

**PERAMALAN HARGA EMAS PADA PASAR BERJANGKA
MENGUNAKAN ALGORITMA *SUPPORT VECTOR
REGRESSION***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Galuh Widhi Gumilar
NIM: 125150100111018



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

PERAMALAN HARGA EMAS PADA PASAR BERJANGKA MENGGUNAKAN
ALGORITMA *SUPPORT VECTOR REGRESSION*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh:
Galuh Widhi Gumilar
NIM: 125150100111018

Skripsi ini telah diuji dan dinyatakan lulus pada
11 Agustus 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Imam Cholissodin, S.Si, M.Kom
NIK: 201201 850719 1 001

Randy Cahya W, S.ST, M.Kom
NIK: 201405 880206 1 001

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

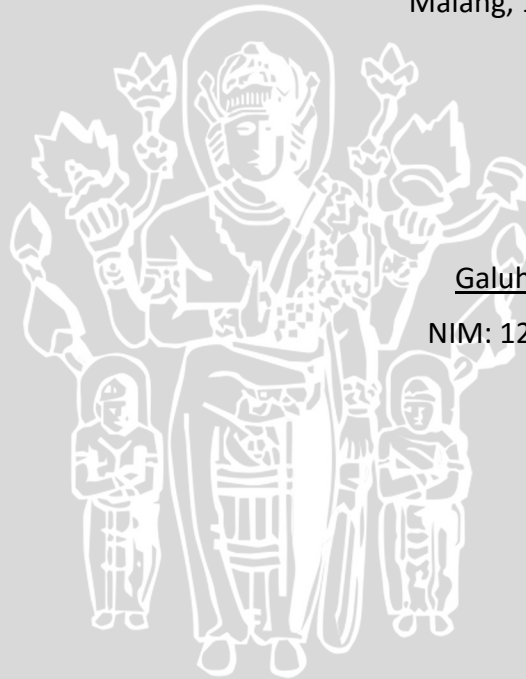
Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan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

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

Malang, 11 Agustus 2016

Galuh Widhi Gumilar

NIM: 125150100111018



KATA PENGANTAR

Dengan menyebut nama Allah Yang Maha Pengasih lagi Maha Penyayang. Segala puji bagi Allah yang telah melimpahkan rahmat serta hidayah-Nya sehingga penulis dapat menyelesaikan Skripsi dengan judul “Peramalan Harga Emas pada Pasar Berjangka menggunakan Algoritma *Support Vector Regression*” yang mana skripsi ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer di Fakultas Ilmu Komputer Universitas Brawijaya Malang.

Melalui kesempatan ini, penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada semua pihak yang telah berkontribusi dan memberikan dukungan lahir dan batin selama penyelesaian skripsi ini, yaitu:

1. Imam Cholissodin, S.Si., M.Kom dan Randy Cahya W, S.ST, M.Kom selaku dosen pembimbing skripsi yang telah sabar dalam membantu, membimbing dan mengarahkan penulis sampai terselesaikannya skripsi ini.
2. Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D, Ir. Heru Nurwasito M.Kom, Drs. Marji, M.T dan Edy Santoso, S.Si, M.Kom selaku Dekan, Wakil Dekan 1, Wakil Dekan 2, Wakil Dekan 3 Fakultas Ilmu Komputer Universitas Brawijaya.
3. Tri Astoto Kurniawan, S.T, M.T, Ph.D selaku Ketua Jurusan Teknik Informatika.
4. Novanto Yudhistira selaku dosen penasehat akademik yang selalu memberikan nasehat kepada penulis selama menempuh masa studi.
5. Seluruh Dosen Informatika yang telah memberikan ilmu kepada penulis dari awal sampai akhir masa studi di Fakultas Ilmu Komputer Universitas Brawijaya.
6. Kedua orangtua Bapak Mudji Wantoro dan Ibu Yuliana Daryati, kakak Sungging Kalpika Widhi, adik Gurit Widhi Anjunari, beserta keluarga besar yang sudah memberikan doa, nasehat, motivasi, perhatian, bantuan lahir dan batin hingga terselesaikannya skripsi ini.
7. Para sahabat Istidyah Nur S., Mamlu’atul Nur K., Nurul Islamiah, Risda Amalia K., Munifatullaili M.S., Randy Ari P., Prihani Pratiwi, Anung Adi N., Ainun Najib E. C., Reza Rahardian, Krisbianto Cahyo S., yang memberikan dukungan dalam penyelesaian skripsi ini.
8. Seluruh Civitas Akademika Informatika Universitas Brawijaya yang telah banyak memberikan dukungan dan bantuan selama masa studi dan penyelesaian skripsi ini.
9. Ujang Prasetyo S.E beserta keluarga besar PT. Victory International Futures Cab. Malang yang bersedia memberikan data untuk keperluan penyelesaian skripsi ini.
10. Kawan-kawan Informatika angkatan 2012 yang telah memberikan bantuan selama masa studi di Universitas Brawijaya.
11. Semua pihak yang tidak bisa penulis sebutkan satu persatu yang terlibat langsung maupun tidak langsung dalam penyelesaian skripsi ini.

Malang, 11 Agustus 2016

Galuh Widhi Gumilar

hulag1922@gmail.com

ABSTRAK

Emas merupakan salah satu jenis komoditi paling diminati untuk tujuan investasi. Salah satu macam investasi emas adalah dengan sistem berjangka atau sistem perdagangan alternatif (SPA) yaitu perdagangan yang berkaitan dengan jual-beli kontrak derivatif. Ketika berinvestasi emas di bursa berjangka yang harus diperhatikan secara hati-hati adalah pergerakan harga emas di pasar fisik, sehingga diperlukan sebuah analisa dan strategi untuk menentukan kapan harus membeli (buy) dan kapan harus menjual (sell). Peramalan harga ini menjadi hal yang diperlukan untuk memudahkan trader/investor untuk menganalisa prospek investasi dan mengatur strategi perdagangan di masa mendatang. *Support Vector Machine* diperkenalkan untuk memecahkan masalah pengenalan pola oleh Vapnik sebagai teknik minimalisasi risiko struktural. *Support Vector Regression* merupakan pengembangan model dari SVM yang memudahkan pemilihan model secara otomatis. Tujuannya adalah untuk mengumpulkan solusi optimal dengan iterasi yang sangat cepat dibandingkan SVM konvensional dan sangat sederhana untuk diimplementasikan bahkan untuk masalah dengan ukuran besar. Data yang digunakan adalah harga emas harian pada pasar berjangka pada tahun 2014 dan 2015, dengan nilai tukarnya terhadap Dolar Amerika Serikat. Parameter yang digunakan pada algoritma SVR yaitu kompleksitas (C), epsilon (ϵ), sigma (σ) untuk Kernel Gaussian RBF, *constant learning rate* (cLR), dan lambda (λ). Hasil pengujian terbaik yang didapatkan berdasarkan nilai evaluasi *Mean Absolute Percentage Error* (MAPE) senilai 0,324378 dengan nilai parameter $C = 500$, $\epsilon = 5 \times 10^{-9}$, $\sigma = 50$, $cLR = 1$, $\lambda = 0,01$, jumlah data latih = 30 hari, jumlah data uji = 10 hari, dan iterasi maksimal = 1000.

Kata kunci: Peramalan, harga emas, pasar berjangka, *Support Vector Regression*

ABSTRACT

Gold is one of the most desirable commodity for investment purposes. A kind of ways to invest in gold is gold investments with futures system or alternative trading system (ATS) that is associated with the trade of buying and selling derivatives contracts. When investing in gold futures, one thing need to be considered carefully is the gold price movements in the physical market, so we need an analysis and strategy to determine the time to buy and sell. This price forecasting becomes necessary to facilitate the trader / investor to analyze the prospects of investment and regulate trade strategy in the future. Support Vector Machine was introduced to solve pattern recognition problems by Vapnik as techniques that have ability to minimize some structural risk. Support Vector Regression is the enhancement of SVM which ease model selection automatically. It's goal is to collect the optimal solution by iterating very quickly than conventional SVM and very simple to implement even for problems with large size. The data used is the daily gold price in the futures market for 2 years, 2014 to 2015, with the exchange rate against the US Dollar. The parameters used in the algorithm SVR namely complexity (C), epsilon (ϵ), sigma (σ) for Gaussian RBF Kernels, constanta learning rate (CLR), and lambda (λ). The best test results obtained by the evaluation value Mean Absolute Percentage Error (MAPE) at 0,324378 with the value of the parameter $C = 500$, $\epsilon = 5 \times 10^{-9}$, $\sigma = 50$, $cLR = 1$, $\lambda = 0,01$, number of data-training = 30 days, number of data-testing = 10 days, maximal iteration = 1000.

Keywords: Forecasting, gold price, the futures market, Support Vector Regression

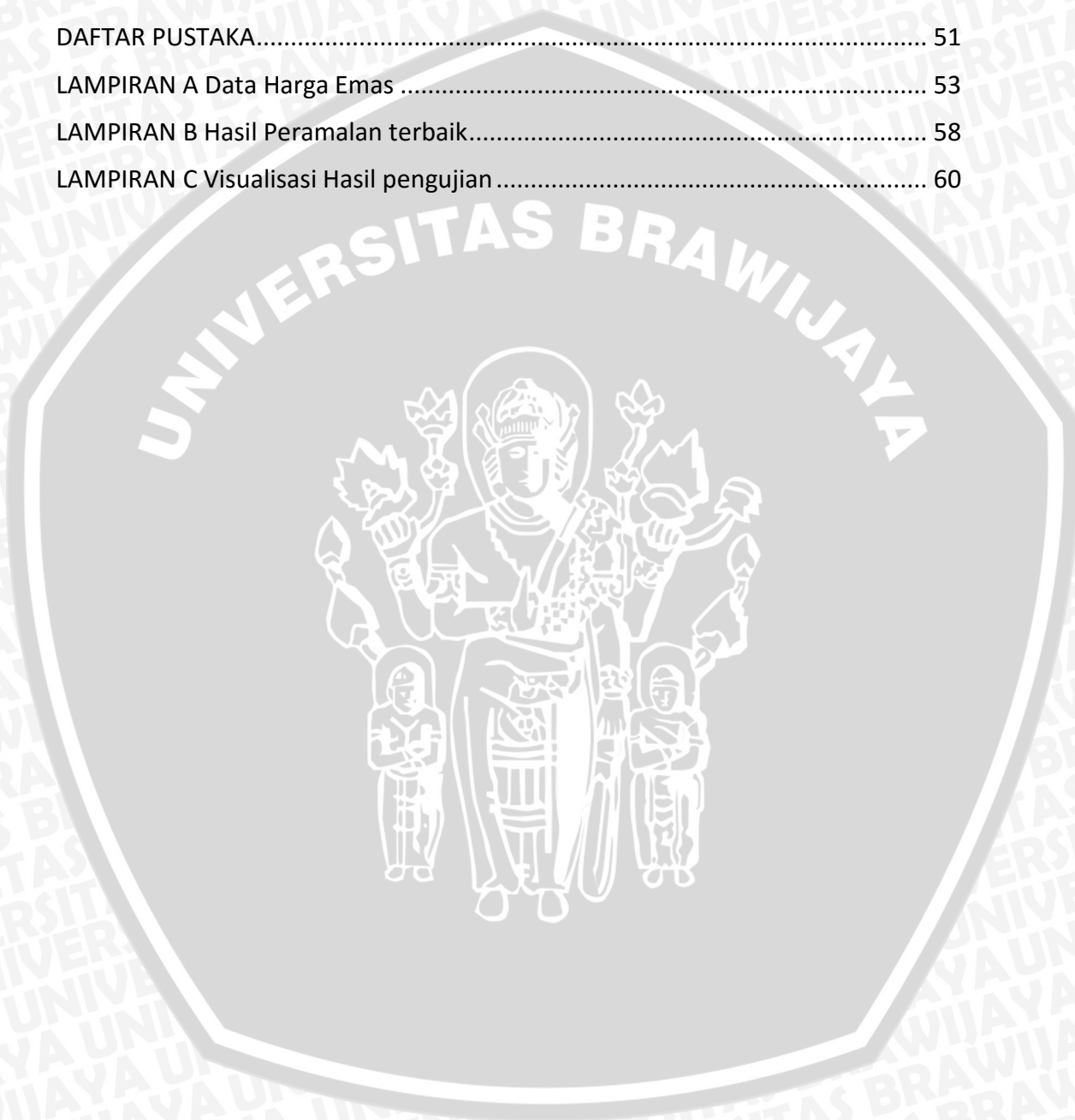
DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xii
DAFTAR KODE PROGRAM	xiii
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Komoditi Emas	7
2.3 Perdagangan Emas Berjangka.....	8
2.4 Algoritma <i>Support Vector Regression</i> (SVR)	8
2.5 Normalisasi Data.....	11
2.6 Nilai Evaluasi	11
BAB 3 METODOLOGI	12
3.1 Studi Literatur	12
3.2 Pengumpulan Data	13
3.3 Perancangan Sistem.....	13
3.4 Implementasi Sistem	13

3.5 Pengujian Sistem.....	13
3.6 Pengambilan Kesimpulan.....	13
BAB 4 PERANCANGAN SISTEM.....	14
4.1 Formulasi Permasalahan.....	14
4.2 Perancangan Antarmuka Pengguna	14
4.3 Siklus Algoritma <i>Support Vector Regression</i> (SVR)	15
4.4 Perancangan Uji Coba dan Evaluasi.....	29
4.4.1 Uji Coba Jumlah Iterasi untuk Pelatihan SVR.....	29
4.4.2 Uji Coba Nilai Parameter C (<i>Complexity</i>)	30
4.4.3 Uji Coba Nilai Parameter ϵ (<i>epsilon</i>) SVR.....	30
4.4.4 Uji Coba Nilai Parameter σ (<i>sigma</i>) Kernel Gaussian RBF	31
4.4.5 Uji Coba Nilai Parameter cLR SVR	31
4.4.6 Uji Coba Nilai Parameter λ (<i>lambda</i>) SVR.....	32
4.4.7 Uji Coba Variasi Jumlah Data Latih.....	32
4.4.8 Uji Coba Variasi Jumlah Data Uji.....	33
BAB 5 IMPLEMENTASI SISTEM.....	34
5.1 Implementasi <i>Source Code</i> Program.....	34
5.1.1 Normalisasi Data	34
5.1.2 Penghitungan Kernel, Matriks Hessian (R_{ij}), dan Nilai γ (<i>gamma</i>)	34
5.1.3 Iterasi <i>Sequential Learning</i>	36
5.1.4 Penghitungan Fungsi Peramalan ($f(x)$)	37
5.1.5 Penghitungan Nilai <i>Error Rate</i>	38
5.2 Implementasi Antarmuka Pengguna	38
BAB 6 PENGUJIAN DAN ANALISIS.....	39
6.1 Hasil dan Analisis Uji Coba Jumlah Iterasi Pelatihan	39
6.2 Hasil dan Analisis Uji Coba Nilai Parameter C (<i>Complexity</i>)	41
6.3 Hasil dan Analisis Uji Coba Nilai Parameter ϵ (<i>epsilon</i>)	42
6.4 Hasil dan Analisis Uji Coba Nilai Parameter σ (<i>sigma</i>)	43
6.5 Hasil dan Analisis Uji Coba Nilai Parameter cLR	45
6.6 Hasil dan Analisis Uji Coba Nilai Parameter λ (<i>lambda</i>)	46
6.7 Hasil dan Analisis Uji Coba Variasi Jumlah Data Latih	47



6.8 Hasil dan Analisis Uji Coba Variasi Jumlah Data Uji	48
BAB 7 Penutup	50
7.1 Kesimpulan.....	50
7.2 Saran	50
DAFTAR PUSTAKA.....	51
LAMPIRAN A Data Harga Emas	53
LAMPIRAN B Hasil Peramalan terbaik.....	58
LAMPIRAN C Visualisasi Hasil pengujian	60



DAFTAR TABEL

Tabel 2. 1 Kajian Pustaka.....	5
Tabel 4. 1 Data Harga Emas	14
Tabel 4. 2 Inisialisasi Parameter SVR.....	16
Tabel 4.3 Fitur yang digunakan	17
Tabel 4.4 Hasil Normalisasi Data Latih.....	18
Tabel 4.5 Hasil Normalisasi Data Uji	18
Tabel 4.6 Matriks <i>Kernel</i> Data Latih.....	20
Tabel 4.7 Matriks <i>Kernel</i> Data Uji.....	20
Tabel 4.8 Matriks <i>Hessian</i> Data Latih.....	21
Tabel 4.9 Matriks <i>Hessian</i> Data Uji	21
Tabel 4.10 Nilai Awal <i>Lagrange Multiplier</i>	22
Tabel 4.11 Hasil Iterasi Pertama	25
Tabel 4.12 Hasil Iterasi ke-10	25
Tabel 4.13 Hasil Peramalan pada Data Latih	27
Tabel 4.14 Hasil Peramalan pada Data Uji.....	28
Tabel 4. 15 Rancangan Uji Coba Jumlah Iterasi	29
Tabel 4. 16 Rancangan Uji Coba Nilai Parameter C (<i>Complexity</i>).....	30
Tabel 4. 17 Rancangan Uji Coba Nilai Parameter ϵ (<i>epsilon</i>).....	30
Tabel 4. 18 Rancangan Uji Coba Nilai Parameter σ (<i>sigma</i>).....	31
Tabel 4.19 Rancangan Uji Coba Nilai Parameter cLR	31
Tabel 4.20 Rancangan Uji Coba Nilai Parameter λ (<i>lambda</i>).....	32
Tabel 4.21 Rancangan Uji Coba Variasi Data Latih	32
Tabel 4.22 Rancangan Uji Coba Variasi Data Uji.....	33
Tabel 6. 1 Hasil Pengujian Jumlah Iterasi SVR.....	39
Tabel 6. 2 Waktu Eksekusi Hasil Pengujian Jumlah Iterasi SVR	40
Tabel 6. 3 Hasil Pengujian Nilai Parameter C (<i>Complexity</i>)	41
Tabel 6. 4 Hasil Pengujian Nilai Parameter ϵ (<i>epsilon</i>)	43
Tabel 6. 5 Hasil Pengujian Nilai Parameter σ (<i>sigma</i>).....	44
Tabel 6. 6 Hasil Pengujian <i>range</i> parameter cLR	45
Tabel 6. 7 Hasil Pengujian Nilai Parameter λ (<i>lambda</i>)	47

Tabel 6. 8 Hasil Pengujian Variasi Jumlah Data Latih..... 48
Tabel 6. 9 Hasil Pengujian Variasi Data Uji..... 49

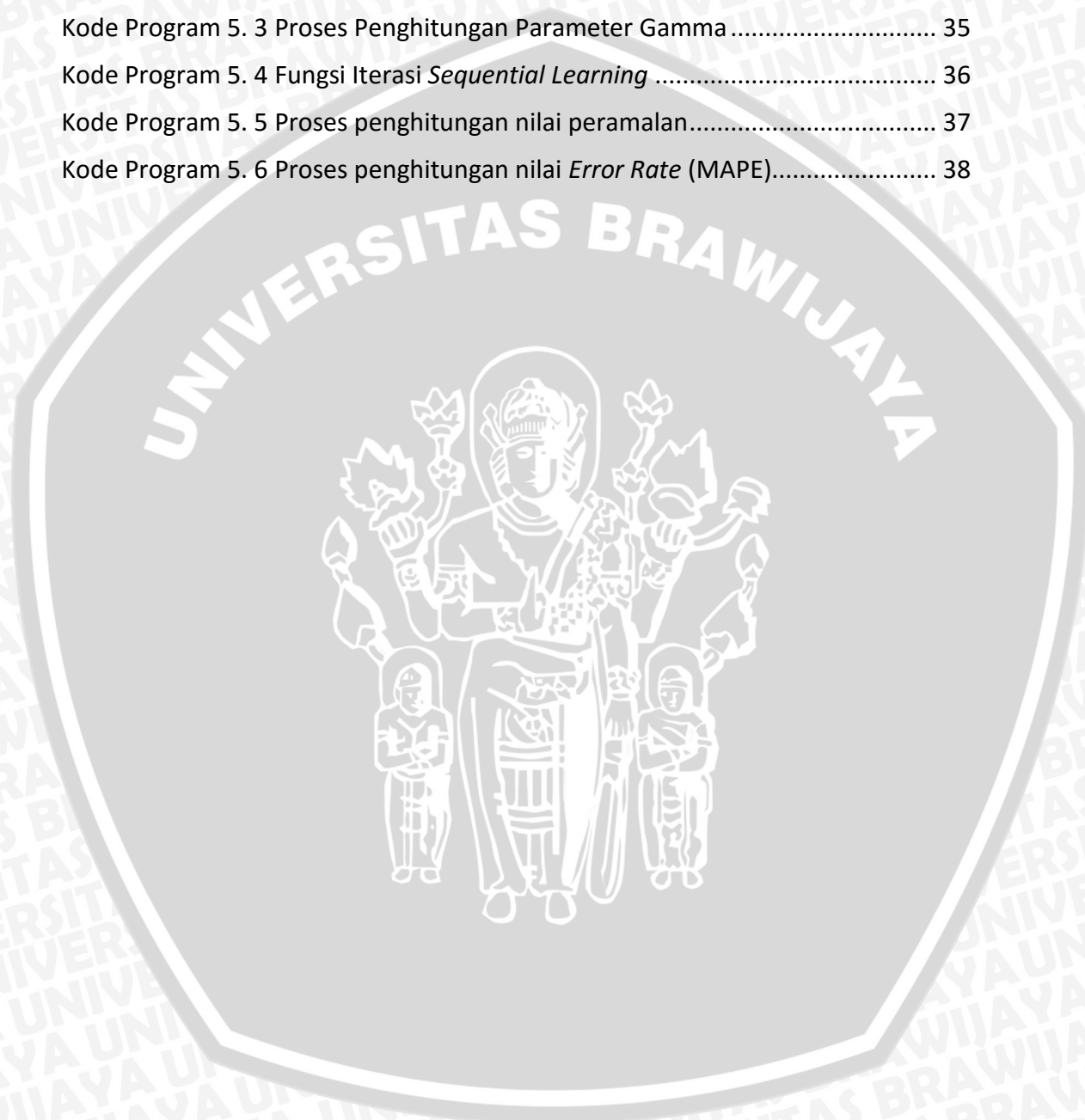


DAFTAR GAMBAR

Gambar 4. 1 Rancangan Antarmuka Pengguna	15
Gambar 4. 2 Diagram Alir Algoritma <i>Support Vector Regression</i>	16
Gambar 4. 3 Diagram Alir Proses Normalisasi Min-Max.....	18
Gambar 4. 4 Diagram Alir Menghitung <i>kernel</i> dan Matriks <i>Hessian</i>	20
Gambar 4. 5 Diagram Alir Menghitung nilai parameter <i>gamma</i> (γ)	22
Gambar 4. 6 Diagram Alir Proses Iterasi <i>Sequential Training</i>	24
Gambar 4. 7 Diagram Alir Menentukan Nilai $f(x)$	26
Gambar 4. 8 Diagram Alir Proses Denormalisasi	27
Gambar 4. 9 Diagram Alir Proses menghitung MAPE	28
Gambar 5. 1 Jendela Aplikasi Peramalan	38
Gambar 6. 1 Grafik Rerata <i>Error Rate</i> Pengujian Iterasi SVR.....	40
Gambar 6. 2 Grafik Rerata Waktu Eksekusi Pengujian Iterasi SVR	41
Gambar 6. 3 Grafik rata-rata <i>error rate</i> pengujian nilai parameter C	42
Gambar 6. 4 Grafik rata-rata <i>error rate</i> pengujian nilai parameter ϵ	43
Gambar 6. 5 Grafik rata-rata <i>error rate</i> pengujian nilai parameter σ	45
Gambar 6. 6 Grafik rata-rata <i>error rate</i> pengujian nilai parameter cLR	46
Gambar 6. 7 Grafik rata-rata <i>error rate</i> pengujian nilai parameter λ	47
Gambar 6. 8 Grafik nilai <i>error rate</i> pengujian variasi jumlah data latih	48
Gambar 6. 9 Grafik nilai <i>error rate</i> pengujian variasi jumlah data uji	49

DAFTAR KODE PROGRAM

Kode Program 5. 1 Proses Normalisasi Data.....	34
Kode Program 5. 2 Proses Penghitungan Kernel dan Matriks Hessian.....	35
Kode Program 5. 3 Proses Penghitungan Parameter Gamma.....	35
Kode Program 5. 4 Fungsi Iterasi <i>Sequential Learning</i>	36
Kode Program 5. 5 Proses penghitungan nilai peramalan.....	37
Kode Program 5. 6 Proses penghitungan nilai <i>Error Rate</i> (MAPE).....	38



DAFTAR LAMPIRAN

LAMPIRAN A Data Harga Emas	53
A.1 Tahun 2014	53
A.2 Tahun 2015	55
LAMPIRAN B Hasil Peramalan terbaik.....	58
B.1 Penggunaan <i>Sequence</i> Data Latih Peramalan	58
B.2 Penggunaan <i>Sequence</i> Data Uji Peramalan.....	58
B.3 Nilai <i>Lagrange Multiplier</i> Terbaik pada Fungsi Regresi SVR.....	59
LAMPIRAN C Visualisasi Hasil pengujian	60
C.1 Visualisasi Hasil Pengujian Jumlah Iterasi terbaik SVR.....	60
C.2 Visualisasi Hasil Pengujian Nilai Parameter C (<i>Complexity</i>)	60
C.3 Visualisasi Hasil Pengujian Nilai Parameter ϵ (<i>epsilon</i>)	61
C.4 Visualisasi Hasil Pengujian Nilai Parameter σ (<i>sigma</i>)	61
C.5 Visualisasi Hasil Pengujian Nilai Parameter cLR (<i>Learning Rate</i>)	62
C.6 Visualisasi Hasil Pengujian Nilai Parameter λ (<i>lambda</i>)	62
C.7 Visualisasi Hasil Pengujian Variasi Jumlah Data Latih	63
C.8 Visualisasi Hasil Pengujian Variasi Jumlah Data Uji.....	63



BAB 1 PENDAHULUAN

1.1 Latar belakang

Emas merupakan salah satu jenis komoditi paling diminati untuk tujuan investasi. Dibandingkan indek (saham), *foreign exchange* (forex) atau valuta asing (valas) dan komoditi lainnya, emas memiliki nilai yang cenderung stabil dan dipercaya sebagai objek investasi yang paling aman dan menguntungkan dengan karakteristiknya yang unik seperti *safe haven*, *hedge against inflation* (Departemen Perdagangan Republik Indonesia, 2009). Investasi emas dapat dilakukan dengan berbagai cara, di antaranya investasi emas batangan, investasi emas simpanan, reksa dana emas, saham pertambangan emas, emas berjangka, perhiasan, dan koin emas (Dwi, 2015).

Indonesia mulai mengenal sistem perdagangan alternatif, perdagangan yang berkaitan dengan jual-beli kontrak derivatif, sejak tahun 90-an. Pada sistem ini tidak ada penyerahan barang secara fisik, melainkan penyelesaian secara tunai yaitu selisih antara harga jual dan harga beli yang harus dibayar atau diterima (Ismail, 2012). Menurut Panggabean, *trading* emas *online* merupakan salah satu bisnis yang memberikan keuntungan tinggi namun memiliki resiko yang tinggi pula (Panggabean, et al., 2013). Maka dari itu diperlukan sebuah analisa dan strategi untuk menentukan kapan harus membeli (buy) dan kapan harus menjual (sell).

Peramalan harga ini menjadi hal yang diperlukan untuk memudahkan *trader/investor* untuk menganalisa prospek investasi dan mengatur strategi perdagangan di masa mendatang. Metode dan periode peramalan bisa beragam tergantung pada waktu dan informasi yang digunakan dari masa lalu yang mana metode prediksi yang teratur akan memberikan keyakinan pada penggunaanya karena dapat dievaluasi secara ilmiah (Syafruddin, 2014). Jangka waktu yang dapat digunakan untuk analisa pergerakan harga emas dibedakan menjadi jangka waktu bulanan, mingguan, dan harian, namun berdasarkan karakteristiknya harga emas level tertinggi dan terendah dapat terulang kembali hanya dalam hitungan jam (Suharto, 2015).

Support Vector Machines (SVM) diperkenalkan untuk memecahkan masalah pengenalan pola oleh Vapnik sebagai teknik minimalisasi risiko struktural. SVMs berkemampuan untuk meringkas informasi dalam data pelatihan dan memberikan representasi yang tersebar menggunakan luaran keputusan non-linear pada dimensi VC yang relatif rendah. Selain itu, ketika diperluas untuk kasus regresi, solusi SVM menyediakan alat untuk beberapa tingkat pemilihan model secara otomatis. Pengembangan yang dinamakan *SV Regression* (SVR) ini ditujukan untuk mengumpulkan solusi optimal dengan iterasi yang sangat cepat dibandingkan solusi SVM konvensional dan sangat sederhana untuk diimplementasikan bahkan untuk masalah dengan ukuran besar. (Vijayakumar & Wu, 1999).

Beberapa penelitian telah membuktikan kemampuan SVR dalam beberapa studi kasus, di antaranya penelitian oleh Suranart yang membandingkan hasil peramalan harga emas antara menggunakan *Neural Network* (NN), *Radial Basis*

Function Network (RBF-NN), dan *Support Vector Regression* (SVR). Dengan menggunakan dataset harga emas antara bulan Juni 2008 sampai April 2013 dan beberapa metode evaluasi seperti *Mean Absolute Deviation* (MAD), *Mean Square Error* (MSE), *Mean Forecast Error* (MFE) dan *Mean Absolute Percentage Error* (MAPE) didapati SVR memiliki akurasi tertinggi disusul RBF-NN dan NN (Suranart, et al., 2014).

Berdasarkan permasalahan dan solusi dari penelitian sebelumnya, peneliti akan menerapkan metode *Support Vector Regression* untuk meramalkan nilai tukar Emas terhadap Dollar Amerika Serikat pada pasar berjangka dengan nilai evaluasi yang digunakan adalah MAPE (*Mean Absolute Percentage Error*). Penerapan metode yang dilakukan bertujuan untuk mencari akurasi terbaik dengan membandingkan variabel-variabel yang berkaitan. Dari hasil peramalan yang dihasilkan diharap dapat meningkatkan efisiensi dan efektifitas serta mempermudah analisa harga sehingga *trader/investor* dapat menekan resiko kerugian.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah dipaparkan, dibuatlah beberapa rumusan masalah sebagai berikut:

1. Bagaimana mengimplementasikan algoritma *Support Vector Regression* (SVR) untuk meramalkan harga emas pada pasar berjangka?
2. Berapakah nilai evaluasi (*error rate*) terbaik dari hasil implementasi algoritma *Support Vector Regression* (SVR) untuk meramalkan harga emas pada pasar berjangka?

1.3 Tujuan

Adapun maksud dan tujuan dari penelitian ini adalah sebagai berikut:

1. Menerapkan algoritma *Support Vector Regression* (SVR) untuk meramalkan harga emas pada pasar berjangka.
2. Mengetahui nilai evaluasi dari hasil implementasi algoritma *Support Vector Regression* (SVR) untuk meramalkan harga emas pada pasar berjangka.

1.4 Manfaat

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Menciptakan suatu rancangan aplikasi peramalaan harga emas yang dapat digunakan untuk membantu analisa dan mengatur strategi bagi investor.
2. Mahasiswa dapat menerapkan pengetahuan teoritis yang telah diperoleh di bangku perkuliahan dalam berbagai kasus di lapangan.
3. Mengenal lebih jauh relevansi ilmu yang diterima selama kuliah, untuk diterapkan dalam situasi yang sesungguhnya.

4. Merupakan salah satu tolak ukur bagi perguruan tinggi untuk menilai kualitas yang dimiliki oleh mahasiswa, untuk acuan bagi perguruan tinggi dalam meningkatkan mutu pendidikan selanjutnya

1.5 Batasan masalah

Pada penelitian ini terdapat beberapa batasan masalah yang harus terpenuhi agar pembahasan penelitian tidak menyimpang dari perumusan masalah, antara lain:

1. Analisis yang digunakan untuk meramalkan harga emas pada pasar berjangka adalah analisis teknikal.
2. Data yang digunakan adalah fluktuasi harga emas harian pada pasar berjangka selama 2 tahun, 2014 sampai 2015, sesuai yang didapat dari PT. Victory International Futures dengan nilai tukarnya terhadap Dolar Amerika Serikat.
3. Metode yang digunakan adalah *Support Vector Regression* (SVR), dan metode kernel yang digunakan di dalamnya adalah *Kernel Gaussian RBF*.
4. *Sequence data* yang digunakan sebagai fitur adalah data 4 hari sebelum data aktual yang ditargetkan.
5. Model peramalan diterapkan secara *supervised* di mana semua nilai untuk membentuk pola dataset latihan dan uji telah diketahui nilainya.
6. Penghitungan nilai evaluasi dilakukan dengan metode *Mean Absolute Percentage Error* (MAPE).

1.6 Sistematika pembahasan

Sistematika penulisan dalam penyusunan Skripsi ini dibagi menjadi 7 bab, uraian masing-masing bab dijabarkan sebagai berikut:

BAB I: PENDAHULUAN

Bab ini menguraikan latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika penulisan skripsi.

BAB II: LANDASAN KEPUSTAKAAN

Bab ini menguraikan tentang landasan teori dan referensi yang mendasari penyusunan Skripsi. Teori yang tercakup terkait temuan dan penelitian sebelumnya terkait metode *Support Vector Regression* (SVR) dan emas pada pasar berjangka.

BAB III: METODOLOGI

Bab ini menguraikan tentang metode dan langkah-langkah yang dilakukan dalam proses penelitian tentang meramalkan harga emas pada pasar berjangka menggunakan algoritma *Support Vector Regression* (SVR). Langkah-langkah tersebut terdiri dari studi literatur, pengumpulan data, analisa dan perancangan sistem, implementasi sistem, pengujian dan evaluasi sistem, penarikan kesimpulan.

BAB IV: PERANCANGAN

Bab ini menguraikan tentang perancangan yang dilakukan dalam proses penelitian ini meliputi formulasi permasalahan, perancangan antarmuka pengguna, siklus algoritma *Support Vector Regression* (SVR) dalam menyelesaikan masalah peramalan harga, dan perancangan uji coba dan evaluasi sistem.

BAB V: IMPLEMENTASI

Bab ini menguraikan tentang proses implementasi dari perancangan yang dilakukan meliputi kode pemrograman sistem dan antarmuka pengguna.

BAB VI: PENGUJIAN DAN ANALISIS

Bab ini menguraikan tentang hasil pengujian dan analisis dari performansi aplikasi peramalan harga emas pada pasar berjangka menggunakan algoritma SVR yang telah dibangun.

BAB VII: PENUTUP

Bab ini menguraikan kesimpulan yang diambil berdasarkan analisa meliputi hasil, kelebihan dan kekurangan, dan saran guna pengembangan penelitian ini.



BAB 2 LANDASAN KEPUSTAKAAN

Landasan kepustakaan terdiri dari kajian pustaka dan dasar teori yang digunakan. Kajian Pustaka adalah pembahasan tentang penelitian yang sudah dilakukan. Dasar Teori adalah pembahasan tentang teori yang diperlukan untuk meramalkan harga emas pada pasar berjangka menggunakan algoritma *support vector regression*.

2.1 Kajian Pustaka

Kajian Pustaka ini berisi penelitian-penelitian terdahulu yang berkaitan atau memiliki kesesuaian dengan penelitian penulis. Pokok-pokok kajian pustaka tersebut dijelaskan pada Tabel 2.1:

Tabel 2. 1 Kajian Pustaka

No.	Judul	Obyek	Metode	Hasil
1.	Sequential Support Vector Classifiers and Regression	Sonar data, USPS, MNIST database	Support Vector Machine	Pengusulan sebuah algoritma yang pada dasarnya merupakan metode peningkatan gradien dalam problem ganda termodifikasi untuk memenuhi kendala linear.
2.	<i>The Mechanisms of the Bullion Markets and the Return of Gold (and Silver) as Money</i>	Emas dan Perak		<i>Digital Gold Currency</i> (DGC) adalah solusi yang diusulkan sebagai alat transaksi pada perdagangan e-commerce dan transaksi secara online sebagai substansi nilai mata uang tertentu. Tidak hanya itu, DGC adalah tren baru yang akan merevolusi dunia bisnis dan keuangan.
3.	<i>Analysis of Comparisons for Forecasting Gold Price using Neural Network, Radial Basis Function</i>	Harga Emas	<i>Neural Network, Radial Basis Function Network,</i>	Ketiga metode digunakan untuk mempelajari rincian harga emas jangka pendek. Hasil akurasinya diukur dengan deviasi,

	<i>Network and Support Vector Regression</i>		<i>Support Vector Regression</i>	<i>complete average value, average squared error, average error, dan nilai absolute average error. Setelah dibandingkan hasilnya, didapati akurasi SVR lebih baik dibandingkan RBF-N dan NN.</i>
4.	Prediksi Kebutuhan Energi Listrik Jangka Panjang untuk Provinsi Lampung hingga Tahun 2030	Konsumsi Energi Listrik Provinsi Lampung	Regresi Linear	Dengan menggunakan metode regresi linier diperoleh hasil prediksi daya listrik tersambung total pada tahun 2028 sebesar 2.841,78 dengan rata-rata pertumbuhannya sebesar 2,38 % dan pada tahun 2023 sebesar 5.934,98 dengan rata-rata pertumbuhannya sebesar 3, 83 %.

Sumber ((Vijayakumar & Wu, 1999), (Godazgar & Leila, 2013), (Suranart, et al., 2014), (Syafuruddin, 2014))

Penelitian pertama oleh Vijayakumar membahas pembenaran teoritis *Support Vector Machine* untuk kasus regresi yang bertujuan untuk mengumpulkan solusi optimal dengan sangat cepat dalam hal jumlah iterasi, yang mana lebih cepat dibandingkan solusi SVM konvensional dan sangat sederhana untuk diimplementasikan bahkan untuk masalah dengan ukuran besar. Ide dasar dari algoritma SV untuk estimasi regresi adalah untuk menghitung fungsi linear dalam ruang fitur dimensi tinggi (yang memiliki titik produk) dan menghitung fungsi non-linear di ruang input data. Evaluasi penelitian pada masalah klasifikasi benchmark pada data sonar, USPS dan database MNIST membuktikan kecepatan dan ketahanan dari prosedur pembelajaran.

Penelitian kedua oleh Godazgar, membahas peranan emas pada era modern sebagai penyokong uang kertas. Penelitiannya menyarankan *Digital Gold Currency* (DGC) sebagai salah satu solusi untuk pasar modern yang berjalan pada perdagangan e-commerce dan transaksi secara online. DGC adalah tren baru yang datang bersamaan dengan lahirnya e-commerce. Membangun hubungan antara emas sebagai uang, dan e-commerce sebagai media, pasti akan merevolusi dunia bisnis dan keuangan. Inilah yang mendorong kebanyakan orang untuk berinvestasi dalam emas sebagai perlindungan terhadap segala jenis krisis. Emas / perak berperan sebagai mata uang universal. Beberapa Keuntungan yang ditawarkan



adalah penekanan risiko bisnis atau individu dalam menjaga "cash" di "mata uang lokal". Dengan keterlibatan agen penukaran uang di tiap negara, penciptaan pasar lokal emas / perak akan membuat perkembangan yang stabil untuk emas / perak atau bentuk lainnya sebagai uang sehingga menjadi dasar penghapusan valuta asing bagi bisnis internasional. Tidak hanya itu, keuntungan lainnya adalah menjadikan pembeli individu juga bisa menjadi penjual dan bersaing dengan vendor.

Penelitian ketiga oleh Suranart bertujuan untuk mempelajari dan menganalisa perbandingan hasil peramalan harga emas antara menggunakan *Neural Network* (NN), *Radial Basis Function Network* (RBF-NN), dan *Support Vector Regression* (SVR). Dataset yang digunakan antara bulan Juni 2008 sampai April 2013 dipisahkan menjadi dua bagian yaitu rincian bulanan yang digunakan memprediksi 3 bulan ke depan dan rincian mingguan digunakan memprediksi 3 minggu berikutnya. Metode evaluasi yang digunakan adalah *Mean Absolute Deviation* (MAD), *Mean Square Error* (MSE), *Mean Forecast Error* (MFE) dan *Mean Absolute Percentage Error* (MAPE). Dari hasil perbandingan yang ditunjukkan, didapati SVR memiliki akurasi yang lebih baik disusul RBF-NN dan terakhir NN.

Penelitian keempat oleh Syafrudin, menerapkan metode regresi linier untuk memprediksi kebutuhan energi listrik di Provinsi Lampung hingga Tahun 2030. Pada penelitiannya, Syafrudin menyatakan bahwa prakiraan beban listrik yang akan ditanggung oleh sistem tenaga listrik menjadi faktor yang sangat mempengaruhi penentuan rencana pengembangan dan operasi sistem dalam jangka waktu berikutnya. Sistem prediksi digunakan untuk merencanakan kebutuhan yang dinyatakan dalam jumlah, sehingga hasil prediksi kebutuhan dan nilai sebenarnya perlu dibandingkan untuk mengevaluasi model peramalan. Metode prediksi adalah cara yang dilakukan untuk memperkirakan kejadian di masa mendatang secara sistematis atau pragmatis. Metode prediksi yang teratur akan memberikan keyakinan pada penggunaannya karena dapat dievaluasi secara ilmiah. Untuk jangka waktu prediksinya dibedakan menjadi 3, yaitu jangka panjang (> 1tahun), jangka menengah (1bulan – 1 tahun), dan jangka pendek (< 1 minggu).

2.2 Komoditi Emas

Komoditas adalah segala sesuatu yang memiliki nilai, dengan kualitas yang seragam dan diproduksi dalam jumlah besar oleh banyak produsen. Komoditi secara umum dikatakan sebagai suatu produk yang diperdagangkan melalui mekanisme bursa berjangka yang mana salah satu di antaranya adalah emas (PT. Victory International Futures, 2016).

Sebagai salah satu jenis komoditas, emas merupakan tujuan investasi yang paling banyak diminati. Bahkan di beberapa negara emas juga digunakan sebagai standar keuangan, cadangan devisa dan alat pembayaran utama. Umumnya investor melakukan perdagangan emas untuk *hedging* atau *safe haven* sebagai bentuk antisipasi dari berbagai krisis ekonomi. Kini dengan bantuan teknologi, emas bisa diperjualbelikan di perdagangan berjangka (*future trading*) yaitu

perdagangan yang bukan mentransaksikan bentuk fisik melainkan hanya bukti administrasi kepemilikannya saja (Ismail, 2012).

2.3 Perdagangan Emas Berjangka

Investasi emas dapat dilakukan dengan berbagai cara, di antaranya investasi emas batangan, investasi emas simpanan, reksa dana emas, saham pertambangan emas, emas berjangka, perhiasan, dan koin emas (Dwi, 2015). Perdagangan emas dengan sistem berjangka atau disebut juga sistem perdagangan alternatif (SPA) merupakan perdagangan yang berkaitan dengan jual-beli kontrak derivatif tanpa adanya penyerahan fisik.

Di dunia terdapat lima pasar emas global yang utama yaitu New York, London, Zurich, Hong Kong, Sydney dengan tingkat liquiditas yang tinggi. Di antara ke lima pusat perdagangan emas tersebut London merupakan yang tertua dan memiliki basis yang sangat kuat. *Loco London Gold* bukanlah nama emas melainkan sebutan untuk merepresentasikan basis perdagangan dan penyelesaian emas dan perak internasional di London yang mana pelaksanaannya dinaungi oleh London Bullion Market Association (LBMA) (PT. Mahadana Asta Berjangka, 2016). Produk emas yang diperdagangkan berdasarkan pasar berjangka *Loco London Gold* berukuran kontrak sebesar 100 troy ounce atau 3.110,35 gram (3,1 kg).

Perdagangan emas berjangka dipengaruhi oleh beberapa faktor seperti supply and demand, kegagalan bank, tingkat suku bunga riil yang rendah/negatif, krisis, pengaruh inflasi dan deflasi (Departemen Perdagangan Republik Indonesia, 2009). Beberapa karakteristik emas pada perdagangan berjangka adalah sebagai berikut (Suharto, 2015):

1. Volatilitas sangat tinggi, range pergerakan dalam satu hari dapat mencapai 1000 hingga 5000 poin.
2. Harga terendah dan tertinggi berkemungkinan terulang lagi hanya dalam tempo hitungan jam.
3. Memiliki hubungan erat dengan kurs Dollar Australia karena Australia merupakan negeri penghasil emas, Poundsterling karena Inggris merupakan pusat perdagangan emas dunia, dan Dollar Amerika Serikat karena mata uangnya menjadi tolok ukur perdagangan emas.

2.4 Algoritma Support Vector Regression (SVR)

Support Vector Regression (SVR) merupakan model pengembangan dari *Support Vector Machines* (SVM) yang ditujukan untuk permasalahan regresi dan time series. SVR mengadaptasi proses analisa regresi dan *time series* dari SVM dan memiliki ide dasar yaitu untuk menemukan persamaan yang memberikan *error* terkecil (Suranart, et al., 2014). Yang menjadi perbedaan diantaranya adalah SVM diterapkan untuk kasus klasifikasi dan menghasilkan keluaran berupa bilangan diskrit, sedangkan SVR diterapkan untuk kasus regresi yang memberikan keluaran berupa bilangan kontinyu (Furi, et al., 2015).

Secara umum fungsi regresi *linear* dirumuskan pada persamaan (2.1) berikut ini:

$$f(x) = \omega\varphi(x) + b \quad (2.1)$$

di mana ω dan b adalah koefisien sedangkan $\varphi(x)$ adalah fungsi pemetaan nilai x pada dimensi fitur yang lebih tinggi. (Vijayakumar & Wu, 1999) pada penelitiannya mengajukan sebuah algoritma sekuensial untuk regresi non-*linear* yang dapat menghasilkan solusi optimal dengan iterasi yang lebih cepat dibandingkan metode konvensional. Tahap-tahap perhitungan algoritma tersebut dijelaskan sebagai berikut:

1. Inisialisasi parameter dan batas iterasi

Parameter yang digunakan dalam SVR diantaranya C (*complexity*), ε (*epsilon*), cLR (*constant learning rate*), λ (*lambda*), dan batas iterasi yang merupakan iterasi maksimal untuk tahap nomor 4.

2. Penghitungan matriks *Hessian*

Matriks *Hessian* dihitung dengan persamaan (2.2) berikut:

$$[R]_{ij} = K(x_i, x_j) + \lambda^2, \text{ untuk } i \text{ dan } j=1,2,\dots,n \quad (2.2)$$

Keterangan:

$[R]_{ij}$ = matriks *Hessian*

$K(x_i, x_j)$ = fungsi *kernel*

λ^2 = variabel skalar

Fungsi *kernel* memetakan data input ke dalam dimensi yang lebih tinggi untuk menyelesaikan permasalahan non-*linear* sehingga setiap data dapat dipisahkan secara linear oleh sebuah *hyperplane* (Furi, et al., 2015). Banyak penelitian sebelumnya berpendapat bahwa SVR akan memberikan kinerja yang baik dengan menggunakan fungsi *kernel Gaussian* (RBF) (Li, et al., 2005), fungsi tersebut dirumuskan pada persamaan berikut:

$$K(x, x_i) = \exp\left(-\frac{\|x-x_i\|^2}{2\sigma^2}\right) \quad (2.3)$$

x dan x_i adalah data yang digunakan, sedangkan σ adalah *bandwith* dari *kernel Gaussian* (RBF) yang bernilai bebas akan tetapi jika terlalu kecil maka kurva peramalan yang dihasilkan menjadi sangat halus, namun jika terlalu besar maka kurva peramalan yang dihasilkan menjadi sangat kasar (Kurniasih, 2013).

Selanjutnya keluaran dari matriks *Hessian*, khususnya dari data latih, adalah nilai parameter gamma (γ) yang akan digunakan pada langkah selanjutnya yaitu penghitungan perubahan nilai *Lagrange Multiplier*. Nilai parameter gamma (γ) dihitung dengan persamaan 2.4 berikut:

$$\gamma = \frac{cLR}{\max(\text{matriks } Hessian)} \quad (2.4)$$

3. Penghitungan nilai *error*, perubahan nilai *Lagrange Multiplier*, dan nilai *Lagrange Multiplier* yang baru

Sebelumnya inialisasikan nilai *Lagrange Multiplier* α_i dan α_i^* awal sebesar 0. Selanjutnya ulangi tahap a, b, dan c berikut untuk setiap data latih

a. Penghitungan nilai *error*

$$E_i = y_i - \sum_{j=1}^n (\alpha_j^* - \alpha_j) R_{ij} \quad (2.5)$$

Keterangan:

- E_i = nilai *error* ke- i
- y_i = nilai aktual data latih ke- i
- $\alpha_j^* - \alpha_j$ = nilai *Lagrange Multiplier* terkini
- n = banyak data latih

b. Penghitungan perubahan nilai *Lagrange Multiplier*

$$\delta\alpha_i^* = \min\{\max(\gamma(E_i - \varepsilon), -\alpha_i^*), C - \alpha_i^*\} \quad (2.6)$$

$$\delta\alpha_i = \min\{\max(\gamma(-E_i - \varepsilon), -\alpha_i), C - \alpha_i\} \quad (2.7)$$

Keterangan:

- $\delta\alpha_i^*$ = perubahan nilai α_i^*
- $\delta\alpha_i$ = perubahan nilai α_i
- γ = nilai *learning rate*
- ε = nilai *epsilon*
- C = nilai kompleksitas

c. Penghitungan nilai *Lagrange Multiplier* yang baru

Nilai *Lagrange Multiplier* yang baru didapat dari nilai *Lagrange Multiplier* yang lama ditambahkan dengan perubahannya

$$\alpha_{i(\text{baru})}^* = \delta\alpha_i^* + \alpha_{i(\text{lama})}^* \quad (2.8)$$

$$\alpha_{i(\text{baru})} = \delta\alpha_i + \alpha_{i(\text{lama})} \quad (2.9)$$

4. Proses iterasi *sequential training*

Tahap (3) diulangi hingga mencapai batas iterasi maksimal yang telah diinisialisasikan di awal, atau telah memenuhi syarat konvergensi di mana $\max(|\delta\alpha_i|) < \varepsilon$ dan $\max(|\delta\alpha_i^*|) < \varepsilon$.

5. Penghitungan nilai hasil peramalan

Nilai hasil peramalan dihitung menggunakan persamaan regresi yang dirumuskan sebagai berikut:

$$f(x) = \sum_{i=1}^n (\alpha_i^* - \alpha_i) (K(x_i, x) + \lambda^2) \quad (2.10)$$

Keterangan:

- $f(x)$ = nilai hasil peramalan
- $K(x_i, x)$ = fungsi *kernel*
- λ^2 = variabel scalar



2.5 Normalisasi Data

Proses normalisasi data adalah tahap *preprocessing* yang digunakan untuk mengolah data terlebih dahulu. Tujuannya adalah untuk menyamakan standar semua data yang digunakan agar berada pada jarak tertentu (Patro & Sahu, 2015). Sebagai proses normalisasi data pada penelitian ini digunakan *Min-Max Normalization* dengan persamaan sebagai berikut:

$$x' = \frac{(x-x_{min})}{(x_{max}-x_{min})} \quad (2.11)$$

Keterangan:

- x' = nilai hasil normalisasi (nilainya berkisar antara 0 dan 1)
- x = nilai yang dinormalisasikan dari data awal
- x_{min} = nilai tertinggi dari data yang digunakan
- x_{max} = nilai terendah dari data yang digunakan

Sedangkan untuk mengembalikan nilai x' menjadi nilai yang sebenarnya dilakukan proses denormalisasi data dengan persamaan sebagai berikut:

$$x = x_{min} + (x'(x_{max} - x_{min})) \quad (2.12)$$

2.6 Nilai Evaluasi

Evaluasi bertujuan untuk mengukur dan menguji seberapa jauh model peramalan itu mampu memprediksi data dengan cara mengevaluasi kesesuaian model peramalan terhadap suatu *dataset* yang diberikan. Model peramalan yang akurat adalah yang mampu memberikan hasil peramalan dengan kesalahan sekecil mungkin.

Nilai evaluasi diukur dalam bentuk tingkat kesalahan (*error rate*) yang mana pada penelitian ini digunakan *Mean Absolute Percentage Error* (MAPE) karena memberikan hasil yang relatif terhadap nilai yang sebenarnya sehingga tingkat kesalahan yang didapat lebih stabil pada batasan prosentase. Persamaan MAPE dirumuskan sebagai berikut (Furi, et al., 2015):

$$MAPE = \frac{1}{n} \left(\sum_{i=1}^n \left| \frac{y_i - y'_i}{y_i} \right| \right) \times 100 \quad (2.13)$$

Keterangan:

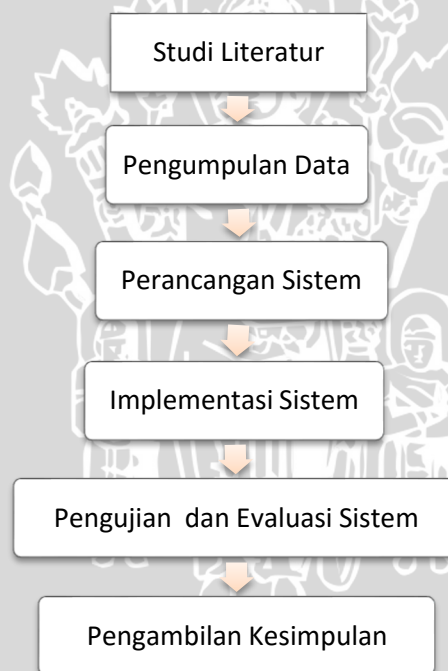
- y_i = nilai sebenarnya
- y'_i = nilai hasil peramalan
- n = banyaknya data

BAB 3 METODOLOGI

Pada bab metodologi penelitian akan dijelaskan metode yang digunakan dan langkah-langkah yang dilakukan dalam penelitian tentang meramalkan harga emas pada pasar berjangka menggunakan algoritma support vector regression. Adapun tahapannya, yaitu:

1. Mempelajari literatur yang berhubungan dengan peramalan menggunakan algoritma *Support Vector Regression* (SVR).
2. Melakukan pengumpulan data .
3. Menganalisa dan melakukan perancangan sistem .
4. Membuat sistem berdasarkan analisis dan perancangan yang telah dilakukan.
5. Melakukan uji coba terhadap sistem yang telah dibuat.
6. Mengevaluasi dan menganalisis hasil peramalan

Adapun gambar tahapannya secara umum digambarkan pada Gambar 3.1 :



Gambar 3.1 Langkah-Langkah Penelitian

3.1 Studi Literatur

Studi literatur dalam penelitian ini dilakukan dengan cara mengumpulkan dan mempelajari berbagai informasi mengenai permasalahan dan metode yang berasal dari jurnal penelitian, media internet, maupun buku. Hal ini dimaksudkan untuk mempelajari segala sesuatu yang berkaitan dengan objek dan algoritma yang digunakan pada penelitian ini sehingga mempermudah proses analisa, perancangan dan pengimplementasian.

3.2 Pengumpulan Data

Pengumpulan data dilakukan dengan mengajukan surat permohonan permintaan data skripsi kepada PT. Victory International Futures cabang Malang dan wawancara kepada beberapa staff ahli di bidang perdagangan emas berjangka.

3.3 Perancangan Sistem

Perancangan merupakan tahap untuk menggambarkan rancangan yang akan dibangun sebelum dilakukan penulisan *source code* kedalam suatu bahasa pemrograman. Rancangan yang dimaksud meliputi rancangan antarmuka pengguna, penghitungan manual untuk *sample data*, dan diagram alir proses penyelesaian masalah.

3.4 Implementasi Sistem

Implementasi dalam penelitian ini dilakukan dengan mengacu pada perancangan yang telah dibuat. Implementasi aplikasi dilakukan dengan menggunakan bahasa pemrograman Java dan tools pendukung lainnya. Implementasi peramalan harga emas pada perdagangan berjangka meliputi:

- Penerapan algoritma SVR dalam program yang dibuat dengan menggunakan bahasa pemrograman Java.
- Pembuatan antarmuka program

3.5 Pengujian Sistem

Pengujian ini merupakan pengujian terhadap aplikasi yang sudah diimplementasikan, pengujian ini menggunakan beberapa teknik-teknik pengujian, dengan tujuan menguji aplikasi apakah sudah bekerja sesuai dengan yang diharapkan sehingga tidak ditemukan lagi kesalahan-kesalahan pada aplikasi. Salah satu uji coba yang akan dilakukan adalah membandingkan pendapatan investor pada tahun tertentu dengan dan tanpa menggunakan peramalan ini.

3.6 Pengambilan Kesimpulan

Pada tahapan ini dilakukan analisa hasil dari pengujian yang telah dilakukan sehingga dapat mengevaluasi hal-hal yang terjadi setelah semua tahapan penelitian mulai dari perancangan, implementasi dan pengujian sistem selesai dilakukan dan didasarkan pada kesesuaian antara teori dan praktik. Kesimpulan diambil untuk menjawab rumusan masalah yang telah ditentukan..

BAB 4 PERANCANGAN SISTEM

Pada bab ini akan dibahas perancangan sistem yang dimulai dari formulasi permasalahan, siklus peramalan harga emas menggunakan algoritma *Support Vector Regression* (SVR), perancangan uji coba, dan perancangan antarmuka pengguna

4.1 Formulasi Permasalahan

Permasalahan yang akan diselesaikan adalah peramalan harga emas dengan masukan berupa data harga emas dari pengguna. Data harga tersebut akan diolah menggunakan algoritma *Support Vector Regression* (SVR), dari hasil pengolahan tersebut akan menghasilkan data harga yang diramalkan dan *error rate*. Dari hasil pengolahan tersebut akan didapatkan nilai evaluasi yang dihitung menggunakan *Mean Absolute Percentage Error* (MAPE).

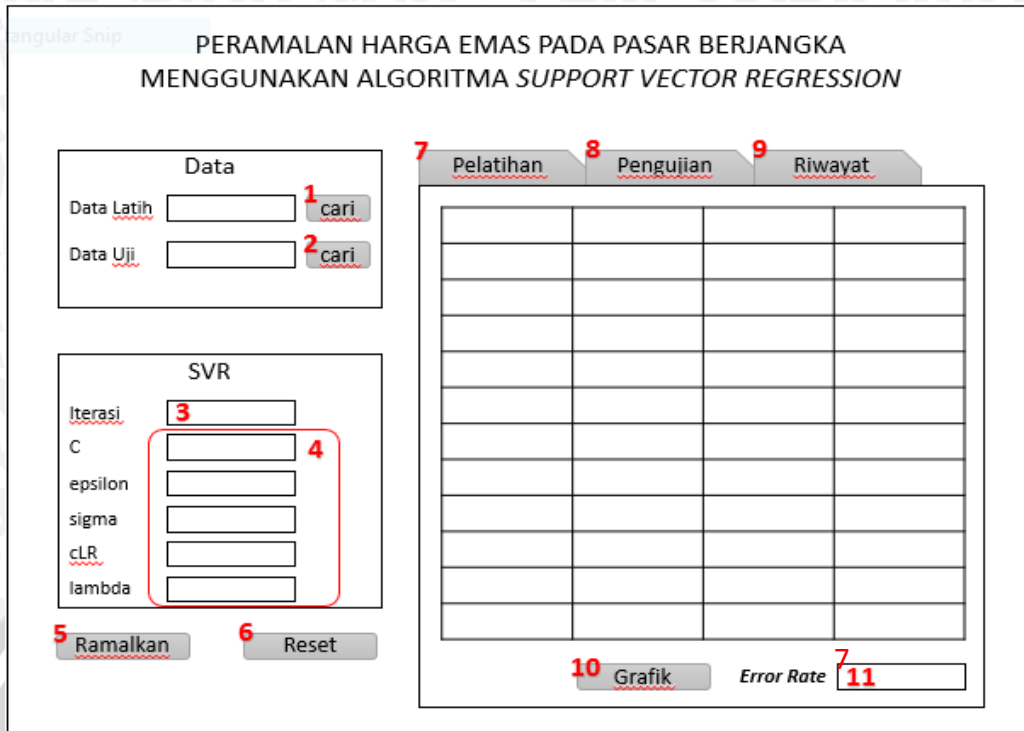
Adapun sampel data harga emas yang digunakan pada penelitian ini berjumlah 14 *record* dari 28 Januari sampai 14 Februari 2014 sebagaimana ditunjukkan pada Tabel 4.1 berikut: (data lengkap disertakan pada Lampiran)

Tabel 4. 1 Data Harga Emas

Tanggal	Harga	Tanggal	Harga
28/01/2014	1253,50	06/02/2014	1258,50
29/01/2014	1254,75	07/02/2014	1260,00
30/01/2014	1254,00	10/02/2014	1273,50
31/01/2014	1246,50	11/02/2014	1282,75
03/02/2014	1246,50	12/02/2014	1286,50
04/02/2014	1253,00	13/02/2014	1290,25
05/02/2014	1257,00	14/02/2014	1308,50

4.2 Perancangan Antarmuka Pengguna

Antarmuka sistem ini terdiri dari satu halaman dengan dua *tab* di dalamnya. Dari dua *tab* tersebut yang pertama berisi data dan hasil pelatihan, sedangkan yang kedua berisi data dan hasil pengujian. Perancangan antarmuka pengguna sistem yang dimaksud ditunjukkan pada Gambar 4.1 pada halaman berikut,



Gambar 4. 1 Rancangan Antarmuka Pengguna

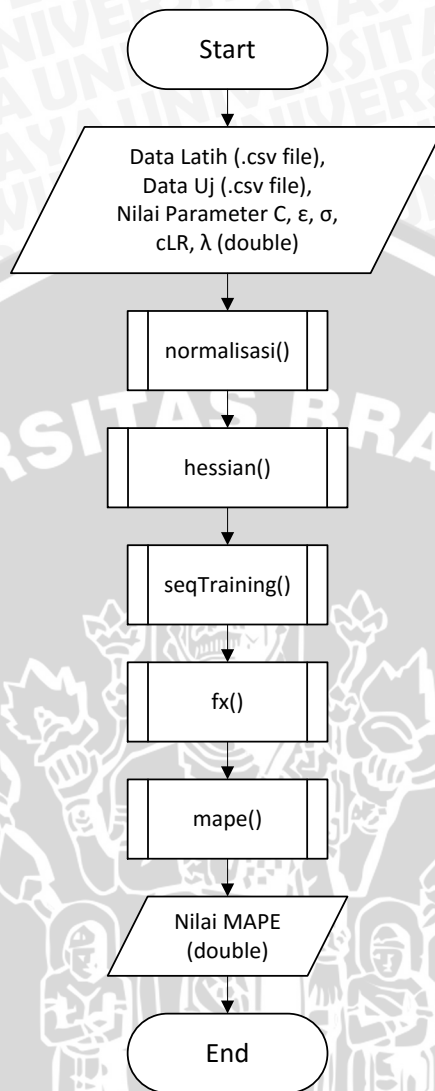
Keterangan:

1. Tombol pencarian *file* data latih
2. Tombol pencarian *file* data uji
3. *Field* untuk memasukkan jumlah iterasi pelatihan SVR
4. *Field* untuk memasukkan nilai parameter
5. Tombol untuk memulai proses peramalan
6. Tombol untuk mengembalikan aplikasi ke keadaan sebelum dilakukan peramalan
7. *Tab* data dan hasil pelatihan
8. *Tab* data dan hasil pengujian
9. *Tab* riwayat hasil peramalan
10. *Tombol* untuk menampilkan grafik peramalan
11. *Field* yang menampilkan nilai *error rate* MAPE

4.3 Siklus Algoritma Support Vector Regression (SVR)

Siklus algoritma SVR adalah urutan penyelesaian masalah menggunakan algoritma *Support Vector Regression* (SVR). Ada beberapa parameter yang digunakan di sini yaitu kompleksitas (C), epsilon (ϵ), sigma (σ) untuk Kernel Gaussian RBF, *constant learning rate* (cLR), dan lambda (λ). Diagram alir

penyelesaian masalah peramalan harga emas menggunakan algoritma SVR secara garis besar digambarkan pada Gambar 4.2,



Gambar 4. 2 Diagram Alir Algoritma *Support Vector Regression*

1. Proses Inisialisasi Parameter

Dari diagram alir di atas disebutkan langkah pertama adalah inisialisasi parameter. Nilai parameter yang digunakan ditentukan secara eksplisit. Sebagai contoh pada perhitungan manual nilai parameter yang didapat ditunjukkan pada Tabel 4.2,

Tabel 4. 2 Inisialisasi Parameter SVR

C	Epsilon (ϵ)	Sigma(σ)	cLR	Lambda (λ)
10	0,0001	0,7	0,08	0,8

2. Proses Pemilihan Fitur

Pada studi kasus ini akan digunakan 4 data sekuensial sebelum tanggal yang ditargetkan sebagai fitur, kemudian dibagi menjadi 2 kelompok data yaitu data latih dan data uji sebagaimana ditunjukkan pada Tabel 4.3

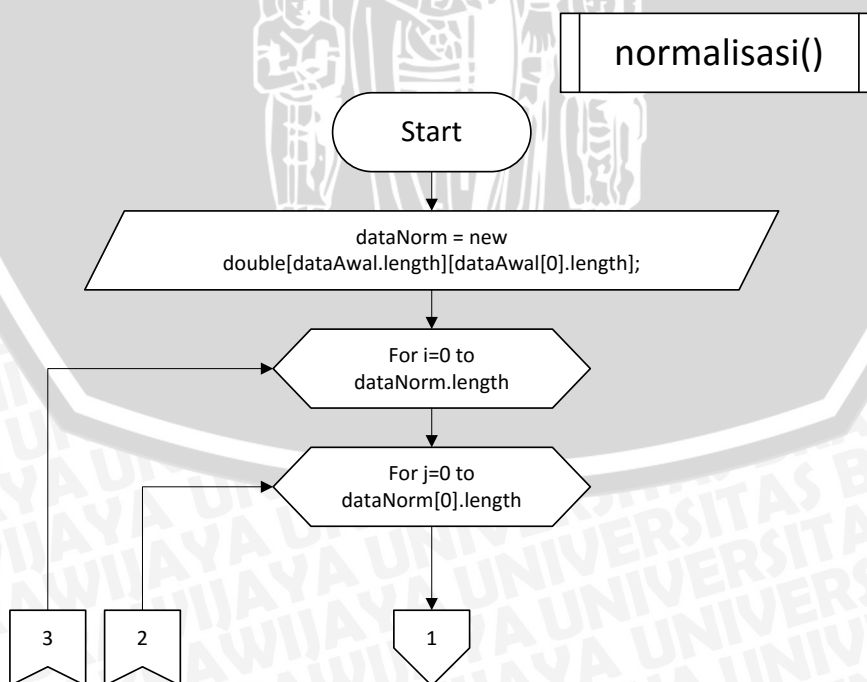
Tabel 4.3 Fitur yang digunakan

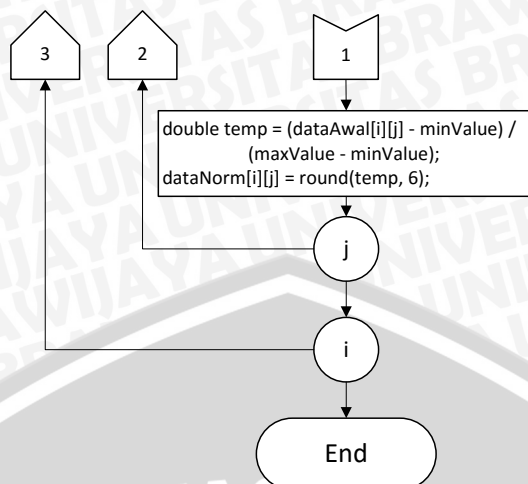
Tanggal	x1	x2	x3	x4	Y
03/02/2014	1253,50	1254,75	1254,00	1246,50	1246,50
04/02/2014	1254,75	1254,00	1246,50	1246,50	1253,00
05/02/2014	1254,00	1246,50	1246,50	1253,00	1257,00
06/02/2014	1246,50	1246,50	1253,00	1257,00	1258,50
07/02/2014	1246,50	1253,00	1257,00	1258,50	1260,00
10/02/2014	1253,00	1257,00	1258,50	1260,00	1273,50
11/02/2014	1257,00	1258,50	1260,00	1273,50	1282,75
12/02/2014	1258,50	1260,00	1273,50	1282,75	1286,50
13/02/2014	1260,00	1273,50	1282,75	1286,50	1290,25
14/02/2014	1273,50	1282,75	1286,50	1290,25	1308,50

Dari data fitur tersebut, data latih yang digunakan adalah data dari tanggal 3 sampai 7 Februari 2014, sedangkan data uji dari tanggal 10 sampai 14 Februari 2014.

3. Normalisasi Data Latih dan Data Uji

Data latih dan data uji yang didapat selanjutnya dinormalisasikan menggunakan metode *Min-Max Normalization*. Diagram alir dari proses normalisasi ditunjukkan pada Gambar 4.3,





Gambar 4. 3 Diagram Alir Proses Normalisasi Min-Max

Sesuai persamaan (2.11), nilai minimal dan maksimal diperoleh secara global dari data harga emas keseluruhan yaitu Min = 1050,60 dan Max = 1379,00. Dari nilai tersebut diberikan selisih 5 di luar rentang min-max untuk mengantisipasi adanya nilai min atau max yang termuat pada data yang digunakan sebagai data latih atau data uji. Dengan pemberian selisih ini maka didapat nilai Min = 1045,60 dan Max = 1384,00.

Sebagai contoh hasil normalisasi data harga untuk data latih pada fitur x1 (1253,50) didapatkan dengan cara berikut,

$$x' = \frac{(1253,50 - 1045,60)}{(1384,00 - 1045,60)} = 0,6144$$

Selengkapnya hasil normalisasi untuk data latih dan data uji ditunjukkan pada Tabel 4.4 dan Tabel 4.5,

Tabel 4.4 Hasil Normalisasi Data Latih

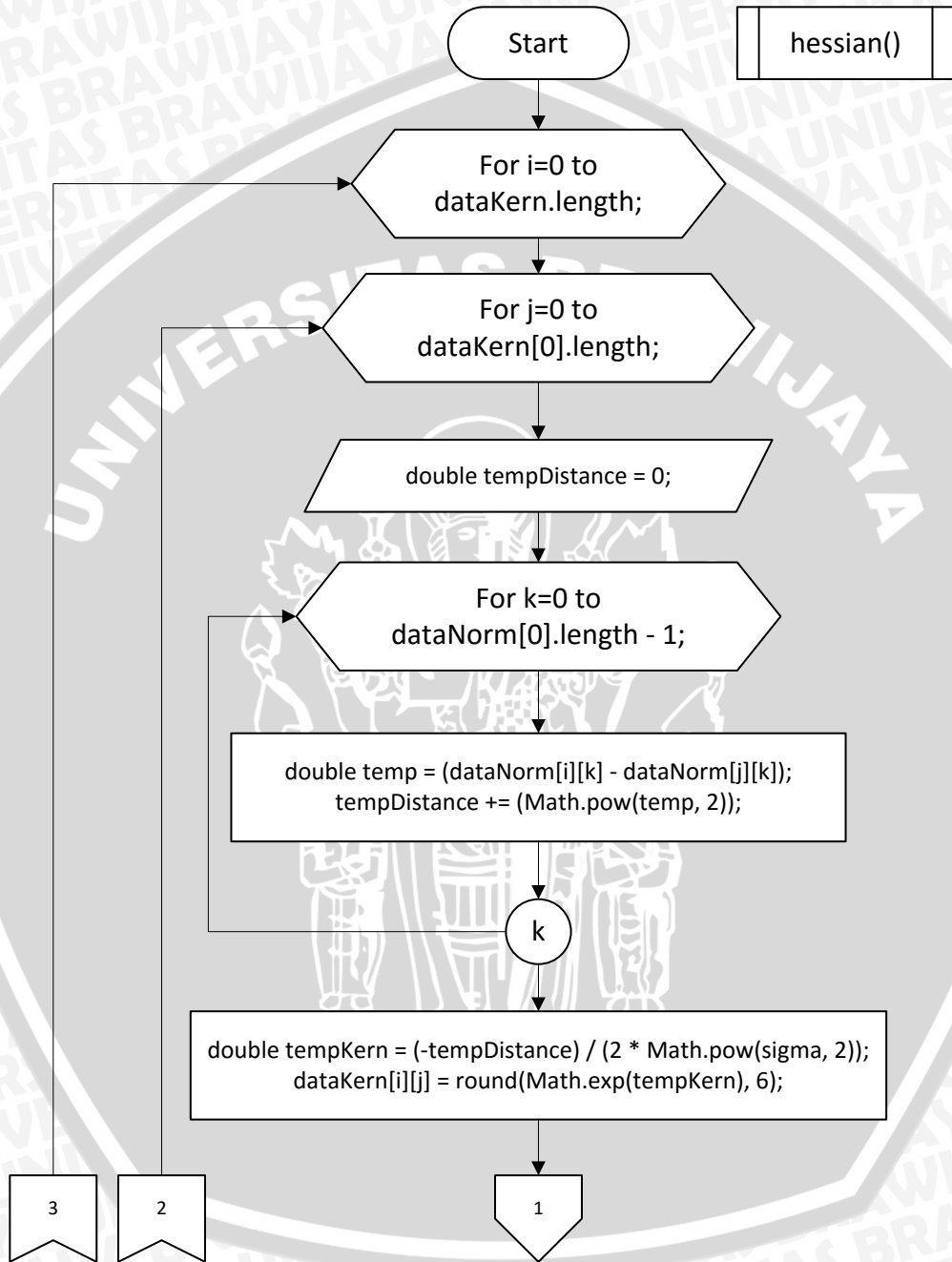
TANGGAL	X1	X2	X3	X4	Y
03/02/2014	0,6144	0,6181	0,6158	0,5937	0,5937
04/02/2014	0,6181	0,6158	0,5937	0,5937	0,6129
05/02/2014	0,6158	0,5937	0,5937	0,6129	0,6247
06/02/2014	0,5937	0,5937	0,6129	0,6247	0,6291
07/02/2014	0,5937	0,6129	0,6247	0,6291	0,6336

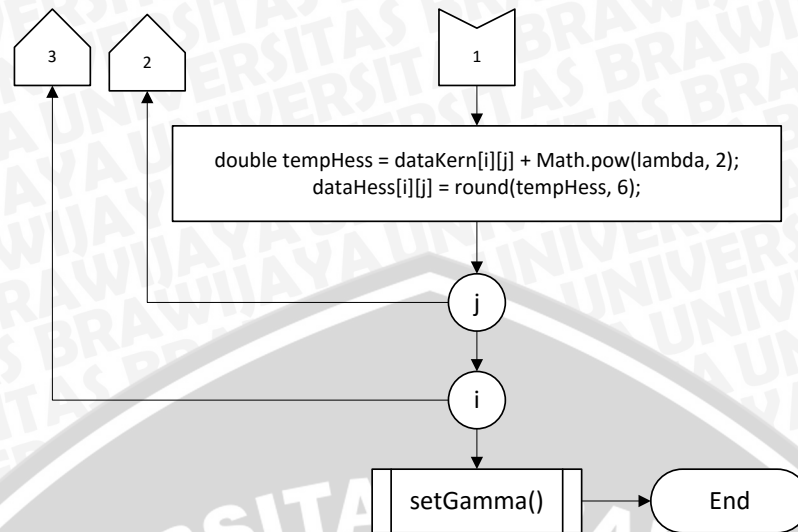
Tabel 4.5 Hasil Normalisasi Data Uji

TANGGAL	X1	X2	X3	X4	Y
10/02/2014	0,6129	0,6247	0,6291	0,6336	0,6735
11/02/2014	0,6247	0,6291	0,6336	0,6735	0,7008
12/02/2014	0,6291	0,6336	0,6735	0,7008	0,7119
13/02/2014	0,6336	0,6735	0,7008	0,7119	0,7230
14/02/2014	0,6735	0,7008	0,7119	0,7230	0,7769

4. Menghitung Matriks *Hessian*

Langkah kedua adalah menghitung matriks *Hessian*. Namun sebelum itu hitung dahulu matriks *kernel*-nya. Diagram alir penghitungan *kernel* dan matriks *Hessian* ditunjukkan pada Gambar 4.4,





Gambar 4. 4 Diagram Alir Menghitung kernel dan Matriks Hessian

Sesuai persamaan (2.3), menghitung nilai *kernel* dimulai dengan menghitung jarak data dengan data latih. Sebagai contoh untuk nilai *kernel* data latih pada index (1,2) dihitung jarak antara fitur data latih ke-1 dengan ke-2 yang telah dinormalisasikan,

$$\begin{aligned} \|x - x_i\|^2 &= (0,6144 - 0,6181)^2 + (0,6181 - 0,6158)^2 \\ &\quad + (0,6158 - 0,5937)^2 + (0,5937 - 0,5937)^2 \\ &= 0,0005 \end{aligned}$$

Setelah didapat nilai jarak tersebut, kemudian dihitung nilai *kernel*-nya,

$$K(x, x_i) = \exp\left(-\frac{0,0005}{2(0,7^2)}\right) = 0,9995$$

Selengkapnya matriks *kernel* untuk data latih dan data uji ditunjukkan pada Tabel 4.6 dan Tabel 4.7,

Tabel 4.6 Matriks Kernel Data Latih

KIJ	1	2	3	4	5
1	1	0,9995	0,9985	0,9980	0,9982
2	0,9995	1	0,9991	0,9975	0,9971
3	0,9985	0,9991	1	0,9990	0,9979
4	0,9980	0,9975	0,9990	1	0,9995
5	0,9982	0,9971	0,9979	0,9995	1

Tabel 4.7 Matriks Kernel Data Uji

KIJ	1	2	3	4	5
1	0,9981	0,9970	0,9973	0,9983	0,9994
2	0,9930	0,9917	0,9933	0,9949	0,9967
3	0,9846	0,9815	0,9839	0,9875	0,9907
4	0,9752	0,9708	0,9719	0,9765	0,9819
5	0,9637	0,9591	0,9591	0,9627	0,9694

Setelah didapatkan nilai *kernel*-nya maka dapat dihitung nilai *Hessian*-nya sesuai persamaan (2.2),

$$[R]_{ij} = 0,9995 + 0,8^2 = 1,6395$$

Selengkapnya matriks *Hessian* untuk data latih dan data uji ditunjukkan pada Tabel 4.8 dan Tabel 4.9,

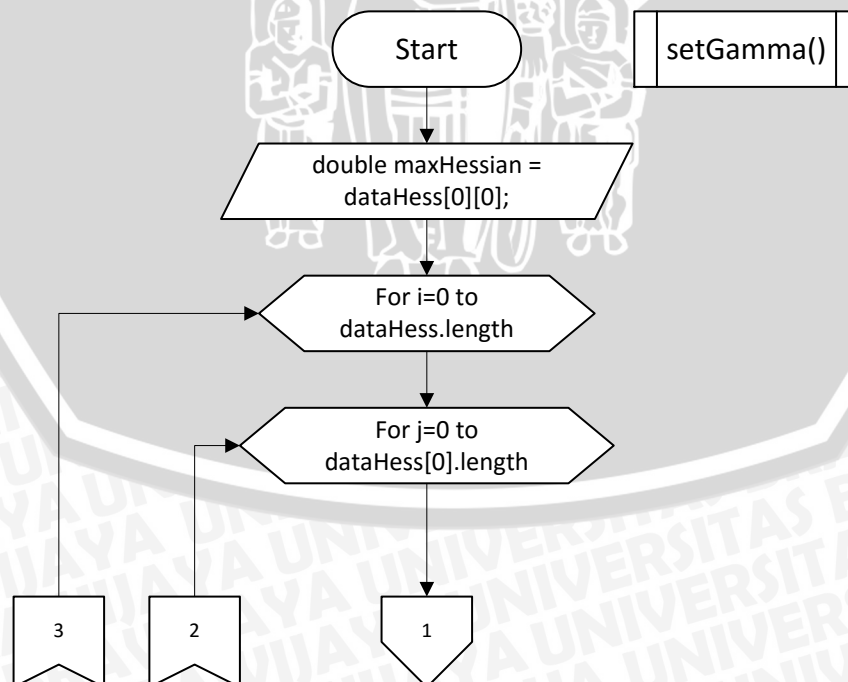
Tabel 4.8 Matriks *Hessian* Data Latih

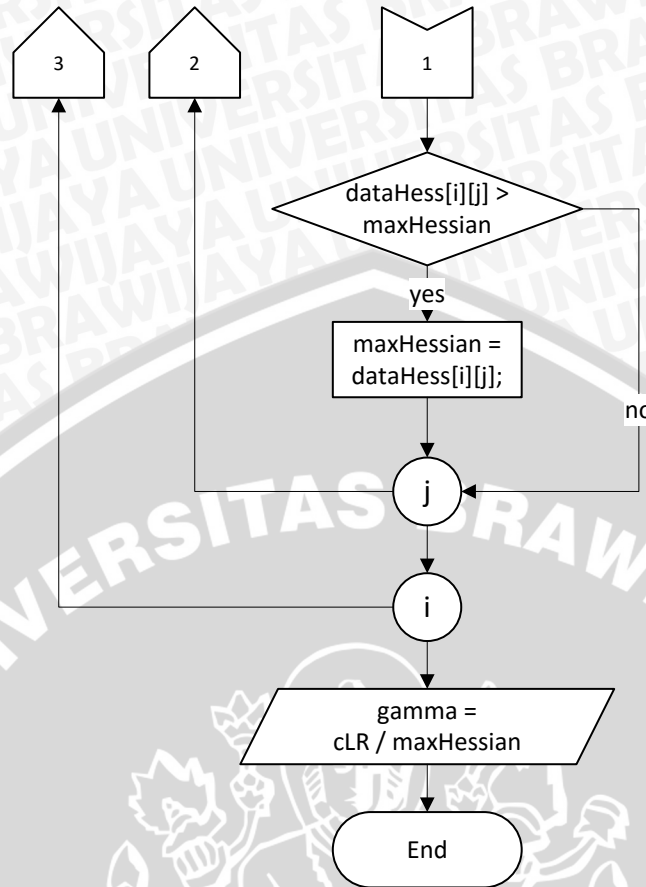
RIJ	1	2	3	4	5
1	1,64	1,6395	1,6385	1,6380	1,6382
2	1,6395	1,64	1,6391	1,6375	1,6371
3	1,6385	1,6391	1,64	1,6390	1,6379
4	1,6380	1,6375	1,6390	1,64	1,6395
5	1,6382	1,6371	1,6379	1,6395	1,64

Tabel 4.9 Matriks *Hessian* Data Uji

RIJ	1	2	3	4	5
1	1,6381	1,6370	1,6373	1,6383	1,6394
2	1,6330	1,6317	1,6333	1,6349	1,6367
3	1,6246	1,6215	1,6239	1,6275	1,6307
4	1,6152	1,6108	1,6119	1,6165	1,6219
5	1,6037	1,5991	1,5991	1,6027	1,6094

Setelah didapat matriks *Hessian* maka diperoleh nilai parameter gamma (γ) untuk data latih dengan diagram alir pada Gambar 4.5,





Gambar 4. 5 Diagram Alir Menghitung nilai parameter γ

sesuai persamaan (2.4) maka didapatkan nilai parameter gamma,

$$\gamma_{latih} = \frac{0,08}{1,64} = 0,04878$$

5. Proses *Sequential Learning*

Seperti dijelaskan pada subbab 2.4, langkah ke-3, terlebih dahulu inialisasikan nilai *Lagrange Multiplier* α_i^* dan α_i di awal sebesar 0 sejumlah banyaknya data latih yang digunakan seperti dicontohkan pada Tabel 4.10,

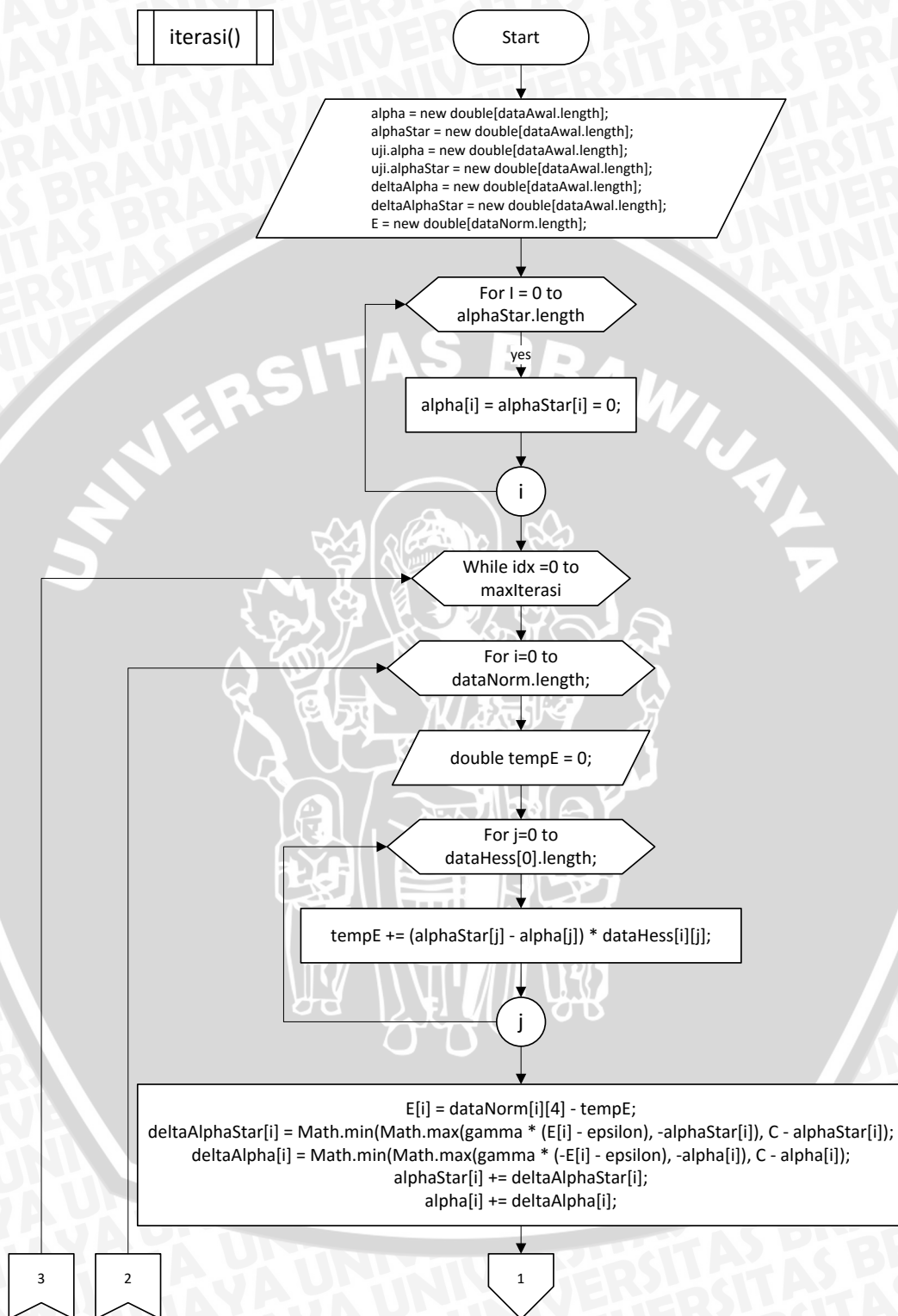
Tabel 4.10 Nilai Awal *Lagrange Multiplier*

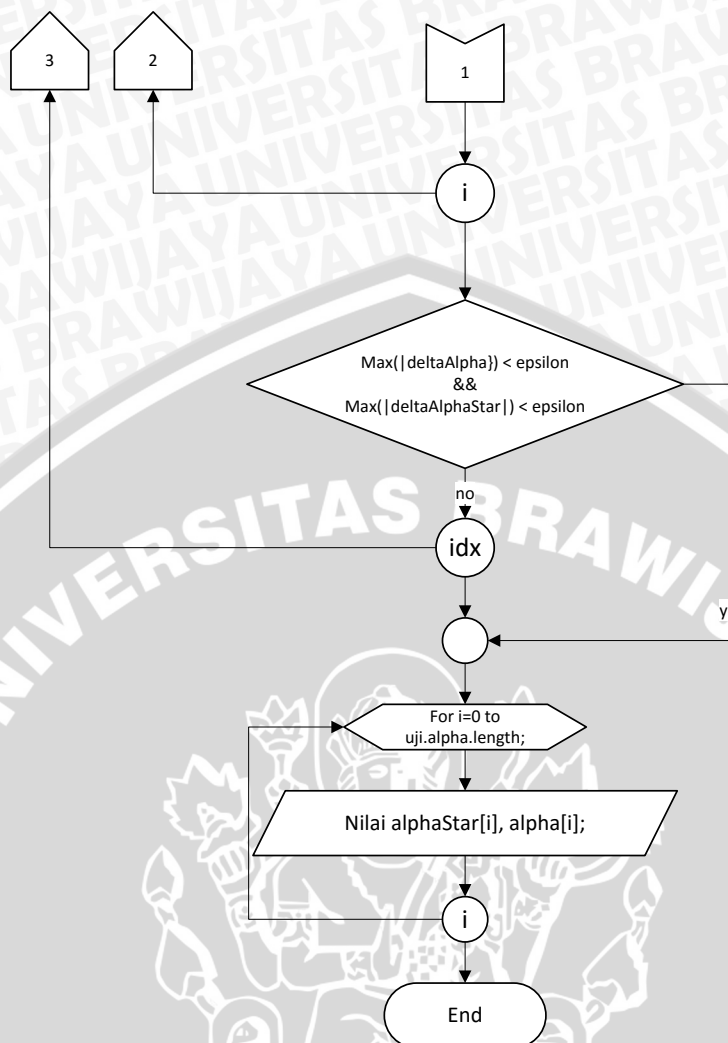
A_i^*	A_i
0	0
0	0
0	0
0	0
0	0

Proses ini merupakan proses utama pada SVR yang mana keluaran dari proses ini adalah nilai *Lagrange Multiplier*. Proses ini diulang sampai pada iterasi maksimal yang diinisialisasikan atau telah mencapai konvergensi (memenuhi



syarat $\max(|\delta\alpha_i|) < \epsilon$ dan $\max(|\delta\alpha_i^*|) < \epsilon$). Diagram alir proses iterasi ini ditunjukkan pada Gambar 4.5,





Gambar 4. 6 Diagram Alir Proses Iterasi *Sequential Training*

Sebagai contoh untuk index pertama dan iterasi pertama, nilai *Error Rate* sesuai persamaan (2.5) diperoleh dengan cara,

$$\begin{aligned}
 E_1 &= y_1 - \sum_{j=1}^n (\alpha_j^* - \alpha_j) R_{1j} \\
 &= 0,5937 - ((0 - 0)1,64 + (0 - 0)1,6395 + (0 - 0)1,6385 \\
 &\quad + (0 - 0)1,6380 + (0 - 0)1,6382) \\
 &= 0,5937
 \end{aligned}$$

Sedangkan nilai $\delta\alpha_i^*$ sesuai persamaan (2.6) dan $\delta\alpha_i$ (2.7) diperoleh dengan cara,

$$\begin{aligned}
 \delta\alpha_i^* &= \min\{\max(\gamma(E_i - \varepsilon), -\alpha_i^*), C - \alpha_i^*\} \\
 &= \min\{\max(0,04878 \times (0,5937 - 0,0001), -0), 10 - 0\} \\
 &= 0,029
 \end{aligned}$$

$$\delta\alpha_i = \min\{\max(\gamma(-E_i - \varepsilon), -\alpha_i), C - \alpha_i\}$$

$$= \min\{ \max(0,04878 \times (-0,5937 - 0,0001), -0), 10 - 0 \}$$

$$= 0$$

Kemudian nilai *Lagrange Multiplier Multiplier* α_i^* dan α_i sesuai persamaan (2.8) dan (2.9) diperoleh dengan cara,

$$\alpha_{i(\text{baru})}^* = \delta\alpha_i^* + \alpha_{i(\text{lama})}^*$$

$$= 0,0265 + 0$$

$$= 0,029$$

$$\alpha_{i(\text{baru})} = \delta\alpha_i + \alpha_{i(\text{lama})}$$

$$= 0 + 0$$

$$= 0$$

Hasil iterasi pertama dan ke-10 dari penghitungan manual ditunjukkan pada Tabel 4.11 dan Tabel 4.12,

Tabel 4.11 Hasil Iterasi Pertama

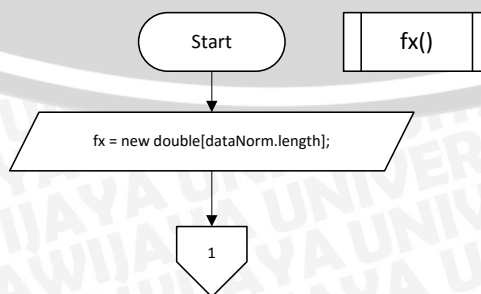
HARGA	E	$\delta\alpha_i^*$	$\delta\alpha_i$	α_i^*	α_i
0,5937	0,5937	0,0290	0,0000	0,0290	0,0000
0,6129	0,6129	0,0299	0,0000	0,0299	0,0000
0,6247	0,6247	0,0305	0,0000	0,0305	0,0000
0,6291	0,6291	0,0307	0,0000	0,0307	0,0000
0,6336	0,6336	0,0309	0,0000	0,0309	0,0000

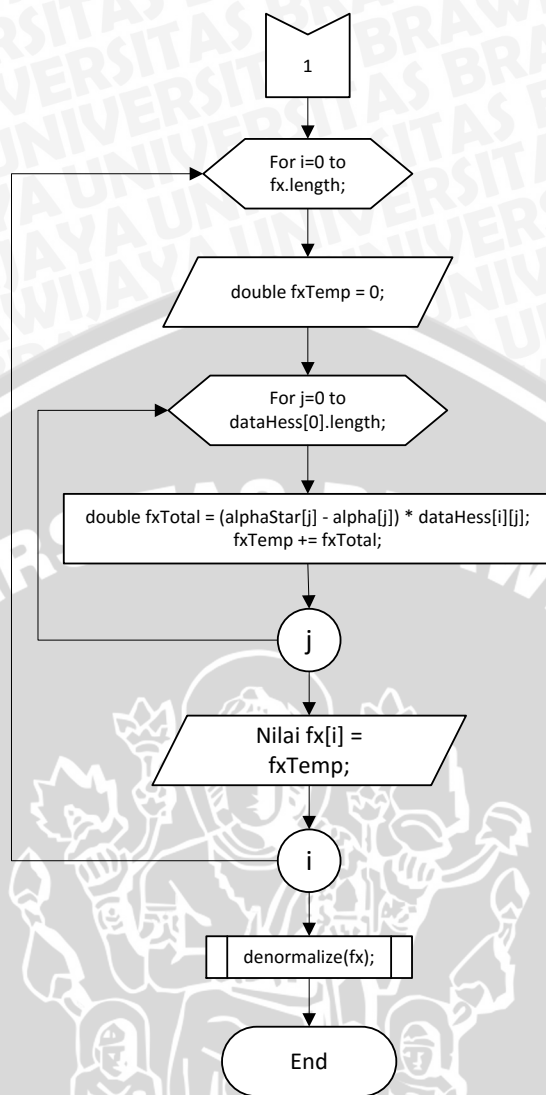
Tabel 4.12 Hasil Iterasi ke-10

HARGA	E	$\delta\alpha_i^*$	$\delta\alpha_i$	α_i^*	α_i
0,5937	-0,0173	-0,0008	0,0010	0,0629	0,0021
0,6129	0,0020	0,0001	0,0000	0,0723	0,0000
0,6247	0,0137	0,0007	0,0000	0,0780	0,0000
0,6291	0,0181	0,0009	0,0000	0,0802	0,0000
0,6336	0,0227	0,0011	0,0000	0,0824	0,0000

6. Penghitungan Nilai Peramalan dan Proses Evaluasi

Proses peramalan (menentukan nilai peramalan $f(x)$) ditunjukkan pada diagram alir berikut,



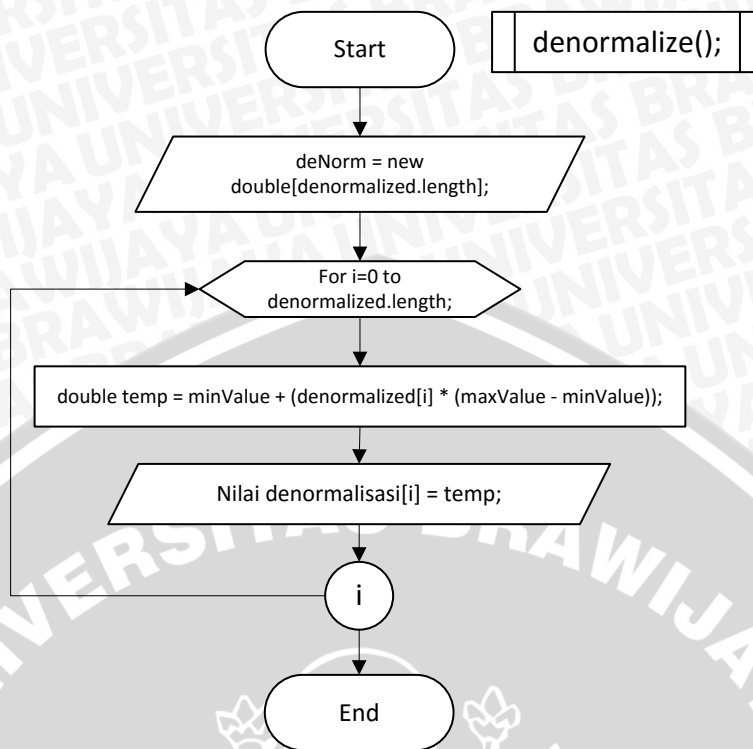


Gambar 4. 7 Diagram Alir Menentukan Nilai f(x)

Sebagai contoh pada index pertama, nilai *Lagrange Multiplier* terakhir yang didapat dari tahap sebelumnya digunakan untuk menghitung nilai peramalan sesuai persamaan (2.10),

$$\begin{aligned}
 f(x) &= \sum_{i=1}^n (\alpha_i^* - \alpha_i)(K(x_i, x) + \lambda^2) \\
 &= \sum_{i=1}^n (\alpha_i^* - \alpha_i) * [R]_{ij} \\
 &= (0,0629 - 0,0021)1,64 + (0,0723 - 0)1,6395 + (0,078 - 0)1,6385 \\
 &\quad + (0,0802 - 0)1,6380 + (0,0824 - 0)1,6382 \\
 &= 0,6123
 \end{aligned}$$

dan selanjutnya dilakukan proses denormalisasi sebagaimana ditunjukkan pada diagram alir berikut,



Gambar 4. 8 Diagram Alir Proses Denormalisasi

sesuai persamaan (2.12) diperoleh nilai denormalisasi sebagai berikut,

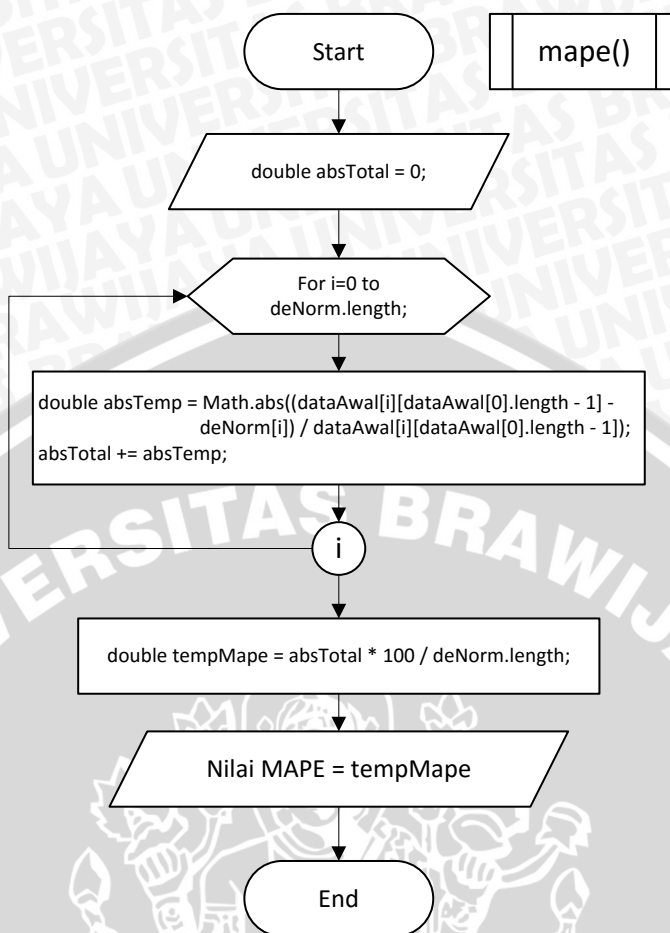
$$\begin{aligned}
 x &= x_{min} + (x'(x_{max} - x_{min})) \\
 &= 1045,6 + (0,6123(1384,0 - 1045,6)) \\
 &= 1252,8164
 \end{aligned}$$

Hasilnya dapat dilihat pada Tabel 4.13,

Tabel 4.13 Hasil Peramalan pada Data Latih

Normalisasi		Denormalisasi		Error Rate
Harga	f(x)	Harga	f(x)	Y-fx / Y
0,5937	0,6123	1246,50	1252,8164	0,0051
0,6129	0,6123	1253,00	1252,7933	0,0002
0,6247	0,6124	1257,00	1252,8352	0,0033
0,6291	0,6124	1258,50	1252,8303	0,0045
0,6336	0,6123	1260,00	1252,7957	0,0057

Terakhir dapat dihitung nilai evaluasi MAPE berdasarkan persamaan (2.13) yang alurnya ditunjukkan pada diagram alir Gambar 4.9,



Gambar 4. 9 Diagram Alir Proses menghitung MAPE

$$\begin{aligned}
 \text{MAPE} &= \frac{1}{n} \left(\sum_{i=1}^n \left| \frac{y_i - y'_i}{y_i} \right| \right) \times 100 \\
 &= \frac{1}{5} (0,0051 + 0,0002 + 0,0033 + 0,0045 + 0,0057) \times 100 \\
 &= 0,3754
 \end{aligned}$$

Nilai *Lagrange Multiplier* terakhir yang didapat dari tahap pelatihan diaplikasikan pada data uji dan hasilnya ditunjukkan pada Tabel 4.14, nilai *error rate* MAPE yang didapatkan adalah sebesar 2,8977.

Tabel 4.14 Hasil Peramalan pada Data Uji

Normalisasi		Denormalisasi		Error Rate
Harga	f(x)	Harga	f(x)	Y-f(x) / Y
0,6735	0,6121	1273,50	1252,7293	0,0163
0,7008	0,6106	1282,75	1252,2177	0,0238
0,7119	0,6075	1286,50	1251,1816	0,0275
0,7230	0,6036	1290,25	1249,8662	0,0313
0,7769	0,5989	1308,50	1248,2789	0,046

4.4 Perancangan Uji Coba dan Evaluasi

Perancangan ini dibuat untuk menguji dan mengevaluasi hasil peramalan harga emas dengan algoritma *Support Vector Regression*. Evaluasi sistem yang diterapkan di antaranya:

1. Uji coba jumlah iterasi untuk pelatihan SVR
2. Uji coba nilai parameter C (*Complexity*) SVR
3. Uji coba nilai parameter ε (*epsilon*) SVR
4. Uji coba nilai parameter σ (*sigma*) Kernel Gaussian RBF
5. Uji coba nilai parameter cLR SVR
6. Uji coba nilai parameter λ (*lambda*) SVR
7. Uji coba variasi jumlah data latih
8. Uji coba variasi jumlah data uji

Pengujian ini perlu dilakukan untuk mengetahui nilai parameter, jumlah iterasi, banyaknya data latih, dan banyaknya data uji yang sesuai dengan dataset yang digunakan. Untuk pengujian pada poin 1 sampai 6 digunakan variasi komposisi data latih dan data uji yang berbeda namun jumlah yang sama yaitu 30 data latih dan 10 data uji, hal ini ditujukan untuk memastikan bahwa jumlah iterasi dan nilai parameter yang diujikan dapat memberikan nilai MAPE yang stabil untuk berbagai variasi data.

4.4.1 Uji Coba Jumlah Iterasi untuk Pelatihan SVR

Uji coba jumlah iterasi digunakan untuk menentukan jumlah iterasi pelatihan SVR yang optimal sehingga didapatkan nilai *lagrange multiplier* (α_i^* dan α_i) yang memberikan hasil peramalan terbaik. Pengujian ini menggunakan 9 variasi iterasi, di mana setiap variasi akan diuji sebanyak 10 kali dengan nilai parameter yang sama dan variasi komposisi data. Rancangan variasi jumlah iterasi pelatihan SVR yang digunakan ditunjukkan pada Tabel 4.15,

Tabel 4. 15 Rancangan Uji Coba Jumlah Iterasi

Banyak Iterasi	<i>Error Rate Percobaan ke – i</i>										Rata-rata <i>Error Rate</i>
	1	2	3	4	5	6	7	8	9	10	
10											
50											
100											
500											
1000											
5000											
10000											
50000											
100000											

4.4.2 Uji Coba Nilai Parameter C (Complexity)

Pencarian nilai parameter C (Complexity) SVR bertujuan untuk menemukan nilai terbaiknya sehingga memberikan hasil peramalan yang lebih baik. Pada penelitian ini akan dilakukan pengujian pada 9 variasi nilai yang masing-masing akan diuji sebanyak 10 kali dengan variasi komposisi data, rancangan pengujiannya ditunjukkan pada Tabel 4.16,

Tabel 4. 16 Rancangan Uji Coba Nilai Parameter C (Complexity)

Nilai C	Error Rate Percobaan ke - i										Rata-rata Error Rate	
	1	2	3	4	5	6	7	8	9	10		
0,05												
0,1												
0,5												
1												
5												
10												
50												
100												
500												

4.4.3 Uji Coba Nilai Parameter ϵ (epsilon) SVR

Pencarian nilai parameter ϵ (epsilon) SVR bertujuan untuk menemukan nilai terbaiknya sehingga memberikan hasil peramalan yang lebih baik. Pengujian akan dilakukan pada 9 variasi nilai yang masing-masing akan diuji sebanyak 10 kali dengan variasi komposisi data, rancangan pengujiannya ditunjukkan pada Tabel 4.17,

Tabel 4. 17 Rancangan Uji Coba Nilai Parameter ϵ (epsilon)

Nilai ϵ	Error Rate Percobaan ke - i										Rata-rata Error Rate	
	1	2	3	4	5	6	7	8	9	10		
1,E-11												
5,E-11												
1,E-10												
5,E-10												
1,E-09												
5,E-09												
1,E-08												
5,E-08												
1,E-07												

4.4.4 Uji Coba Nilai Parameter σ (*sigma*) Kernel Gaussian RBF

Pencarian nilai parameter σ (*sigma*) Kernel Gaussian RBF bertujuan untuk menemukan nilai terbaiknya sehingga memberikan hasil peramalan yang lebih baik. Pengujian yang akan dilakukan terdiri dari 9 variasi nilai yang masing-masing akan diuji sebanyak 10 kali dengan variasi komposisi data, rancangan pengujiannya ditunjukkan pada Tabel 4.18,

Tabel 4. 18 Rancangan Uji Coba Nilai Parameter σ (*sigma*)

Nilai σ	Error Rate Percobaan ke - <i>i</i>										Rata-rata Error Rate	
	1	2	3	4	5	6	7	8	9	10		
0,05												
0,1												
0,5												
1												
5												
10												
50												
100												
500												

4.4.5 Uji Coba Nilai Parameter *cLR* SVR

Pencarian nilai parameter *cLR* SVR bertujuan untuk menemukan nilai terbaiknya sehingga memberikan hasil peramalan yang lebih baik. Terdapat 9 variasi nilai optimal yang masing-masing akan diuji sebanyak 10 kali dengan variasi komposisi data, rancangan pengujiannya ditunjukkan pada Tabel 4.19,

Tabel 4.19 Rancangan Uji Coba Nilai Parameter *cLR*

Nilai <i>cLR</i>	Error Rate Percobaan ke - <i>i</i>										Rata-rata Error Rate	
	1	2	3	4	5	6	7	8	9	10		
0,0001												
0,0005												
0,001												
0,005												
0,01												
0,05												
0,1												
0,5												
1												

4.4.6 Uji Coba Nilai Parameter λ (*lambda*) SVR

Pencarian nilai parameter λ (*lambda*) SVR bertujuan untuk menemukan nilai terbaiknya sehingga memberikan hasil peramalan yang lebih baik. Pengujian akan dilakukan pada 9 variasi nilai parameter λ (*lambda*) yang masing-masing akan diuji sebanyak 10 kali dengan variasi komposisi data, rancangan pengujiannya ditunjukkan pada Tabel 4.20,

Tabel 4.20 Rancangan Uji Coba Nilai Parameter λ (*lambda*)

Nilai λ	Error Rate Percobaan ke - <i>i</i>										Rata-rata Error Rate
	1	2	3	4	5	6	7	8	9	10	
0,001											
0,005											
0,01											
0,05											
0,1											
0,5											
1											
5											
10											

4.4.7 Uji Coba Variasi Jumlah Data Latih

Uji coba ini akan membandingkan *error rate* hasil peramalan dengan variasi banyaknya data latih yang kesemuanya digunakan untuk meramalkan satu bulan yang sama. Variasi pada uji coba ini menggunakan data harga 10 sampai 100 hari sebelum awal bulan Desember 2015, variasi data harga tersebut akan digunakan untuk meramalkan harga pada 10 hari pertama pada bulan Desember 2015. Rancangan uji coba variasi jumlah data latih ditunjukkan pada Tabel 4.21,

Tabel 4.21 Rancangan Uji Coba Variasi Data Latih

Jumlah Data Latih (hari)	Nilai Error Rate
10	
20	
30	
40	
50	
60	
70	
80	
90	
100	

4.4.8 Uji Coba Variasi Jumlah Data Uji

Uji coba ini akan membandingkan hasil *error rate* pada variasi banyaknya data uji yang diramalkan dengan jumlah data latih yang sama. Variasi pada uji coba ini menggunakan data harga 10 sampai 100 hari setelah akhir bulan Agustus 2014 sebagai data uji, variasi data harga tersebut akan diramalkan menggunakan data latih sejumlah 80 hari terakhir sebelum akhir bulan Agustus 2014. Rancangan uji coba variasi jumlah data uji ditunjukkan pada Tabel 4.22,

Tabel 4.22 Rancangan Uji Coba Variasi Data Uji

Jumlah Data Uji (hari)	Nilai <i>Error Rate</i>
10	
15	
20	
25	
30	
35	
40	
45	
50	

BAB 5 IMPLEMENTASI SISTEM

Bab ini menjelaskan implementasi sistem yang telah dirancang pada bab sebelumnya, perancangan sistem. Implementasi yang dijelaskan adalah bagian *source code* program dan antarmuka pengguna.

5.1 Implementasi *Source Code* Program

Aplikasi peramalan harga emas menggunakan algoritma *Support Vector Regression* (SVR) ini diimplementasikan dengan bahasa pemrograman JAVA pada IDE *Netbeans* 8.1.

5.1.1 Normalisasi Data

Normalisasi data bertujuan untuk menyamakan standar semua data agar berada pada jarak [0,1]. Metode normalisasi data yang digunakan adalah *min-max normalization*, proses keseluruhannya disajikan pada Kode Program 5.1,

```

1 void normalize() {
2     dataNorm = new double[dataAwal.length]
3         [dataAwal[0].length];
4     for (int i = 0; i < dataNorm.length; i++) {
5         for (int j = 0; j < dataNorm[0].length; j++) {
6             double temp = (dataAwal[i][j] - minValue) /
7                 (maxValue - minValue);
8             dataNorm[i][j] = round(temp, 6);
9         }
10    }
11 }

```

Kode Program 5. 1 Proses Normalisasi Data

Penjelasan dari Kode Program di atas adalah sebagai berikut:

1. Baris 2 adalah inialisasi ordo matriks normalisasi sebagai penampung nilai normalisasi dari data awal.
2. Baris 5 adalah proses normalisasi.
3. Baris 6 digunakan untuk membulatkan nilai hasil normalisasi menjadi 6 angka di belakang koma sekaligus menyimpannya ke dalam array `dataNorm[i][j]`.

5.1.2 Penghitungan Kernel, Matriks Hessian (R_{ij}), dan Nilai γ (*gamma*)

Fungsi *Kernel* bertugas untuk memetakan data input ke dalam dimensi yang lebih tinggi. Fungsi *kernel* yang digunakan adalah *Kernel Gaussian* (RBF) yang mana kode program untuk pemrosesan nilai *distance*, *kernel*, dan *Hessian* disajikan tergabung menjadi 1 fungsi, untuk lebih jelasnya dapat dilihat pada Kode Program 5.2,

```

1 void kernelToHessian() {
2     dataKern = new double[dataAwal.length]
3         [dataAwal.length];
4     dataHess = new double[dataKern.length]
5         [dataKern.length];
6     for (int i = 0; i < dataKern.length; i++) {
7         for (int j = 0; j < dataKern[0].length; j++) {
8             double tempDistance = 0;
9             for (int k = 0; k < dataNorm[0].length - 1; k++) {
10                double temp = (dataNorm[i][k] - dataNorm[j][k]);
11                tempDistance += (Math.pow(temp, 2));
12            }
13            double tempKern = (-tempDistance) /
14                (2 * Math.pow(sigma, 2));
15            dataKern[i][j] = round(Math.exp(tempKern), 6);
16            double tempHess = dataKern[i][j] +
17                Math.pow(lambda, 2);
18            dataHess[i][j] = round(tempHess, 6);
19        }
20    }
21    gamma = setGamma();
22 }

```

Kode Program 5. 2 Proses Penghitungan Kernel dan Matriks Hessian

Penjelasan kode program di atas adalah sebagai berikut:

1. Baris 2 adalah inialisasi ordo matriks kernel sesuai dengan ukuran array `dataAwal`.
2. Baris 3 adalah inialisasi ordo matriks Hessian sesuai dengan ukuran array `dataKern`.
3. Dari baris 4-16 dijalankan berulang dari index 0 sampai memenuhi nilai `dataKern.length`.
4. Baris 8-9 menghitung nilai *distance*.
5. Baris 11 menghitung nilai *kernel*.
6. Baris 13 menghitung nilai *Hessian*.
7. Baris 17 berfungsi menginisialisasikan nilai parameter `gamma`, fungsi rinciannya dapat dilihat pada kode program berikut,

```

1 double setGamma() {
2     double maxHess = dataHess[0][0];
3     for (int i = 0; i < dataHess.length; i++) {
4         for (int j = 0; j < dataHess[0].length; j++) {
5             if (dataHess[i][j] > maxHess) {
6                 maxHess = dataHess[i][j];
7             }
8         }
9     }
10    return round((cLR / maxHess), 6);
11 }

```

Kode Program 5. 3 Proses Penghitungan Parameter Gamma

Penjelasan kode program di atas adalah sebagai berikut:

1. Baris 2-9 merupakan proses pencarian nilai maksimal dari matriks *hessian*.

- Baris 10 adalah inisialisasi nilai parameter gamma yang dibulatkan 6 angka di belakang koma.

5.1.3 Iterasi *Sequential Learning*

Fungsi *sequential learning* bertugas untuk mencari nilai *Lagrange Multiplier* α_i dan α_i^* . Di mana prosesnya akan diulang sebanyak iterasi maksimal sesuai masukan dari pengguna. Fungsi inti dari *Sequential Learning* ini membutuhkan parameter `maxIterasi` berupa integer yang menginisialkan banyaknya iterasi (proses perulangan *sequential learning*) dan parameter `uji` bertipe objek yang menginisialkan objek data uji. Proses fungsi inti ini disajikan pada Kode Program 5.4.

```

1 void iterasiSeqLearn(int maxIterasi, SVRGoldUI uji) {
2     alpha = new double[dataAwal.length];
3     alphaStar = new double[dataAwal.length];
4     uji.alpha = new double[dataAwal.length];
5     uji.alphaStar = new double[dataAwal.length];
6     deltaAlpha = new double[dataAwal.length];
7     deltaAlphaStar = new double[dataAwal.length];
8     E = new double[dataNorm.length];
9     for (int i = 0; i < alphaStar.length; i++) {
10        alpha[i] = alphaStar[i] = 0;
11    }
12    int idx = 0;
13    while (idx < maxIterasi) {
14        for (int i = 0; i < dataNorm.length; i++) {
15            double tempE = 0;
16            for (int j = 0; j < dataHess[0].length; j++) {
17                tempE += (alphaStar[j] - alpha[j]) * dataHess[i][j];
18            }
19            E[i] = dataNorm[i][4] - tempE;
20            deltaAlphaStar[i] = Math.min(Math.max(gamma * (E[i] -
21                epsilon), -alphaStar[i]), C - alphaStar[i]);
22            deltaAlpha[i] = Math.min(Math.max(gamma * (-E[i] -
23                epsilon), -alpha[i]), C - alpha[i]);
24            alphaStar[i] += deltaAlphaStar[i];
25            alpha[i] += deltaAlpha[i];
26        }
27        if (setMaxDelta(deltaAlpha) < epsilon &&
28            setMaxDelta(deltaAlphaStar) < epsilon) {
29            break;
30        }
31        idx++;
32    }
33    rchItr = idx;
34    for (int i = 0; i < uji.alpha.length; i++) {
35        uji.alphaStar[i] = alphaStar[i];
36        uji.alpha[i] = alpha[i];
37    }
38 }

```

Kode Program 5. 4 Fungsi Iterasi *Sequential Learning*

Penjelasan kode program di atas adalah sebagai berikut:

- Baris 2-8 adalah inisialisasi ordo matriks alpha, alphaStar, deltaAlpha, deltaAlphaStar, dan E sesuai dengan ukuran i array `dataAwal`.

2. Baris 9-11 adalah proses inialisasi nilai awal matriks alpha dan alphaStar bernilai 0.
3. Baris 13-30 menjalankan fungsi iterasi sebanyak `maxIterasi` (masukan dari user).
4. Baris 16-18 melakukan perhitungan sesuai bagian dari persamaan 2.4 ($\sum_{j=1}^n (\alpha_j^* - \alpha_j) R_{ij}$) dan nilainya disimpan pada variabel `tempE`.
5. Baris 19 mengisikan nilai E dari perhitungan yang didapat ke dalam matriks `E`.
6. Baris 20 menginisalkan nilai $\delta\alpha_i^*$ sesuai dengan persamaan 2.5.
7. Baris 21 menginisalkan nilai $\delta\alpha_i$ sesuai dengan persamaan 2.6.
8. Baris 22 merupakan fungsi penghitungan α_i^* yang baru sesuai persamaan 2.7.
9. Baris 23 merupakan fungsi penghitungan α_i yang baru sesuai persamaan 2.8
10. Baris 25-27 bertugas memeriksa kondisi konvergensi.
11. Baris 31-34 bertugas untuk menduplikasikan nilai *lagrange multiplier* terakhir yang didapat dari proses iterasi ke array `alphaStar` dan `alpha` pada objek uji yang dimasukkan melalui parameter fungsi ini. Array tersebut nantinya digunakan pada fase pengujian (*testing phase*).

5.1.4 Penghitungan Fungsi Peramalan ($f(x)$)

```

1 void fx() {
2     fx = new double[dataNorm.length];
3     for (int i = 0; i < fx.length; i++) {
4         double fxTemp = 0;
5         for (int j = 0; j < dataHess[0].length; j++) {
6             double fxTotal = (alphaStar[j] - alpha[j]) *
                               dataHess[i][j];
7             fxTemp += fxTotal;
8         }
9         fx[i] = round(fxTemp, 8);
10    }
11 }

```

Kode Program 5. 5 Proses penghitungan nilai peramalan

Penjelasan kode program di atas adalah sebagai berikut:

1. Baris 2 menginisialisasikan ordo matriks `fx` sesuai dengan ukuran `i` matriks `dataNorm` (data yang telah dinormalisasi).
2. Baris 3-10 adalah proses penghitungan nilai $f(x)$ (hasil peramalan) untuk masing-masing data dan akan dijlankan sampai semua index array `fx` terisi semua.
3. Baris 5-8 adalah proses penghitungan nilai $f(x)$ (hasil peramalan) untuk satu data sesuai dengan persamaan 2.9, hasilnya disimpan pada variabel `fxTemp`.
4. Baris 9 mengisikan nilai hasil pembulatan dari proses 3 di atas ke dalam array `fx`.

5.1.5 Penghitungan Nilai *Error Rate*

```
1 void mape() {
2     double absTotal = 0;
3     for (int i = 0; i < deNorm.length; i++) {
4         double absTemp = Math.abs((dataAwal[i]
5             [dataAwal[0].length - 1] - deNorm[i]) /
6             dataAwal[i][dataAwal[0].length - 1]);
7         absTotal += absTemp;
8     }
9     double tempMape = absTotal * 100 / deNorm.length;
10    mape = round(tempMape, 8);
11 }
```

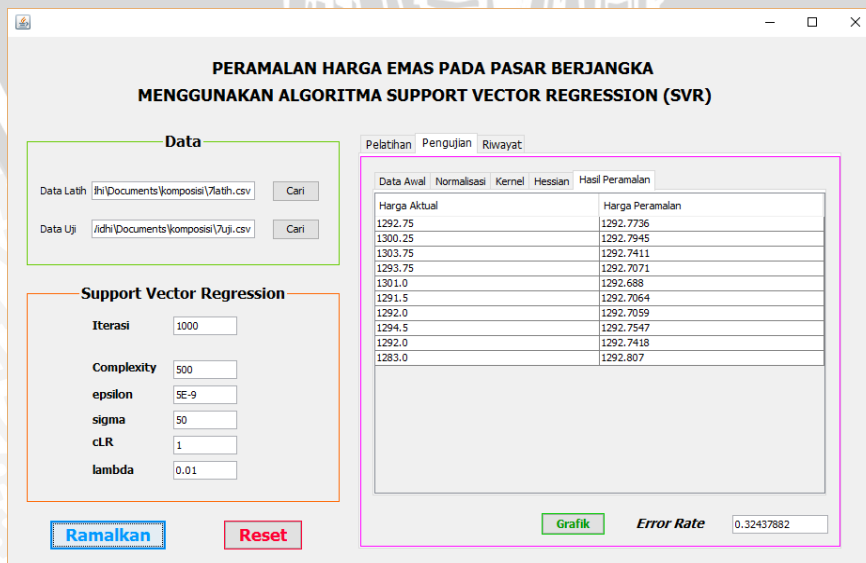
Kode Program 5. 6 Proses penghitungan nilai *Error Rate* (MAPE)

Penjelasan kode program di atas adalah sebagai berikut:

1. Baris 3-6 menghitung nilai absolut kesalahan dari hasil peramalan (array $f(x)$) sesuai dengan bagian dari persamaan 2.12 yaitu $\left(\sum_{i=1}^n \left| \frac{y_i - y'_i}{y_i} \right| \right)$. Hasilnya disimpan pada variabel `absTotal`.
2. Baris 7 menghitung nilai kesalahan sesuai persamaan 2.12. hasilnya disimpan pada variabel `tempMape`.
3. Baris 8 memberikan nilai pembulatan dari variabel `tempMape` pada variabel global `mape` sebagai nilai evaluasi MAPE.

5.2 Implementasi Antarmuka Pengguna

Implementasi antarmuka aplikasi sesuai dengan yang telah dibuat pada Bab 4. Aplikasi terdiri dari 1 jendela dengan beberapa tab untuk menampilkan hasil pelatihan, pengujian, dan riwayat peramalan. Yang perlu dilakukan sebelum melakukan peramalan adalah memilih berkas data latih dan data uji pada bagian Data, kemudian mengatur jumlah iterasi dan nilai parameter pada bagian Support Vector Regression. Setelah itu 'klik' pada tombol Ramalkan untuk memulai proses peramalan. Hasil MAPE dapat dilihat pada field *Error Rate*.



Gambar 5. 1 Jendela Aplikasi Peramalan

BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini dibahas mengenai hasil pengujian dan analisis dari performansi aplikasi peramalan harga emas pada pasar berjangka menggunakan algoritma SVR yang telah dibangun. Pengujian yang dilakukan sesuai dengan yang telah dirancang pada bab 4, perancangan sistem.

6.1 Hasil dan Analisis Uji Coba Jumlah Iterasi Pelatihan

Jumlah iterasi SVR untuk pelatihan yang diuji adalah 50, 100, 500, 1000, 5000, 10000, 50000, 100000, 500000 sesuai dengan perancangan yang telah dibuat pada Bab 4. Rincian parameter lainnya yang digunakan pada proses pengujian ini adalah sebagai berikut:

- a. Nilai C : 10
- b. Nilai ϵ : 0,0000000002
- c. Nilai σ : 0,07
- d. Nilai cLR : 0,85
- e. Nilai λ : 0,0009
- f. Jumlah data latih : 30 hari
- g. Jumlah data uji : 10 hari

Pengujian jumlah iterasi pelatihan SVR ini dilakukan sebanyak 10 kali di mana masing-masing iterasi diuji menggunakan 10 variasi pasangan data latih dan data uji. Hasil pengujiannya ditunjukkan pada Tabel 6.1,

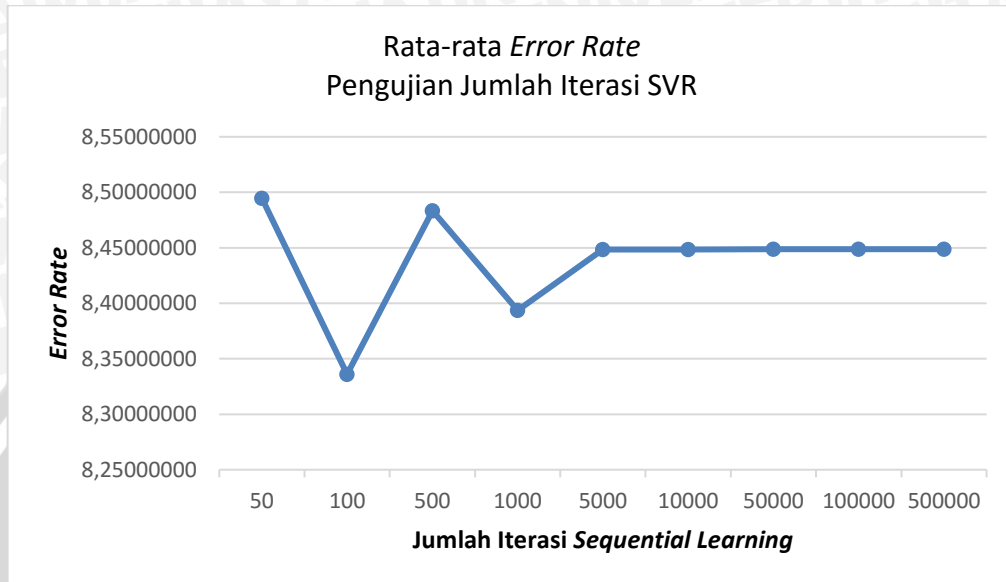
Tabel 6. 1 Hasil Pengujian Jumlah Iterasi SVR

Jumlah Iterasi	Error Rate Percobaan ke - i										Rata-rata
	1	2	3	4	5	6	7	8	9	10	Error Rate
50	18,7756	11,9415	13,2864	5,2654	1,7923	1,7868	1,3841	12,1363	4,3147	14,2631	8,49461367
100	18,7765	11,8766	13,0506	5,3683	1,8432	1,8027	0,7775	12,2044	3,9275	13,7327	8,33601524
500	18,7828	11,8557	13,0246	5,4198	2,0796	1,6041	1,1104	12,1223	5,2730	13,5621	8,48343301
1000	18,7857	11,8555	13,0252	5,4258	2,0758	1,5448	0,7360	11,8379	5,1472	13,5014	8,39352895
5000	18,7871	11,8551	13,0256	5,4257	2,0786	1,5522	0,7325	12,3144	5,2081	13,5041	8,44832561
10000	18,7871	11,8551	13,0256	5,4257	2,0786	1,5521	0,7350	12,3143	5,2081	13,5041	8,44856895
50000	18,7871	11,8551	13,0256	5,4257	2,0786	1,5521	0,7352	12,3143	5,2081	13,5041	8,44858807
100000	18,7871	11,8551	13,0256	5,4257	2,0786	1,5521	0,7352	12,3143	5,2081	13,5041	8,44858807
500000	18,7871	11,8551	13,0256	5,4257	2,0786	1,5521	0,7352	12,3143	5,2081	13,5041	8,44858807

Jumlah iterasi pelatihan berpengaruh pada presisi yang dihasilkan, semakin banyak iterasi maka bertambah pula kemampuan SVR dalam mengobservasi pola data. Iterasi yang bertambah juga akan mengoptimalkan proses *trade-off* dalam membentuk garis regresi sehingga *lagrange multiplier* semakin konvergen pada optimum global dan nilai *error rate* membaik. Namun untuk beberapa percobaan sempat terjadi anomali seperti pada percobaan 9 di mana iterasi 100 dapat menghasilkan *error rate* yang lebih baik dibandingkan iterasi 500, hal ini disebabkan oleh fluktuasi data yang diujikan tidak begitu besar.

Dari Tabel 6.1 ditunjukkan rerata *error rate* terkecil didapatkan pada 100 iterasi SVR dengan nilai 8,33601524. Grafik pada Gambar 6.1 menunjukkan rerata *error*

rate yang cukup berfluktuasi, namun apabila dicermati detail setiap percobaan pada Tabel 6.1 nilai *error rate* yang didapat cenderung menurun sampai pada jumlah iterasi maksimal tertentu kemudian menjadi konstan. Titik konstan tiap percobaan berbeda-beda, hal inilah yang menyebabkan fluktuasi rerata *error rate* pada Gambar 6.1.

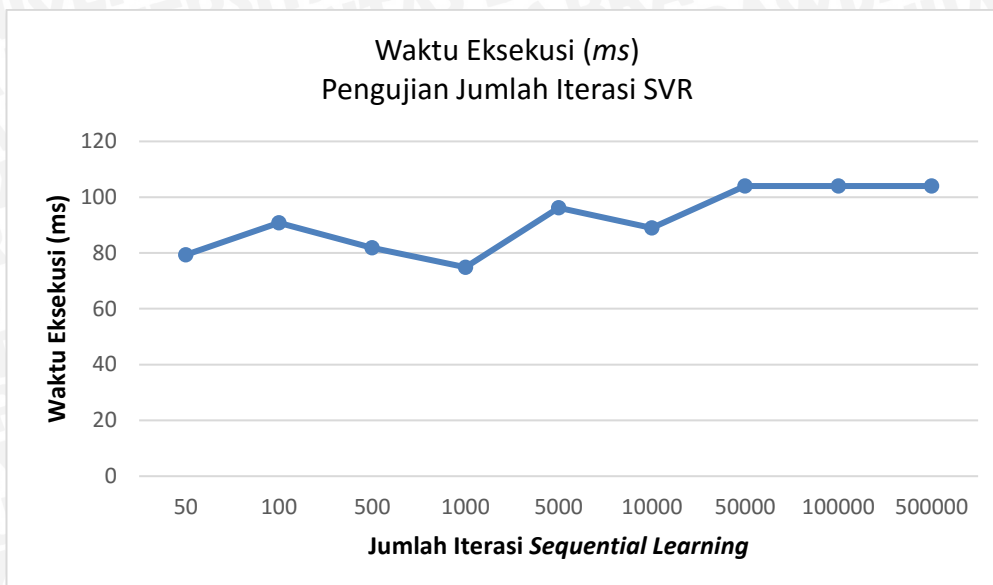


Gambar 6. 1 Grafik Rerata *Error Rate* Pengujian Iterasi SVR

Waktu yang dibutuhkan SVR untuk menjalankan iterasi pada tiap batas maksimalnya tidak berbeda secara signifikan, dalam hitungan *millisecond* jumlah waktu eksekusi yang dibutuhkan untuk tiap iterasinya ditunjukkan pada Tabel 6.2 dan direpresentasikan reratanya pada Gambar 6.2. Dari grafik tersebut dapat dilihat rata-rata waktu eksekusi yang berfluktuatif, namun mulai terlihat stabil ketika iterasi diatur sejumlah 50000. Jika dibandingkan dengan grafik rerata *error rate* pada Gambar 6.1 yang juga mulai menunjukkan nilai konstan mulai dari iterasi 5000 maka model peramalan diperkirakan sudah memenuhi syarat konvergensi sebelum mencapai iterasi maksimal yang diatur.

Tabel 6. 2 Waktu Eksekusi Hasil Pengujian Jumlah Iterasi SVR

Jumlah Iterasi	Waktu Eksekusi Percobaan ke - i (milisecond)										Rata-rata Waktu Eksekusi
	1	2	3	4	5	6	7	8	9	10	
50	153	137	85	81	60	60	46	65	50	56	79,3
100	197	169	108	73	62	68	57	60	63	51	90,8
500	137	127	84	71	115	64	65	59	45	52	81,9
1000	143	105	69	79	60	64	81	47	51	50	74,9
5000	267	121	73	73	79	69	91	65	63	61	96,2
10000	185	117	73	81	78	74	88	70	63	61	89
50000	185	129	73	81	78	93	207	70	63	61	104
100000	185	129	73	81	78	93	207	70	63	61	104
500000	185	129	73	81	78	93	207	70	63	61	104



Gambar 6. 2 Grafik Rerata Waktu Eksekusi Pengujian Iterasi SVR

6.2 Hasil dan Analisis Uji Coba Nilai Parameter C (Complexity)

Nilai parameter C (Complexity) yang diuji sesuai dengan perancangan yang telah dibuat pada Bab 4. Rincian parameter lainnya yang digunakan pada proses pengujian ini adalah sebagai berikut:

- a. Nilai ϵ : 0,0000000002
- b. Nilai σ : 0,07
- c. Nilai cLR : 0,85
- d. Nilai λ : 0,0009
- e. Iterasi SVR : 1000
- f. Jumlah data latih : 30 hari
- g. Jumlah data uji : 10 hari

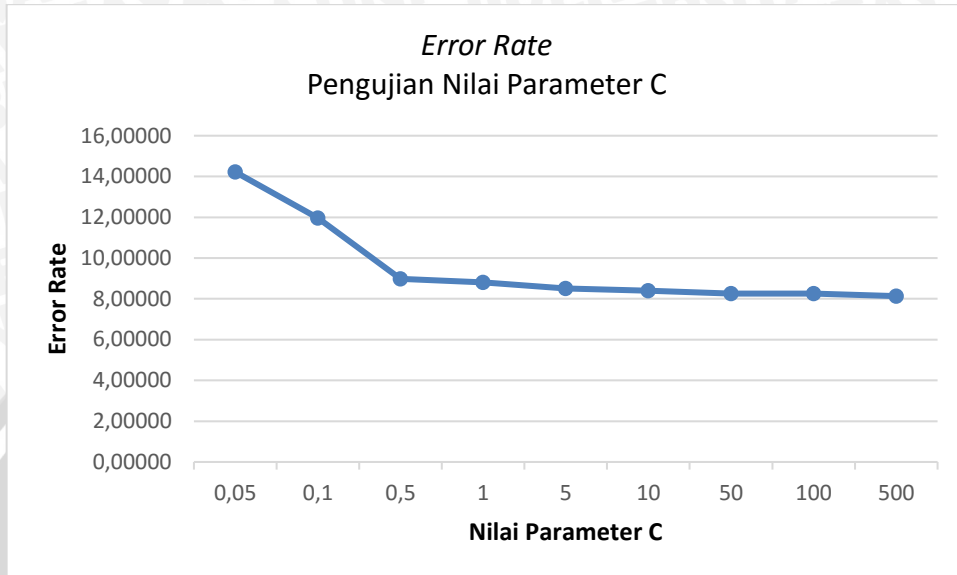
Pengujian C (Complexity) dilakukan sebanyak 10 kali untuk setiap nilai yang diujikan. Hasil pengujiannya ditunjukkan pada tabel berikut:

Tabel 6. 3 Hasil Pengujian Nilai Parameter C (Complexity)

Nilai C	Error Rate Percobaan ke - i										Rata-rata Error Rate
	1	2	3	4	5	6	7	8	9	10	
0,05	21,1279	19,8710	18,6053	15,3878	13,0412	8,5678	0,9193	13,4813	13,0375	18,1824	14,22216
0,1	20,8674	17,7955	16,1550	12,0118	7,2021	3,7561	2,3673	13,2534	9,4456	16,7996	11,96537
0,5	19,1143	12,2001	14,2795	5,2045	1,7532	1,7583	1,8050	12,6630	5,9641	15,1115	8,98534
1	18,7734	11,9443	14,0401	5,3403	1,7836	1,8057	1,1121	12,6684	5,6974	14,9036	8,80689
5	18,7801	11,8554	13,2352	5,4386	1,9157	1,5624	0,6708	12,2223	5,0888	14,3859	8,51553
10	18,7857	11,8555	13,0252	5,4258	2,0758	1,5448	0,7360	11,8379	5,1472	13,5014	8,39353
50	18,7857	11,8555	13,0252	5,4258	2,0758	1,5448	0,8242	12,1190	5,8911	10,9563	8,25034
100	18,7857	11,8555	13,0252	5,4258	2,0758	1,5448	0,8242	12,1190	5,8911	10,9563	8,25034
500	18,7857	11,8555	11,8554	5,4258	2,0758	1,5448	0,8242	12,1190	5,8911	10,9563	8,13336

Parameter C merepresentasikan batas penalti toleransi terhadap kesalahan peramalan, semakin besar nilai parameter C menjadikan model peramalan

semakin tidak mentoleransi kesalahan sehingga memberikan hasil peramalan yang bagus begitu pula sebaliknya (Furi, et al., 2015). Pada pengujian ini teori tersebut benar adanya, dibuktikan dengan nilai parameter C tertinggi yang memberikan *error rate* terbaik dibandingkan dengan nilai lainnya yang lebih kecil.



Gambar 6. 3 Grafik rata-rata *error rate* pengujian nilai parameter C

Gambar 6.3 menunjukkan rerata *error rate* yang menurun cukup tajam ketika nilai $C < 1$ dan cenderung mendatar sampai $C = 500$. Dari hasil pengujian sebagaimana ditunjukkan pada tabel di atas, didapatkan nilai rata-rata *error rate* terkecil sebesar 8,13336 saat nilai parameter C (*Complexity*) = 500.

6.3 Hasil dan Analisis Uji Coba Nilai Parameter ϵ (*epsilon*)

Nilai parameter ϵ (*epsilon*) yang diuji sesuai dengan perancangan yang telah dibuat pada Bab 4. Rincian nilai parameter lainnya yang digunakan pada proses pengujian ini adalah sebagai berikut:

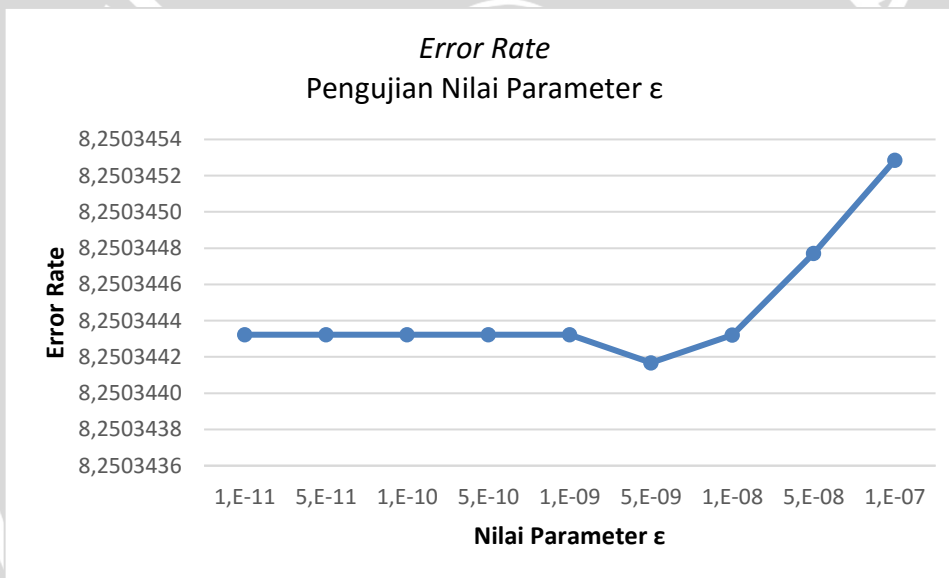
- a. Nilai C : 500
- b. Nilai σ : 0,07
- c. Nilai cLR : 0,85
- d. Nilai λ : 0,0009
- e. Iterasi SVR : 1000
- f. Jumlah data latih : 30 hari
- g. Jumlah data uji : 10 hari

Pengujian nilai ϵ (*epsilon*) dilakukan sebanyak 10 kali untuk setiap nilai yang diujikan. Hasil pengujiannya ditunjukkan pada Tabel 6.4,

Tabel 6. 4 Hasil Pengujian Nilai Parameter ϵ (*epsilon*)

Nilai ϵ	Error Rate Percobaan ke - i										Rata-rata
	1	2	3	4	5	6	7	8	9	10	Error Rate
1,E-11	18,7856727	11,8554720	13,0252427	5,4258371	2,075814	1,54480826	0,82421580	12,11895478	5,8910917	10,9563343	8,2503443
5,E-11	18,7856727	11,8554720	13,0252427	5,4258371	2,075814	1,54480826	0,82421580	12,11895478	5,8910917	10,9563343	8,2503443
1,E-10	18,7856727	11,8554720	13,0252427	5,4258371	2,075814	1,54480826	0,82421580	12,11895478	5,8910917	10,9563343	8,2503443
5,E-10	18,7856727	11,8554720	13,0252427	5,4258371	2,075814	1,54480826	0,82421580	12,11895478	5,8910917	10,9563343	8,2503443
1,E-09	18,7856727	11,8554720	13,0252427	5,4258371	2,075814	1,54480826	0,82421580	12,11895478	5,8910917	10,9563343	8,2503443
5,E-09	18,7856727	11,8554720	13,0252427	5,4258364	2,075814	1,54480749	0,82421580	12,11895478	5,8910909	10,9563351	8,2503442
1,E-08	18,7856727	11,8554720	13,0252427	5,4258364	2,075814	1,54480749	0,82421580	12,11895478	5,8910909	10,9563366	8,2503443
5,E-08	18,7856727	11,8554720	13,0252427	5,4258364	2,075814	1,54480904	0,82421426	12,11895478	5,8910886	10,9563434	8,2503448
1,E-07	18,7856727	11,8554720	13,0252434	5,4258356	2,075814	1,54480982	0,82421425	12,11895318	5,8910855	10,9563525	8,2503453

Parameter ϵ (*epsilon*) digunakan untuk mengatur batas kesalahan fungsi $f(x)$, nilai tersebut menyelubungi nilai dari fungsi $f(x)$ sehingga membentuk daerah yang disebut *error-zone* untuk tiap nilai dataset. Jika nilai $f(x)$ melebihi batas zona yang terbentuk maka akan dikenai penalti sebesar nilai parameter C yang telah diatur (Furi, et al., 2015). Dari hasil pengujian sebagaimana ditunjukkan pada tabel di atas, didapatkan nilai rata-rata *error rate* terkecil = 8,2503442 pada saat parameter $\epsilon = 5 \times 10^{-9}$.



Gambar 6. 4 Grafik rata-rata *error rate* pengujian nilai parameter ϵ

Gambar 6.4 di atas merepresentasikan fluktuasi nilai rata-rata *error rate* dari tabel hasil pengujian parameter ϵ di atas, dari gambar tersebut terlihat nilai rata-rata *error rate* cenderung stabil ketika nilai $\epsilon < 5 \times 10^{-9}$ namun meningkat setelah nilai $\epsilon > 5 \times 10^{-9}$, ini menandakan *error-zone* semakin melebar dan batas penalti juga meningkat.

6.4 Hasil dan Analisis Uji Coba Nilai Parameter σ (*sigma*)

Nilai *range* parameter σ (*sigma*) yang diuji sesuai dengan perancangan yang telah dibuat pada Bab 4. Rincian parameter lainnya yang digunakan pada proses pengujian ini adalah sebagai berikut:

- a. Nilai C : 500



- b. Nilai ϵ : 0,000000005
- c. Nilai cLR : 0,85
- d. Nilai λ : 0,0009
- e. Iterasi SVR : 1000
- f. Jumlah data latih : 30 hari
- g. Jumlah data uji : 10 hari

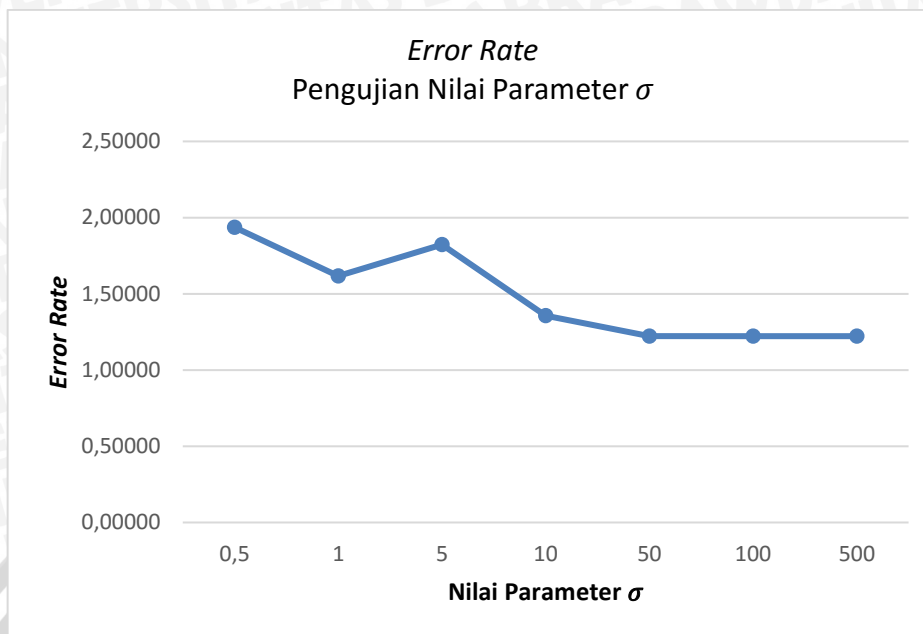
Pengujian parameter σ (*sigma*) dilakukan sebanyak 10 kali untuk setiap nilai yang diujikan. Hasil pengujianya ditunjukkan pada Tabel 6.5,

Tabel 6. 5 Hasil Pengujian Nilai Parameter σ (*sigma*)

Nilai σ	Error Rate Percobaan ke - <i>i</i>										Rata-rata
	1	2	3	4	5	6	7	8	9	10	Error Rate
0,05	20,6123	16,5056	17,0833	8,2723	4,9402	2,7719	1,0760	13,2557	10,8952	17,4601	11,28726
0,1	15,3001	7,3059	6,6286	2,9816	0,9644	1,3604	0,5013	9,5849	1,0976	6,4526	5,21774
0,5	0,9458	0,7272	3,0213	1,5394	0,5474	1,0715	0,5474	3,6274	1,4115	5,9236	1,93624
1	0,9855	0,7683	2,4671	1,0723	0,4908	1,1785	0,3943	3,5773	1,9215	3,3165	1,61722
5	0,8359	0,9378	1,9027	2,5078	0,9282	0,8783	0,3761	4,0338	2,5922	3,2279	1,82207
10	0,4690	0,6583	1,7222	2,1870	1,0204	0,6096	0,3146	3,0902	2,5002	0,9874	1,35589
50	1,0811	0,5556	1,6976	1,4969	1,1147	0,5367	0,3303	2,6460	2,3515	0,4156	1,22261
100	1,1196	0,5579	1,7064	1,4643	1,1160	0,5367	0,3303	2,6312	2,3422	0,4230	1,22276
500	1,1296	0,5589	1,7095	1,4561	1,1162	0,5354	0,3308	2,6272	2,3417	0,4260	1,22313

Parameter σ (*sigma*) merupakan konstanta dari fungsi Kernel Gaussian RBF untuk mengatur persebaran data ke dalam dimensi fitur yang lebih tinggi (Furi, et al., 2015). Jika σ bernilai terlalu kecil maka kurva peramalan yang dihasilkan menjadi sangat halus, namun jika terlalu besar maka kurva peramalan yang terbentuk menjadi sangat kasar (Kurniasih, 2013). Dari hasil pengujian sebagaimana ditunjukkan pada Tabel 6.5 di atas, didapatkan rata-rata *error rate* terkecil senilai 1,22261 dengan parameter $\sigma = 50$.

Hal ini menunjukkan bahwa untuk data harga emas yang digunakan, pada saat nilai $\sigma < 1$ maka persebaran data yang terjadi terlalu dekat sehingga kurva peramalan yang dihasilkan sangat halus dan nilai *error rate* meningkat. Untuk mendapatkan visualisasi grafik yang lebih jelas, nilai rerata *error rate* pada saat $\sigma = 0,05$ dan $\sigma = 0,1$ dapat diabaikan karena rentangnya yang cukup jauh bila dibandingkan dengan lainnya, maka didapatlah visualisasi grafik *error rate* yang ditunjukkan pada Gambar 6.5,



Gambar 6. 5 Grafik rata-rata *error rate* pengujian nilai parameter σ

6.5 Hasil dan Analisis Uji Coba Nilai Parameter *cLR*

Nilai parameter *cLR* yang diuji sesuai dengan perancangan yang telah dibuat pada Bab 4. Rincian parameter lainnya yang digunakan pada proses pengujian ini adalah sebagai berikut:

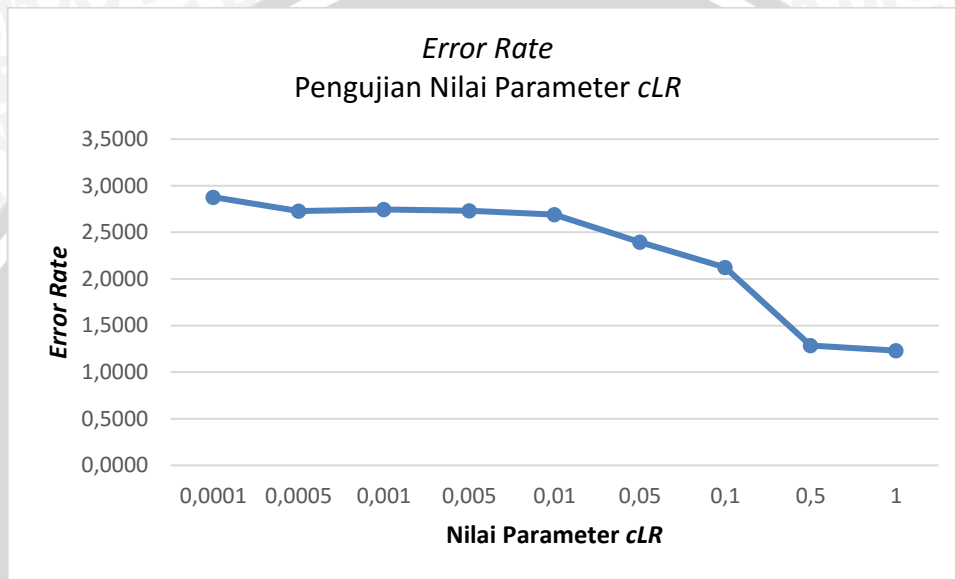
- a. Nilai C : 500
- b. Nilai ϵ : 0,000000005
- c. Nilai σ : 50
- d. Nilai λ : 0,0009
- e. Iterasi SVR : 1000
- f. Jumlah data latih : 30 hari
- g. Jumlah data uji : 10 hari

Pengujian nilai parameter *cLR* dilakukan sebanyak 10 kali untuk setiap nilai yang diujikan. Hasil pengujiannya ditunjukkan pada Tabel 6.6:

Tabel 6. 6 Hasil Pengujian *range* parameter *cLR*

Nilai <i>cLR</i>	<i>Error Rate</i> Percobaan ke - <i>i</i>										Rata-rata <i>Error Rate</i>
	1	2	3	4	5	6	7	8	9	10	
0,0001	6,2178	5,7164	4,0890	1,8302	1,1248	0,7581	0,7553	2,6424	1,5163	4,1228	2,8773
0,0005	5,5218	4,8356	2,8349	2,7082	1,7350	1,2381	0,3369	3,4187	1,3853	3,2750	2,7289
0,001	5,3734	4,6571	2,7320	2,9291	1,7985	1,5109	0,3823	3,5167	1,3857	3,1584	2,7444
0,005	5,2603	4,5460	2,6393	3,0108	1,8073	1,4737	0,4275	3,6268	1,3847	3,1348	2,7311
0,01	5,1809	4,4378	2,5246	3,0396	1,7453	1,4054	0,4293	3,6103	1,3834	3,1371	2,6894
0,05	4,5610	3,5635	1,8767	3,1869	1,3658	1,0476	0,4068	3,4957	1,3867	3,0628	2,3953
0,1	3,8001	2,5812	1,6323	3,1846	1,1214	0,8157	0,3752	3,3921	1,5591	2,7696	2,1231
0,5	1,2265	0,5547	1,6168	1,7845	1,0644	0,5310	0,3236	2,9238	2,3916	0,4392	1,2856
1	1,1647	0,5729	1,7217	1,4795	1,1107	0,5803	0,3244	2,5368	2,3683	0,4506	1,2310

Parameter cLR merupakan konstanta untuk laju pembelajaran, semakin kecil nilainya maka proses *learning* semakin lambat namun dapat memberikan hasil peramalan yang lebih bagus namun apabila nilai cLR semakin besar laju proses *learning* semakin cepat namun rawan memberikan hasil yang buruk (Vijayakumar & Wu, 1999). Hasil pengujian di atas menunjukkan parameter cLR terbaik berada pada nilai 1 yang ditandai dengan nilai rata-rata *error rate* terkecil yaitu 1,231, ini menunjukkan bahwa untuk nilai $cLR = 1$ dapat memberikan perpaduan antara laju proses *learning* dan hasil peramalan yang bagus. Hasil visualnya ditunjukkan pada Gambar 6.5,



Gambar 6. 6 Grafik rata-rata *error rate* pengujian nilai parameter cLR

6.6 Hasil dan Analisis Uji Coba Nilai Parameter λ (*lambda*)

Nilai *range* parameter λ (*lambda*) yang diuji sesuai dengan perancangan yang telah dibuat pada Bab 4. Rincian parameter lainnya yang digunakan pada proses pengujian ini adalah sebagai berikut:

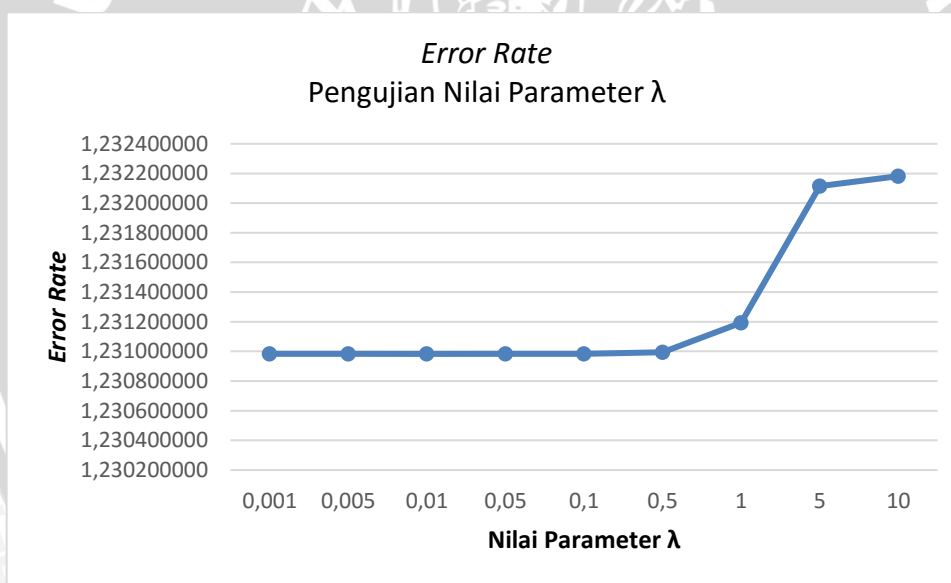
- a. Nilai C : 500
- b. Nilai ϵ : 0,000000005
- c. Nilai σ : 50
- d. Nilai cLR : 1
- e. Iterasi SVR : 1000
- f. Jumlah data latih : 30 hari
- g. Jumlah data uji : 10 hari

Pengujian λ (*lambda*) dilakukan sebanyak 10 kali untuk setiap nilai yang diujikan. Hasil pengujiannya ditunjukkan pada Tabel 6.7,

Tabel 6. 7 Hasil Pengujian Nilai Parameter λ (*lambda*)

Nilai λ	Error Rate Percobaan ke - <i>i</i>										Rata-rata
	1	2	3	4	5	6	7	8	9	10	Error Rate
0,001	1,16465	0,57287	1,72170	1,47945	1,11074	0,58033	0,32438	2,53679	2,36830	0,45062	1,230983738
0,005	1,16466	0,57286	1,72171	1,47945	1,11074	0,58033	0,32438	2,53679	2,36830	0,45062	1,230983804
0,01	1,16466	0,57286	1,72171	1,47945	1,11074	0,58033	0,32438	2,53678	2,36830	0,45063	1,230983165
0,05	1,16477	0,57285	1,72173	1,47935	1,11074	0,58033	0,32438	2,53674	2,36826	0,45068	1,230983424
0,1	1,16512	0,57281	1,72181	1,47903	1,11074	0,58036	0,32437	2,53658	2,36817	0,45085	1,230984293
0,5	1,17420	0,57169	1,72377	1,47096	1,11065	0,58098	0,32430	2,53254	2,36567	0,45519	1,230995470
1	1,18863	0,56991	1,72691	1,45816	1,11048	0,58199	0,32419	2,52615	2,36172	0,46378	1,231192520
5	1,21109	0,56711	1,73189	1,44167	1,11016	0,58360	0,32401	2,51630	2,35562	0,47970	1,232114994
10	1,21248	0,56693	1,73220	1,44075	1,11014	0,58370	0,32400	2,51570	2,35524	0,48069	1,232181728

Parameter λ (*lambda*) berpengaruh pada ukuran penskalaan ruang pemetaan *kernel* pada SVR. Nilai λ yang tinggi akan memberikan hasil peramalan yang baik, namun apabila terlalu besar maka akan memengaruhi waktu komputasi karena proses *learning* menjadi tidak stabil dan konvergensi yang melambat (Vijayakumar & Wu, 1999). Hasil pengujian memberikan nilai *error rate* terkecil = 1,230983165 di saat $\lambda = 0,01$. Gambar 6.7 menunjukkan peningkatan *error rate* yang lebih tajam saat nilai $\lambda = 0,5$, ini menandakan penskalaan yang terjadi mulai memburuk dan mempengaruhi kestabilan proses *learning* dan konvergensi.



Gambar 6. 7 Grafik rata-rata *error rate* pengujian nilai parameter λ

6.7 Hasil dan Analisis Uji Coba Variasi Jumlah Data Latih

Variasi jumlah data latih yang diujikan sesuai dengan perancangan yang telah dibuat pada Bab 4. Nilai parameter dan jumlah iterasi yang digunakan adalah nilai terbaik dari yang telah diujikan pada subbab sebelumnya yaitu sebagai berikut:

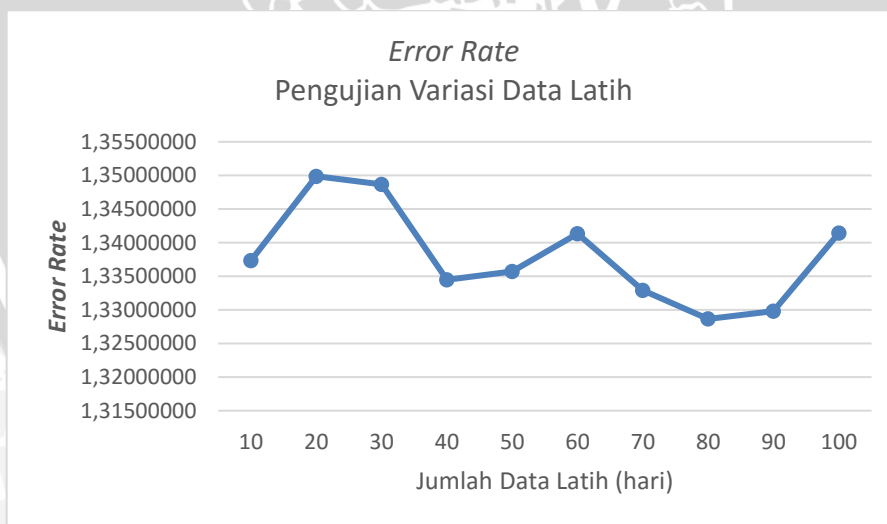
- a. Nilai C : 500
- b. Nilai ϵ : 0,000000005
- c. Nilai σ : 50
- d. Nilai cLR : 1

- e. Nilai λ : 0,01
- f. Iterasi SVR : 1000
- g. Jumlah data uji : 10 hari

Hasil pengujian memberikan nilai error rate terbaik senilai 1,32864642 saat jumlah data latih yang digunakan adalah 80 hari. Tabel 6.8 menunjukkan nilai *error rate* yang relatif kecil yaitu berkisar 1,32 – 1,35, namun Grafik 6.8 menunjukkan fluktuasi yang cukup beragam pada setiap perbedaan jumlah data latih yang digunakan. Hal ini menunjukkan pengaruh jumlah data latih yang relatif kecil terhadap data uji yang digunakan.

Tabel 6. 8 Hasil Pengujian Variasi Jumlah Data Latih

Jumlah Data Latih (hari)	Rata-rata Error Rate
10	1,33734882
20	1,34987895
30	1,34868925
40	1,33450038
50	1,33572413
60	1,34134623
70	1,33289154
80	1,32864642
90	1,32980743
100	1,34141387



Gambar 6. 8 Grafik nilai *error rate* pengujian variasi jumlah data latih

6.8 Hasil dan Analisis Uji Coba Variasi Jumlah Data Uji

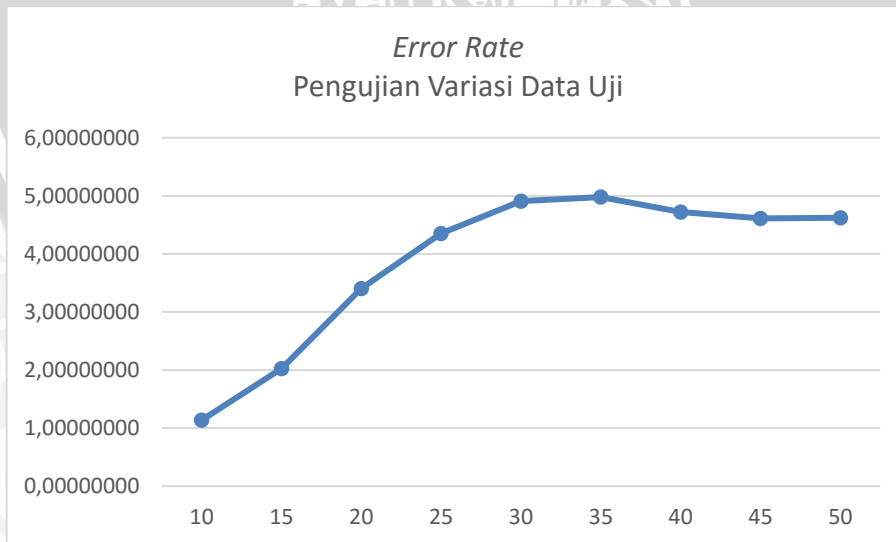
Variasi jumlah data uji yang diujikan sesuai dengan perancangan yang telah dibuat pada Bab 4. Nilai parameter dan jumlah iterasi yang digunakan adalah nilai terbaik dari yang telah diujikan pada subbab sebelumnya yaitu sebagai berikut:

- a. Nilai C : 500
- b. Nilai ϵ : 0,000000005
- c. Nilai σ : 50
- d. Nilai cLR : 1
- e. Nilai λ : 0,01
- f. Iterasi SVR : 1000
- g. Jumlah data latih : 80 hari

Dari Tabel 6.9 didapatkan nilai error rate terkecil senilai 1,13670285 saat jumlah data uji sebanyak 10 hari. Sedangkan Gambar 6.9 menunjukkan nilai *error rate* yang meningkat secara eksponensial untuk data uji berjumlah 10 – 25 dan cenderung stabil untuk data uji lebih dari 25. Hal ini menunjukkan keakuratan yang bisa diperoleh dari model peramalan SVR semakin menurun ketika jumlah data uji yang digunakan semakin banyak.

Tabel 6. 9 Hasil Pengujian Variasi Data Uji

Jumlah Data Uji (hari)	Rata-rata Error Rate
10	1,13670285
15	2,02648098
20	3,40189812
25	4,35054136
30	4,91076965
35	4,97862872
40	4,72153263
45	4,61167341
50	4,62135158



Gambar 6. 9 Grafik nilai *error rate* pengujian variasi jumlah data uji

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan perancangan, implementasi, dan pengujian yang telah dilakukan pada penelitian ini dapat ditarik beberapa kesimpulan yaitu:

1. Proses awal sebelum mengimplementasikan metode SVR untuk meramalkan harga emas pada pasar berjangka adalah menentukan seperangkat data yang akan digunakan baik data latih maupun data uji. Fitur yang digunakan berupa data sekuensial, yaitu data harga pada hari sebelum hari yang ditargetkan. Proses selanjutnya adalah normalisasi data baik untuk data latih maupun data uji. Kemudian menghitung matriks *kernel* dan *Hessian* untuk menentukan nilai parameter *gamma*. Nilai *lagrange multiplier* yang dikeluarkan dari proses *sequential learning* digunakan untuk menghitung nilai peramalan ($f(x)$). Untuk proses evaluasi digunakan MAPE untuk menghitung nilai *error rate*, nilai inilah yang menyatakan akurasi model peramalan harga emas menggunakan algoritma *support vector regression*.
2. Berdasarkan pengujian yang telah dilakukan, nilai *error rate* terkecil yang didapatkan senilai 0,324378 dengan nilai parameter $C = 500$, $\varepsilon = 5 \times 10^{-9}$, $\sigma = 50$, $cLR = 1$, $\lambda = 0,01$, jumlah data latih = 30 hari, jumlah data uji = 10 hari, iterasi maksimal = 1000.

7.2 Saran

Saran yang dapat digunakan untuk penelitian selanjutnya terkait objek dan metode pada penelitian ini adalah sebagai berikut:

1. Dapat ditambahkan metode untuk optimasi nilai parameter yang digunakan sehingga memberikan hasil peramalan yang lebih baik dan mengurangi nilai kesalahan. Hal ini dikarenakan nilai-nilai parameter SVR yang digunakan mudah sekali terjebak pada nilai *local optimum*, sehingga apabila digunakan metode optimasi model peramalan menggunakan algoritma SVR maka diharapkan dapat mengatasi kekurangan SVR dalam pemilihan nilai parameter.
2. Dilakukan pengujian terhadap jumlah data sekuensial untuk fitur data latih dan data uji yang digunakan, seperti pada penelitian ini digunakan 4 data sekuensial untuk meramalkan data ke-lima. Hal ini ditujukan untuk menguji seberapa kompleks fitur yang dibutuhkan untuk mendapatkan hasil peramalan terbaik.
3. Diujicobakan pada model peramalan *semi-supervised*, dimana hanya terdapat 1 dataset yang diatur sedemikian hingga membentuk pola data fitur dan data aktual. Dari pengaturan tersebut nantinya terdapat pola dengan bagian kosong yang akan diramalkan nilainya berdasarkan pola penuh sebelumnya.

DAFTAR PUSTAKA

- Departemen Perdagangan Republik Indonesia, 2009. *Berjangka*. November penyunt. Jakarta: Badan Pengawas Perdagangan Berjangka Komoditi.
- Dwi, N., 2015. Penerapan Algoritma Support Vector Machine untuk Prediksi Harga Emas. *Jurnal Informatika*, Volume 1, pp. 8-10.
- Furi, R. P., J. & Saepudin, D., 2015. *Prediksi Financial Time Series Menggunakan Independent Component Analysis dan Support Vector Regression Studi Kasus: IHSB dan JII*, Bandung: s.n.
- Godazgar, V. & Leila, H., 2013. *The Mechanisms of the Bullion Markets and the Return of Gold (and Silver) as Money*. Kish Island, IEEE.
- Ismail, J., 2012. *Hukum Jual Beli Komoditi Emas Berjangka (Perspektif Normatif dan Yuridis)*, Yogyakarta: Universitas Islam Negeri Sunan Kalijaga.
- Kurniasih, D., 2013. *Efisiensi Relatif Estimator Fungsi Kernel Gaussian terhadap Estimator Polinomial dalam Peramalan Usd terhadap Jpy*, Semarang: s.n.
- Li, C.-H., Lu, Z.-d. & Zhou, K., 2005. *SVR-Parameters Selection for Image Watermarking*. Wuhan, IEEE.
- Panggabean, V., Nababan, E. & Bu, F., 2013. Analisis Fundamental Dan Analisis Teknikal Pada Investasi Trading Emas Online Dengan Value At Risk. *Saintia Matematika*, Volume 1, pp. 369-382.
- Patro, S. G. K. & Sahu, K. K., 2015. Normalization: A Preprocessing Stage. 19 Maret.
- PT. Mahadana Asta Berjangka, 2016. *Artikel: Investasi Emas*. [Online] Available at: <http://www.mahadana.co.id/id/index.php/artikel/investasi-emas> [Diakses 1 Maret 2016].
- PT. Victory International Futures, 2016. *Our Products: Commodity Futures*. [Online] Available at: <http://www.vifcorps.com/our-products/commodity> [Diakses 10 Februari 2016].
- Suharto, F. T., 2015. *Lebih Mudah dan Untung Berdagang Emas ketimbang Main Forex*. Jakarta: PT Elex Media Komputindo.
- Suranart, K., Kiattisin, S. & Leelasantitham, A., 2014. *Analysis of Comparisons for Forecasting Gold Price using Neural Network, Radial Basis Function Network and Support Vector Regression*. Mahidol, IEEE.
- Syafruddin, M., 2014. *Prediksi Kebutuhan Energi Listrik Jangka Panjang untuk Provinsi Lampung hingga Tahun 2030*, Bandar Lampung: Universitas Lampung.

Vijayakumar, S. & Wu, S., 1999. *Sequential Support Vector Classifiers and Regression*. Genoa, Italy, Saitama: RIKEN Brain Science Institute, The Institute for Physical and Chemical Research, pp. 610-619.



LAMPIRAN A DATA HARGA EMAS

SURAT KETERANGAN PENGAMBILAN DATA

Dalam mendukung penyelesaian skripsi mahasiswa berikut :

Nama : Galuh Widhi Gumilar
 NIM : 125150100111018
 Dosen Pembimbing : 1. Imam Cholissodin, S.Si, M.Kom
 2. Randy Cahya W, S.ST, M.Kom
 Judul Skripsi : Peramalan Harga Emas pada Pasar Berjangka menggunakan
 Algoritma *Support Vector Regression*

Menyatakan bahwa mahasiswa tersebut telah melakukan wawancara dengan staf ahli di PT. Victory International Futures cab. Malang untuk mendapatkan data seperti di bawah ini,

Data Harga Emas tahun 2014

Tanggal	Harga (USD)				
02/01/2014	1219,75	14/02/2014	1308,50	01/04/2014	1286,50
03/01/2014	1232,25	17/02/2014	1326,00	02/04/2014	1284,00
06/01/2014	1238,00	18/02/2014	1314,00	03/04/2014	1287,25
07/01/2014	1237,50	19/02/2014	1318,75	04/04/2014	1293,50
08/01/2014	1226,50	20/02/2014	1313,75	07/04/2014	1299,00
09/01/2014	1226,00	21/02/2014	1320,75	08/04/2014	1314,75
10/01/2014	1232,25	24/02/2014	1333,00	09/04/2014	1309,75
13/01/2014	1246,00	25/02/2014	1332,75	10/04/2014	1321,50
14/01/2014	1248,75	26/02/2014	1340,00	11/04/2014	1317,25
15/01/2014	1238,00	27/02/2014	1331,00	14/04/2014	1324,50
16/01/2014	1237,25	28/02/2014	1327,75	15/04/2014	1311,50
17/01/2014	1241,00	03/03/2014	1344,25	16/04/2014	1299,00
20/01/2014	1254,75	04/03/2014	1339,50	17/04/2014	1299,25
21/01/2014	1247,75	05/03/2014	1333,50	22/04/2014	1290,75
22/01/2014	1239,50	06/03/2014	1334,25	23/04/2014	1283,50
23/01/2014	1244,25	07/03/2014	1348,25	24/04/2014	1283,50
24/01/2014	1259,25	10/03/2014	1334,25	25/04/2014	1294,25
27/01/2014	1270,00	11/03/2014	1348,00	28/04/2014	1302,00
28/01/2014	1253,50	12/03/2014	1355,75	29/04/2014	1289,75
29/01/2014	1254,75	13/03/2014	1371,00	30/04/2014	1292,00
30/01/2014	1254,00	14/03/2014	1370,00	01/05/2014	1283,00
31/01/2014	1246,50	17/03/2014	1379,00	02/05/2014	1285,00
03/02/2014	1246,50	18/03/2014	1362,50	06/05/2014	1308,50
04/02/2014	1253,00	19/03/2014	1346,00	07/05/2014	1311,00
05/02/2014	1257,00	20/03/2014	1327,00	08/05/2014	1291,25
06/02/2014	1258,50	21/03/2014	1338,50	09/05/2014	1289,00
07/02/2014	1260,00	24/03/2014	1322,00	12/05/2014	1292,75
10/02/2014	1273,50	25/03/2014	1314,75	13/05/2014	1292,75
11/02/2014	1282,75	26/03/2014	1314,50	14/05/2014	1300,25
12/02/2014	1286,50	27/03/2014	1295,00	15/05/2014	1303,75
13/02/2014	1290,25	28/03/2014	1295,75	16/05/2014	1293,75
		31/03/2014	1294,00	19/05/2014	1301,00

20/05/2014	1291,50
21/05/2014	1292,00
22/05/2014	1294,50
23/05/2014	1292,00
27/05/2014	1283,00
28/05/2014	1265,25
29/05/2014	1254,00
30/05/2014	1254,00
02/06/2014	1244,75
03/06/2014	1244,25
04/06/2014	1246,00
05/06/2014	1244,75
06/06/2014	1254,00
09/06/2014	1255,00
10/06/2014	1253,50
11/06/2014	1262,50
12/06/2014	1261,75
13/06/2014	1273,00
16/06/2014	1281,75
17/06/2014	1264,50
18/06/2014	1269,00
19/06/2014	1282,00
20/06/2014	1310,00
23/06/2014	1313,50
24/06/2014	1323,00
25/06/2014	1312,00
26/06/2014	1311,50
27/06/2014	1315,25
30/06/2014	1313,00
01/07/2014	1325,75
02/07/2014	1326,75
03/07/2014	1322,50
04/07/2014	1321,50
07/07/2014	1313,25
08/07/2014	1318,25
09/07/2014	1322,50
10/07/2014	1343,25
11/07/2014	1336,50
14/07/2014	1321,25
15/07/2014	1312,00
16/07/2014	1297,50
17/07/2014	1302,75
18/07/2014	1310,25
21/07/2014	1312,75

22/07/2014	1307,00
23/07/2014	1307,50
24/07/2014	1300,00
25/07/2014	1292,50
28/07/2014	1305,00
29/07/2014	1307,50
30/07/2014	1297,50
31/07/2014	1295,00
01/08/2014	1284,50
04/08/2014	1293,50
05/08/2014	1292,75
06/08/2014	1288,50
07/08/2014	1302,00
08/08/2014	1317,50
11/08/2014	1308,25
12/08/2014	1311,00
13/08/2014	1309,25
14/08/2014	1315,00
15/08/2014	1313,60
18/08/2014	1302,75
19/08/2014	1300,25
20/08/2014	1294,50
21/08/2014	1280,50
22/08/2014	1281,00
26/08/2014	1286,50
27/08/2014	1285,00
28/08/2014	1288,00
29/08/2014	1285,75
01/09/2014	1287,25
02/09/2014	1277,75
03/09/2014	1268,50
04/09/2014	1271,00
05/09/2014	1264,00
08/09/2014	1267,25
09/09/2014	1256,00
10/09/2014	1254,25
11/09/2014	1247,00
12/09/2014	1237,25
15/09/2014	1234,75
16/09/2014	1238,75
17/09/2014	1236,50
18/09/2014	1223,00
19/09/2014	1222,50
22/09/2014	1214,00

23/09/2014	1225,00
24/09/2014	1224,00
25/09/2014	1210,50
26/09/2014	1222,25
29/09/2014	1217,75
30/09/2014	1210,00
01/10/2014	1208,50
02/10/2014	1214,50
03/10/2014	1207,50
06/10/2014	1193,25
07/10/2014	1207,50
08/10/2014	1220,00
09/10/2014	1227,50
10/10/2014	1222,25
13/10/2014	1228,00
14/10/2014	1233,00
15/10/2014	1223,50
16/10/2014	1241,00
17/10/2014	1238,00
20/10/2014	1241,00
21/10/2014	1251,75
22/10/2014	1246,75
23/10/2014	1240,50
24/10/2014	1231,75
27/10/2014	1230,50
28/10/2014	1228,25
29/10/2014	1228,00
30/10/2014	1205,75
31/10/2014	1173,25
03/11/2014	1170,75
04/11/2014	1169,25
05/11/2014	1145,25
06/11/2014	1144,50
07/11/2014	1145,00
10/11/2014	1172,00
11/11/2014	1151,25
12/11/2014	1163,25
13/11/2014	1161,00
14/11/2014	1154,00
17/11/2014	1187,00
18/11/2014	1202,00
19/11/2014	1200,75
20/11/2014	1194,00
21/11/2014	1193,25



24/11/2014	1196,00
25/11/2014	1202,25
26/11/2014	1195,75
27/11/2014	1196,50
28/11/2014	1184,50
01/12/2014	1178,75
02/12/2014	1197,00
03/12/2014	1203,25
04/12/2014	1204,00

05/12/2014	1204,50
08/12/2014	1195,25
09/12/2014	1206,50
10/12/2014	1228,25
11/12/2014	1219,50
12/12/2014	1223,50
15/12/2014	1210,75
16/12/2014	1199,25
17/12/2014	1199,00

18/12/2014	1210,75
19/12/2014	1197,50
22/12/2014	1195,25
23/12/2014	1179,50
24/12/2014	1177,00
29/12/2014	1194,00
30/12/2014	1186,50
31/12/2014	1199,25

Data Harga Emas tahun 2015

Tanggal	Harga (USD)
02/01/2015	1184,25
05/01/2015	1192,00
06/01/2015	1211,00
07/01/2015	1213,75
08/01/2015	1206,50
09/01/2015	1211,25
12/01/2015	1222,00
13/01/2015	1239,00
14/01/2015	1228,75
15/01/2015	1235,25
16/01/2015	1258,25
19/01/2015	1275,50
20/01/2015	1292,25
21/01/2015	1298,00
22/01/2015	1287,00
23/01/2015	1293,50
26/01/2015	1282,75
27/01/2015	1279,00
28/01/2015	1287,00
29/01/2015	1275,50
30/01/2015	1263,50
02/02/2015	1274,25
03/02/2015	1281,00
04/02/2015	1269,25
05/02/2015	1263,75
06/02/2015	1264,00
09/02/2015	1242,25
10/02/2015	1237,50
11/02/2015	1235,50
12/02/2015	1225,25
13/02/2015	1225,75
16/02/2015	1233,50

17/02/2015	1221,75
18/02/2015	1206,50
19/02/2015	1217,75
20/02/2015	1203,50
23/02/2015	1193,50
24/02/2015	1195,50
25/02/2015	1206,50
26/02/2015	1220,00
27/02/2015	1205,00
02/03/2015	1216,75
03/03/2015	1207,75
04/03/2015	1204,25
05/03/2015	1199,75
06/03/2015	1196,50
09/03/2015	1173,75
10/03/2015	1161,00
11/03/2015	1158,75
12/03/2015	1161,25
13/03/2015	1156,50
16/03/2015	1157,00
17/03/2015	1154,75
18/03/2015	1149,00
19/03/2015	1164,00
20/03/2015	1171,75
23/03/2015	1181,40
24/03/2015	1193,25
25/03/2015	1192,55
26/03/2015	1209,40
27/03/2015	1198,00
30/03/2015	1187,40
31/03/2015	1179,25
01/04/2015	1181,25
02/04/2015	1201,50

07/04/2015	1208,50
08/04/2015	1211,10
09/04/2015	1196,00
10/04/2015	1201,90
13/04/2015	1197,85
14/04/2015	1191,45
15/04/2015	1189,85
16/04/2015	1204,60
17/04/2015	1204,55
20/04/2015	1203,25
21/04/2015	1197,70
22/04/2015	1202,40
23/04/2015	1187,75
24/04/2015	1192,15
27/04/2015	1182,75
28/04/2015	1201,40
29/04/2015	1204,80
30/04/2015	1204,30
01/05/2015	1179,00
05/05/2015	1187,40
06/05/2015	1191,25
07/05/2015	1183,00
08/05/2015	1185,25
11/05/2015	1184,75
12/05/2015	1184,45
13/05/2015	1193,00
14/05/2015	1214,75
15/05/2015	1216,30
18/05/2015	1228,15
19/05/2015	1219,65
20/05/2015	1206,75
21/05/2015	1209,60
22/05/2015	1211,00



26/05/2015	1194,00
27/05/2015	1187,85
28/05/2015	1189,45
29/05/2015	1190,40
01/06/2015	1187,30
02/06/2015	1188,75
03/06/2015	1186,60
04/06/2015	1182,45
05/06/2015	1175,90
08/06/2015	1173,40
09/06/2015	1181,00
10/06/2015	1186,00
11/06/2015	1180,50
12/06/2015	1179,25
15/06/2015	1178,25
16/06/2015	1182,10
17/06/2015	1178,50
18/06/2015	1198,00
19/06/2015	1198,15
22/06/2015	1193,70
23/06/2015	1183,35
24/06/2015	1175,75
25/06/2015	1174,60
26/06/2015	1174,40
29/06/2015	1176,50
30/06/2015	1175,00
01/07/2015	1171,70
02/07/2015	1164,30
03/07/2015	1168,25
06/07/2015	1164,25
07/07/2015	1166,25
08/07/2015	1154,25
09/07/2015	1162,10
10/07/2015	1162,40
13/07/2015	1154,95
14/07/2015	1153,20
15/07/2015	1154,75
16/07/2015	1145,10
17/07/2015	1143,00
20/07/2015	1115,00
21/07/2015	1108,00
22/07/2015	1096,80
23/07/2015	1101,65
24/07/2015	1083,75

27/07/2015	1098,60
28/07/2015	1095,60
29/07/2015	1096,75
30/07/2015	1085,65
31/07/2015	1080,05
03/08/2015	1093,00
04/08/2015	1092,60
05/08/2015	1086,50
06/08/2015	1085,00
07/08/2015	1091,35
10/08/2015	1094,80
11/08/2015	1113,25
12/08/2015	1116,80
13/08/2015	1117,35
14/08/2015	1116,75
17/08/2015	1117,30
18/08/2015	1119,15
19/08/2015	1123,20
20/08/2015	1137,95
21/08/2015	1149,35
24/08/2015	1153,50
25/08/2015	1154,25
26/08/2015	1134,40
27/08/2015	1128,50
28/08/2015	1125,50
01/09/2015	1141,90
02/09/2015	1140,00
03/09/2015	1130,05
04/09/2015	1125,00
07/09/2015	1121,00
08/09/2015	1120,85
09/09/2015	1122,30
10/09/2015	1107,75
11/09/2015	1106,35
14/09/2015	1108,00
15/09/2015	1105,50
16/09/2015	1109,75
17/09/2015	1118,15
18/09/2015	1136,00
21/09/2015	1136,85
22/09/2015	1129,30
23/09/2015	1124,60
24/09/2015	1134,45
25/09/2015	1145,50

28/09/2015	1137,50
29/09/2015	1124,60
30/09/2015	1122,50
01/10/2015	1114,20
02/10/2015	1106,30
05/10/2015	1134,35
06/10/2015	1136,90
07/10/2015	1147,90
08/10/2015	1143,30
09/10/2015	1151,50
12/10/2015	1164,20
13/10/2015	1154,40
14/10/2015	1173,70
15/10/2015	1183,35
16/10/2015	1176,35
19/10/2015	1171,65
20/10/2015	1173,70
21/10/2015	1174,40
22/10/2015	1166,45
23/10/2015	1171,55
26/10/2015	1168,50
27/10/2015	1165,75
28/10/2015	1171,50
29/10/2015	1159,00
30/10/2015	1147,75
02/11/2015	1135,80
03/11/2015	1130,90
04/11/2015	1118,00
05/11/2015	1107,30
06/11/2015	1107,70
09/11/2015	1095,60
10/11/2015	1092,50
11/11/2015	1088,60
12/11/2015	1087,60
13/11/2015	1083,75
16/11/2015	1094,50
17/11/2015	1080,80
18/11/2015	1070,50
19/11/2015	1073,10
20/11/2015	1085,15
23/11/2015	1068,35
24/11/2015	1073,00
25/11/2015	1072,20
26/11/2015	1070,50



27/11/2015	1064,65
30/11/2015	1055,65
01/12/2015	1069,25
02/12/2015	1066,90
03/12/2015	1050,60
04/12/2015	1063,00
07/12/2015	1082,70
08/12/2015	1071,75

09/12/2015	1078,40
10/12/2015	1072,00
11/12/2015	1067,20
14/12/2015	1068,00
15/12/2015	1069,15
16/12/2015	1065,75
17/12/2015	1065,85
18/12/2015	1055,25

21/12/2015	1071,15
22/12/2015	1077,00
23/12/2015	1071,50
24/12/2015	1071,90
29/12/2015	1067,25
30/12/2015	1067,80
31/12/2015	1062,25

Yang bertandatangan di bawah ini,

Ujang Prasetyo S.E

LAMPIRAN B HASIL PERAMALAN TERBAIK

B.1 Penggunaan *Sequence Data* Latih Peramalan

Tabel berikut merupakan data latih yang menghasilkan peramalan terbaik di mana X1 – X4 merupakan fitur peramalan SVR, dan Y adalah harga aktual.

Tanggal	X1	X2	X3	X4	Y
27/03/2014	1338,50	1322,00	1314,75	1314,50	1295,00
28/03/2014	1322,00	1314,75	1314,50	1295,00	1295,75
31/03/2014	1314,75	1314,50	1295,00	1295,75	1294,00
01/04/2014	1314,50	1295,00	1295,75	1294,00	1286,50
02/04/2014	1295,00	1295,75	1294,00	1286,50	1284,00
03/04/2014	1295,75	1294,00	1286,50	1284,00	1287,25
04/04/2014	1294,00	1286,50	1284,00	1287,25	1293,50
07/04/2014	1286,50	1284,00	1287,25	1293,50	1299,00
08/04/2014	1284,00	1287,25	1293,50	1299,00	1314,75
09/04/2014	1287,25	1293,50	1299,00	1314,75	1309,75
10/04/2014	1293,50	1299,00	1314,75	1309,75	1321,50
11/04/2014	1299,00	1314,75	1309,75	1321,50	1317,25
14/04/2014	1314,75	1309,75	1321,50	1317,25	1324,50
15/04/2014	1309,75	1321,50	1317,25	1324,50	1311,50
16/04/2014	1321,50	1317,25	1324,50	1311,50	1299,00
17/04/2014	1317,25	1324,50	1311,50	1299,00	1299,25
22/04/2014	1324,50	1311,50	1299,00	1299,25	1290,75
23/04/2014	1311,50	1299,00	1299,25	1290,75	1283,50
24/04/2014	1299,00	1299,25	1290,75	1283,50	1283,50
25/04/2014	1299,25	1290,75	1283,50	1283,50	1294,25
28/04/2014	1290,75	1283,50	1283,50	1294,25	1302,00
29/04/2014	1283,50	1283,50	1294,25	1302,00	1289,75
30/04/2014	1283,50	1294,25	1302,00	1289,75	1292,00
01/05/2014	1294,25	1302,00	1289,75	1292,00	1283,00
02/05/2014	1302,00	1289,75	1292,00	1283,00	1285,00
06/05/2014	1289,75	1292,00	1283,00	1285,00	1308,50
07/05/2014	1292,00	1283,00	1285,00	1308,50	1311,00
08/05/2014	1283,00	1285,00	1308,50	1311,00	1291,25
09/05/2014	1285,00	1308,50	1311,00	1291,25	1289,00
12/05/2014	1308,50	1311,00	1291,25	1289,00	1292,75

B.2 Penggunaan *Sequence Data* Uji Peramalan

Tabel berikut merupakan data uji yang menghasilkan peramalan terbaik di mana X1 – X4 merupakan fitur peramalan SVR, dan Y adalah harga aktual.

Tanggal	X1	X2	X3	X4	Y
13/05/2014	1311,00	1291,25	1289,00	1292,75	1292,75
14/05/2014	1291,25	1289,00	1292,75	1292,75	1300,25
15/05/2014	1289,00	1292,75	1292,75	1300,25	1303,75
16/05/2014	1292,75	1292,75	1300,25	1303,75	1293,75
19/05/2014	1292,75	1300,25	1303,75	1293,75	1301,00
20/05/2014	1300,25	1303,75	1293,75	1301,00	1291,50
21/05/2014	1303,75	1293,75	1301,00	1291,50	1292,00
22/05/2014	1293,75	1301,00	1291,50	1292,00	1294,50
23/05/2014	1301,00	1291,50	1292,00	1294,50	1292,00
27/05/2014	1291,50	1292,00	1294,50	1292,00	1283,00

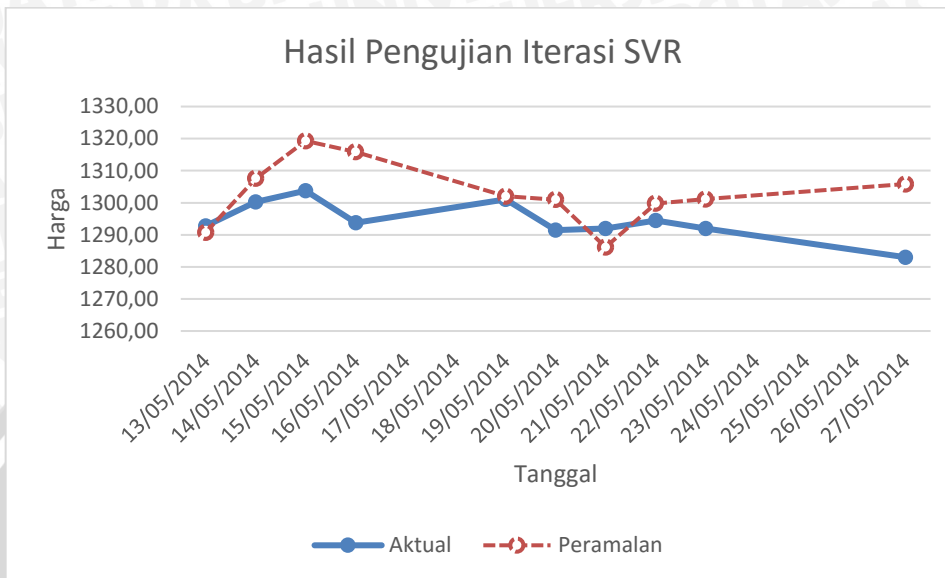
B.3 Nilai Lagrange Multiplier Terbaik pada Fungsi Regresi SVR

Tabel berikut menampilkan nilai Lagrange Multiplier α_i^* dan α_i yang memberikan hasil peramalan terbaik yang didapat dari iterasi *sequential learning* terakhir yaitu 1000:

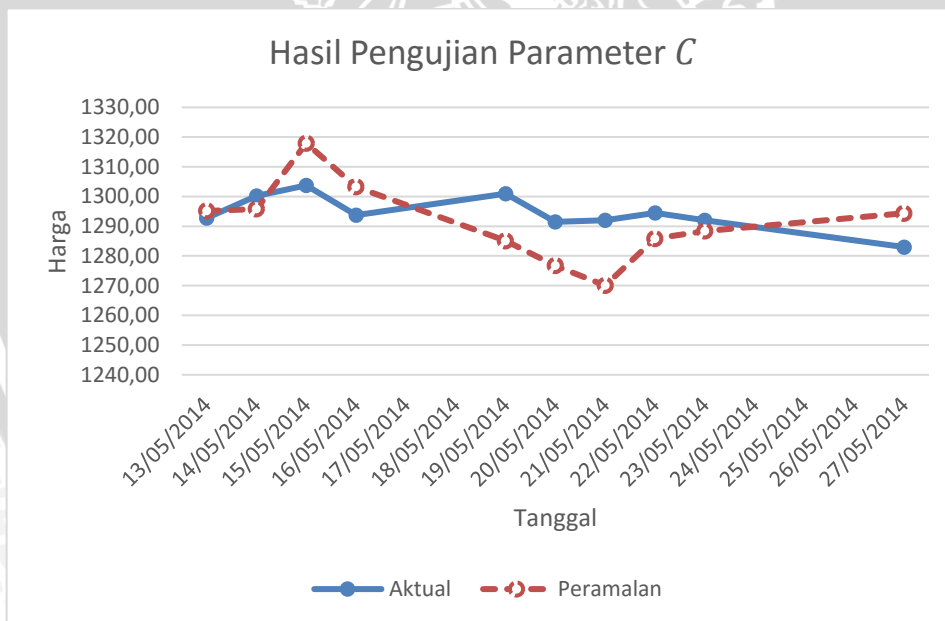
No.	α_i^*	α_i	No.	α_i^*	α_i
1	7,883731	0	16	0,601116	0
2	2,018709	0	17	0	25,17951
3	0	5,328567	18	0	21,63526
4	0	22,24343	19	0	0,144939
5	0	7,516615	20	31,71039	0
6	9,529028	0	21	22,878	0
7	18,45993	0	22	0	36,11024
8	16,24694	0	23	6,68393	0
9	46,6109	0	24	0	26,55544
10	0	14,5649	25	5,793727	0
11	34,77463	0	26	69,40016	0
12	0	12,35618	27	7,496728	0
13	21,47693	0	28	0	58,23027
14	0	38,36651	29	0	6,637564
15	0	36,95752	30	10,99214	0

LAMPIRAN C VISUALISASI HASIL PENGUJIAN

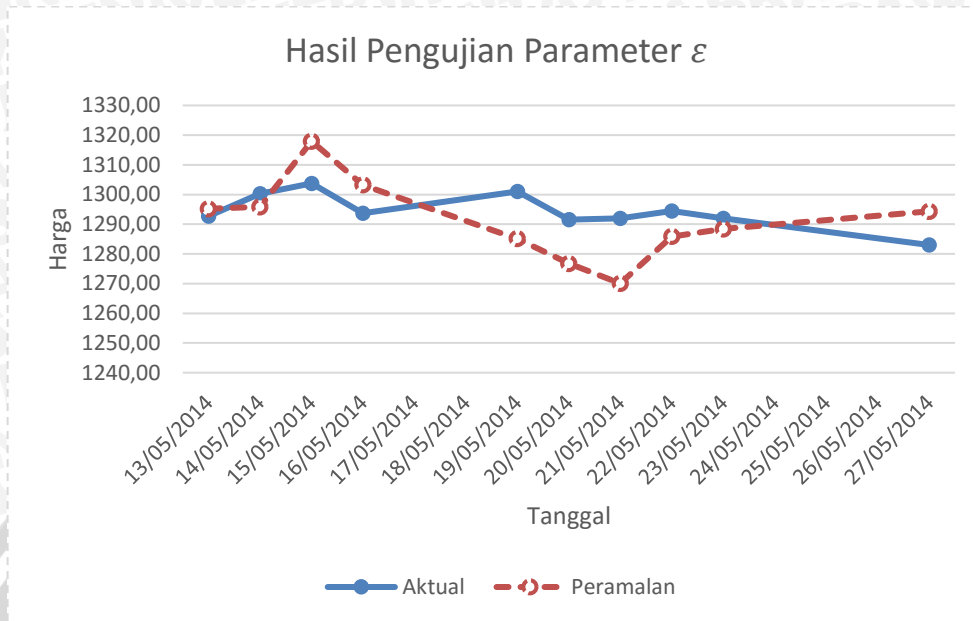
C.1 Visualisasi Hasil Pengujian Jumlah Iterasi terbaik SVR



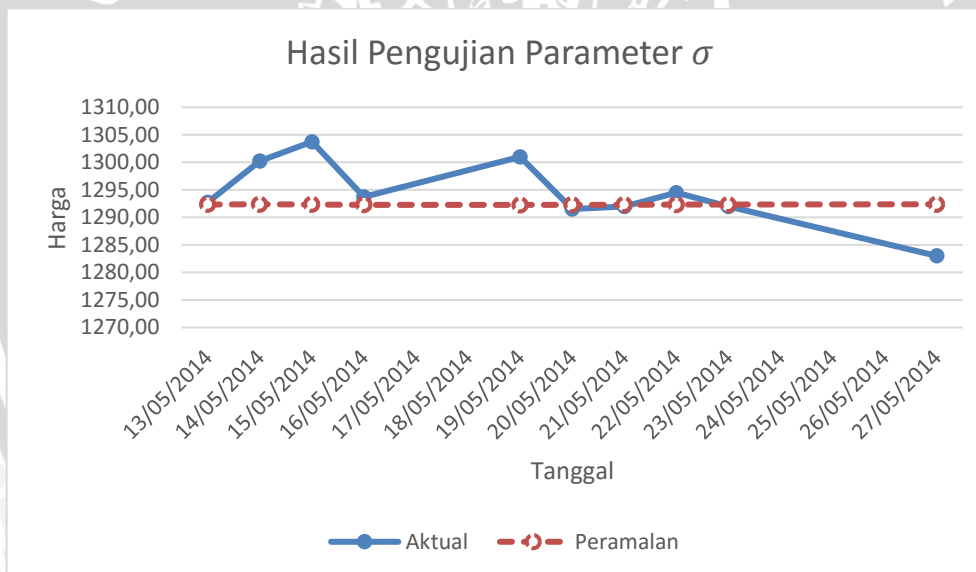
C.2 Visualisasi Hasil Pengujian Nilai Parameter C (Complexity)



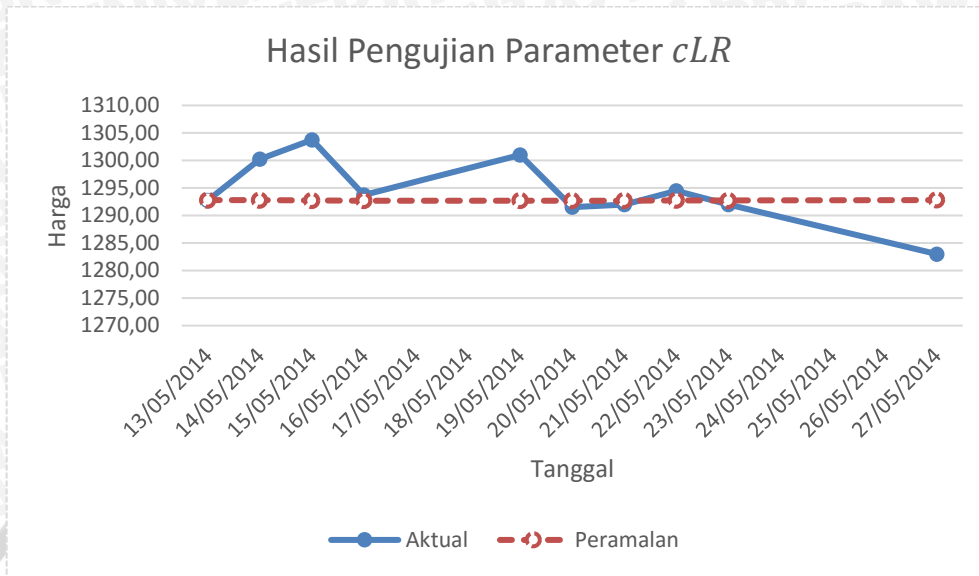
C.3 Visualisasi Hasil Pengujian Nilai Parameter ϵ (epsilon)



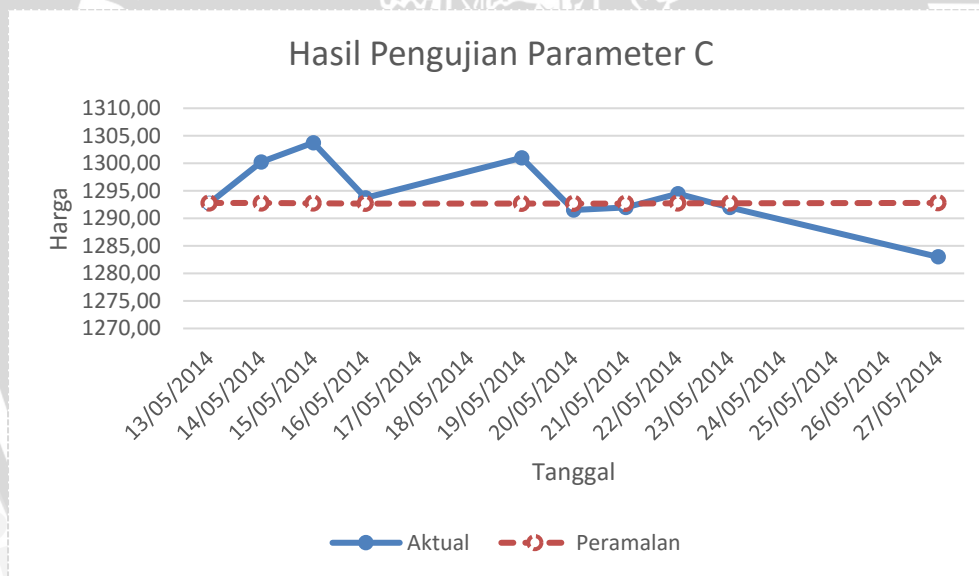
C.4 Visualisasi Hasil Pengujian Nilai Parameter σ (sigma)



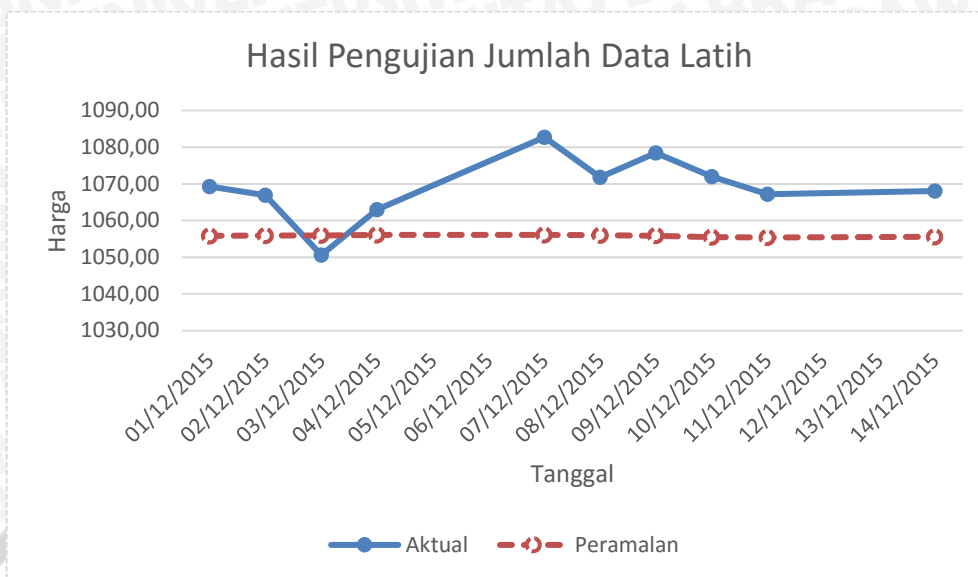
C.5 Visualisasi Hasil Pengujian Nilai Parameter cLR (Learning Rate)



C.6 Visualisasi Hasil Pengujian Nilai Parameter λ (lambda)



C.7 Visualisasi Hasil Pengujian Variasi Jumlah Data Latih



C.8 Visualisasi Hasil Pengujian Variasi Jumlah Data Uji

