Perancangan Model Peramalan

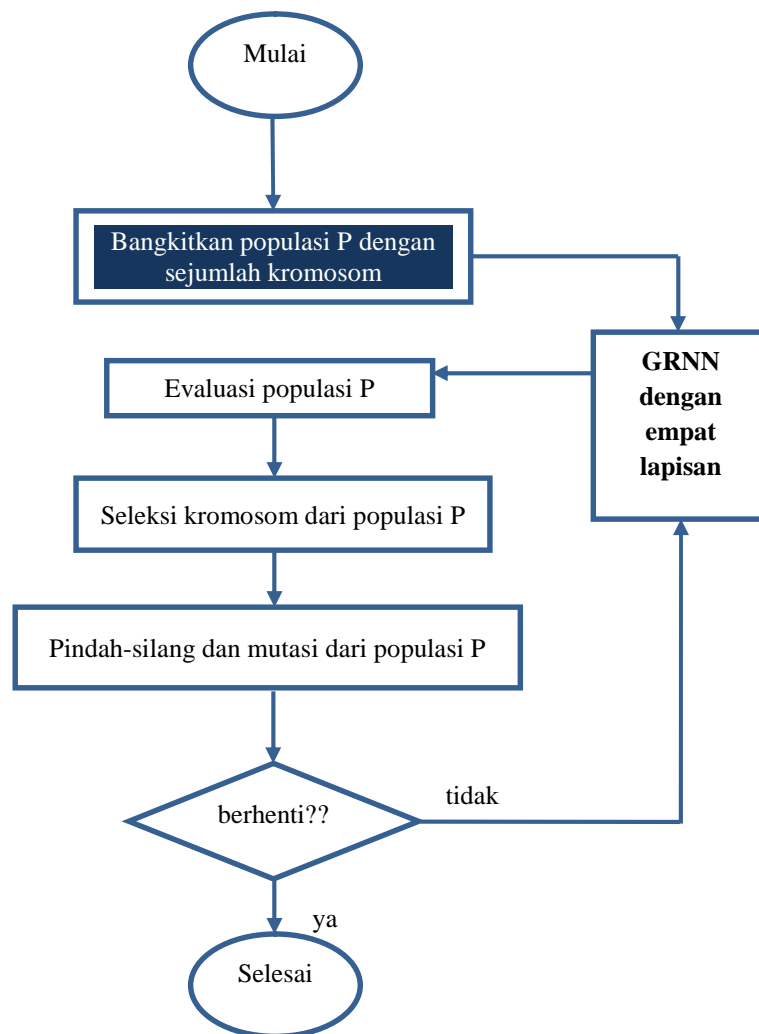
Tujuan dari penelitian ini adalah mengimplementasikan *generalized regression neural network* (GRNN) dan metode theta untuk menghasilkan peramalan harga emas yang lebih akurat. GRNN akan dikombinasikan dengan algoritma genetika untuk seleksi data latih dengan tujuan meningkatkan akurasi peramalan oleh GRNN. Untuk itu, langkah-langkah pengerjaan dapat dilihat pada Gambar 3.2.



Gambar 3.2 Perancangan model peramalan

Pada langkah satu, data harga emas didekomposisi menggunakan metode STL seperti yang telah dijelaskan di sub-bab 2.5.1. Data $x(t)$ akan dipisahkan menjadi komponen data siklus-tren (*trend-cycle*) $T(t)$, komponen musiman (*seasonal*) $S(t)$, dan komponen residual (*irregular*) $I(t)$. Langkah kedua adalah meramalkan ketiga komponen hasil dekomposisi. Komponen $S(t)$ dan $I(t)$ diramalkan menggunakan GRNN yang dikombinasikan dengan algoritma genetika (GA), sedangkan komponen $T(t)$ diprediksi dengan model theta karena menurut (Theodosiou, 2011a) GRNN kurang sesuai untuk meramalkan komponen tren. Prediksi dengan model theta mengikuti langkah-langkah yang telah diuraikan pada sub-bab 2.6. Adapun model kombinasi GRNN-GA dapat dilihat pada Gambar 3.3. Algoritma genetika akan digunakan untuk memilih vektor latih dari

keseluruhan vektor latih yang ada. Untuk itu, akan dicari format kromosom yang dapat merepresentasikan solusi yang diharapkan. Penentuan kromosom merupakan langkah penting dalam GA. Kromosom-kromosom tersebut merupakan representasi dari data latih bagi GRNN. Seleksi kromosom didasarkan pada galat yang dihasilkan dari proses pelatihan. Proses ini berlangsung secara terus menerus hingga kondisi berhenti terpenuhi. Kondisi berhentinya algoritma genetika adalah terpenuhinya iterasi atau nilai galat yang tidak lagi berubah. Adapun langkah ketiga adalah menggabungkan hasil peramalan dari ketiga komponen. Metode yang digunakan untuk menggabungkan komponen data musiman, tren, dan residual adalah jaringan syaraf tiruan propagasi balik. Tahapan ini mengacu pada penelitian Hansen, dkk (2003).



Gambar 3.3 Model peramalan GRNN-AG (langkah II)

3.1.1. Dekomposisi STL

Data runtut waktu harga emas didekomposisi menggunakan STL menjadi komponen data musiman, tren, dan residual. Kode program menggunakan R ditunjukkan ada Gambar 3.4. Parameter yang ditentukan pada metode STL adalah jendela musiman, nilai parameter *low-pass filter*, jumlah iterasi dalam, dan jumlah iterasi luar. Nilai jendela musiman ditentukan berdasarkan rata-rata jumlah hari dalam satu tahun yaitu 261, nilai parameter *low-pass filter* disamakan dengan frekuensi data runtut waktu yaitu 261, jumlah iterasi dalam adalah 0, dan jumlah iterasi luar adalah 0. Pada penelitian ini nilai iterasi luar adalah nol karena terdeteksinya *outlier* pada data. Adapun nilai parameter lainnya mengacu pada penelitian oleh Cleveland (1990).

Hasil dekomposisi harga emas harian menggunakan STL dapat dilihat pada Gambar 3.5. Gambar 3.5 menunjukkan data asli yaitu harga emas harian pada baris pertama, komponen data musiman pada baris kedua, komponen data tren pada baris ketiga, dan komponen data residual pada baris keempat. Keempat jenis data tersebut berada pada rentang waktu yang sama yaitu mulai tanggal 1 Januari 2003 hingga 31 Desember 2013.

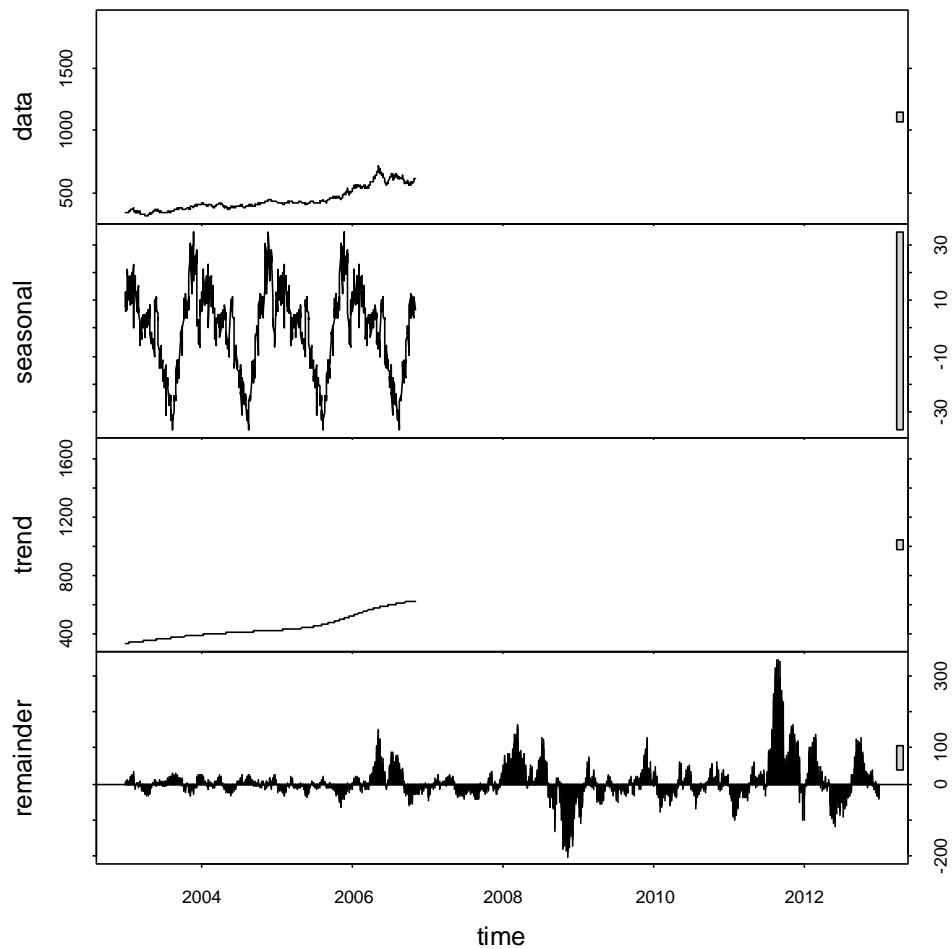


```
E:\unmi nitip\MyThesis\programs\STL - R Editor
File Edit Package Help
gold_daily<- read.csv("E:/unmi nitip/MyThesis/data/09_13/harga2000.dat")
goldmts <- ts(gold_daily)
goldmts <- ts(gold_daily, frequency=261, start=c(2003,1))
d <- stl(goldmts,s.window=261, l.window=261,inner=2, outer=0 )

write(d$time.series[, "trend"],file="E:/unmi nitip/MyThesis/data/09_13/trend.dat", ncolumns=1, append=FALSE)
write(d$time.series[, "seasonal"],file="E:/unmi nitip/MyThesis/data/09_13/seasonal.dat", ncolumns=1, append=FALSE)
write(d$time.series[, "remainder"],file="E:/unmi nitip/MyThesis/data/09_13/residual.dat", ncolumns=1, append=FALSE)

plot(d)
```

Gambar 3.4 Kode Program Dekomposisi STL



Gambar 3.5 Hasil dekomposisi harga emas harian menggunakan metode STL

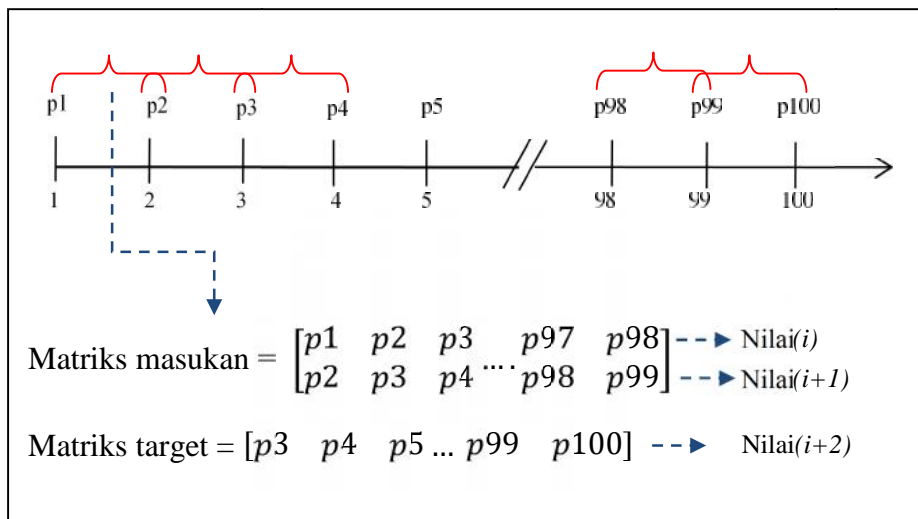
3.1.2. Peramalan Komponen Data Musiman dan Residual

Peramalan komponen data musiman dan residual dilakukan menggunakan metode GRNN-GA. Adapun alur secara garis besar dapat dilihat pada Gambar 3.3. Tahap-tahap peramalan menggunakan GRNN-GA adalah penyiapan data latih, pengkodean kromosom, evaluasi populasi, seleksi, rekombinasi, dan mutasi. Penjelasan setiap tahap adalah sebagai berikut:

a. Penyiapan Data Latih

Untuk keperluan pelatihan dan validasi data, digunakan data mulai tanggal 1 Januari 2003 sampai dengan 31 Desember 2013 dengan total sebesar

2.869 data. Proses pelatihan GRNN-GA menggunakan sebanyak 2.609 data sedangkan validasi menggunakan 260 data. Selanjutnya, untuk dapat diproses menggunakan GRNN-GA data runut waktu harus ditransformasikan menjadi matriks yang berisi data masukan dan target/keluaran. Untuk menggambarkan proses transformasi ini, misalnya akan diramalkan nilai entitas esok hari berdasarkan nilai hari ini dan kemarin, maka nilai entitas esok hari menjadi variabel target sedangkan nilai hari ini dan kemarin menjadi variabel masukan. Prosedur ini dilakukan secara berulang dengan menggeser jendela masukan dan target sesuai dengan banyaknya data runut waktu. Berdasarkan prosedur ini, disusun matriks masukan dan target dari data runut waktu. Untuk 100 data observasi dan dua variabel masukan, jika nilai(t), nilai($t+1$) dan nilai($t+2$) berturut-turut adalah nilai entitas kemarin, hari ini, dan esok hari dengan $t = 1, 2, 3, 4, 5, \dots, 98, 99, 100$, maka prosedur pembentukan matriks ditunjukkan pada Gambar 3.6. Pembentukan matriks data validasi juga mengikuti prosedur yang sama.



Gambar 3.6 Pembentukan matriks dari data runut waktu

b. Pengkodean Kromosom

Kromosom merupakan representasi dari solusi yang diharapkan pada persoalan optimasi menggunakan algoritma genetika. Pada penelitian ini,

kromosom merupakan representasi dari data latih GRNN. Data dalam bentuk runut waktu diubah terlebih dahulu menjadi bentuk pasangan masukan dan keluaran sesuai pada tahap sebelumnya. Jumlah data masukan akan menunjukkan banyaknya simpul masukan pada GRNN, sedangkan banyaknya data keluaran menunjukkan banyaknya simpul keluaran GRNN. Banyaknya simpul masukan GRNN menunjukkan n hari sebelum hari prediksi, sedangkan banyaknya simpul keluaran menunjukkan m hari yang akan diprediksi. Matriks latih yang terbentuk selanjutnya akan dibagi menjadi data latih dan data uji dengan perbandingan 70:30.

Pada penelitian ini, nilai n dicari melalui uji coba, sedangkan nilai m ditetapkan satu. Kode program untuk mengubah data runut waktu menjadi matriks masukan dan keluaran ditunjukkan pada Gambar 3.7. Setelah diubah menjadi pasangan masukan dan keluaran, maka akan terbentuk dua matriks yaitu matrik yang berisi data masukan (mIn) dan matriks data keluaran ($mOut$). Untuk jumlah data latih sebanyak l , maka matriks data masukan akan berukuran $l \times n$, sedangkan matriks data keluaran berukuran $l \times m$. Nilai l dipengaruhi oleh ukuran simpul masukan n .

```

1  function [mIn, mOut] = setMatriks(data, awalPos, interval, n)
2  % fungsi untuk mengambil subset dari time series dan membentuk matriks latih
3  % latih
4  % mIn : matriks hasil load data
5  % awalPos : posisi data awal subset
6  % interval : banyaknya data u/diambil
7  % n : jumlah input jaringan
8
9  rentang = awalPos+interval-1;
10 m = data([awalPos:rentang],:);
11
12 for i=1:1:(n+1)
13     tmp(:,i) = m([i:(end-n+(i-1))],:);
14 end
15
16 mIn = [];
17 for i=1:1:n
18     mIn = vertcat(mIn, tmp(:, i)');
19 end
20
21 mOut = tmp(:, (n+1))';

```

Gambar 3.7 Kode program untuk mengubah data seri waktu menjadi matriks

Format kromosom menggunakan pengkodeaan biner dengan panjang kromosom sebanyak data latih awal (l). Nilai satu atau nol menunjukkan masuk atau tidaknya baris pada matriks latih awal ke dalam matriks latih seleksi. Populasi yang terdiri atas sejumlah kromosom yang dibangkitkan secara acak ditunjukkan pada kode di Gambar 3.8.

```

1 function pop = encodingKrom (popsize, stringlength)
2     % fungsi untuk membangkitkan populasi acak
3     pop = round(rand(popsize, stringlength+1));
4

```

Gambar 3.8 Kode program pembangkitan kromosom

c. Evaluasi Populasi

Tiap kromosom pada populasi selanjutnya akan dievaluasi untuk ditentukan nilai *fitness*-nya. Fungsi obyektif yang digunakan adalah fungsi perhitungan *root mean squared error* (RMSE). Pada proses evaluasi, dihitung RMSE peramalan menggunakan GRNN berbasis matriks latih seleksi dan berbasis matriks latih keseluruhan. RMSE diperoleh dari hasil pengujian *out-sample* yaitu pengujian dengan menggunakan 30% data yang berbeda dari data latih. Adapun fungsi fitness untuk individu $f(i)$ yang digunakan adalah pembagian antara RMSE GRNN menggunakan matriks latih keseluruhan dan RMSE GRNN menggunakan matriks latih seleksi yang ditunjukkan pada persamaan 2.18. Kode program proses evaluasi dapat dilihat pada Gambar 3.9.

$$f(i) = \frac{RMSE_{(GRNN)}}{RMSE_{(GRNN-GA)}} \quad (2.18)$$

d. Seleksi

Seleksi merupakan mekanisme untuk membentuk populasi yang mempertahankan individu dengan fitness tinggi dan mengeliminasi individu dengan fitness yang lebih rendah. Kode program untuk proses seleksi dapat dilihat pada Gambar 3.10.

```

1  function pop = decodingKrom(pop, stringlength, inLatih, outLatih, inUji, outUji, spread)
2  % pop: populasi hasil encoding
3  % stringlength : panjang tiap gen
4  % dataLatihAwal : data latih komplit
5  % dataUji : data untuk pengujian jaringan
6
7  popsize = size(pop,1); % menghasilkan baris x kolom dari pop
8
9  for i=1:popsize
10     dataSelIn = setDataSeleksi(inLatih, pop(i,(1:end-1)));
11     dataSelOut = setDataSeleksi(outLatih, pop(i,1:end-1));
12
13     %GRNN dengan data komplit
14     net1 = newgrnn(inLatih, outLatih, spread);
15     net1.performFcn = 'mse';
16     %GRNN dengan data yang sudah diseleksi
17     net2 = newgrnn(dataSelIn, dataSelOut, spread);
18     net2.performFcn = 'mse';
19
20     y1 = sim(net1, inUji);%data komplit
21     y2 = sim(net2, inUji);%data seleksi
22
23     e1 = outUji - y1;%error dengan data komplit
24     e2 = outUji - y2;%error dengan data seleksi
25
26     perf = mse(e1)/mse(e2);
27
28     pop(i,stringlength+1) = perf;
29 end

```

Gambar 3.9 Kode program evaluasi kromosom

```

1  function selected = selection(pop, popsize, stringlength, dimension)
2  popsize_new =size(pop,1);
3  r = rand(1, popsize);
4  fitness= pop(:, dimension*stringlength+1);
5  fitness = fitness/sum(fitness);
6  fitness = cumsum(fitness);
7
8  for i=1:popsize
9      for j=1:popsize_new
10         if r(i) <= fitness(j)
11             selected(i,:) = pop(j,:);
12             break;
13         end
14     end
15 end

```

Gambar 3.10 Kode program proses seleksi

e. Rekombinasi dan Mutasi

Mekanisme rekombinasi yang digunakan adalah memilih dua individu secara acak dari populasi kemudian mengubah bit pada posisi yang sama dari

kedua individu tersebut. Adapun proses mutasi dilakukan dengan secara acak memilih salah satu bit pada kromosom kemudian membalikinya yaitu mengubah nol menjadi satu atau mengubah satu menjadi nol. Kode program proses rekombinasi dan mutasi dapat dilihat pada Gambar 3.11 dan Gambar 3.12.

```

1  function new_pop = cross_over (pop, popsize, stringlength, dimension)
2  % fungsi untuk melakukan cross over terhadap populasi
3  - match = round (rand(1, popsize)*(popsize-1))+1;
4  - for i=1:popsize
5  -     [child1, child2] = cross_running(pop(i,:), pop(match(i,:), stringlength, dimension);
6  -     new_pop(2*i-1:2*i, :) = [child1; child2];
7  - end

```

Gambar 3.11 Kode program proses rekombinasi

```

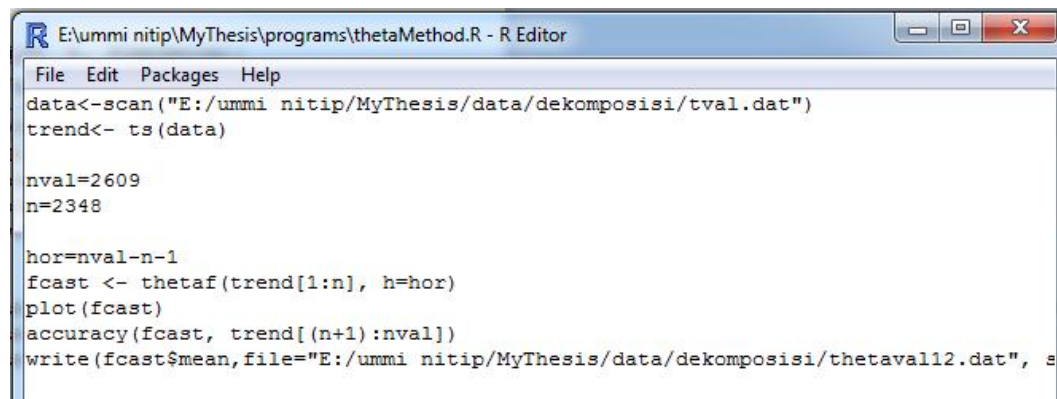
1  function new_pop = mutation(new_pop, stringlength, dimension, pm)
2  % pm - probability of mutation
3  - new_popsize = size(new_pop, 1);
4  - for i=1:new_popsize
5  -     if rand<pm
6  -         mpoint = round(rand(1, dimension)*(stringlength-1))+1;
7  -         for j=1:dimension
8  -             new_pop(i, (j-1)*stringlength+mpoint(j)) = 1 - new_pop(i, (j-1)*stringlength+mpoint(j));
9  -         end
10 -     end
11 - end

```

Gambar 3.12 Kode program proses mutasi

3.1.3. Peramalan Komponen Tren

Peramalan untuk komponen tren dilakukan menggunakan metode theta yang diimplementasikan menggunakan paket *forecast* pada R. Fungsi yang digunakan adalah fungsi `thetaf()` dengan parameter data runtut waktu untuk pemodelan dan rentang waktu peramalan. Data runtut waktu yang digunakan adalah 70% dari keseluruhan data yaitu sebanyak 1.826 observasi sedangkan lamanya rentang hari peramalan adalah sebanyak 783 hari yaitu keseluruhan data dikurangi hari observasi. Kode program implementasi metode theta dapat dilihat pada Gambar 3.13.



```
E:\ummi nitip\MyThesis\programs\thetaMethod.R - R Editor
File Edit Packages Help
data<-scan("E:/ummi nitip/MyThesis/data/dekomposisi/tval.dat")
trend<- ts(data)

nval=2609
n=2348

hor=nval-n-1
fcast <- thetaf(trend[1:n], h=hor)
plot(fcast)
accuracy(fcast, trend[(n+1):nval])
write(fcast$mean,file="E:/ummi nitip/MyThesis/data/dekomposisi/thetaval12.dat", s
```

Gambar 3.13 Implementasi metode theta

3.1.4. Penggabungan menggunakan Algoritma JST Propagasi Balik

Hasil peramalan pada langkah 3.1.2 dan 3.1.3 digabungkan menggunakan JST propagasi balik (BPNN) untuk memperoleh peramalan tunggal. Data yang diramalkan adalah 30% dari keseluruhan data. Untuk komponen data musiman dan residual, model GRNN dibangun menggunakan matriks data latih hasil seleksi GA. Model yang terbangun selanjutnya digunakan untuk meramalkan data uji. Hasilnya digabungkan dengan hasil peramalan oleh metode theta. Kode program penggabungan dapat dilihat pada Gambar 3.14.

Pada Gambar 3.14, pembentukan matriks masukan untuk BPNN ditunjukkan pada baris delapan dan sembilan dimana baris delapan merupakan matriks latih seleksi dan baris sembilan matriks latih keseluruhan. Selanjutnya dilakukan normalisasi terhadap kedua matriks sebelum dimasukkan ke fungsi pembentukan model BPNN pada baris 15. Setelah pengaturan beberapa nilai parameter, dilakukan pelatihan untuk memperoleh model terbaik. Peramalan untuk data uji ditunjukkan pada baris ke-25 dari program. Hasil peramalan kemudian dihitung akurasi menggunakan RMSE dan dibandingkan antara menggunakan matriks latih seleksi dan matriks latih keseluruhan.

```

8 - inputga = [s1; r1; t];
9 - input = [s2; r2; t];
10
11 - [inputn, meanin1, stdin1, targetn1, meant1, stdt1] = prestd(input, target);
12 - [inputgan, meanin2, stdin2, targetn2, meant2, stdt2] = prestd(inputga, target);
13
14 %desain ANN dg parameter default tanpa GA
15 - bpnn=newff(inputn,targetn1);
16 - bpnn.trainParam.epochs=100;
17 - bpnn.trainParam.goal=0.01;
18 - bpnn.trainParam.lr=0.5;
19 - bpnn.trainParam.show=10;
20 - bpnn.trainParam.mc=0.7;
21 %train network
22 - bpnn=train(bpnn, inputn, targetn1);
23 - save bpnn;
24 %tes
25 - res = sim(bpnn, inputn);
26 %denormalisasi hasil tes
27 - res = poststd(res, meant1, stdt1);
28

```

Gambar 3.14 Kode program rekombinasi

3.2. Pengujian dan Analisis Hasil

Pada sub-bab ini, dijelaskan mengenai data uji coba dan analisis hasil uji coba, lingkungan uji coba dan skenario uji coba dalam penelitian ini. Skenario uji coba yang dilakukan yaitu uji coba validasi dan uji coba perbandingan dengan metode peramalan lain untuk harga emas.

3.2.1. Data Uji Coba

Pada penelitian ini, data uji coba diperoleh dari situs World Gold Council (<http://www.gold.org/>). Data berupa harga emas harian selama 10 tahun mulai dari tahun 2003 sampai dengan tahun 2012 dengan total jumlah data adalah 2.609 data. Nominal yang digunakan adalah dollar (USD) yang menunjukkan harga emas per *troy ounce*. Dari seluruh data yang ada, data 3 tahun terakhir akan digunakan sebagai data uji sehingga total data latih awal adalah 1.826 data dan 783 data sebagai data uji. Pembagian ini mengacu pada penelitian Braddock, dkk (1998) dalam Bowden, dkk (2002) yang membagi data dengan komposisi 70:30 yaitu 70% data latih dan 30% data uji.

3.2.2. Lingkungan Uji Coba

Pada penelitian ini, data uji coba dilakukan menggunakan laptop dengan spesifikasi Processor Intel ® Dual Core 1.86 GHz dan RAM 2.5 GB. Program penelitian dibuat menggunakan perangkat lunak R versi 2.15.2 dan Matlab Versi R2009a. Selain itu, digunakan juga perangkat lunak Microsoft Excel 2010 untuk pembacaan, perekayasa statistik, dan penyimpanan data.

3.2.3. Jenis Uji Coba

Pada penelitian ini akan dilakukan beberapa jenis uji coba untuk mengetahui kinerja model peramalan yang diusulkan. Terdapat tiga jenis pengujian yaitu uji coba penentuan parameter model peramalan GRNN dan algoritma genetika, uji coba peramalan menggunakan GRNN tanpa algoritma genetika, dan uji coba peramalan menggunakan model Arima. Dua uji coba terakhir ditujukan untuk membandingkan kinerja model peramalan yang diusulkan dengan model peramalan yang ada. Penjelasan ringkas mengenai setiap jenis uji coba adalah sebagai berikut :

a. Uji coba penentuan parameter model peramalan STL-GRNN-GA-Theta

Uji coba ini mempunyai tujuan untuk menentukan parameter-parameter model peramalan. Parameter-parameter tersebut yaitu jumlah simpul masukan GRNN, nilai penyebaran, jumlah populasi, nilai probabilitas mutasi, dan jumlah generasi. Tahap ini dilakukan dengan mengujikan beberapa nilai pada tiap parameter tersebut sehingga diperoleh akurasi terbaik. Akurasi akan diukur menggunakan *root mean square error* (RMSE). Nilai yang diperoleh dari pengujian ini akan digunakan untuk membangun model peramalan GRNN-GA.

b. Uji coba perbandingan beberapa metode peramalan

Metode peramalan yang digunakan sebagai pembanding adalah metode STL-GRNN-Theta dan ARIMA. Metode STL-GRNN-Theta merupakan metode peramalan sebelumnya yang digunakan oleh Theodosiu (2011a), sedangkan ARIMA dipilih sebagai pembanding karena pada penelitian Hussein, dkk, (2011) akurasi peramalannya lebih baik daripada peramalan menggunakan jaringan syaraf tiruan. Untuk

metode pembandingan pertama, uji coba dilakukan dengan cara melatih GRNN menggunakan keseluruhan data yang dibagi menjadi dua bagian yaitu 1826 data pertama sebagai data latih dan sisanya (783 data) sebagai data uji. Selanjutnya model yang dihasilkan akan dibandingkan akurasi dan waktunya dengan model yang diperoleh pada langkah (a) untuk menganalisis kekurangan dan kelebihanannya.

Uji coba kedua yaitu menggunakan ARIMA dilakukan dengan membandingkan akurasi hasil peramalan dengan GRNN dan algoritma genetika (pada langkah a) dan hasil peramalan dengan model ARIMA. Hasil yang diharapkan adalah bahwa model peramalan yang mengintegrasikan GRNN dan algoritma genetika memberikan akurasi yang lebih baik dibandingkan dengan model ARIMA. Selanjutnya akan diberikan analisis sesuai dengan hasil perbandingan.

3.2.4. Analisis Hasil Uji Coba

Berdasarkan hasil uji coba pada tahap 3.2.3, dilakukan analisis untuk mengetahui kinerja model yang diusulkan. Adapun jenis analisis uji coba adalah sebagai berikut:

- a. Analisis pengujian penentuan parameter model peramalan STL-GRNN-GA-Theta

Analisis dilakukan berdasarkan nilai parameter-parameter yang dihasilkan dari pengujian pada langkah 3.2.3.(a). Dari beberapa nilai yang diujikan untuk setiap parameter, akan ditentukan nilai yang menghasilkan RMSE terkecil. Nilai-nilai ini kemudian digunakan untuk membentuk model jaringan syaraf GRNN. Dari hasil pengujian menggunakan 30% data, akan diperoleh akurasi peramalan menggunakan GRNN-GA.

- b. Analisis pengujian perbandingan beberapa metode peramalan

Analisis dilakukan dengan membandingkan akurasi dan kecepatan peramalan menggunakan STL-GRNN-GA-Theta yang dihasilkan dari

pengujian (a) dengan peramalan menggunakan metode STL-GRNN-Theta dan ARIMA. Akurasi dibandingkan berdasarkan nilai RMSE hasil pengujian dari kedua model, sedangkan kecepatan diperoleh dari waktu pembentukan model dan waktu peramalannya. Hasil peramalan dari kedua model juga akan dibandingkan dengan data aktual untuk mengetahui kinerjanya.

BAB 4

UJI COBA DAN ANALISIS HASIL

Bab uji coba dan analisis hasil menjelaskan lingkungan uji coba, data uji coba, jenis uji coba, dan analisis hasil uji coba. Jenis uji coba menjelaskan tentang prosedur penentuan parameter model peramalan dan perbandingan dengan beberapa metode peramalan lain.

4.1. Lingkungan Uji Coba

Pada penelitian ini, uji coba dilakukan menggunakan laptop dengan spesifikasi Processor Intel ® Dual Core 1.86 GHz dan RAM 2.5 GB. Program penelitian dibuat menggunakan perangkat lunak R versi 2.15.2 dan Matlab Versi R2009a. Selain itu, digunakan juga perangkat lunak Microsoft Excel 2010 untuk pembacaan, perekayasaan statistik, dan penyimpanan data.

4.2. Data Uji Coba

Pada penelitian ini, data uji coba diperoleh dari situs World Gold Council (<http://www.gold.org/>). Data berupa harga emas harian selama 10 tahun mulai tanggal 1 Januari 2003 hingga 31 Desember 2012 dengan total jumlah data adalah 2609 data tidak termasuk hari libur. Nominal yang digunakan adalah dollar (USD) yang menunjukkan harga emas per *troy ounce*. Dari seluruh data yang ada, data 3 tahun terakhir atau sebanyak 30% dari keseluruhan data digunakan sebagai data uji dan data 7 tahun pertama atau 70% digunakan sebagai data latih awal. Banyaknya data latih awal adalah 1826 data dan 783 data sebagai data uji. Pembagian ini mengacu pada penelitian Braddock, dkk (1998) dalam Bowden, dkk (2002) yang menggunakan data 3 tahun terakhir sebagai validasi dan juga penelitian Hussein, dkk (2011) yang menyatakan bahwa komposisi data uji adalah 30% dari keseluruhan data.

4.3. Jenis Uji Coba

Jenis uji coba menjelaskan mengenai prosedur uji coba yang dilakukan. Tujuannya adalah untuk mengetahui kinerja dan keandalan metode peramalan dalam penelitian ini. Uji coba yang dilakukan adalah penentuan parameter model peramalan dan uji coba perbandingan beberapa metode peramalan.

4.3.1. Uji Coba Penentuan Parameter Model Peramalan

Berdasarkan metodologi yang telah diuraikan pada bab sebelumnya, hasil dari dekomposisi data diprediksi menggunakan metode *generalized regression neural network* (GRNN) dan metode theta. Untuk itu, GRNN harus dilatih terlebih dahulu dalam rangka menentukan model yang terbaik. Terdapat 2 model GRNN yang akan digunakan yaitu untuk memodelkan dan meramalkan komponen data musiman (*seasonal*) dan residu (*remainder*), sedangkan prediksi komponen data tren menggunakan metode theta. Untuk masing-masing model GRNN, parameter yang diestimasi melalui pengujian adalah jumlah masukan jaringan, faktor penyebaran (*spread*), jumlah individu dalam populasi, nilai probabilitas mutasi, dan jumlah generasi. Kinerja hasil peramalan diukur menggunakan *root mean squared error* (RMSE) yang menunjukkan akurasi hasil peramalan dibandingkan data asli. Kinerja peramalan dinyatakan baik jika RMSE bernilai minimum.

4.3.2. Uji Coba Perbandingan Metode Peramalan

Uji coba dilakukan untuk mengukur kinerja metode peramalan pada penelitian ini dibandingkan dengan beberapa metode peramalan lain. Metode lain yang akan digunakan adalah model peramalan yang diusulkan oleh Theodosiou (2011a) dan metode ARIMA. Metode peramalan Theodosiou akan diimplementasikan ulang untuk mengetahui perbandingan kinerjanya dengan metode peramalan pada penelitian ini yaitu metode peramalan yang menggunakan GRNN dan algoritma genetika.

4.4. Pelaksanaan dan Hasil Uji Coba

Sub bab ini menjelaskan tentang pelaksanaan uji coba sesuai dengan rencana pada sub bab 4.3 dan hasilnya. Detail hasil uji coba dapat dilihat pada lampiran.

4.4.1. Pelaksanaan Uji Coba Penentuan Parameter Model Peramalan

Pada bagian ini, uji coba penentuan parameter model peramalan dilakukan terhadap metode GRNN. Untuk pemilihan model GRNN, parameter yang dicari adalah jumlah masukan jaringan, faktor penyebaran (*spread*), jumlah individu dalam populasi algoritma genetika, nilai probabilitas mutasi, dan jumlah generasi. GRNN akan digunakan untuk memodelkan 2 komponen yaitu komponen data musiman dan komponen residual. Penentuan parameter model GRNN untuk kedua komponen ini dibuat terpisah karena masing-masing komponen memiliki karakteristik data tertentu. Sebelum dimodelkan menggunakan GRNN, data harga emas harian didekomposisi terlebih dahulu menggunakan metode STL.

4.4.1.1. Penentuan Parameter Model GRNN untuk Komponen Data Musiman

a. Penentuan jumlah masukan jaringan GRNN

Uji coba dilakukan untuk menemukan jumlah simpul pada lapisan masukan GRNN yang paling baik. Uji coba diimplementasikan pada GRNN menggunakan data harga emas harian yang telah didekomposisi sebelumnya. GRNN diimplementasikan tanpa menggunakan algoritma genetika (AG) dengan asumsi bahwa hasil yang diperoleh juga dapat diterapkan pada kombinasi GRNN dan AG. Uji coba dilakukan dengan mengujikan beberapa nilai sebagai jumlah simpul masukan GRNN yaitu 7, 15, 30, 45, 60, 75, dan 90, sedangkan jumlah simpul keluaran adalah satu. Untuk tiap nilai jumlah simpul masukan diujikan beberapa nilai penyebaran yaitu 6, 8, 10, 12, 14, dan 16.

Hasil uji coba dapat dilihat pada Tabel 4.1. Nilai RMSE terbaik tiap nilai masukan uji ditandai dengan sel berwarna abu-abu pada setiap baris. Untuk seluruh jumlah simpul masukan yang diujikan, nilai penyebaran 6 memberikan RMSE terkecil yaitu 3,1046 untuk jumlah simpul 7, RMSE 0,48039 untuk jumlah simpul 15, RMSE 0,028625 untuk jumlah simpul 30, RMSE 0,0030236 untuk jumlah simpul 45, RMSE 0,0027364 untuk jumlah simpul 60, RMSE 0,0027283 untuk jumlah simpul 75, dan RMSE 0,0027309 untuk jumlah simpul 90.

Tabel 4.1 Hasil pengujian penentuan jumlah simpul masukan GRNN untuk komponen data musiman

N simpul masukan	RMSE untuk nilai penyebaran					
	6	8	10	12	14	16
7	3,1046	3,9489	4,5288	4,9488	5,258	5,5
15	0,48039	1,3193	2,3814	3,2817	3,9657	4,5306
30	0,028625	0,19932	0,55426	1,0051	1,5436	2,182
45	0,0030236	0,027578	0,13682	0,35946	0,66752	1,0217
60	0,0027364	0,0034005	0,026306	0,1174	0,29568	0,54167
75	0,0027283	0,0027669	0,0079397	0,045513	0,14194	0,30235
90	0,0027309	0,0027333	0,0032843	0,016477	0,067511	0,17301

b. Penentuan jumlah individu pada populasi algoritma genetika

Pengujian ini bertujuan untuk menentukan jumlah individu terbaik pada penggabungan GRNN dan AG. Langkah-langkah pengujian adalah sebagai berikut :

- Menentukan nilai penyebaran sama dengan 6, jumlah simpul lapisan masukan sebanyak 90, probabilitas mutasi 0.05, dan jumlah generasi 10
- Melatih GRNN dengan komponen data musiman hasil dekomposisi dengan beberapa nilai populasi yaitu 10, 30, 50, 70, dan 90.

- Mengukur RMSE tiap model yang terbentuk dan memilih model dengan RMSE terkecil.

Masing-masing nilai uji populasi dilakukan sebanyak lima kali. Hasil pengujian dapat dilihat pada Tabel 4.2. Nilai RMSE terbaik untuk tiap jumlah populasi ditandai oleh warna abu-abu pada sel di tabel tersebut. Berdasarkan tabel 4.2 dapat disimpulkan bahwa untuk populasi sebanyak 10 individu diperoleh RMSE terkecil sebesar 0,0026873, untuk populasi 30 individu nilai RMSE terkecil adalah 0,0025783, untuk populasi 50 individu diperoleh nilai RMSE terbaik adalah 0,0026451, untuk populasi 70 individu nilai RMSE terbaiknya adalah 0,002604, dan untuk populasi 90 individu diperoleh nilai RMSE terbaik sebesar 0,0025969.

Tabel 4.2 Hasil pengujian penentuan jumlah populasi untuk komponen data musiman

Uji Coba ke	RMSE Pop = 10	RMSE Pop = 30	RMSE Pop = 50	RMSE Pop = 70	RMSE Pop = 90
1	0,0026874	0,0025938	0,0026505	0,002604	0,0026323
2	0,0027702	0,0026935	0,002701	0,0026197	0,0026341
3	0,0027123	0,0025783	0,0026882	0,0026593	0,0025969
4	0,0027376	0,0026852	0,0026451	0,0026283	0,002667
5	0,0026873	0,0026729	0,0026851	0,0026499	0,0026049

c. Penentuan nilai probabilitas mutasi

Uji coba ini dilakukan bertujuan untuk menentukan nilai probabilitas mutasi yang paling sesuai pada algoritma genetika. Adapun langkah-langkah uji coba adalah sebagai berikut :

- Menggunakan jumlah masukan 90, nilai penyebaran 6, jumlah populasi 30, dan jumlah generasi 10.
- Melatih dan menguji model berdasarkan parameter-parameter di atas dengan beberapa nilai probabilitas mutasi yaitu 0.1, 0.2, 0.3, 0.4, dan 0.5.

- Mengukur RMSE dari kedua model di atas dan menentukan yang terkecil

Untuk tiap nilai probabilitas mutasi yang diujikan, dilakukan pelatihan model sebanyak lima kali. Hasil pengujian dapat dilihat pada Tabel 4.3. Nilai RMSE terbaik untuk tiap probabilitas mutasi yang diujikan ditunjukkan oleh warna abu-abu pada sel yang bersesuaian. Dari tabel 4.3 terlihat bahwa nilai probabilitas mutasi yang memberikan RMSE terkecil untuk probabilitas mutasi 0.1 adalah 0,0026534, untuk probabilitas mutasi 0.2 adalah 0,0026166, probabilitas mutasi 0.3 sebesar 0,0025653, probabilitas mutasi 0.4 sebesar 0,0026092, dan probabilitas mutasi 0.5 sebesar 0,0025685.

Tabel 4.3 Hasil uji coba penentuan probabilitas mutasi untuk komponen data musiman

Uji coba ke	RMSE pm = 0.1	RMSE pm = 0.2	RMSE pm = 0.3	RMSE pm = 0.4	RMSE pm = 0.5
1	0,0026583	0,0026295	0,0026958	0,0026092	0,0027075
2	0,0026912	0,0026166	0,0026709	0,0026549	0,0025685
3	0,0026534	0,0026397	0,0026625	0,0026432	0,002754
4	0,0026778	0,0026986	0,0025653	0,0026597	0,0026766
5	0,0027064	0,0026397	0,0026116	0,0027237	0,0026974

d. Penentuan jumlah generasi

Tujuan dari uji coba ini adalah menentukan jumlah generasi yang memberikan akurasi terbaik yaitu model dengan nilai RMSE terkecil. Model yang digunakan adalah kombinasi antara GRNN dengan AG. Akurasi dari model ini juga akan dibandingkan dengan akurasi dari model GRNN tanpa AG yang berdasarkan perhitungan memiliki nilai RMSE sebesar 0,0027309. Langkah-langkah uji coba adalah sebagai berikut :

- Menentukan jumlah masukan 90, nilai penyebaran 6, jumlah populasi 30, dan probabilitas mutasi 0.5.

- Menghitung akurasi model GRNN dengan AG berdasarkan beberapa nilai generasi yaitu 15, 30, 45, 60, 75, dan 90 kemudian menentukan model dengan RMSE terkecil.
- Membandingkan akurasi dari model di atas dengan model GRNN tanpa AG.

Untuk tiap nilai generasi yang diujikan, dilakukan lima kali pengujian. Hasil uji coba dapat dilihat pada Tabel 4.4. Nilai RMSE terkecil untuk tiap nilai uji ditunjukkan oleh sel yang berwarna abu-abu. Dari tabel tersebut, terlihat bahwa untuk 15 generasi nilai RMSE terbaik adalah 0,002648, untuk 30 generasi diperoleh RMSE terkecil adalah 0,0026183, untuk 45 generasi RMSE terkecil adalah 0,002553, untuk 60 generasi diperoleh RMSE terbaik sebesar 0,002553, untuk 75 generasi 75 nilai RMSE terkecil adalah 0,0024612, dan untuk 90 generasi nilai RMSE terkecilnya adalah 0,0025015.

4.4.1.2. Penentuan Parameter Model GRNN untuk Komponen Residual

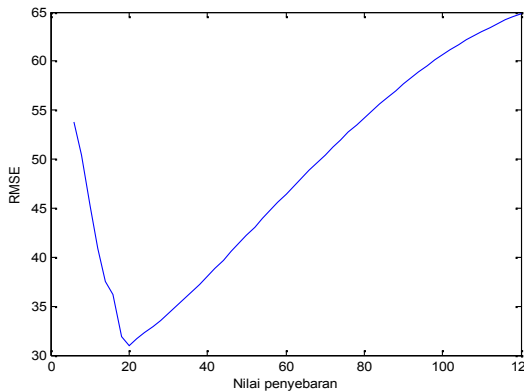
Komponen residual hasil dekomposisi memiliki karakteri data yang flukutasinya tidak beraturan. GRNN akan digunakan untuk memodelkan komponen ini. Parameter yang sama dengan komponen data musiman akan ditentukan melalui pengujian ini yaitu jumlah masukan jaringan, nilai penyebaran, jumlah populasi, nilai probabilitas mutasi, dan jumlah generasi.

a. Penentuan jumlah masukan jaringan

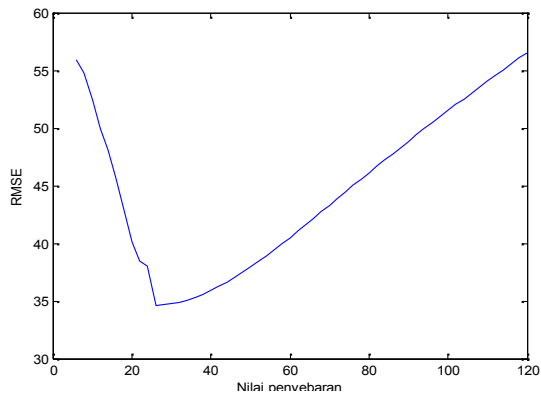
Tujuan dari uji coba ini adalah mengetahui jumlah masukan jaringan yang paling sesuai bagi komponen residual. Uji coba ini dilakukan terhadap GRNN tanpa kombinasi dengan algoritma genetika dengan asumsi bahwa hasil yang sama dapat diterapkan terhadap model GRNN dengan algoritma genetika. Adapun langkah-langkah uji coba adalah sebagai berikut :

- Melatih GRNN dengan beberapa jumlah masukan yaitu 1, 2, 7, 15, dan 30.

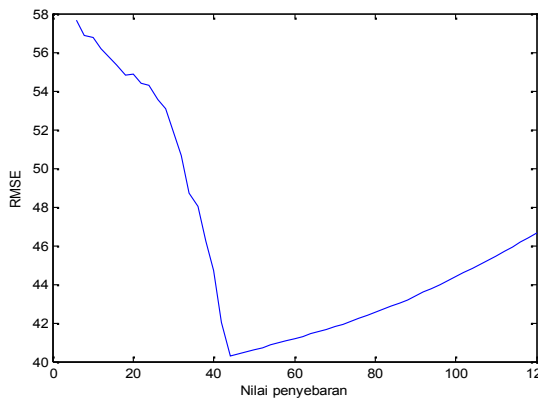
- Memberikan beberapa nilai penyebaran untuk tiap nilai simpul masukan uji coba yaitu mulai dari 6, 8, 10, 12, ..., hingga 120.
- Mengukur RMSE tiap percobaan dan mencari nilai RMSE terkecil



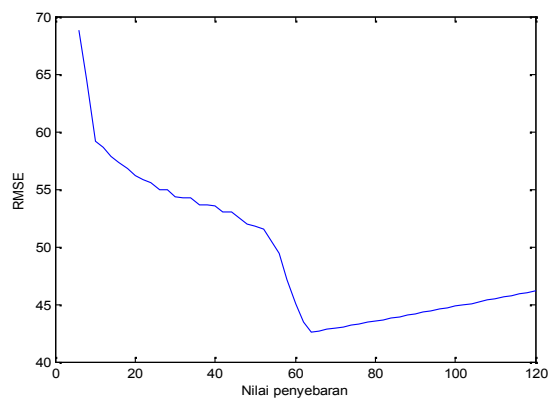
(a) Jumlah simpul masukan = 1



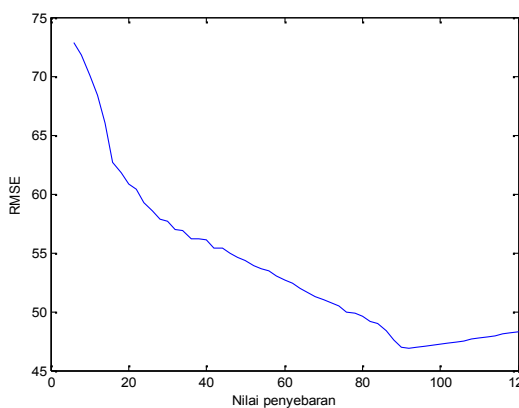
(b) Jumlah simpul masukan = 2



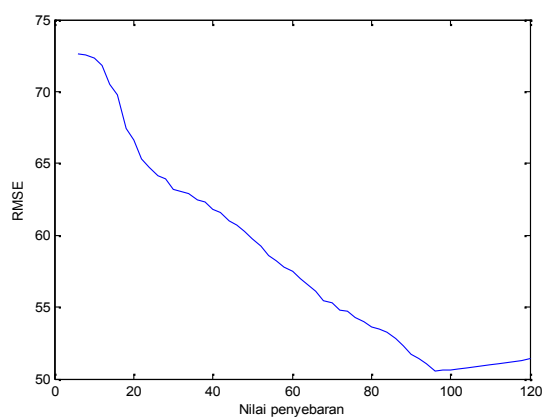
(c) Jumlah simpul masukan = 7



(d) Jumlah simpul masukan = 15



(e) Jumlah simpul masukan = 30



(f) Jumlah simpul masukan = 45

Gambar 4.1 Hasil uji coba penentuan nilai penyebaran untuk pemodelan komponen residual

Adapun hasil uji coba untuk tiap jumlah simpul masukan dapat dilihat pada Gambar 4.1 dan Tabel 4.5. Gambar 4.2 menggambarkan grafik perubahan nilai RMSE untuk nilai penyebaran sesuai kisaran nilai uji coba. Tabel 4.5 menunjukkan nilai RMSE terkecil untuk tiap jumlah simpul masukan dan nilai penyebaran yang bersesuaian. Berdasarkan tabel tersebut, untuk jumlah simpul masukan satu diperoleh RMSE terbaik sebesar 31,0141 pada nilai penyebaran 20. Untuk simpul masukan berjumlah dua diperoleh RMSE terkecil adalah 34,5988 pada nilai penyebaran 26. Untuk simpul masukan berjumlah tujuh diperoleh RMSE terkecil adalah 40,2821 pada nilai penyebaran 44. Untuk simpul masukan berjumlah 15 diperoleh RMSE terkecil adalah 42,5635 pada nilai penyebaran 64. Untuk simpul masukan berjumlah 30 diperoleh RMSE terkecil adalah 46,8841 pada nilai penyebaran 92. Untuk simpul masukan berjumlah 45 diperoleh RMSE terkecil adalah 50,5522 pada nilai penyebaran 96.

Tabel 4.4 Hasil uji coba penentuan jumlah generasi untuk komponen data musiman

Uji coba ke	RMSE gen = 15	RMSE gen = 30	RMSE gen = 45	RMSE gen = 60	RMSE gen = 75	RMSE gen = 90
1	0,0026963	0,0026677	0,002614	0,0026469	0,002577	0,0025627
2	0,0026842	0,0026323	0,002553	0,0025496	0,002577	0,0025015
3	0,002648	0,0026474	0,002585	0,0025837	0,0025546	0,0025627
4	0,0027019	0,0026183	0,002585	0,0026485	0,002490	0,002561
5	0,0026642	0,0026275	0,0026349	0,0026245	0,0024612	0,002456

Tabel 4.5 Hasil uji coba penentuan jumlah masukan GRNN untuk komponen residual

N simpul masukan	Nilai penyebaran	RMSE terkecil
1	20	31,0141
2	26	34,5988
7	44	40,2821
15	64	42,5635
30	92	46,8841
45	96	50,5522

b. Penentuan populasi algoritma genetika

Uji coba ini bertujuan untuk menentukan jumlah individu pada populasi algoritma genetika yang dikombinasikan dengan GRNN. Adapun langkah-langkah pengujian adalah sebagai berikut :

- Memodelkan komponen residual menggunakan GRNN yang dikombinasikan dengan algoritma genetika dimana jumlah masukan 1, jumlah keluaran 1, jumlah generasi 10, probabilitas mutasi 0.05 dan nilai penyebaran 18.5
- Menguji model dengan beberapa jumlah populasi yaitu 10, 30, 50, 70, dan 90.
- Mengukur RMSE tiap model uji dan menentukan nilai terkecil.

Untuk tiap jumlah populasi yang diujikan, dilakukan uji coba sebanyak lima kali. Tabel 4.6 menunjukkan hasil uji coba penentuan populasi. Nilai RMSE terbaik untuk tiap nilai uji ditandai dengan sel berwarna abu-abu. Dari Tabel 4.6 dapat dilihat bahwa untuk populasi sebanyak 10 individu nilai RMSE terkecil yang diperoleh sebesar 28,795, untuk jumlah populasi 30 nilai RMSE terbaik adalah 28,7367, untuk populasi 50 individu nilai RMSE terkecil adalah 28,7635, untuk populasi 70 individu diperoleh RMSE terbaik sebesar 28,6827, dan untuk populasi 90 individu nilai RMSE terbaik adalah 28,6727.

Tabel 4.6 Hasil uji coba penentuan populasi untuk komponen residual

Uji Coba ke	RMSE Pop 10	RMSE Pop 30	RMSE Pop 50	RMSE Pop 70	RMSE Pop 90
1	28,829	28,9004	28,7652	28,7197	28,7942
2	28,795	28,7367	28,8187	28,8123	28,7543
3	28,8804	28,912	28,8089	28,6827	28,7298
4	28,9506	28,852	28,776	28,8207	28,7773
5	28,8134	28,7695	28,7635	28,6989	28,6727

c. Penentuan nilai probabilitas mutasi

Pengujian ini bertujuan untuk menentukan nilai probabilitas mutasi yang paling baik untuk memodelkan komponen residual menggunakan GRNN dan algoritma genetika. Langkah-langkah pengujian adalah sebagai berikut :

- Memodelkan komponen residual menggunakan GRNN dan algoritma genetika dimana jumlah masukan dan keluaran GRNN adalah 1, jumlah generasi 10, probabilitas mutasi 0.05, nilai penyebaran 18.5, dan jumlah populasi 70.
- Menguji model dengan beberapa nilai probabilitas mutasi yaitu 0.1, 0.2, 0.3, 0.4 dan 0.5.
- Mengukur RMSE tiap pengujian dan menentukan nilai terkecil

Untuk tiap nilai probabilitas mutasi yang diujikan, dilakukan uji coba sebanyak lima kali. Hasil uji coba ditunjukkan pada Tabel 4.7. Dari tabel tersebut dapat dilihat bahwa untuk probabilitas mutasi 0.1 nilai RMSE terkecil adalah 28,6864, untuk probabilitas mutasi 0.2 diperoleh RMSE terbaik sebesar 28,7392, untuk probabilitas mutasi 0.3 diperoleh RMSE terbaik sebesar 28,6556, untuk probabilitas mutasi 0.4 nilai RMSE terkecil adalah 28,6763 dan untuk probabilitas mutasi 0.5 diperoleh RMSE terbaik sebesar 28,6972.

Tabel 4.7 Hasil uji coba penentuan probabilitas mutasi untuk komponen residual

Uji coba ke	RMSE pm = 0.1	RMSE pm = 0.2	RMSE pm = 0.3	RMSE pm = 0.4	RMSE pm = 0.5
1	28,7456	28,8768	28,706	28,867	28,6972
2	28,7094	28,799	28,6556	28,7664	28,7676
3	28,7282	28,7392	28,7487	28,7726	28,7416
4	28,6864	28,7789	28,7184	28,7008	28,7392
5	28,736	28,828	28,7247	28,6763	28,7275

d. Penentuan jumlah generasi

Uji coba ini bertujuan untuk menentukan jumlah generasi yang paling sesuai untuk pemodelan komponen residual menggunakan GRNN dan algoritma genetika. Langkah-langkah uji coba adalah sebagai berikut :

- Menentukan jumlah masukan 1, nilai penyebaran 18.5, jumlah populasi 70, dan probabilitas mutasi 0.05.
- Menghitung akurasi model GRNN dengan AG berdasarkan beberapa nilai generasi yaitu 10, 30, 50, 70 dan 90, kemudian menentukan model dengan RMSE terkecil.
- Membandingkan akurasi dari model di atas dengan model GRNN tanpa AG.

Untuk tiap nilai uji coba, dilakukan pengujian sebanyak lima kali. Hasil uji coba dapat dilihat pada Tabel 4.8. Nilai RMSE terkecil untuk tiap nilai uji generasi ditunjukkan oleh warna abu-abu pada sel. Untuk 10 generasi nilai RMSE terkecil adalah 28,6531, untuk 30 generasi nilai RMSE terbaik adalah 28,5864, untuk 50 generasi nilai RMSE terbaik adalah 28,5662, untuk 70 generasi diperoleh nilai RMSE terkecil 28,5546, dan untuk 90 generasi diperoleh RMSE terbaik sebesar 28,5121.

Tabel 4.8 Hasil uji coba penentuan generasi untuk komponen residual

Uji coba ke	RMSE gen = 10	RMSE gen = 30	RMSE gen = 50	RMSE gen = 70	RMSE gen = 90
1	28,6531	28,7187	28,612	28,6466	28,601
2	28,7985	28,7512	28,6445	28,5546	28,6174
3	28,6865	28,671	28,5712	28,6509	28,6284
4	28,8092	28,5864	28,6308	28,5651	28,5121
5	28,8745	28,7139	28,5662	28,6416	28,584

4.4.2. Pelaksanaan Uji Coba Beberapa Metode Peramalan

Uji coba dilakukan dengan tujuan mengetahui kinerja model peramalan yang diusulkan dibandingkan beberapa metode peramalan lainnya. Kinerja akan diukur berdasarkan nilai RMSE masing-masing model peramalan. Peramalan akan diterapkan terhadap data emas harian mulai tanggal 1 Januari 2003 hingga 31 Desember 2012 sebanyak 2.609 data. Perbandingan dilakukan antara metode STL, theta, dan GRNN berbasis algoritma genetika dengan metode STL dan GRNN-theta serta metode ARIMA. Metode GRNN dan algoritma genetika menggunakan hasil pengujian pada sub-bab 4.4.1. Peramalan menggunakan metode theta diimplementasikan menggunakan fungsi `thetaf()` pada perangkat lunak R. Adapun langkah-langkah uji coba adalah sebagai berikut :

- Peramalan menggunakan STL, GRNN, dan theta
- Peramalan menggunakan STL, GRNN berbasis algoritma genetika dan theta
- Peramalan menggunakan ARIMA

Langkah pertama adalah menguji hasil peramalan menggunakan STL-GRNN-GA-Theta. Metode GRNN-GA digunakan untuk meramalkan komponen data musiman dan residual, sedangkan komponen data tren diramalkan menggunakan metode theta. Untuk menemukan model GRNN-GA yang terbaik, dilakukan iterasi sebanyak lima kali untuk masing-masing komponen data musiman dan residual. Untuk komponen data musiman, parameter GRNN yang diatur adalah jumlah simpul masukan sebanyak 75 yang menunjukkan lebar jendela observasi dan nilai penyebaran sebesar enam. Sedangkan untuk parameter algoritma genetika yang digunakan untuk peramalan komponen data musiman menggunakan GRNN yaitu jumlah populasi 30, probabilitas mutasi 0,4 dan jumlah generasi 90. Untuk komponen data residual, parameter GRNN yang diatur adalah jumlah simpul masukan sebanyak satu dan nilai penyebaran sebesar 20. Sedangkan untuk algoritma genetika, parameter yang digunakan yaitu jumlah populasi 90, probabilitas mutasi 0,4 dan jumlah generasi 90. Hasil uji coba model peramalan yang diusulkan terhadap komponen data musiman dapat dilihat pada

Tabel 4.9, sedangkan hasil untuk komponen data residual dapat dilihat pada Tabel 4.10. Peramalan kedua komponen selanjutnya digabungkan dengan hasil peramalan komponen data tren yang menggunakan metode theta. Penggabungan dilakukan menggunakan metode JST propagasi balik.

Langkah kedua adalah melakukan peramalan menggunakan STL-GRNN-Theta. Peramalan diterapkan terhadap komponen data musiman dan residual dengan menggunakan metode GRNN dan komponen data tren menggunakan metode theta. Parameter untuk GRNN untuk komponen data residual adalah jumlah simpul masukan sebanyak satu dan nilai parameter penyebaran 20. Untuk komponen data musiman, jumlah simpul masukan adalah 75 dan nilai penyebaran enam. Hasil peramalan komponen data musiman, residual, dan tren digabungkan menggunakan metode JST propagasi balik.

Peramalan menggunakan ARIMA dilakukan dengan menerapkan pemilihan model secara otomatis menggunakan algoritma pemilihan *stepwise* yang diimplementasikan dalam perangkat lunak R yaitu fungsi `auto.arima()`. Untuk pemodelan, digunakan 70% data yaitu sebanyak 1.826 observasi untuk meramalkan 783 hari ke depan. Parameter Arima terbaik yang diperoleh adalah $p=0$, $d=1$, dan $q = 0$.

Tabel 4.9 Uji coba STL-GRNN-AG untuk komponen data musiman

N simpul masukan	RMSE STL-GRNN-AG	RMSE STL-GRNN
75	0,0025139	0,0027174
	0,0025585	
	0,0024631	
	0,0024783	
	0,0024783	

Tabel 4.10 Uji coba STL-GRNN-AG untuk komponen data residual

N masukan	Nilai Penyebaran	RMSE STL-GRNN-AG	RMSE STL-GRNN
1	20	28,4187	31,0141
		28,4951	
		28,3836	
		28,3183	
		28,5354	

Hasil pengujian peramalan ditunjukkan pada Tabel 4.11. Nilai terbaik untuk tiap nilai uji ditunjukkan oleh sel pada kolom ketiga yang berwarna abu-abu. Selanjutnya, hasil terbaik untuk komponen data musiman dan residual digabungkan dengan prediksi oleh metode theta. Hasil uji coba dapat dilihat pada Tabel 4.9. Pada tabel tersebut dapat diketahui bahwa akurasi terbaik diperoleh oleh model STL, theta, dan GRNN berbasis algoritma genetika dengan RMSE sebesar 121,8702. Berdasarkan hasil pengujian pada Tabel 4.9 dan 4.10, dilakukan peramalan harga emas dengan menggabungkan peramalan komponen data musiman, residual, dan tren.

Tabel 4.11 Hasil uji coba beberapa metode peramalan

Metode Peramalan	RMSE	N data latih	Waktu proses (s)
STL- THETA - GRNN – AG	51,9161	Seasonal : 907 Residual : 911	0,88921
STL – THETA – GRNN	53,1904	Seasonal : 1,751 Residual : 1,825	1,794
ARIMA (0,1,0)	195,3653	1,826	0,21

4. 5. Analisis Hasil Uji Coba

Sub-bab ini menjelaskan analisis hasil uji coba yang telah dilakukan pada sub-bab 4.4. Analisis ini dilakukan terhadap uji coba penentuan parameter model peramalan dan uji coba perbandingan beberapa metode peramalan.

4.5.1. Analisis Uji Coba Penentuan Parameter Model Peramalan

Dalam sub-bab berikut akan dijelaskan analisis uji coba penentuan parameter model peramalan untuk komponen data musiman dan residual.

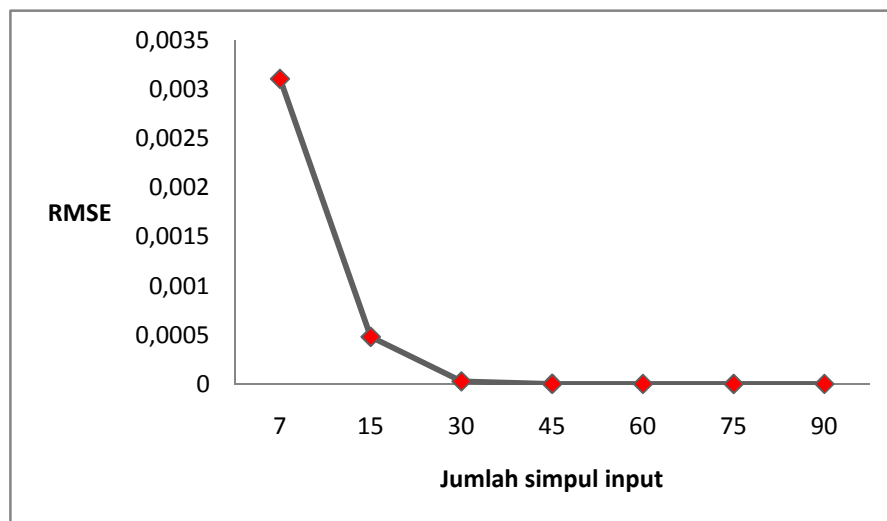
4.5.1.1. Analisis Uji Coba Komponen Data Musiman

Uji coba terhadap komponen data musiman untuk menemukan parameter model GRNN dan algoritma genetika yang terbaik dilakukan dengan mencari nilai parameter jumlah simpul lapisan masukan, nilai penyebaran, jumlah individu

dalam populasi, nilai probabilitas mutasi, dan jumlah generasi. Adapun hasil uji coba adalah sebagai berikut:

a. Penentuan parameter jumlah simpul masukan

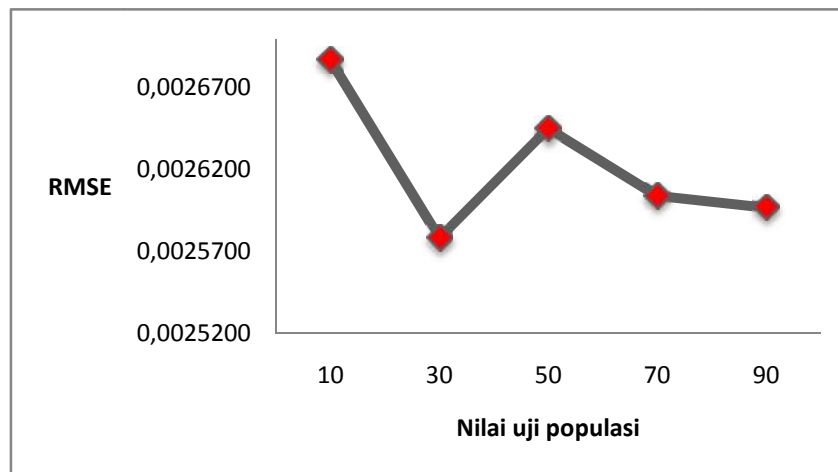
Hasil pengujian penentuan jumlah simpul masukan dan nilai penyebaran pada pemodelan komponen data musiman menggunakan GRNN dapat dilihat pada Tabel 4.1, Untuk seluruh jumlah simpul masukan yang diujikan, nilai penyebaran 6 memberikan RMSE terkecil yaitu 3,1046 untuk jumlah simpul 7, RMSE 0,48039 untuk jumlah simpul 15, RMSE 0,028625 untuk jumlah simpul 30, RMSE 0,0030236 untuk jumlah simpul 45, RMSE 0,0027364 untuk jumlah simpul 60, RMSE 0,0027283 untuk jumlah simpul 75, dan RMSE 0,0027309 untuk jumlah simpul 90, Oleh karena itu, jumlah simpul masukan dengan RMSE terkecil adalah 75 dengan nilai penyebaran 6 seperti ditunjukkan dari gambar 4.2.



Gambar 4.2 Hasil uji penentuan jumlah simpul masukan GRNN untuk komponen data musiman

b. Penentuan parameter jumlah individu populasi

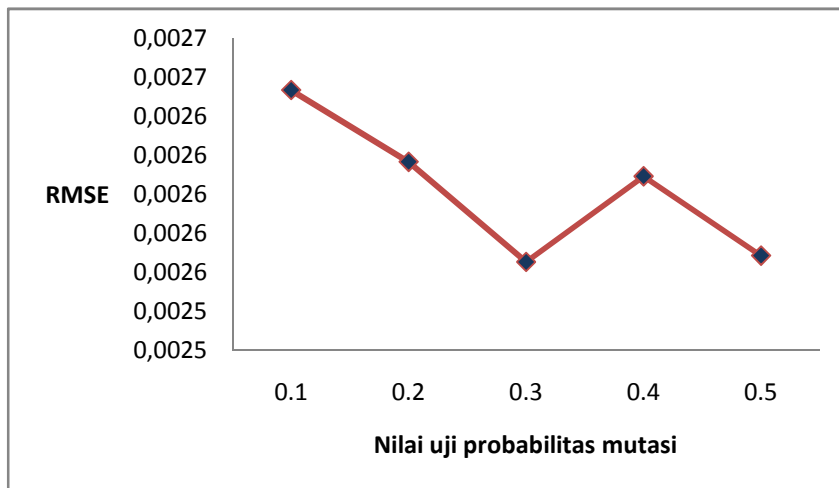
Hasil pengujian jumlah individu dalam populasi menggunakan GRNN dan algoritma genetika ditunjukkan oleh Tabel 4.2. Berdasarkan Tabel 4.2 dapat disimpulkan bahwa untuk populasi sebanyak 10 individu diperoleh RMSE terkecil sebesar 0,0026873, untuk populasi 30 individu nilai RMSE terkecil adalah 0,0025783, untuk populasi 50 individu diperoleh nilai RMSE terbaik adalah 0,0026451, untuk populasi 70 individu nilai RMSE terbaiknya adalah 0,002604, dan untuk populasi 90 individu diperoleh nilai RMSE terbaik sebesar 0,0025969, Berdasarkan grafik pada Gambar 4.3, populasi terbaik dengan RMSE terkecil adalah sebanyak 30.



Gambar 4.3 Hasil uji coba penentuan populasi GRNN-AG untuk komponen data musiman

c. Penentuan parameter nilai probabilitas mutasi

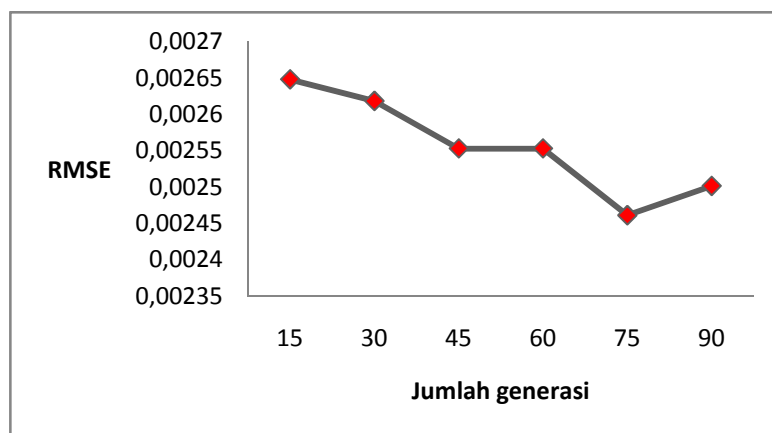
Hasil uji coba probabilitas mutasi ditunjukkan pada Tabel 4.3, Dari Tabel 4.3 terlihat bahwa nilai probabilitas mutasi yang memberikan RMSE terkecil untuk probabilitas mutasi 0,1 adalah 0,0026534, untuk probabilitas mutasi 0,2 adalah 0,0026166, probabilitas mutasi 0,3 sebesar 0,0025653, probabilitas mutasi 0,4 sebesar 0,0026092, dan probabilitas mutasi 0,5 sebesar 0,0025685, Berdasarkan grafik pada Gambar 4.4, disimpulkan bahwa probabilitas mutasi terbaik adalah 0,3.



Gambar 4.4 Hasil uji coba probabilitas mutasi pada pemodelan komponen data musiman

d. Penentuan parameter jumlah generasi

Hasil uji coba dapat dilihat pada Tabel 4.4, Dari tabel tersebut, terlihat bahwa untuk 15 generasi nilai RMSE terbaik adalah 0,002648, untuk 30 generasi diperoleh RMSE terkecil adalah 0,0026183, untuk 45 generasi RMSE terkecil adalah 0,002553, untuk 60 generasi diperoleh RMSE terbaik sebesar 0,002553, untuk 75 generasi 75 nilai RMSE terkecil adalah 0,0024612, dan untuk 90 generasi nilai RMSE terkecilnya adalah 0,0025015. Nilai-nilai terbaik untuk setiap uji coba ditunjukkan pada Gambar 4.5, Oleh karena itu, akurasi terbaik diperoleh pada jumlah generasi 75.



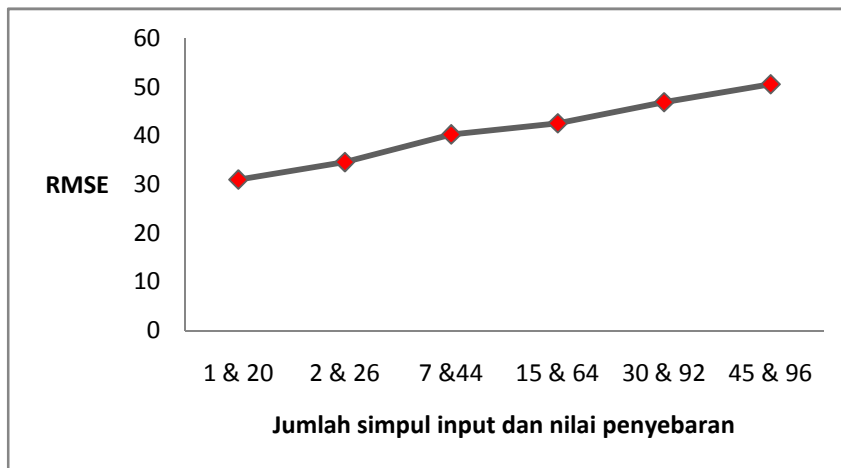
Gambar 4.5 Hasil uji coba generasi untuk komponen data musiman

4.5.1.2. Analisis Uji Coba Komponen Data Residual

Uji coba terhadap komponen data residual untuk menemukan parameter model GRNN dan algoritma genetika yang terbaik dilakukan dengan mencari nilai parameter jumlah simpul lapisan masukan, nilai penyebaran, jumlah individu dalam populasi, nilai probabilitas mutasi, dan jumlah generasi. Hasil uji coba dapat dilihat pada sub bab 4.4.1.2. Nilai terbaik untuk tiap pengujian juga dapat dilihat di tiap elemen pengujian. Sub bab ini menjelaskan mengenai analisis dari tiap hasil pengujian.

a. Penentuan parameter jumlah simpul masukan

Hasil uji coba penentuan jumlah simpul masukan dan nilai penyebaran dapat dilihat pada Tabel 4.5. Berdasarkan tabel tersebut, untuk jumlah simpul masukan 1 diperoleh RMSE terbaik sebesar 31,0141 pada nilai penyebaran 20, Untuk simpul masukan berjumlah 2 diperoleh RMSE terkecil adalah 34,5988 pada nilai penyebaran 26. Untuk simpul masukan berjumlah 7 diperoleh RMSE terkecil adalah 40,2821 pada nilai penyebaran 44. Untuk simpul masukan berjumlah 15 diperoleh RMSE terkecil adalah 42,5635 pada nilai penyebaran 64. Untuk simpul masukan berjumlah 30 diperoleh RMSE terkecil adalah 46,8841 pada nilai penyebaran 92. Untuk simpul masukan berjumlah 45 diperoleh RMSE terkecil adalah 50,5522 pada nilai penyebaran 96. Oleh karena itu, nilai RMSE terbaik diperoleh pada jumlah simpul 1 dan nilai penyebaran 20 seperti ditunjukkan pada Gambar 4.6. Dari hasil uji coba ini juga terlihat bahwa untuk pemodelan komponen data residual menggunakan GRNN, semakin besar jumlah simpul masukan maka nilai RMSE terbaik diperoleh pada nilai penyebaran yang juga semakin besar.



Gambar 4.6 Hasil uji coba jumlah simpul masukan dan nilai penyebaran pemodelan komponen data residual menggunakan GRNN

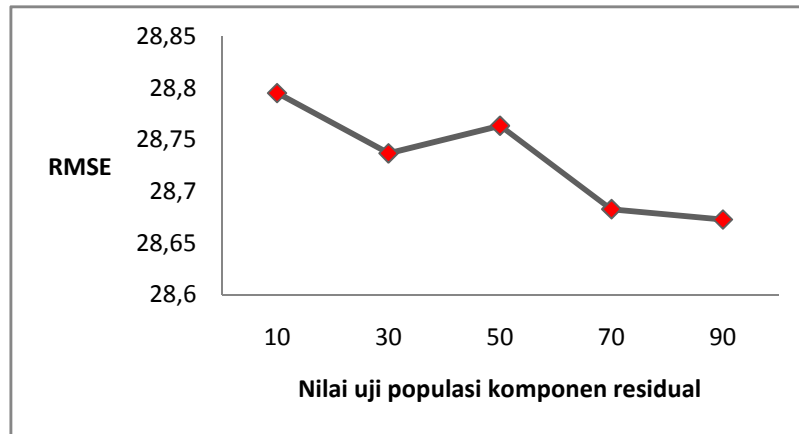
b. Penentuan parameter jumlah individu dalam populasi

Hasil pengujian jumlah individu dalam populasi menggunakan GRNN dan algoritma genetika ditunjukkan oleh Tabel 4.6. Dari Tabel 4.6 dapat dilihat bahwa untuk populasi sebanyak 10 individu nilai RMSE terkecil yang diperoleh sebesar 28,795, untuk jumlah populasi 30 nilai RMSE terbaik adalah 28,7367, untuk populasi 50 individu nilai RMSE terkecil adalah 28,7635, untuk populasi 70 individu diperoleh RMSE terbaik sebesar 28,6827, dan untuk populasi 90 individu nilai RMSE terbaik adalah 28,6727. Grafik perubahan RMSE berdasarkan nilai populasi yang diujikan ditunjukkan pada Gambar 4.7 dimana dapat dilihat bahwa nilai populasi terbaik adalah 90.

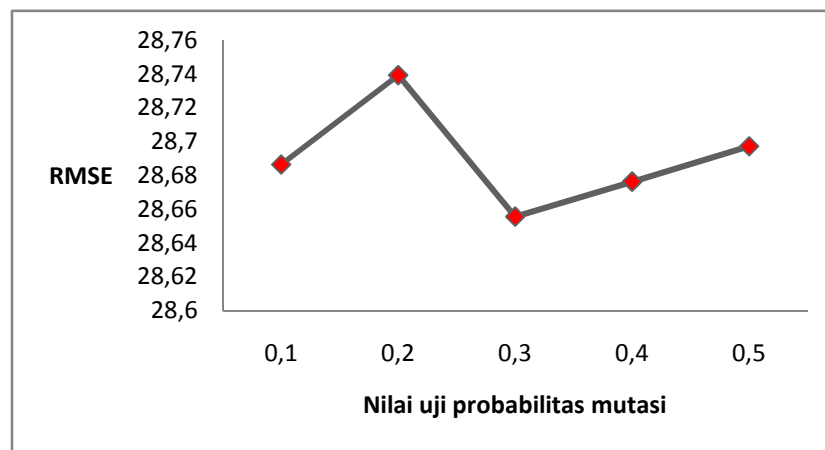
c. Penentuan parameter nilai probabilitas mutasi

Hasil uji coba ditunjukkan pada Tabel 4.7. Dari tabel tersebut dapat dilihat bahwa untuk probabilitas mutasi 0,1 nilai RMSE terkecil adalah 28,6864, untuk probabilitas mutasi 0,2 diperoleh RMSE terbaik sebesar 28,7392, untuk probabilitas mutasi 0,3 diperoleh RMSE terbaik sebesar 28,6556, untuk probabilitas mutasi 0,4 nilai RMSE terkecil adalah 28,6763 dan untuk probabilitas mutasi 0,5 diperoleh RMSE terbaik sebesar 28,6972.

Berdasarkan Gambar 4.8, probabilitas mutasi terbaik untuk komponen residual adalah 0,3.



Gambar 4.7 Hasil uji populasi GRNN-AG untuk komponen data residual

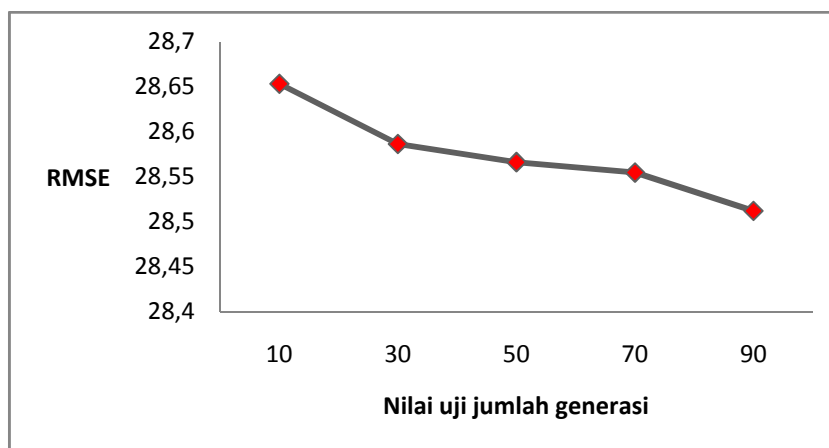


Gambar 4.8 Hasil uji probabilitas mutasi untuk komponen data residual

d. Penentuan parameter jumlah generasi

Hasil uji coba dapat dilihat pada Tabel 4.8. Untuk 10 generasi nilai RMSE terkecil adalah 28,6531, untuk 30 generasi nilai RMSE terbaik adalah

28,5864, untuk 50 generasi nilai RMSE terbaik adalah 28,5662, untuk 70 generasi diperoleh nilai RMSE terkecil 28,5546, dan untuk 90 generasi diperoleh RMSE terbaik sebesar 28,5121. Berdasarkan grafik pada Gambar 4.9, diperoleh jumlah generasi yang paling baik adalah 90.



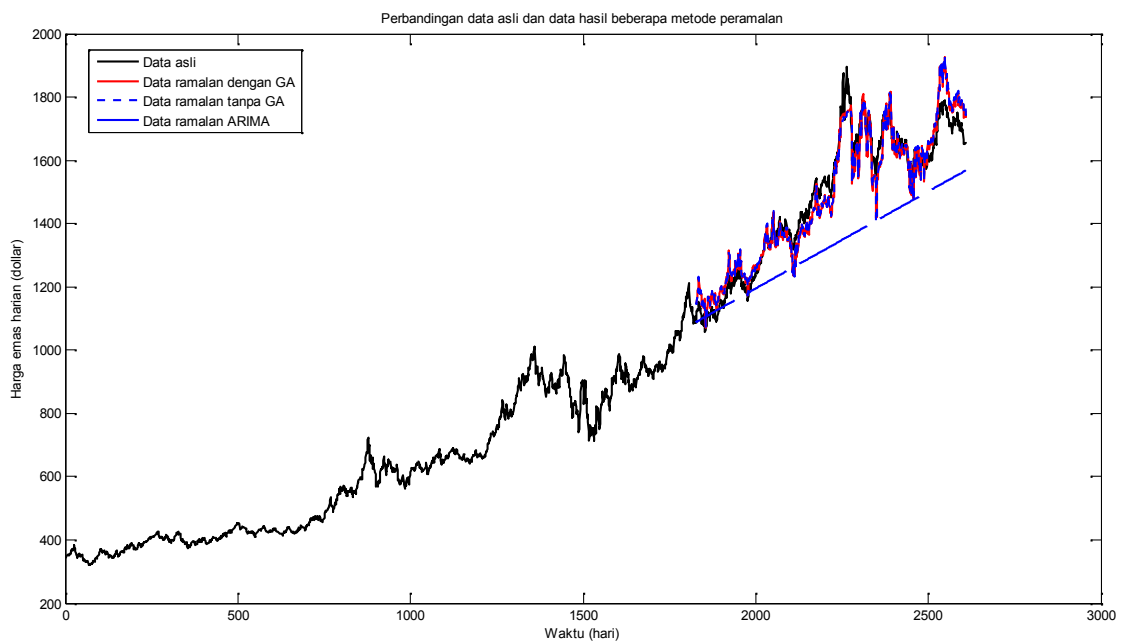
Gambar 4.9 Hasil uji coba generasi GRNN-AG untuk komponen data residual

4.5.2. Analisis Uji Coba Perbandingan Metode Peramalan

Berdasarkan uji coba pada sub-bab 4.4.2, diperoleh parameter-parameter dengan nilai terbaik untuk meramalkan komponen data musiman, tren, dan residual, Berdasarkan parameter-parameter ini, dilakukan peramalan harga emas sesuai dengan data uji yaitu 30% dari keseluruhan data. Metode peramalan menggunakan STL-GRNN-GA-Theta dibandingkan dengan peramalan menggunakan metode STL-GRNN-Theta dan metode ARIMA. Hasil peramalan ditunjukkan pada Tabel 4.10. Perbandingan ketiga metode peramalan tersebut dapat dilihat pada Gambar 4.10.

Berdasarkan Gambar 4.10, terlihat bahwa peramalan menggunakan metode STL-GRNN-GA-Theta (RMSE=51,9178) menghasilkan tingkat akurasi yang lebih baik dibandingkan dengan metode STL-GRNN-Theta (RMSE=53,2424) dan ARIMA (RMSE=195,3653). Hal ini menunjukkan bahwa dengan menggunakan menambahkan algoritma genetika pada metode peramalan

STL-GRNN-Theta dapat diperoleh tingkat akurasi yang setara atau lebih baik. Kelebihan lain dari penambahan algoritma genetika adalah diperoleh data latih yang lebih sedikit. Berdasarkan hasil pengujian pada Tabel 4.10, jika tanpa menggunakan algoritma genetika pembentukan model membutuhkan data latih sebanyak 1.751 *tuple* data untuk pemodelan komponen data musiman dan 1.825 *tuple* data untuk komponen data residual menggunakan GRNN, sedangkan dengan algoritma genetika hanya dibutuhkan 907 *tuple* data untuk pemodelan komponen data musiman dan 911 *tuple* data untuk pemodelan komponen data residual atau terjadi reduksi data latih lebih dari 50%. Pengurangan data latih ini juga mengakibatkan berkurangnya waktu yang dibutuhkan untuk pembentukan model dan simulasi yaitu dari 1,4976 detik menjadi 0,5148 detik atau berkurang hampir 66%.

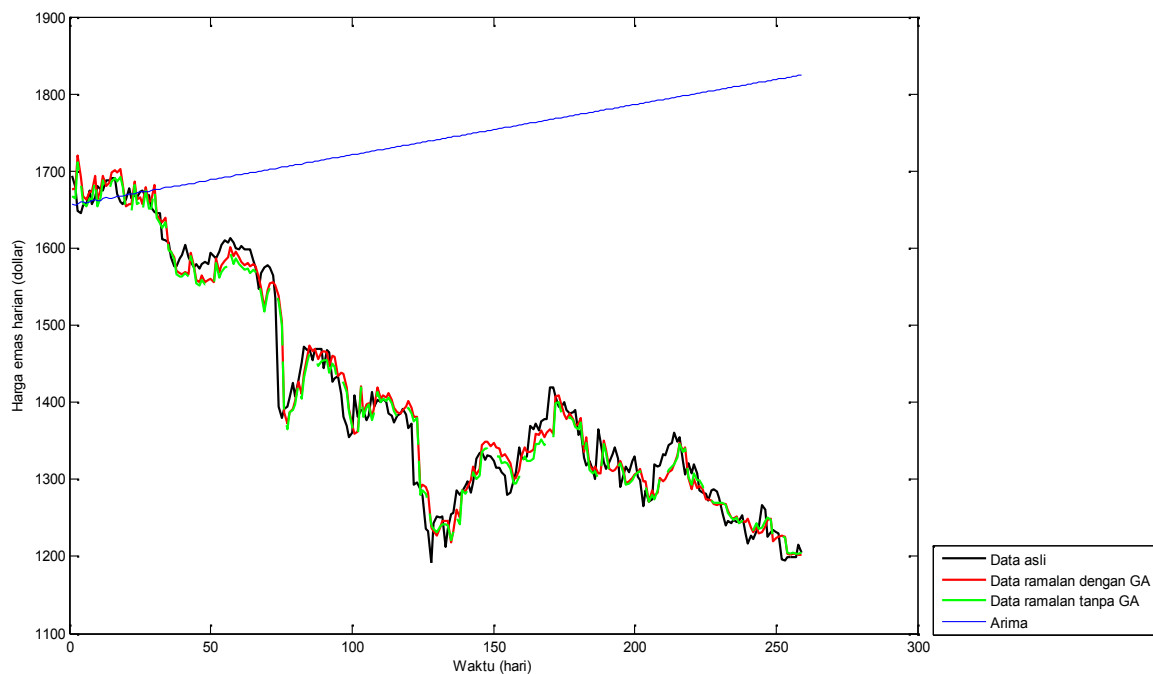


Gambar 4.10 Perbandingan hasil pengujian peramalan harga emas harian

Untuk mengetahui kinerja model, dilakukan pengujian menggunakan data harian harga emas mulai Januari sampai dengan bulan Desember 2013. Hasil validasi dapat dilihat pada Gambar 4.11. Nilai galat (RMSE) dan waktu yang

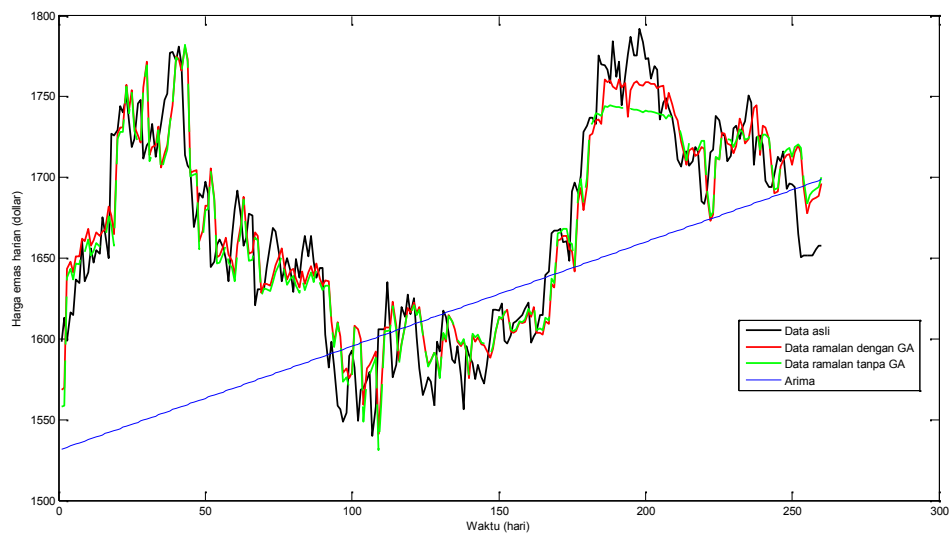
diperlukan dari ketiga metode yang diperbandingkan yaitu STL-GRNN-GA-Theta sebesar 26,4169 dan membutuhkan waktu 0,2496 detik, STL-GRNN-Theta sebesar 26,3546 dan membutuhkan waktu 0,6396 detik, dan ARIMA sebesar 382,2303. Terlihat bahwa akurasi peramalan menggunakan STL-GRNN-GA-Theta hampir setara dengan STL-GRNN-Theta dan keduanya lebih baik dibandingkan ARIMA. Meskipun RMSE peramalan menggunakan STL-GRNN-GA-Theta sedikit lebih besar dibandingkan peramalan menggunakan STL-GRNN-Theta, namun waktu yang dibutuhkan lebih kecil yaitu 61% lebih kecil.

Untuk mengetahui validitas model, digunakan perhitungan uji variasi amplitudo atau % *error variance*. Model disebut valid jika nilai *error variance* kurang dari 30%. Berdasarkan perhitungan, model peramalan STL-GRNN-GA-Theta memiliki nilai *error variance* sebesar 0,0058368 atau 0,5% sedangkan model STL-GRNN-Theta nilai memberikan *error variance* sebesar 0,019873 atau 1,9%. Dengan demikian, model peramalan STL-GRNN-GA-Theta adalah valid.



Gambar 4.11 Hasil validasi peramalan harga emas Jan – Des 2013

Untuk menguji konsistensi validasi model, dilakukan uji coba menggunakan data latih mulai 1 Januari 2003 sampai dengan 31 Desember 2011 dan data validasi mulai tanggal 1 Januari 2012 hingga 31 Desember 2012. Dari hasil uji coba, diperoleh RMSE peramalan untuk model STL-GRNN-GA-Theta sebesar 24,8043 (waktu = 0,3276 detik), model STL-GRNN-Theta sebesar 25,9564 (waktu = 1,4664 detik), dan model ARIMA sebesar 87,6749. Hasil uji coba dapat dilihat pada gambar 4.12. Berdasarkan uji variasi amplitudo, diperoleh nilai *error variance* sebesar 0,075272 atau 7,5% yang menunjukkan bahwa model STL-GRNN-GA-Theta adalah valid. Dengan demikian, dapat dikatakan bahwa model yang diusulkan pada penelitian ini konsisten untuk digunakan dalam peramalan.



Gambar 4.12 Hasil validasi menggunakan data Jan - Des 2012

BAB 5

PENUTUP

Bab ini menjelaskan mengenai kesimpulan yang diperoleh dari hasil uji coba dan analisis hasil uji coba. Selain itu, dalam bab ini juga diuraikan saran kemungkinan pengembangan lebih lanjut dari penelitian yang telah dilakukan.

5.1. Kesimpulan

Berdasarkan hasil uji coba dan analisis hasil dapat ditarik kesimpulan sebagai berikut:

- a. Penelitian ini telah berhasil mengintegrasikan algoritma genetika (GA) ke dalam metode prediksi harga emas berbasis kombinasi metode *Seasonal Trend based on LOESS* (STL) dan *Generalized Regression Neural Network* (GRNN) serta metode Theta (STL-GRNN-Theta) sehingga dihasilkan metode prediksi berbasis STL-GRNN-GA-Theta. Metode prediksi yang dihasilkan dalam penelitian ini mampu mereduksi kebutuhan data latih secara signifikan, tetapi dapat memberikan hasil prediksi yang relatif lebih akurat.
- b. Dari hasil uji coba penentuan parameter model peramalan STL-GRNN-GA-Theta dengan menggunakan data latih harga emas harian sebanyak 2,609 untuk periode mulai 1 Januari 2003 sampai dengan 31 Desember 2012 dapat disimpulkan bahwa:
 - i. Hasil terbaik peramalan untuk komponen data musiman diperoleh dengan menggunakan rentang jendela observasi sebesar 75, sedangkan hasil terbaik untuk komponen data residual diperoleh menggunakan rentang jendela observasi sebesar satu. Dalam hal ini, rentang jendela observasi menunjukkan jumlah hari sebelum hari peramalan.

- ii. Parameter yang paling berpengaruh pada akurasi model peramalan adalah nilai penyebaran (*spread*) data. Hal ini ditunjukkan oleh perubahan nilai akurasi yang signifikan seiring dengan perubahan nilai penyebaran data dibandingkan dengan perubahan akurasi yang dipengaruhi oleh perubahan nilai parameter GA. Dalam pengujian, nilai penyebaran terbaik untuk komponen data musiman dan residual berturut-turut diperoleh untuk nilai penyebaran sebesar enam dan 20.
- c. Dari hasil uji coba validasi terhadap model peramalan yang telah dilakukan (menggunakan data validasi mulai 1 Januari 2013 sampai dengan 31 Desember 2013) dapat disimpulkan bahwa:
- i. Jika dibandingkan dengan model peramalan STL-GRNN-Theta, model peramalan STL-GRNN-GA-Theta mampu mereduksi data latih sebesar 50%. Dampak dari reduksi data latih ini adalah pengurangan waktu peramalan sebesar 34%, tetapi akurasi hasil peramalan tetap dapat dipertahankan.
 - ii. Jika dibandingkan dengan hasil peramalan menggunakan model ARIMA, model peramalan STL-GRNN-GA-Theta mampu memberikan hasil akurasi 93% lebih baik.
 - iii. Berdasarkan pengujian terhadap variasi amplitudo, model peramalan STL-GRNN-GA-Theta memberikan nilai *error variance* sebesar 1,9%. Hal ini menunjukkan bahwa model peramalan yang dikembangkan dalam penelitian ini adalah valid.
- d. Dari hasil uji coba konsistensi diketahui bahwa model peramalan STL-GRNN-GA-Theta valid untuk meramalkan harga emas selama satu tahun. Hal ini ditunjukkan oleh nilai akurasi yang konsisten setara atau lebih baik dibandingkan dengan nilai akurasi yang dihasilkan oleh model tanpa GA. Selain itu, persentase reduksi data latih dan pengurangan waktu proses peramalan juga menunjukkan hasil yang konsisten.

5.2. Saran

Dalam penelitian ini, proses reduksi data latih menggunakan GA membutuhkan waktu yang cukup lama untuk memperoleh parameter GA yang baik dan stabil. Oleh karena untuk meramal data satu tahun ke depan diperlukan data latih beberapa data dari tahun-tahun sebelumnya, maka upaya untuk memperoleh model pembelajaran berkelanjutan tanpa harus mengulangi proses pelatihan secara menyeluruh merupakan salah satu alternatif pengembangan yang menantang.

Model peramalan STL-GRNN-GA-Theta yang dikembangkan dalam penelitian ini hanya mampu memberikan hasil akurasi yang sedikit lebih baik dibandingkan dengan model peramalan tanpa menggunakan GA, tetapi memerlukan waktu pelatihan yang cukup lama. Salah satu kemungkinan kelemahan ini dikarenakan penggunaan rentang waktu data latih yang digunakan terlalu panjang (sekitar 10 tahun). Oleh karena itu, salah satu pengembangan yang dapat dilakukan untuk mengurangi waktu peramalan dapat dilakukan dengan memilih rentang data yang paling sesuai agar proses reduksi data menggunakan algoritma genetika dapat dibuat lebih efisien tanpa harus menurunkan akurasi hasil peramalan. Alternatif pengembangan ini dimungkinkan atas pertimbangan bahwa peramalan runut waktu data terbaru dapat memberikan pengaruh yang signifikan dibandingkan data yang lebih lama.

DAFTAR PUSTAKA

- Assimakopoulos, V. dan Nikolopoulos, K. 2000. The theta model : a decomposition approach for forecasting. *International Journal of Forecasting* 16, pp. 521 – 530.
- Bowden, G.J., Maier, H.R., Dandy, G.C. 2002. Optimal division of data for neural network models in water resources applications. *Water Resources Research*, vol. 38, no. 2.
- Casey, K., Lim, A., dan Dozier, G. 2006. Evolving general regression neural networks for tsunami detection and response. 2006 IEEE Congress on Evolutionary Computations, Canada, 16-21 Juni 2006, pp 2451-2458.
- Chatfield, Chris. 2000. *Time Series Forecasting*. Chapman and Hall/CRC.
- Cigizoglu, H.K. 2005. Generalized regression neural network in monthly flow forecasting. *Civil Engineering and Environmental Systems*, vol. 22, no. 2, Juni 2005, pp. 71 – 84.
- Cleveland, R.B., Cleveland, W.S., McRae, J.E., dan Terpenning, I. 1990. STL : A seasonal-trend decomposition procedure based on Loess. *Journal of Official Statistics*, vol. 6, no. 1, 1990, 3-73.
- Crone, S.F. 2005. *Forecasting with Artificial Neural Networks*. Materi kuliah.
- Duzgun, R. 2010. Generalized regression neural network for inflation forecasting. *International Research Journal of Finance and Economics* 51, hal. 59 – 70.

- Fiszelew, A., Britos, P., Ochoa, A., Merlino, H., Fernandez, E., dan Garcia-Martinez, R. 2007. Finding optimal neural network architecture using genetic algorithm. *Research in Computing Science*, vol. 27, pp 15-24.
- Gen M and Cheng R (1997), “Genetic Algorithms and Engineering Design”, John Wiley and Sons, Inc., USA.
- Hannan, S.A., Manza, R.R., dan Ramteke, R.J. 2010. Generalized regression neural network and radial basis function for heart disease diagnosis. *International Journal of Computer Applications* 7, 7 – 13.
- Hansen, J.V., Nelson, R.D. 2003. Forecasting and recombining time series components by using neural network. *Journal of the Operational Research Society* (2003) 54, 307–317.
- Hussein, S.F.M, Shah, M.B.N., Jalal, M.R.A., dan Abdullah, S.S. 2011. Gold Price Prediction using Radial Basis Function Neural Network. *Proceeding of International Conference of Modelling, Simulation and Applied Optimization (ICMSAO)*, Kuala Lumpur, Malaysia, 19-21 April 2011, pp. 1-11.
- Ismail, Z., Yahya, A., dan Shabri, A. 2009. Forecasting gold prices using multiple linear regression method. *American Journal of Applied Sciences* vol. 6, issue 8, pp. 1509 – 1514.
- Kaastra, I., Boyd, M. 1996. Designing a neural network for forecasting financial and economic time series. *Neurocomputing* vol. 10, 215 – 236.
- Kamp, R.G., Savenije, H.H.G. 2006. Optimizing training data for ANNs with Genetic Algorithms. *Hydrology and Earth Systems Sciences*, 10, pp. 603-608.

- Kim, S. dan Kim, H.S. 2008. Neural networks and genetic algorithm approach for nonlinear evaporation and evapotranspiration modeling. *Journal of Hydrology*, vol. 351, pp. 299-317.
- Kosasih, D. dan Rinaldo. Analisis aplikasi algoritma genetika untuk pencarian nilai fungsi maksimum. Makalah tidak diterbitkan.
- Linde, Damon van der. 2011. The Seasonal Trends of Gold Prices. Artikel di Gold Investing News (<http://goldinvestingnews.com/14894/the-seasonal-trends-of-gold-prices.html>)
- Ladlani, I., Houichi, L., Djemili, L., Heddami, S., dan Belouz, K. 2012. Modeling daily reference evapotranspiration (ET₀) in the north of Algeria using generalized regression neural networks (GRNN) and radial basis function neural networks (RBFNN): a comparative study. *Jurnal Meteorol Atmos Phys* 118, 163 – 178.
- McCann, P., Kalman, B., 1994. A Neural Network Model for the Gold Market. Washington University in St. Louis, Part of the Philosophy Neuroscience and Psychological Technical Report Series.
- Mutakhirah, I., Saptono, F., Hasanah, N., dan Wiryadinata, R. 2007. Pemanfaatan metode heuristic dalam pencarian jalur terpendek dengan algoritma semut dan algoritma genetika. Prosiding dalam Seminar Nasional Aplikasi Teknologi Informasi 2007, Yogyakarta, 16 Juni 2007, pp. B-33 s/d B-39.
- Parisi, A., Parisi, F., Diaz, D. 2008. Forecasting gold price changes : Rolling and recursive neural network. *Journal of Multinational Financial Management* 18, 477 – 487.
- Popescu, I., Constantinou, P., Naornita, M., dan Naornita, I. 2004. Generalized regression neural network prediction model for indoor environment.

Proceeding of 9th International Symposium on Computers and Communications, 28 Juni – 1 Juli 2004, 657 – 661.

Reeves, C.R., Taylor, S.J. 1998. Selection of training data for neural networks by a genetic algorithm. *Parallel Problem Solving from Nature, Lecture Notes in Computer Science* vol. 1498, pp 633-642.

Shafiee, S. dan Topal, E. 2010. An overview of global gold market and gold price forecasting. *Jurnal Resources Policy* vol. 35, 178 – 189.

Sopipan, N., Sattayatham, P., dan Premanode, B. 2012. Forecasting volatility of gold price using markov regime switching and trading strategy. *Journal of mathematic finance* 2, 121 – 131.

Specht, D.F. 1991. A general regression neural network. *IEEE Transactions Neural Networks* 2, 568 – 576.

Suhartono dan Subanar, S.G. 2005. A comparative study of forecasting models for trend and seasonal time-series : Does complex models always yield better forecast than simple models?. *Jurnal Teknik Industri* vol. 7, 22-30.

Theodosiou, M. 2010. Forecasting issues : Ideas of decomposition and combination. *Working Paper Series 2010-4*, Juni 2010, Central Bank of Cyprus.

Theodosiou, M. 2011a. Disaggregation & aggregation of time series components : A hybrid forecasting approach using generalized neural networks and the theta method. *Neurocomputing* vol. 74, 896 – 905.

Theodosiou, M. 2011b. Forecasting monthly and quarterly time-series using STL decomposition. *International Journal of Forecasting* 27 (2011), 1178-1195.

- Tully, E. dan Lucey, B.M. 2007. A power GARCH examination of the gold market. *Research in International Business and Finance* vol. 21, Juni 2007, pp 316 – 325.
- Wang, Y.S. dan Chueh, Y.L. 2012. Dynamic transmission effects between the interest rate, the US dollar, and gold and oil prices. *Jurnal Economic Modelling* 30, 792 – 798.
- Wibowo, A.T. dan Widhiantika, B.A. 2010. Peramalan harga dinar di Indonesia menggunakan grammatical evolution. *Konferensi Nasional Sistem dan Informatika*, Bali, November 13, 2010, pp. 45 – 50.
- Witjaksono, A.A. 2010. Analisis pengaruh tingkat suku bunga SBI, harga minyak dunia, harga emas dunia, kurs rupiah, indeks Nikkei 225, dan indeks Dow Jones terhadap IHSG. Laporan tesis.
- World Gold Council (WGC). URL : <http://www.gold.org>
- Yao, Xin. 1999. Evolving artificial neural networks. *Proceeding of the IEEE*, vol. 87. no. 9, pp 1423-1447, Sep. 1999.
- Yazici, G., Polat, O., Yildirim, T. 2006. Genetic optimizations for radial basis function and general regression neural networks. *Proceeding of the 5th Mexican International Conference on Artificial Intelligence*, Apicazo, Mexico, 13-17 Nopember 2006, pp. 348-356.
- Zhang, G., Patuwo, B.E., Hu, M.Y. 1998. Forecasting with artificial neural networks : The state of the art. *International Journal of Forecasting* vol. 14, 35 – 62.
- Zhao, S., Zhang, J., Li, X., Song, W. 2007. A generalized regression neural network based on fuzzy means clustering and its application in system

identification. 2007 International Symposium on Information Technology Convergence, Jeonju, Korea, 23-24 November 2007, pp. 13-16.

Zhou, J. dan Yang, K. 2010. General regression neural network forecasting model based on PSO algorithm in water demand. Proceeding of 3rd International Symposium on Knowledge Acquisition and Modeling, Wuhan, China, 20-21 Oktober 2010, pp. 51-54.

Zou, H.F., Xia, G.P., Yang, F.T., Wang, H.Y. 2007. An investigation and comparison of artificial neural network and time series model for Chinese food grain price forecasting. *Neurocomputing* vol. 70, 2913-2923.

BIOGRAFI PENULIS



Penulis dilahirkan di Malang, 20 Maret 1981. Merupakan anak pertama dari dua bersaudara. Penulis menempuh pendidikan formal di SDK Mardiyata I, Malang, SMPN 3 Malang, dan SMUN 3 Malang. Pada tahun 1999, melanjutkan pendidikan di Teknik Informatika Institut Teknologi Bandung dan lulus pada tahun 2004. Sejak tahun 2006, mengabdikan sebagai staf pengajar di Jurusan Teknik Informatika Fakultas Teknik Universitas Muhammadiyah Malang. Menikah sejak tahun 2006 dengan Dian Ferianto dan dikaruniai dua putri cantik, Hana Syahidah dan Fahmida Sabrina. Pada tahun 2011, penulis mengambil program pascasarjana di Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya dengan Program Studi Sistem Informasi dan bidang keahlian Sistem Pendukung Keputusan serta terdaftar dengan NRP 5111202002.