

TUGAS AKHIR - SS234862

***FORECASTING INDEKS HARGA KONSUMEN KOTA
SURABAYA MENGGUNAKAN METODE *GENETIC
ALGORITHM-SUPPORT VECTOR REGRESSION****

PUTRI HERLIANI NABILLA

NRP 5003201014

Dosen Pembimbing

Irhamah, S.Si., M.Si., Ph.D.

NIP 19780406 200112 2 002

Program Studi Sarjana Statistika

Departemen Statistika

Fakultas Sains dan Analitika Data

Institut Teknologi Sepuluh Nopember

Surabaya

2024



TUGAS AKHIR - SS234862

***FORECASTING INDEKS HARGA KONSUMEN KOTA
SURABAYA MENGGUNAKAN METODE GENETIC
ALGORITHM-SUPPORT VECTOR REGRESSION***

PUTRI HERLIANI NABILLA

NRP 5003201014

Dosen Pembimbing

Irhamah, S.Si., M.Si., Ph.D.

NIP 19780406 200112 2 002

Program Studi Sarjana Statistika

Departemen Statistika

Fakultas Sains dan Analitika Data

Institut Teknologi Sepuluh Nopember

Surabaya

2024



FINAL PROJECT - SS234862

**FORECASTING CONSUMER PRICE INDEX OF
SURABAYA USING METHOD GENETIC ALGORITHM-
SUPPORT VECTOR REGRESSION**

PUTRI HERLIANI NABILLA

NRP 5003201014

Advisor

Irhamah, S.Si., M.Si., Ph.D.

NIP 19780406 200112 2 002

Undergraduate Study Program of Statistics

Department of Statistics

Faculty of Science and Data Analytics

Institut Teknologi Sepuluh Nopember

Surabaya

2024

LEMBAR PENGESAHAN

FORECASTING INDEKS HARGA KONSUMEN KOTA SURABAYA MENGUNAKAN METODE *GENETICS ALGORITHM-SUPPORT VECTOR REGRESSION*

TUGAS AKHIR


Diajukan untuk memenuhi salah satu syarat
memperoleh gelar Sarjana Statistika pada
Program Studi S-1 Statistika
Departemen Statistika
Fakultas Sains dan Analitika Data
Institut Teknologi Sepuluh Nopember

Oleh : **PUTRI HERLIANI NABILLA**
NRP. 5003201014

Disetujui oleh :

Pembimbing :

1. **Irhamah, S.Si., M.Si., Ph.D.**
NIP. 19780406 200112 2 002

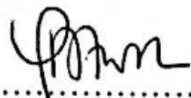

.....

Penguji :

2. **Prof. Dr.rer.pol Heri Kuswanto, S.Si., M.Si.**
NIP. 19820326 200312 1 002


.....

3. **Widhianingsih Tintrim Dwi Ary, S.Si., M.Stat., Ph.D.**
NIP. 2022199512001


.....

**Kepala Departemen Statistika
Fakultas Sains dan Analitika Data**



Dr. Dra. Kartika Fithriasari, M.Si.

NIP. 19691212 199303 2 002

(Halaman ini sengaja dikosongkan)

APPROVAL SHEET

FORECASTING CONSUMER PRICE INDEX OF SURABAYA USING METHOD GENETIC ALGORITHM-SUPPORT VECTOR REGRESSION

FINAL PROJECT


Submitted to fulfill one of the requirements
for obtaining a degree in Bachelor of Statistics at
Undergraduate Program of Statistics
Department of Statistics
Faculty of Science and Data Analytics
Institut Teknologi Sepuluh Nopember

by : **PUTRI HERLIANI NABILLA**
NRP. 5003201014

Approved by :

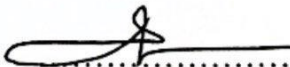
Advisor :

1. **Irhamah, S.Si., M.Si., Ph.D.**
NIP. 19780406 200112 2 002

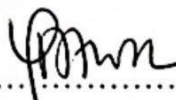

.....

Examiners :

2. **Prof. Dr.rer.pol Heri Kuswanto, S.Si., M.Si.**
NIP. 19820326 200312 1 002


.....

3. **Widhianingsih Tintrim Dwi Ary, S.Si., M.Stat., Ph.D.**
NIP. 2022199512001


.....

Head of Statistics Department
Faculty of Science and Data Analytics



Dr. Dra. Kartika Fithriasari, M.Si.

NIP. 19691212 199303 2 002

(Halaman ini sengaja dikosongkan)

PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini:

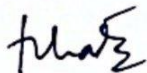
Nama mahasiswa / NRP : Putri Herliani Nabilla / 5003201014
Program studi : Statistika
Dosen Pembimbing / NIP : Irhamah, S.Si., M.Si., Ph.D.

dengan ini menyatakan bahwa Tugas Akhir dengan judul “*Forecasting Indeks Harga Konsumen Kota Surabaya Menggunakan Metode Algorithm Support Vector Regression*” adalah hasil karya sendiri, bersifat orisinal, dan ditulis dengan mengikuti kaidah penulisan ilmiah.

Bilamana di kemudian hari ditemukan ketidaksesuaian dengan pernyataan ini, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku di Institut Teknologi Sepuluh Nopember.

Surabaya, 13 Juni 2024

Mengetahui
Dosen Pembimbing



Irhamah, S.Si., M.Si., Ph.D.
NIP. 19780406 200112 2 002

Mahasiswa



Putri Herliani Nabilla
NRP. 5003201014

(Halaman ini sengaja dikosongkan)

STATEMENT OF ORIGINALITY

The undersigned below :

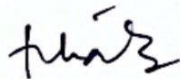
Name of student / NRP : Putri Herliani Nabilla / 5003201014
Department : Statistics
Advisor / NIP : Irhamah, S.Si., M.Si., Ph.D.

hereby declare that the Final Project with the title of “**Forecasting Consumer Price Index of Surabaya Using Method Genetic Algorithm-Support Vector Regression**” is the result of my own work, is original, and is written by following the rules of scientific writing.

If in the future there is discrepancy with this statement, then I am willing to accept sanctions in accordance with the provisions that apply at Institut Teknologi Sepuluh Nopember.

Surabaya, 13th June 2024

Acknowledged
Advisor



Irhamah, S.Si., M.Si., Ph.D.
NIP. 19780406 200112 2 002

Student



Putri Herliani Nabilla
NRP. 5003201014

(Halaman ini sengaja dikosongkan)

ABSTRAK

FORECASTING INDEKS HARGA KONSUMEN KOTA SURABAYA MENGUNAKAN METODE GENETIC ALGORITHM SUPPORT VECTOR REGRESSION

Nama Mahasiswa / NRP : Putri Herliani Nabilla / 5003201014
Departemen : Statistika FSAD - ITS
Dosen Pembimbing : Irhamah, S.Si., M.Si., Ph.D.

Abstrak

Indeks Harga Konsumen menunjukkan ukuran harga rata-rata barang maupun jasa yang dikonsumsi oleh rumah tangga dalam kurun waktu tertentu. Indeks ini dipengaruhi berbagai faktor mulai dari gejolak harga jenis barang yang tidak menentu, nilai tukar rupiah, tingkat inflasi, kebijakan pemerintah dan permintaan masyarakat. Pertumbuhan angka Indeks Harga Konsumen dapat menggambarkan harga barang atau jasa kebutuhan masyarakat yang dapat mempengaruhi perekonomian negara. Salah satu metode yang digunakan untuk *forecasting* pergerakan nilai prediksi ini adalah *Support Vector Regression* (SVR). SVR merupakan metode menganalisis dengan mencari *hyperplane* terbaik suatu fungsi regresi. Dalam kasus ini menggunakan pendekatan *kernel* linear, *polynomial*, dan RBF, dimana dilakukan hibridisasi algoritma SVR dengan optimasi menggunakan *Genetic Algorithm* (GA) untuk meningkatkan akurasi peramalan. Kriteria evaluasi model menggunakan RMSE dan sMAPE selama enam bulan hingga menghasilkan model terbaik. Hasil peramalan cukup akurat pada model terbaik GA-SVR dengan parameter nilai RMSE dan sMAPE baik *in sample* maupun *out sample* lebih kecil dibandingkan SVR sebelum optimasi. Berdasarkan hasil peramalan diketahui bahwa Indeks Harga Konsumen Kota Surabaya menunjukkan stabilitas penurunan pada enam bulan mendatang. Dengan demikian, metode ini sudah tepat digunakan untuk meramalkan Indeks Harga Konsumen Kota Surabaya.

Kata kunci: *Forecasting, Indeks Harga Konsumen, Support Vector Regression, Genetic Algorithm.*

(Halaman ini sengaja dikosongkan)

ABSTRACT

FORECASTING CONSUMER PRICE INDEX OF SURABAYA USING METHOD GENETIC ALGORITHM-SUPPORT VECTOR REGRESSION

Student Name / NRP : Putri Herliani Nabilla / 5003201014
Department : Statistika FSAD - ITS
Advisor : Irhamah, S.Si., M.Si., Ph.D.

Abstract

The consumer price index shows a measure of the average price of goods and services consumed by households in a certain period of time. This index is influenced by various factors ranging from volatile price fluctuations for various types of goods, the rupiah exchange rate, inflation rate, government policy and public demand. The growth in consumer price index figures can describe the prices of goods or services needed by society. One of the methods used for forecasting the movement of predicted values is Support Vector Regression (SVR). SVR is an analysis method by finding the best hyperplane of regression function. In this case, used by a linear, polynomial and RBF kernel approach, where the SVR algorithm is hybridized with optimization using a Genetic Algorithm (GA) to increase forecasting accuracy. The model evaluation criteria used RMSE and sMAPE for six months to produce the best model. The forecasting results are accurate in the best GA-SVR model with the smallest RMSE and sMAPE parameter values both in sample and out sample compared to SVR before optimization. Based on forecasting results, it is known that the Surabaya consumer price index shows a stable decline for the next six months. Thus, this method is appropriate use to predict the consumer price index of Surabaya.

Keywords: Forecasting, Consumer Price index, Support Vector Regression, Genetic Algorithm.

(Halaman ini sengaja dikosongkan)

KATA PENGANTAR

Alhamdulillah, segala puji bagi Allah SWT yang telah melimpahkan karunia nikmat dan rahmat serta hidayah-Nya kepada penulis sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul “*Forecasting* Indeks Harga Konsumen Kota Surabaya Menggunakan Metode *Genetic Algorithm-Support Vector Regression*” dengan baik dan lancar. Kelancaran dalam penyusunan Tugas Akhir ini tidak lepas dari berbagai bantuan, dukungan, dan doa dari berbagai pihak. Oleh karena itu, penulis menyampaikan rasa terima kasih kepada:

1. Kedua orang tua penulis, mama tercinta Sumarlin dan ayah Heri Siswanto yang selalu memberikan dukungan, doa, serta kasih sayang untuk penulis sehingga dapat menjadi motivasi bagi penulis dalam menyelesaikan Tugas Akhir ini.
2. Kakak laki-laki penulis yang penulis banggakan dan selalu menjadi *role model* bagi penulis, Adimas Yahya yang telah memberikan berbagai dukungan afirmasi selama pengerjaan hingga penyelesaian Tugas Akhir, adik penulis tersayang Andhini Rahmadani yang selalu memberi tawa dan menemani saat penulis merasa sendiri.
3. Ibu Dr. Dra. Kartika Fithriasari, M.Si. selaku Kepala Departemen Statistika ITS, Ibu Santi Wulan Purnami, M.Si., Ph.D. selaku Sekretaris Departemen I dan Prof. Dr. Vita Ratnasari, S.Si, M.Si. selaku Sekretaris Departemen II yang telah memfasilitasi sarana dan prasarana untuk kegiatan belajar dan mengajar selama studi di ITS.
4. Ibu Irhamah, S.Si., M.Si., Ph.D. selaku dosen pembimbing utama yang penuh keikhlasan dan kesabaran telah meluangkan waktu, mengarahkan, dan membimbing serta memberikan saran dan masukan kepada penulis selama penyusunan Tugas Akhir.
5. Prof Dr.rer.pol Heri Kuswanto, S.Si., M.Si. selaku dosen penguji pertama atas arahan, saran dan kritik yang membangun terkait penulisan Tugas Akhir.
6. Ibu Widhianingsih Tintrim Dwi Ary, S.Si., M.Stat., Ph.D. selaku dosen penguji kedua atas arahan, saran dan kritik yang membangun terkait penulisan Tugas Akhir.
7. Bapak Dr. Sutikno, S.Si., M.Si. selaku dosen wali yang selalu memberikan bantuan, arahan, dan memberikan banyak masukan yang membangun kepada penulis.
8. Seluruh dosen Statistika ITS yang telah mencurahkan ilmu dan pengalaman yang bermanfaat kepada penulis serta segenap karyawan Departemen Statistika ITS atas segala dukungan dan bantuan yang diberikan kepada penulis.
9. Para sahabat penulis Nevisra Najwa, Annisa Nathania, Rahma Zidny yang senantiasa mendampingi perjalanan penulis selama belajar di Departemen Statistika ITS serta memberikan dukungan dan semangat dalam penyelesaian laporan Tugas Akhir.
10. Teman-teman Pekan Raya Statistika 2023 terutama Farsya Riyanti, Nidya Putri, Turfah Latifah, Dwima Ardhia, Adristy Rizki dan seluruh panitia yang telah berjuang dalam pelaksanaan *big event* Statistika hingga memberikan banyak pelajaran, ilmu, dan pengalaman yang berkesan bagi Penulis.
11. Teman-teman Statistika ITS Angkatan 2020 “Aryasena” yang telah berjuang bersama dari awal perkuliahan hingga akhir.
12. Semua pihak yang turut membantu dalam pelaksanaan Tugas Akhir yang tidak dapat penulis sebutkan satu persatu dimana telah membantu dalam pengerjaan Tugas Akhir penulis baik secara langsung maupun tidak langsung.

Penulis menyadari bahwa dalam penulisan tugas akhir ini masih jauh dari kata sempurna. Ketidaksempurnaan tersebut menjadi celah untuk pembelajaran dan pengembangan keilmuan kedepannya. Semoga Tugas Akhir ini bisa menjadi bahan referensi dan memberikan manfaat bagi pengembangan keilmuan statistika.

Surabaya, Juli 2024

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	i
APPROVAL SHEET	iii
PERNYATAAN ORISINALITAS	v
STATEMENT OF ORIGINALITY	vii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah	5
1.4 Tujuan	5
1.5 Manfaat	5
BAB 2 TINJAUAN PUSTAKA	7
2.1 Penelitian Terdahulu	7
2.2 Uji Nonlinearitas Terasvirta	7
2.3 <i>Support Vector Regression</i>	8
2.4 Fungsi <i>Kernel</i>	12
2.5 <i>Grid Search</i>	13
2.6 <i>Genetic Algorithm</i>	13
2.7 Evaluasi Model	17
2.8 <i>Forecasting</i>	17
2.9 Indeks Harga Konsumen	18
BAB 3 METODOLOGI	19
3.1 Sumber Data	19
3.2 Variabel Penelitian	19
3.3 Langkah Analisis	19
3.4 Diagram Alir	21
BAB 4 HASIL DAN PEMBAHASAN	25
4.1 Karakteristik IHK Kota Surabaya	25

4.2	Pengujian Nonlinearitas Terasvirta.....	28
4.3	Pemodelan IHK Kota Surabaya Menggunakan <i>Support Vector Regression</i>	28
4.3.1	Penentuan Data <i>Input</i> dan Data <i>Output</i>	29
4.3.2	Penentuan Parameter Optimal	29
4.4	Pemodelan IHK Kota Surabaya Menggunakan <i>Genetic Algorithm-Support Vector Regression</i>	33
4.5	<i>Forecasting</i> IHK Kota Surabaya Menggunakan Model Terbaik	37
BAB 5	KESIMPULAN DAN SARAN.....	41
5.1	Kesimpulan	41
5.2	Saran	42
	DAFTAR PUSTAKA	43
	LAMPIRAN.....	47
	BIODATA PENULIS	69

DAFTAR GAMBAR

Gambar 2.1 Ilustrasi Error SVR	9
Gambar 2.2 Kernel Mapping	12
Gambar 3.1 Diagram Alir Penelitian	21
Gambar 3.2 Diagram Alir Tahapan SVR.....	22
Gambar 3.3 Diagram Alir Tahapan GA-SVR	23
Gambar 4.1 Time Series Plot IHK.....	25
Gambar 4.2 <i>Box-Cox Plot</i> IHK.....	27
Gambar 4.3 Transformasi <i>Box-Cox</i> IHK	27
Gambar 4.4 (a) Plot ACF Sebelum <i>Differencing</i> dan (b) Plot ACF <i>Differencing</i> Orde 1	27
Gambar 4.5 Plot PACF Data IHK Kota Surabaya.....	28
Gambar 4.6 <i>Time Series Plot</i> Model SVR Terbaik pada Data (a) <i>In Sample</i> dan (b) <i>Out Sample</i>	32
Gambar 4.7 Ilustrasi Kromosom Parameter SVR Tiga Gen.....	33
Gambar 4.8 Ilustrasi Pindah Silang	34
Gambar 4.9 Ilustrasi Kromosom Melakukan <i>Crossover</i>	34
Gambar 4.10 Ilustrasi Proses Mutasi	35
Gambar 4.11 Nilai <i>Fitness</i> Konvergen	35
Gambar 4.12 <i>Time Series Plot</i> Model GA-SVR Terbaik Data (a) <i>In Sample</i> dan (b) <i>Out Sample</i>	36
Gambar 4.13 Perbandingan Visual Data <i>Out Sample</i> IHK Kota Surabaya	37
Gambar 4.14 Hasil <i>Forecasting</i> IHK 6 Bulan Mendatang	38
Gambar 4.15 <i>Forecasting Plot</i> Model terbaik	39

(Halaman ini sengaja dikosongkan)

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	7
Tabel 3.1 Struktur Data	19
Tabel 4.1 Hasil Uji Terasvirta	28
Tabel 4.2 <i>Range</i> Parameter	29
Tabel 4.3 Kombinasi Nilai Parameter <i>Kernel Linear</i>	29
Tabel 4.4 Kombinasi Nilai Parameter <i>Kernel Polynomial</i>	30
Tabel 4.5 Kombinasi Nilai Parameter <i>Kernel RBF</i>	31
Tabel 4.6 Nilai RMSE Kombinasi Parameter Terbaik.....	32
Tabel 4.7 Parameter Optimum GA-SVR	36
Tabel 4.8 Pemilihan Model Terbaik <i>Forecasting</i> IHK Kota Surabaya.....	37
Tabel 4.9 Hasil <i>Forecast</i> IHK Kota Surabaya 6 Bulan.....	38

(Halaman ini sengaja dikosongkan)

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Masalah perekonomian terbesar suatu negara biasanya disebabkan oleh inflasi. Inflasi terjadi ketika laju kenaikan harga barang dan jasa terus meningkat dalam kurun waktu tertentu. Kenaikan harga barang dan jasa dari waktu ke waktu dapat meningkat secara terus menerus apabila tidak dikendalikan dengan baik. Penyebab kenaikan harga barang dan jasa karena adanya perkembangan harga komoditas pasar yang semakin meningkat dengan adanya pendugaan dari penekanan pertumbuhan pasar. Salah satu indikator yang digunakan untuk melihat keberhasilan moneter dalam mengendalikan inflasi adalah Indeks Harga Konsumen (IHK). IHK merupakan angka indeks yang menunjukkan tingkat barang dan jasa yang dibeli konsumen dalam suatu periode tertentu (Mankiw, Quah, & Wilson, 2012). IHK digunakan untuk mengelola data-data yang ada sehingga dapat mengetahui perkembangan usaha atau kegiatan yang dilakukan, contohnya untuk mengetahui faktor-faktor yang mempengaruhi kemajuan ekonomi, sebagai ukuran tingkat kemajuan ekonomi, atau sebagai alat bagi pemerintah untuk menetapkan harga yang sesuai standar.

Perhitungan IHK sangat penting dalam memperlihatkan tingkat harga suatu barang dan atau jasa yang dibeli masyarakat. IHK mampu menggambarkan tingkat kenaikan (inflasi) apabila angka IHK naik dan menggambarkan tingkat penurunan (deflasi) dari suatu barang dan atau jasa apabila angka IHK turun dari periode sebelumnya. Inflasi dan deflasi keduanya sangat berdampak dalam perekonomian suatu negara. Hasil perhitungan IHK berasal dari hasil observasi harga komoditas di tingkat konsumen rumah tangga pada jangka waktu harian, mingguan, serta bulanan. Komoditas pangan sebagai bahan integral dari IHK yang memiliki dampak signifikan pada kehidupan sehari-hari. Inflasi tidak terjadi apabila kenaikan harga yang dialami hanya pada satu atau dua barang saja, kecuali apabila kenaikan itu menyebar atau mengakibatkan kenaikan harga pada barang lainnya. Kenaikan harga barang dan jasa tersebut menyebabkan penurunan nilai mata uang. Menurunnya daya beli mata uang mengakibatkan kemampuan masyarakat berpenghasilan tetap dalam membeli barang dan jasa kebutuhan sehari-hari akan semakin rendah. Laju kenaikan harga yang tidak stabil juga menyulitkan perencanaan bagi dunia usaha dan menyebabkan dampak negatif lainnya yang tidak kondusif bagi perekonomian secara keseluruhan. Adapun memprediksi IHK beberapa periode ke depan merupakan salah satu usaha dalam mengendalikan inflasi di masa mendatang serta sebagai masukan dalam memecahkan masalah perekonomian.

Badan Pusat Statistik (BPS) secara rutin setiap bulannya menghitung dan melaporkan indeks harga konsumen, Penghitungan IHK dimulai dengan mengumpulkan harga dari ribuan barang dan jasa, jika Produk Domestik Bruto (PDB) mengubah jumlah berbagai barang dan jasa menjadi sebuah angka tunggal yang mengukur nilai produksi, maka IHK mengubah berbagai harga barang dan jasa menjadi sebuah indeks tunggal yang mengukur seluruh tingkat harga. Laporan IHK ini digunakan untuk memantau perubahan biaya hidup dari waktu ke waktu, perubahan IHK menggambarkan kenaikan atau penurunan harga barang atau jasa yang dibutuhkan oleh rumah tangga. IHK dapat diartikan sebagai indeks harga dari biaya sekumpulan barang konsumsi yang masing-masing diberi bobot menurut proporsi belanja masyarakat untuk komoditi yang bersangkutan, diantaranya adalah bahan makanan pokok, sandang, perumahan, dan aneka barang dan jasa yang dibeli konsumen. Perhitungan IHK sangatlah penting dalam kehidupan sehari-hari terutama dalam perekonomian, karena ini

menyangkut biaya hidup yang dikeluarkan setiap orang untuk barang dan jasa tertentu yang dikonsumsi, serta dapat mengetahui gambaran inflasi atau deflasi suatu barang dan jasa. Besarnya pengaruh IHK terhadap laju inflasi ekonomi, maka yang selanjutnya akan berdampak besar terhadap maju tidaknya perekonomian di suatu kota.

Konsumen rumah tangga dapat mengetahui perubahan rata-rata dalam harga barang atau jasa selama periode waktu tertentu dengan menggunakan indeks harga. Bila perubahan IHK meningkat dalam jangka waktu tertentu dikatakan inflasi, dan deflasi terjadi ketika perubahan turun. Di samping menghitung tingkat inflasi, upah, tingkat gaji, dan besaran pensiun didasarkan pada IHK juga. Badan Pusat Statistik (BPS) di Indonesia ditugaskan untuk menghitung IHK. Menurut Indeks Harga Konsumen (IHK), kenaikan harga barang dan jasa yang terjadi dari waktu ke waktu dan menjadi permasalahan yang menghambat 5 konsumsi dan kegiatan usaha. IHK adalah harga sekelompok barang dan jasa dibandingkan dengan harga kelompok barang dan jasa yang sama pada tahun acuan, sehingga mengakibatkan berkurangnya daya beli konsumen dari segi kuantitas dan kualitas barang serta dalam rangka memenuhi kebutuhan subsisten mereka. Pelayanan yang diinginkan tidak lagi tersedia, sehingga mengakibatkan penurunan kesejahteraan. Meningkatnya IHK juga berdampak negatif pada perusahaan yang memproduksi dan mendistribusikan barang dan jasa. Menurunnya daya beli pelanggan akibatnya menyebabkan naiknya IHK akan mempengaruhi produk tidak terjual perusahaan dalam aktivitas pasar produk dan jasanya, menurunkan keuntungan yang diperoleh, dan akibatnya menurunkan kemampuan finansial untuk menutupi biaya operasional usaha. Akibat kenaikan harga yang terkait dengan kenaikan IHK sebagai indikator inflasi, maka daya beli pelaku usaha terhadap sumber daya yang dimiliki konsumen menurun. IHK umumnya menghitung biaya barang dan jasa tertentu yang dibayar oleh pelanggan.

Di Indonesia, perhitungan inflasi dilakukan Badan Pusat Statistik dimana setiap bulan BPS mempublikasikan tingkat inflasi dan nilai IHK dalam suatu periode. Perhitungan IHK didasarkan dari hasil survei harga biaya hidup yang dikonsumsi atau dibutuhkan dalam kebutuhan rumah tangga di suatu daerah (Suarsih & Achsani, 2016). Dalam menghasilkan nilai IHK, BPS melakukan *survey* di beberapa daerah perwakilan di Indonesia. Sebelum tahun 2014 inflasi dihitung dari 66 kota di Indonesia sedangkan sejak tahun 2014 perhitungan inflasi menggunakan data 82 kota di Indonesia. Pengukuran IHK berdasarkan *the Classification of Individual Consumption by Purpose* (COICOP), yaitu terdiri dari tujuh kelompok pengeluaran. Tujuh kelompok pengeluaran tersebut, yaitu kelompok bahan makanan, makanan jadi, minuman dan tembakau, perumahan, sandang, pendidikan, kesehatan, serta transportasi dan komunikasi (BPS, 2022). Di Jawa Timur BPS mengambil delapan kota dalam perhitungan statistik inflasi, yaitu Kota Surabaya, Kota Madiun, Kota Kediri, Kabupaten Sumenep, Kabupaten Jember, Kota Probolinggo, Kabupaten Banyuwangi, dan Kota Malang.

Badan Pusat Statistik Provinsi Jawa Timur pada Desember 2023 mencatat terdapat inflasi *year on year* pada gabungan delapan kota sebesar 2,92%. Sementara tingkat inflasi *month to month* gabungan delapan kota sebesar 0,29% dengan IHK sebesar 117,59. Hal ini tercatat pada level kota dengan salah satu inflasi tertinggi terjadi di Kota Surabaya sebesar 3,03% dengan IHK sebesar 118,17. Inflasi terjadi karena adanya kenaikan harga yang ditunjukkan oleh 10 kelompok pengeluaran mengalami inflasi *year on year* dan satu kelompok pengeluaran mengalami deflasi *year on year*. Secara detailnya, kelompok pengeluaran yang mengalami inflasi *year on year* tertinggi, yaitu kelompok makanan, minuman dan tembakau sebesar 6,36% dan yang terendah, yaitu kelompok transportasi sebesar 0,35%. Sedangkan yang mengalami

deflasi *year on year* hanya pada kelompok informasi, komunikasi dan jasa keuangan sebesar 0,21% (BPS Kota Surabaya, 2023). Kenaikan inflasi dapat menyebabkan persentase pertumbuhan ekonomi menjadi tidak berarti yang kemudian berdampak pada naiknya angka kemiskinan. Dengan adanya kenaikan IHK dapat menyebabkan peningkatan suku bunga, uang beredar, dan daya tarik mata uang yang juga dapat menyebabkan terjadinya fluktuasi.

Prediksi IHK memiliki dampak yang besar dalam menunjang kegiatan sosial ekonomi di Indonesia, khususnya pada wilayah Kabupaten/Kota. Metode dalam memprediksi IHK mempertimbangkan data yang digunakan. Dimana data IHK termasuk dalam data runtun waktu (*time series*) dimana merupakan serangkaian data pengamatan yang berasal dari satu sumber tetap dan terjadi berdasarkan indeks waktu secara beruntuk dengan interval waktu yang tetap. Penelitian dalam bidang peramalan *time series* adalah salah satu metode yang telah digunakan dalam berbagai implementasi dikarenakan oleh kemampuannya untuk memprediksi nilai masa depan. Peramalan IHK dapat dilakukan menggunakan algoritma data mining salah satunya Metode *time series* telah banyak digunakan. Penelitian oleh Ayudhiah, dkk (2020) menggunakan VARIMA dalam prediksi nilai IHK dengan hasil MAPE 10,67%. Penelitian Dimashanti, dkk (2021) dengan SARIMA, namun pada penelitian ini masih belum memiliki skenario evaluasi model. Selanjutnya terdapat penelitian oleh Mukron, dkk (2021) memanfaatkan ARIMA dengan nilai MSE 0,1744. Selain itu, terdapat penelitian yang menggunakan *Exponential Smoothing* dan *Backpropagation*. Data IHK pada umumnya merupakan data non-linear sehingga metode *time series* yang sering digunakan pada kasus non-linear diantaranya *Artificial Neural Network* (ANN), *Threshold Autoregressive* (TAR), *Autoregressive Conditional Heteroscedastic* (ARCH), dan *Support Vector Regression* (SVR) (Alwee, Shamsuddin, & Sallehuddin, 2013). SVR merupakan salah satu metode dalam memprediksi data non-linear dengan keunggulan dapat mengatasi kasus *overfitting* dengan konsep *risk minimization*. SVR telah diterapkan pada berbagai bidang *time series* dan prediksi keuangan, pendekatan teknik analisis kompleks, pemrograman kurva konveks kuadrat, fungsi kemungkinan kerugian, dan sebagainya (Basak, Debasish, Pal, & Patranabis, 2007). SVR memiliki kegunaan terbesar saat dimensi dari ruang *input* dan urutan pendekatan yang menciptakan dimensi dari representasi ruang fitur yang lebih besar (Drucker, Harris, Burges, & Linda Kaufman, 1996).

Metode *Support Vector Regression* (SVR) telah menunjukkan potensinya dalam memprediksi data dengan pola yang kompleks dan nonlinear. SVR, sebagai salah satu metode pembelajaran mesin, mampu menangani masalah regresi dengan mencari hyperplane terbaik yang meminimalkan kesalahan prediksi. Namun, pemilihan parameter yang tepat dalam SVR menjadi tantangan tersendiri yang mempengaruhi kinerja model. Untuk mengatasi masalah pemilihan parameter dalam SVR, *Genetic Algorithm* (GA) dapat digunakan sebagai solusi. GA adalah algoritma optimasi berbasis evolusi yang dapat mengeksplorasi ruang parameter secara efektif. Dengan mengkombinasikan SVR dengan GA, diharapkan dapat diperoleh model prediksi IHK yang lebih akurat dan *robust*. Kota Surabaya, sebagai salah satu kota metropolitan terbesar di Indonesia, memiliki dinamika ekonomi yang kompleks. Perubahan harga konsumen di kota ini dipengaruhi oleh berbagai faktor, termasuk perkembangan industri, perubahan demografi, perubahan laju investasi lokal maupun kebijakan ekonomi lokal. Oleh karena itu, penelitian di Kota Surabaya memberikan tantangan yang menarik untuk penerapan metode GA-SVR dalam memprediksi IHK .

Dalam penelitian ini, data historis IHK Kota Surabaya akan dianalisis dan digunakan untuk membangun model prediksi. Proses pengolahan data meliputi tahap *pre-processing*, pemilihan fitur, dan pembagian data menjadi *set training* dan *testing*. Tahap ini penting untuk memastikan bahwa model yang dibangun memiliki performa yang baik dalam memprediksi IHK. Pemilihan parameter SVR akan dioptimalkan menggunakan GA. Dengan memanfaatkan kekuatan eksploratif GA, diharapkan dapat ditemukan kombinasi parameter yang optimal untuk SVR, sehingga model prediksi memiliki akurasi yang tinggi. Hasil dari penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam bidang ekonomi dan manajemen risiko. Prediksi IHK yang akurat akan membantu pemerintah dan pengambil kebijakan dalam merumuskan kebijakan ekonomi yang lebih efektif dan tepat sasaran. Selain itu, hasil penelitian ini juga dapat digunakan oleh pelaku bisnis untuk merencanakan strategi bisnis yang lebih baik.

Dalam implementasi SVR terdapat parameter-parameter yang menentukan peran dalam meningkatkan akurasi peramalan. Oleh karena itu digunakan algoritma genetika (GA) dalam penentuan parameter-parameter tersebut. Algoritma genetika merupakan percobaan inovatif dengan mempekerjakan proses mutasi GA untuk meningkatkan proses pencarian. Dengan demikian, pada penelitian ini menerapkan metode *Genetic Algorithm-Support Vector Regression* dalam *forecasting* IHK di Kota Surabaya. Berdasarkan pada hasil *forecast* bulan selanjutnya dengan harapan dapat memberikan tambahan informasi bagi pemerintah Kota Surabaya dalam menghadapi kenaikan inflasi sehingga mampu memberikan kebijakan ekonomi yang tepat. Dengan demikian, penelitian ini tidak hanya bertujuan untuk memprediksi IHK Kota Surabaya dengan akurat, tetapi juga untuk mengembangkan metode prediksi yang inovatif dan aplikatif. Semoga hasil penelitian ini dapat memberikan manfaat yang luas bagi berbagai pihak dan menjadi referensi penting dalam bidang ekonomi dan keuangan.

1.2 Rumusan Masalah

Kota Surabaya merupakan salah satu kota besar sebagai ibukota Provinsi Jawa Timur dengan penduduk yang cukup padat dan pusat industri yang cukup besar. Pada Desember 2023 Kota Surabaya mengalami kenaikan harga pada sebagian komoditas bahan pokok. Hal ini mendorong terjadinya kenaikan IHK. Pemerintah Kota Surabaya perlu bergerak cepat dalam menekan kenaikan harga sejumlah komoditas bahan pokok seperti beras. Oleh karena itu, dibutuhkan antisipasi berupa peramalan IHK lebih awal guna mengantisipasi terjadinya kenaikan inflasi. Beberapa metode peramalan yang dapat digunakan pada IHK adalah *Support Vector Regression* (SVR). Metode SVR dapat mengatasi kasus nonlinear yang merupakan ciri karakteristik data IHK dengan menambahkan fungsi kernel. Dalam mengoptimalkan parameter SVR digunakan *Genetic Algorithm* (GA) dengan harapan dapat meningkatkan akurasi peramalan. Berdasarkan permasalahan yang telah diuraikan, maka dapat dirumuskan suatu permasalahan, yaitu peramalan IHK di Kota Surabaya menggunakan metode GA-SVR, proses pemilihan parameter SVR yang optimal dengan menggunakan GA dalam meningkatkan akurasi prediksi IHK sehingga dapat memberikan hasil yang akurat dan andal, proses evaluasi kinerja model prediksi GA-SVR dibandingkan dengan metode prediksi lainnya pada data IHK Kota Surabaya. Pada model SVR akan digunakan beberapa fungsi *kernel* dengan model terbaik dilakukan optimasi menggunakan GA dalam meramalkan IHK Kota Surabaya enam bulan kedepan.

1.3 Batasan Masalah

Batasan masalah penelitian yang digunakan pada penelitian ini ada pada data yang digunakan hanya bagian wilayah Provinsi Jawa Timur, yaitu pada Kota Surabaya dan data yang digunakan merupakan data IHK Kota Surabaya tahun 2019-2023. Pada pengaplikasian metode dilakukan pemotongan data pada tahun 2020-2023 setelah terjadinya *Covid-19*. Metode yang digunakan adalah GA-SVR dimana metode SVR dipilih karena dapat mengatasi kasus nonlinear serta tidak memerlukan asumsi. Pada optimasi parameter SVR menggunakan *Genetic Algorithm* (GA) dengan harapan untuk meningkatkan akurasi peramalan agar memperoleh nilai RMSE dan sMAPE yang paling minimum.

1.4 Tujuan

Berdasarkan uraian rumusan masalah dapat diambil tujuan penelitian ini sebagai berikut.

1. Mendeskripsikan karakteristik IHK di Kota Surabaya.
2. Mendapatkan model terbaik antara SVR dan GA-SVR untuk meramalkan IHK di Kota Surabaya.
3. Mendapatkan hasil ramalan IHK di Kota Surabaya enam bulan mendatang berdasarkan model terbaik.

1.5 Manfaat

Berdasarkan penelitian ini, manfaat yang dapat diambil adalah sebagai berikut.

1. Memberikan tambahan informasi bagi:
 - a. BPS Kota Surabaya dan Provinsi Jawa Timur dalam memprediksi IHK serta mengantisipasi terjadinya inflasi di masa mendatang sehingga dapat menetapkan kebijakan ekonomi yang sesuai dan akurat.
 - b. Dinas Koperasi UMKM dan Perdagangan Kota Surabaya dalam menyusun strategi perencanaan ekonomi yang berkaitan dengan IHK sehingga dapat meminimalkan dampak kerugian apabila terjadi ketidakstabilan IHK di Kota Surabaya.
2. Memberikan informasi tambahan mengenai laju IHK di Kota Surabaya sebagai pembuat kebijakan bagi masyarakat dalam memenuhi kebutuhan.
3. Memberikan informasi bagi pihak-pihak terkait dan dapat dijadikan referensi tambahan untuk penelitian selanjutnya.

(Halaman ini sengaja dikosongkan)

BAB 2 TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Berikut merupakan beberapa penelitian terdahulu mengenai IHK.

Tabel 2.1 Penelitian Terdahulu

Judul	Periode	Rentang Waktu	Metode	Kesimpulan
(Rosdianawati & Surjanto, 2023)	Bulanan	Januari 2012 – Desember 2021	<i>Exponential Smoothing</i>	Model terbaik meramalkan IHK Kota Kediri adalah model <i>Holt-Winters exponential smoothing</i>
(Hidayatullah, Cholissodin, & Perdana, 2018)	Bulanan	Januari 2011 – Mei 2017	<i>Support Vector Regression-Chaotic Genetic Algorithm-Simulated Annealing (CGASA)</i>	Hasil peramalan menggunakan SVR-CGASA didapatkan nilai <i>error</i> terkecil sebesar 0,304462 dan <i>fitness</i> sebesar 0,7666.
(Budiastuti, 2017)	Bulanan	Januari 2012 – Desember 2016	<i>Support Vector Regression</i>	Hasil dari prediksi IHK adalah nilai MSE sebesar 0,3454 dengan koefisien determinan sebesar 0,9991.

2.2 Uji Nonlinearitas Terasvirta

Uji nonlinearitas dilakukan untuk mendeteksi adanya hubungan nonlinear pada data. Salah satu uji nonlinearitas, yaitu uji Terasvirta dimana merupakan pengujian linearitas dengan tipe *Lagrange Multiplier* yang dikembangkan dari model *neural network* dengan ekspansi Taylor. Dalam uji Terasvirta ini, m prediktor tambahan yang digunakan adalah suku kuadrat dan kubik yang merupakan hasil dari pendekatan ekspansi Taylor. Uji linearitas Terasvirta yang diperkenalkan oleh Terasvirta, dkk (1993) dapat dilakukan melalui dua statistik uji, yaitu uji χ^2 atau uji F . Uji Terasvirta dilakukan dengan cara regresi polinomial dan menguji suku-suku polinomialnya. Apabila dimisalkan terdapat dua variabel, yaitu Y_{t-1} dan Y_{t-2} maka model yang diuji sebagai berikut.

$$\begin{aligned}
 Y_t = & \beta_0 + \beta_1 Y_{(t-1)} + \beta_2 Y_{(t-2)} + \delta_1 Y_{(t-1)} + \delta_2 Y_{(t-2)}^2 + \delta_3 Y_{(t-1)} Y_{(t-2)} + \delta_4 Y_{(t-1)}^3 + \delta_5 Y_{(t-2)}^3 \\
 & + \delta_6 Y_{(t-1)}^2 Y_{(t-2)} + \delta_7 Y_{(t-1)}^2 Y_{(t-2)}^2
 \end{aligned}
 \tag{2.1}$$

Berikut merupakan hipotesis uji Terasvirta dan prosedur dalam mendapatkan uji χ^2 .

H_0 : Parameter pada regresi polinomial (δ) bersifat linier ($\delta_1 = \delta_2 = \delta_3 = \delta_4 = \delta_5 = \delta_6$)

H_0 : Salah satu parameter tidak linier ($\delta_i \neq \delta_j$)

Statistik uji yang digunakan adalah nilai F dari hasil pemodelan regresi polinomial seperti pada persamaan (2.3) dengan prosedur uji Terasvirta lebih lanjut sebagai berikut.

1. Meregresikan Y_t dengan $1, Y_{t-1}, Y_{t-2}, \dots, Y_{t-k}$ dan menghitung residual \hat{a}_t serta menghitung jumlah kuadrat residual.

$$SSR_0 = \sum \hat{a}_t^2 \quad (2.2)$$

2. Meregresikan \hat{a}_t dengan $1, Y_{t-1}, Y_{t-2}, \dots, Y_{t-k}$ dan m prediktor tambahan dan menghitung residual $\hat{v}_t = \hat{a}_t - \hat{a}_t$ dan jumlah kuadrat residual $SSR_1 = \sum \hat{v}_t^2$ dimana m dan prediktor-prediktor yang terlibat bervariasi untuk suatu uji dengan uji yang lain.
3. Menghitung nilai uji.

$$F = \frac{MSR}{MSE} \quad (2.3)$$

$$F = \frac{(SSR_0 - SSR_1) / m}{(1 - SSR_1) / (n - p - 1 - m)} \quad (2.4)$$

Di bawah hipotesis linearitas dalam *mean*, nilai uji F ini mendekati distribusi F dengan derajat bebas m dan $(n - p - 1 - m)$. Tolak H_0 apabila nilai statistik uji lebih besar dari nilai F -tabel. Penggunaan dari uji F menggantikan uji χ^2 ini berdasarkan rekomendasi dari teori asimtotis dalam sampel kecil, yaitu karena uji ini memiliki sifat-sifat kuasa dan ukuran yang baik (Terasvirta, 1993).

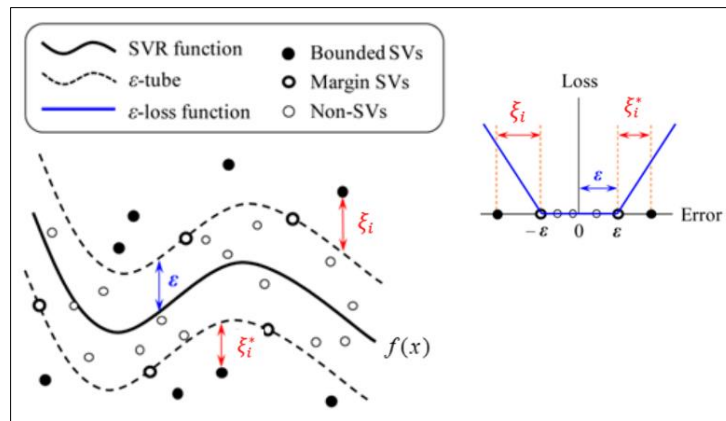
2.3 Support Vector Regression

SVR merupakan bagian dari *Support Vector Machine* (SVM) menggunakan sistem ruang hipotesis berupa fungsi-fungsi linear dalam *feature space* berdimensi tinggi dengan algoritma yang didasarkan pada teori optimasi *biased learning*. Konsep SVM menggunakan ε -*insentive loss function*. SVM merupakan metode yang menggabungkan metode pada komputasi seperti *hyperplane*, *kernel*, dan konsep lainnya yang dapat meminimalkan kesalahan klasifikasinya serta metode yang tergolong dalam *supervised learning* dimana hanya digunakan pada data kategorik dengan kelas $+1$ dan -1 (Prasetyo, 2014). SVM pada dasarnya suatu metode klasifikasi dimana variabel respon menjadi variabel numerik bilangan *real* dan kontinu serta dapat mengatasi masalah *overfitting* sehingga memiliki performa kerja yang baik (Suprayogi & Pardede, 2022). *Overfitting* merupakan kondisi dimana suatu model tidak menggambarkan hubungan utama antara variabel *input* dan *output* melainkan menggambarkan *random error* atau *noise* sehingga pada kondisi ini akan mengakibatkan hasil prediksi yang buruk.

Menurut Cortez dan Vapnik (1995), SVR menggunakan model linear untuk menerapkan konsep kelas nonlinear melalui pemetaan vektor masukan x ke dalam ruang fitur *high-dimensional*. SVR bertujuan untuk menemukan suatu fungsi $f(x)$ sebagai garis pemisah (*hyperplane*) berupa fungsi regresi pada semua *input* data yang memiliki deviasi paling besar ε dengan target aktual y_i untuk semua data *training* dan membuat *error* sekecil mungkin (Smola & Scholkopf, 2004). SVR tidak hanya mampu mengatasi masalah data linear tetapi juga dapat digunakan untuk mengatasi data yang bersifat nonlinear. Konsep SVR mempertimbangkan satu *set* data *vector input* model dan nilai aktual yang merepresentasikan hubungan *output* dengan total jumlah pola data (Guo, 2018). Dalam mengestimasi suatu fungsi dengan meminimalkan batas dari *generalization error* dapat mengatasi kasus *overfitting* dari konsep SVR yang disebut *structural risk minimization*. Model SVR memberikan fleksibilitas untuk mendefinisikan

berapa banyak *error* yang dapat diterima dalam model dan dapat mencari garis atau *hyperplane* yang cocok untuk *fitting* data (Sharp, 2020).

Dalam SVR dikenal ε –*insentive loss* function untuk mengukur resiko empiris atau selisih target dengan hasil estimasi. Oleh karena itu, parameter ε perlu diatur sebagai batas toleransi kesalahan hasil estimasi. Berikut merupakan ilustrasi *error* pada SVR.



Gambar 2.1 Ilustrasi *Error* SVR

Gambar 2.1 menunjukkan perhitungan nilai *error* pada SVR. Nilai *error* dianggap sama dengan nol apabila masih berada di antara nol dan garis $f(x) + \varepsilon$ sedangkan apabila nilai *error* berada di luar garis batas toleransi maka akan dihitung sebagai variabel *slack* ξ atau ξ^* . Variabel *slack* ini menunjukkan jarak dari nilai aktual ke nilai batas yang sesuai dari ε –*tube*. Semakin sempurna suatu fungsi regresi SVR, maka batas deviasinya akan bernilai mendekati nol. Parameter utama SVR yang bertujuan untuk menyeimbangkan antara *training error* dan *generalization*. Berikut penjelasan masing-masing parameter (Zhao, Tao, & Zio, 2015).

- Faktor C atau *Cost*, mendeskripsikan derajat dari *punishment error*. Apabila nilai C semakin tinggi, maka *fitting* semakin tinggi sehingga *training error* dan *test error* semakin rendah. Namun, apabila nilai C terlalu besar, maka akan terjadi fenomena *overfitting* dan akan meningkatkan *error* prediksi. Domain pada parameter C adalah $0 < C < \infty$ artinya apabila mendekati 0 maka semakin kecil nilai *error* yang ditoleransi. Apabila semakin besar nilai parameter C yang digunakan maka semakin tinggi garis batas toleransi.
- Parameter ε , apabila semakin besar maka akurasi *fitting* akan semakin kecil, jumlah *support vector* akan semakin rendah, dan kompleksitas akurasi dari model akan berkurang. Nilai ε semakin kecil menyebabkan jarak *hyperplane* semakin mendekati fungsi regresi artinya model semakin baik karena semakin mengikuti data aktual.
- Parameter γ , apabila semakin kecil maka *fitting* kernel pada *radian basis function* akan semakin buruk, apabila semakin besar maka performa *fitting* baik, namun dapat menyebabkan kemampuan generalisasi yang buruk.

Adapun penjelasan SVR menurut Cao dan Wu (2016). Diketahui data yang diramalkan adalah (x_i, y_i) dimana $1 = 1, 2, \dots, n$; $x_i \in R^n$; $y_i \in R$ dimana x_i adalah vektor masukan, y_i adalah nilai luaran untuk memprediksi n atau disebut juga jumlah *training set*. Berikut fungsi sebagai regresi yang merepresentasikan rumus SVR.

$$f(x) = w^T \varphi(x) + b, \varphi: R^n \rightarrow F \quad (2.5)$$

Dimana w adalah elemen dalam ruang *high-dimensional* atau vektor pembobot yang akan diatur, $\varphi(x)$ adalah fungsi pemetaan nonlinear, b adalah koefisien ambang batas atau *threshold*, $f(x)$ dinotasikan sebagai nilai luaran dari peramalan atau prediksi (Smola & Scholkopf, 2004).

Tujuan dari SVR ialah mencari fungsi $f(x)$ yang memiliki deviasi ε terbanyak dari target y_i untuk semua data *training* dan membuat kurva persamaan (2.5) menjadi selandai mungkin. Dibandingkan dengan OLS, tujuan SVR adalah meminimalkan *norm Euclidean* dari w bukan *error* kuadrat seperti OLS. SVR mencari *hyperplane* optimum yang memiliki batasan maksimum antara sampel positif dan negatif terdekat. Dalam mengestimasi koefisien w dan b dilakukan dengan meminimumkan fungsi risiko (*risk function*) dari persamaan berikut.

$$R(f(x)) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n L_{\varepsilon}(y_i, f(x_i)) \quad (2.6)$$

Dimana $\|w\|$ adalah regularisasi, yaitu fungsi yang diminimalkan agar fungsinya sehalus mungkin (*flat*). Konstanta $C > 0$ adalah kompromi (*trade off*) antara ketipisan fungsi dan batas atas deviasi. (Smola & Scholkopf, 2004). Fungsi kerugian $\varepsilon - insensitive loss function$ didefinisikan sebagai berikut.

$$L_{\varepsilon}(y_i, f(x_i)) = \begin{cases} 0 & ; |y_i - f(x_i)| \leq \varepsilon \\ |y_i - f(x_i)| - \varepsilon & ; |y_i - f(x_i)| > \varepsilon \end{cases} \quad (2.7)$$

Fungsi regresi $f(x)$ adalah fungsi yang dapat mengaproksimasi semua titik (x_i, y_i) dengan ketelitian ε . Semua titik pada interval $f(x) \pm \varepsilon$ disebut sebagai titik yang dapat diterima (*feasible*). Titik di luar interval tersebut disebut *infeasible*. $\varepsilon - insensitive loss function$ digunakan untuk mengukur risiko empiris, selisih target dengan hasil estimasi. Parameter C mengontrol pertukaran antara keempirisan dan kesalahan generalisasi. Oleh karena itu parameter ε harus diatur untuk meminimalisasi risiko empiris menggunakan variabel *slack* ξ atau ξ^* yang menggambarkan simpangan dari data *training* di luar zona $\varepsilon - insensitive$.

Disamping meminimalisasi kesalahan empiris dengan fungsi $\varepsilon - insensitive$, juga harus meminimalisasi norma *Euclidean* dari beban yang linier $\|w\|$ yang mana berhubungan dengan kemampuan generalisasi model SVR yang pada data *training*. Tujuannya untuk memperlebar (*maximize*) margin sehingga kelandaian kurva serta kompleksitas model dapat dipastikan. Permasalahan regresi dapat dinyatakan seperti masalah optimasi kuadratik berikut.

$$L(w, \xi) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*), C > 0 \quad (2.8)$$

dengan batasan,

$$\text{subject to } \begin{cases} y_i - w^T \varphi(x_i) - b \leq \varepsilon + \xi_i \\ w^T \varphi(x_i) + b - y_i \leq \varepsilon + \xi_i^* \\ \varphi_i, \xi_i^* \geq 0 \end{cases} \quad (2.9)$$

Dimana konstanta C menyatakan koefisien penalti yang mendeterminasikan *trade off* antara keempirisan dengan kesalahan generalisasi yang mana nilai tersebut perlu diatur (Zhao, Tao, & Zio, 2015). Dalam SVR, ε ekuivalen dengan akurasi dari aproksimasi terhadap data *training*. Nilai ε yang kecil akan dikaitkan dengan nilai yang tinggi pada variabel *slack* ξ_i^* dan akurasi aproksimasi yang tinggi. Sebaliknya, nilai yang tinggi untuk ε berkaitan dengan nilai ξ_i^* yang kecil dan aproksimasi yang rendah.

Pada persamaan (2.8) nilai yang tinggi untuk variabel *slack* akan membuat kesalahan empiris mempunyai pengaruh yang besar terhadap faktor regulasi. Dalam SVR, *support vector* adalah data *training* yang terletak diluar batas $f(x)$ dari fungsi keputusan sehingga jumlah *support vector* menurun dengan naiknya ε . Hong (2008) menjelaskan bahwa untuk mengoptimalkan batasan dapat diselesaikan menggunakan *primal lagrangian* seperti pada persamaan (2.10).

$$L(w, b, \xi, \xi^*, a_i, a_i^*, \beta_i, \beta_i^*) = \frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^n (\xi_i + \xi_i^*) \right) - \sum_{i=1}^n \beta_i [w\phi(x_i) + b - y_i + \varepsilon + \xi_i^*] - \sum_{i=1}^n \beta_i^* [y_i - w\phi(x_i) - b + \varepsilon + \xi_i^*] - \sum_{i=1}^n [a_i \xi_i - a_i^* \xi_i^*] \quad (2.10)$$

Dimana β_i merupakan koefisien *Lagrange* dengan $0 \leq \beta_i \leq C$ dan $0 \leq \beta_i^* \leq C$. Persamaan (2.5) kemudian diminimalkan pada variabel *primal* w, b, ξ, ξ^* dan memaksimalkan dalam bentuk *lagrangian multiplier non-negative* $a_i, a_i^*, \beta_i, \beta_i^*$ seperti pada persamaan berikut.

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^n (\beta_i - \beta_i^*) \phi(x_i) = 0 \quad (2.11)$$

$$\frac{\partial L}{\partial \beta} = \sum_{i=1}^n (\beta_i - \beta_i^*) = 0 \quad (2.12)$$

$$\frac{\partial L}{\partial \xi} = c - \beta_i - a_i = 0 \quad (2.13)$$

$$\frac{\partial L}{\partial \xi^*} = c - \beta_i^* - a_i^* = 0 \quad (2.14)$$

Kondisi *Karush-Kuhn-Tucker* diaplikasikan untuk model regresi. *Dual lagrangian* pada persamaan (2.10) didapatkan dengan mensubstitusikan persamaan (2.11), (2.12), (2.13), dan (2.14) ke persamaan (2.15).

$$\partial(\beta_i, \beta_i^*) = \sum_{i=1}^n y_i (\beta_i - \beta_i^*) - \varepsilon \sum_{i=1}^n (\beta_i - \beta_i^*) - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\beta_i - \beta_i^*) (\beta_j - \beta_j^*) k(x_i, x_j) \quad (2.15)$$

Batasan pada persamaan (2.10) sebagai berikut.

$$\begin{aligned} \sum_i (\beta_i - \beta_i^*) &= 0 \\ 0 \leq \beta_i &\leq C, i = 1, 2, \dots, n \\ 0 \leq \beta_i^* &\leq C, i = 1, 2, \dots, n \end{aligned} \quad (2.16)$$

Persamaan *dual lagrangian* didapatkan ketika C didefinisikan dengan fungsi *kernel dot-product* $k(x_i, x_j) = \phi(x_i)\phi(x_j)$ (Hardle, Prasetyo, & Hefner, 2014). Dengan menggunakan *lagrange multiplier* dan kondisi optimalitas, fungsi regresi secara eksplisit ditunjukkan sebagai berikut.

$$f(x) = \sum_i (\beta_i - \beta_i^*) k(x_i, x_j) + b \quad (2.17)$$

Performansi model prediksi SVR tergantung pada penggunaan *hyperparameter* C , epsilon (ε), *degree*, parameter *kernel* (γ) *gamma*, dan fungsi *kernel* yang digunakan. Parameter C memberikan nilai penalti pada data apabila terjadi *error* pada prediksi, parameter ε

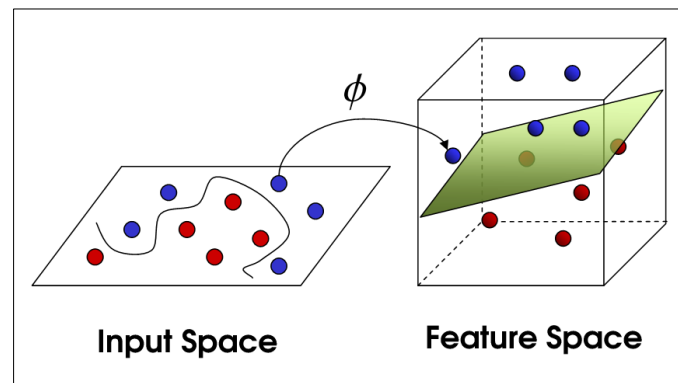
menandakan *epsilon tube* yang berarti tidak ada penalti di dalam *loss function* terhadap poin prediksi selama masih berada di dalam jarak *epsilon* dari nilai aktual. *Degree* hanya digunakan oleh fungsi *kernel polynomial* dan tidak digunakan pada *kernel* lainnya, sedangkan *gamma* koefisien kernel RBF mengontrol seberapa jauh pengaruh satu titik data *training* dengan pengaruh yang menurun seiring meningkatnya jarak dari titik tersebut.

2.4 Fungsi Kernel

Fungsi *kernel* merupakan suatu fungsi yang dapat mengatasi masalah nonlinear yang pada umumnya sering terjadi. Fungsi ini diaplikasikan pada setiap data untuk memetakan data asli nonlinear ke dalam ruang dimensi yang lebih tinggi (*higher dimensional space*) dimana fungsi *k* untuk semua vektor *input* x_i, x_j akan memenuhi kondisi berikut.

$$k(x_i, x_j) = \phi(x_i)^T \phi(x_j) \quad (2.18)$$

Dimana $\phi(\cdot)$ merupakan pemetaan dari *input space* ke *feature space*. Dengan kata lain fungsi *kernel* merupakan fungsi *inner product* pada *feature space*.



Gambar 2.2 Kernel Mapping

Gambar 2.2 merupakan ilustrasi *mapping kernel* dimana data berwarna biru dengan kelas satu dan data berwarna merah dengan kelas nol. Hal ini berarti menunjukkan bagaimana data warna merah dan biru dari *input space* tidak bisa dipisahkan sehingga perlu ditransformasikan ke dimensi yang lebih tinggi, *feature space* menggunakan fungsi *kernel*. Fungsi *kernel* dapat menghitung *inner product* pada *feature space* secara langsung dari ruang *input* tanpa secara eksplisit menghitung koordinat proyeksi masing-masing vektor *input* pada *feature space*. Beberapa fungsi matriks *kernel* untuk menyelesaikan permasalahan nonlinear sebagai berikut.

a. *Kernel Linear*

$$k(x_i, x_j) = x_i \cdot x_j \quad (2.19)$$

b. *Kernel Polynomial*

$$k(x_i, x_j) = (x_i \cdot x_j + 1)^d \quad (2.20)$$

c. *Kernel Radial Basis Function (RBF)*

$$k(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2) \quad (2.21)$$

Penggunaan fungsi *kernel* yang sesuai, hasil prediksi yang didapatkan dari metode SVR dapat menjadi lebih baik (Santosa, 2007).

2.5 Grid Search

Metode *grid search* secara umum merupakan kombinasi parameter yang diujikan pada suatu model SVR untuk mencari nilai *error* dalam klasifikasi. Tujuan dari metode ini adalah mengidentifikasi parameter optimal pada data *training* sehingga model tersebut dapat memprediksi data *testing* secara akurat. Salah satu pendekatan metode *grid search*, yaitu menggunakan *cross validation*. Prosedur *cross validation* adalah membagi secara acak data *training* menjadi n subset yang berukuran sama (Hsu, Chang, & Lin, 2016). Pada pengaplikasian data *time series*, jenis *train control* yang umum digunakan, yaitu *time slice*. Prosedur teknik *time slice* pembagian data menjadi beberapa "*slice*" atau potongan waktu, dimana setiap potongan digunakan secara bergantian sebagai *set training* dan *testing* dalam proses *cross-validation*. *Time slice* memastikan bahwa urutan temporal data tetap terjaga, menghindari kebocoran informasi dari masa depan ke masa lalu yang bisa terjadi dalam pembagian data secara acak (S. Yadav, 2016).

Pada dasarnya, *time slice cross-validation* membagi data historis menjadi beberapa subset secara berurutan, dimana model dilatih pada *subset* awal dan divalidasi pada *subset* berikutnya, terus bergerak maju dalam waktu. Proses ini diulang untuk berbagai kombinasi parameter yang ditentukan dalam *grid search*, sehingga memungkinkan pemilihan parameter terbaik berdasarkan kinerja model pada data yang tidak terlihat selama pelatihan. Dengan demikian, *grid search time slice* membantu dalam menemukan parameter yang menghasilkan model dengan generalisasi yang baik dan performa prediksi yang optimal pada data deret waktu, sambil menjaga integritas temporal dari data tersebut.

2.6 Genetic Algorithm

Genetic Algorithm (GA) merupakan salah satu metode optimasi yang ditemukan oleh John Holan pada tahun 1975 (Sivanandam & Deepa, 2008). Konsep GA diselesaikan dengan proses evolusi dengan prinsip seleksi alam dan merepresentasikan sebuah teknik identifikasi pendekatan solusi masalah optimasi yang paling baik. Pada proses evolusi akan didapatkan individu-individu yang harus mampu menyesuaikan diri dengan lingkungannya untuk bertahan hidup. Individu-individu ini telah secara berulang kali mengalami perubahan gen untuk dapat menyesuaikan diri dengan lingkungan hidupnya. Perubahan gen terjadi melalui proses perkembangbiakan dimana pada GA proses ini merupakan dasar pemikiran dalam mendapatkan anak yang lebih sesuai. GA adalah sebuah algoritma optimasi *metaheuristik* yang berdasarkan populasi solusi dan menggunakan mekanisme spesifik yang terinspirasi dari evolusi secara biologi, seperti individu (kromosom), reproduksi, mutasi, rekombinasi, seleksi, ketahanan *fitness* (Gorunescu, 2011).

Parameter kontrol yang digunakan dalam optimasi GA, yaitu probabilitas *crossover* (P_c) dan probabilitas mutasi (P_m) (Windarto & Wijaya, 2012). Menurut Schaffer dkk (1989) menyatakan bahwa probabilitas *crossover* yang optimal ada pada *range* 0,75 – 0,95 sedangkan probabilitas mutasi yang baik berada pada *range* 0,05 – 0,01. Salah satu keunggulan GA, yaitu mampu mengatasi berbagai jenis fungsi objektif dari berbagai kromosom (Gen & Cheng, 1997). Keunggulan lainnya dari GA adalah kemampuan dalam menangani permasalahan kompleks dan paralel. GA dapat digunakan pada jumlah variabel yang besar, menangani berbagai macam optimasi tergantung pada fungsi objektifnya (*fitness*), seimbang atau tidak seimbang, linear atau linear, berkesinambungan atau tidak berkesinambungan, atau

dengan *random noise* (Zhang, 2011). Prosedur yang harus terlebih dahulu ditetapkan (inisialisasi) sebelum melakukan analisis menggunakan GA sebagai berikut (Setiawan, 2003).

1. Menentukan Representasi Genetik. Bentuk kromosom harus ditentukan paling awal karena setiap nilai yang terkandung dalam gen (*allele*) yang bergabung harus disediakan tempat penampungnya. Panjang kromosom dari program yang dibuat harus sesuai dengan banyaknya parameter yang ingin dioptimasi.
2. Menentukan Rumusan *Fitness Function*. Fungsi *fitness* digunakan untuk mengetahui seberapa baik solusi yang diperoleh satu individu. Melalui nilai *fitness function* dapat dicari probabilitas kumulatif dalam pemilihan individu sebagai *parent*. Salah satu permasalahan yang sering terjadi adalah terdapat beberapa kromosom yang mendominasi populasi sehingga mengakibatkan kondisi konvergen terlalu dini sehingga nilai *fitness* tinggi tetapi tidak optimal akan terjebak pada kondisi lokal optimal.
3. Menetapkan Nilai Parameter. Beberapa parameter yang perlu ditetapkan nilainya sebagai berikut.
 - a. Ukuran Populasi. Banyaknya individu yang terdapat dalam populasi. Populasi merupakan sekumpulan individu yang akan diproses dalam satu siklus iterasi. Semakin banyak dan beragam individu yang terdapat dalam populasi akan memberikan peluang yang lebih besar untuk menemukan individu yang mendekati sempurna.
 - b. Jumlah Generasi. Generasi dalam GA merupakan satuan siklus dalam iterasi. Semakin besar jumlah generasi, maka individu yang dihasilkan baik dan sempurna. Namun, tidak berarti semakin besar jumlah generasi maka individu yang dihasilkan selalu lebih baik ada saat dimana nilai *fitness* semua individu akan menjadi sama. Apabila hal ini terjadi maka generasi selanjutnya cenderung memiliki nilai *fitness* yang sama,
 - c. Probabilitas *Crossover* (P_c). Semakin besar probabilitas pindah silang maka semakin banyak kemungkinan terjadi perkawinan silang antara dua individu menghasilkan individu baru.
 - d. Probabilitas Mutasi (P_m). Semakin besar probabilitas mutasi maka semakin besar kemungkinan operasi mutasi suatu kromosom akan menghasilkan individu baru.
 - e. Probabilitas Reproduksi. Probabilitas menentukan individu yang pantas bertahan dalam generasi selanjutnya karena *offspring* yang dihasilkan dengan *parent*.

GA memiliki tujuh tahapan utama yang perlu dilakukan setelah inisialisasi awal (Ismail & Irhamah, 2008). Berikut tujuh tahapan yang dimaksud.

1. Definisi. Tahapan dimana didefinisikan variabel yang dianggap sebagai kromosom. Optimasi yang dilakukan pada penelitian ini ada pada model SVR. Kromosom akan terdiri dari tiga sel yang terdiri dari parameter fungsi *kernel*, yaitu C , γ dan ϵ .
2. Inisialisasi. Tahapan dimana dilakukan inisialisasi jumlah populasi sebesar P dari N kromosom yang mengandung solusi. Mutasi yang dilakukan secara *random* pada kromosom yang terpilih. Skema inisialisasi yang paling umum digunakan dalam pengkodean kromosom antara lain.
 - i. *Binary Encoding*. Setiap gen hanya bernilai 0 dan 1.
 - ii. *Real Number Encoding*. Nilai gen berada dalam interval $[0, R]$ dimana R merupakan bilangan *real* positif dan umumnya $R = 1$.
 - iii. *Discrete Encoding*. Nilai gen berada dalam interval bilangan bulat $[0, 9]$.

- iv. *Value Encoding*. Nilai gen yang berasal dari nilai sembarang yang sesuai dengan permasalahan baik bilangan bulat maupun bilangan riil.
- 3. *Fitness Function*. Fungsi yang digunakan untuk mengevaluasi daya tahan untuk hidup dari sebuah kromosom. Semakin tinggi nilai $f(C_i)$ dari setiap kromosom (C_i) memiliki arti bahwa kromosom tersebut dapat bertahan hidup dan tidak akan punah. *Fitness function* yang akan digunakan dalam penelitian ini adalah sMAPE.
- 4. Seleksi. Tahapan yang dilakukan untuk mendapatkan calon orang tua yang akan dipindah silangkan pada tahapan berikutnya. Seleksi orang tua ini bertujuan untuk memberikan kesempatan reproduksi bagi anggota populasi yang memiliki *fitness* rendah. Nilai *fitness* setiap observasi akan dibandingkan dengan bilangan *random* yang telah dibangkitkan. Metode seleksi yang digunakan merupakan *Roulette Wheel*. Pada metode ini, masing-masing kromosom akan menempati potongan lingkaran pada roda *roulette* secara proporsional sesuai dengan nilai *fitness*. Tahapan seleksi dengan *Roulette Wheel* adalah sebagai berikut.
 - i. Menghitung nilai *fitness* masing-masing kromosom dengan ukuran populasi sebesar n .

a. Kromosom untuk seleksi variabel					
Kromosom ke-	X_1	X_2	...	X_n	Nilai <i>Fitness</i>
1	1	0	...	1	...
...
n
b. Kromosom untuk optimasi parameter					
Kromosom ke-	X_1	X_2	...	X_n	Nilai <i>Fitness</i>
1	1	0	...	1	...
...
n

- ii. Menghitung total nilai *fitness* dalam populasi.

$$F = \sum_{h=1}^{n_{pop}} f(h) \quad (2.22)$$

Dimana F merupakan total nilai *fitness* semua kromosom dalam populasi, N merupakan jumlah kromosom dalam satu populasi, $f(h)$ merupakan nilai *fitness* kromosom ke- h .

- iii. Menghitung proporsi masing-masing kromosom.

$$S_h = \sum_{q=1}^h P_q \quad (2.23)$$

Dimana P_q merupakan nilai proporsi *fitness* kromosom ke- q , S_h merupakan nilai *fitness* kumulatif kromosom ke- h .

- iv. Menghitung nilai kumulatif proporsi untuk masing-masing kromosom.
- v. Membangkitkan sebuah r angka dengan range (0,1) dimana r merupakan sebuah bilangan *random*.
- vi. Apabila $r \leq S_1$ maka pilih kromosom v_1 lainnya pilih v_h sehingga $S_{h-1} < r \leq S_h$ dimana v_i merupakan kromosom ke- i .
- vii. Mengulangi tahapan v hingga semua kromosom yang berjumlah N terpilih semuanya.

5. Pindah Silang (*Crossover*). Tahapan memasang dua kromosom yang telah terpilih pada tahap sebelumnya dan menukarkan satu sel dari tiga sel antar kromosom tersebut sehingga menghasilkan dua kromosom baru. Kromosom baru yang telah terbentuk akan dihitung nilai $f(C_i)$. Macam-macam proses pindah silang diantaranya, yaitu pindah silang satu titik, dua titik, seragam. Salah satu contoh proses pindah silang satu titik potong sebagai berikut.

Orang tua 1 : [0 1 0 1 1 1 0 0]

Orang tua 2 : [1 0 1 0 0 0 1 1]

Apabila dilakukan pindah silang pada titik ke-4 maka anak yang akan dihasilkan adalah sebagai berikut.

Anak 1 : [0 1 0 1 0 0 1 1]

Anak 2 : [1 0 1 0 1 1 0 0]

Pindah silang dilakukan dengan suatu nilai probabilitas tertentu. Nilai probabilitas pindah silang merupakan seberapa sering proses pindah silang akan terjadi antara dua kromosom orang tua. Berdasarkan hasil penelitian *Genetic Algorithm* yang sudah pernah dilakukan sebaiknya nilai probabilitas pindah silang tinggi, yaitu antara 0,8 – 0,9 sehingga memberikan hasil yang baik.

6. Mutasi. Tahapan setiap kromosom diberi sebuah bilangan *random* yang berdistribusi $U(0,1)$. Apabila nilai bilangan *random* yang dimiliki kromosom tersebut bernilai lebih kecil dari nilai probabilitas mutasi, maka kromosom tersebut merupakan kromosom yang mengalami mutasi. Mutasi dilakukan dengan mengganti nilai salah satu dari sel dengan sebuah nilai lain. Nilai probabilitas mutasi menyatakan seberapa sering gen dalam kromosom akan mengalami mutasi. Proses mutasi bersifat acak sehingga tidak menjamin akan diperoleh kromosom dengan *fitness* yang lebih baik setelah terjadinya mutasi tersebut. Solusi optimum dengan konvergensi dini dapat terjadi apabila proses pencarian solusi terperangkap dalam salah satu ruang pencarian kromosom dengan *fitness* yang rendah yang terus bertahan. Hal ini mengakibatkan tidak adanya eksplorasi bagian lain. Oleh karena itu, diperlukan operator mutasi dengan probabilitas mutasi bernilai kecil antara 0,001 hingga 0,01 dengan tujuan menjaga perbedaan kromosom dalam populasi dan memastikan agar solusi terbaik tidak menyimpang.
7. *Replace*. Tahapan dimana semua kromosom, hasil pindah silang dan mutasi, yang dimiliki akan dihitung nilai dari *fitness function*. Kemudian akan diurutkan kromosom dari nilai *fitness* terbesar hingga terkecil dimana akan diambil N kromosom dengan nilai *fitness* tertinggi dan dianggap sebagai generasi baru. Pembentukan populasi baru dengan *crossover* dan mutasi memungkinkan adanya kromosom yang paling baik hilang. Oleh karena itu, untuk menjaga agar kromosom tidak hilang selama proses evolusi perlu dibuat satu atau beberapa kopinya (*elitism*).
8. *Test*. Apabila *stopping criteria* telah terpenuhi, maka iterasi dapat dihentikan dan apabila belum, maka akan kembali pada tahap kedua.

Dalam algoritma genetika, kriteria stabil nilai *fitness* sering digunakan untuk menentukan kapan proses evolusi harus dihentikan. Beberapa kriteria umum stabilitas nilai *fitness* diantaranya pertama, konvergensi *fitness*. Apabila nilai *fitness* terbaik atau rata-rata populasi tidak berubah secara signifikan selama sejumlah generasi tertentu, hal ini dapat dianggap sebagai tanda stabilitas. Misalnya, jika perbedaan nilai *fitness* terbaik antara generasi berturut-turut kurang dari ambang batas yang telah ditentukan (misalnya 0,01) selama 20 generasi terakhir. Kedua, stagnasi *fitness* dimana Jika nilai *fitness* terbaik tidak meningkat setelah

sejumlah generasi tertentu, algoritma bisa dianggap telah mencapai stabilitas. Misalnya, jika tidak ada peningkatan nilai *fitness* terbaik selama 50 generasi terakhir. Ketiga, perbedaan *fitness* dimana Mengamati distribusi nilai *fitness* dalam populasi. Jika variasi atau perbedaan antara nilai *fitness* individu-individu dalam populasi sangat kecil (misalnya, semua individu memiliki nilai *fitness* yang hampir sama), ini bisa dianggap sebagai tanda stabilitas. Keempat, jumlah generasi maksimal, apabila algoritma telah mencapai jumlah generasi yang telah ditentukan sebelumnya, maka proses dihentikan. Hal ini sering digunakan sebagai batas atas untuk mencegah algoritma berjalan terlalu lama tanpa konvergensi. Selain itu, terdapat evaluasi eksternal dimana kriteria eksternal atau *domain-specific* dapat digunakan untuk menentukan stabilitas. Misalnya, mencapai solusi yang memuaskan berdasarkan penilaian eksternal yang tidak hanya bergantung pada nilai *fitness* (Yuliani, Irhamah, & Prastyo, 2014).

2.7 Evaluasi Model

Pemilihan model terbaik melalui pendekatan *in sample* dan *out sample* dalam mengukur tingkat akurasi peramalan dengan menggunakan *Root Mean Square Error* (RMSE) dan *Symmetric Mean Absolute Percentage Error* (sMAPE). Dengan kata lain, RMSE dan sMAPE akan menyatakan seberapa terkonsentrasi data di sekitar garis prediksi. Model dikatakan baik apabila memiliki nilai RMSE dan sMAPE terkecil. Berikut rumus RMSE dan sMAPE.

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (Y_t - \hat{Y}_t)^2} \quad (2.24)$$

$$sMAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{(Y_t - \hat{Y}_t)}{(Y_t + \hat{Y}_t) / 2} \right| \times 100 \quad (2.25)$$

Dimana n menunjukkan banyaknya data, Y_t merupakan data aktual pada waktu ke- t , dan \hat{Y}_t merupakan data prediksi pada waktu ke- t .

2.8 Forecasting

Istilah *forecasting* pertama kali digunakan dalam bidang meteorologi dimana berkaitan dengan ramalan cuaca pada periode *mesoscale* yang sangat singkat hingga dua jam menurut *World Meteorological Organization* (WMO) dan hingga enam jam menurut penelitian lainnya. *Forecasting* dalam bidang ekonomi merupakan prediksi masa kini, masa depan yang sangat dekat, dan masa lalu yang sangat baru. Sekarang ini menjadi populer di bidang ekonomi sebagai langkah standar yang digunakan untuk menilai keadaan ekonomi salah satunya IHK. Model *forecasting* diterapkan oleh banyak institusi untuk memantau keadaan ekonomi. proses membuat prediksi tentang peristiwa masa depan berdasarkan data historis dan analisis tren. Dalam konteks analisis data dan ilmu statistik, *forecasting* sering digunakan untuk memperkirakan berbagai indikator seperti penjualan, permintaan produk, harga pasar, dan lainnya. Teknik *forecasting* sangat penting dalam berbagai bidang seperti ekonomi, bisnis, keuangan, dan meteorologi. Dalam penelitian ini, *forecasting* merupakan peramalan IHK untuk masa lalu, saat ini dan dalam waktu dekat. IHK dirilis dengan *delay* waktu pada data bulanan sehingga banyak *stakeholder* yang peduli dengan masalah pemanfaatan informasi bulanan untuk mendapatkan estimasi awal dari IHK bulan terakhir dan bulan ini serta perkiraan hingga enam bulan kedepan.

2.9 Indeks Harga Konsumen

IHK merupakan suatu nilai indeks yang berasal dari data harga konsumen harian (Meivita, 2014). Peran IHK menjadi indikator ekonomi penting dalam memberikan informasi mengenai perkembangan harga barang dan atau jasa yang dibayarkan oleh konsumen pada suatu wilayah. Perhitungan IHK ditujukan untuk mengetahui perubahan harga dari suatu kelompok tetap barang atau jasa yang umumnya dikonsumsi masyarakat setempat (Damanik & Sinaga, 2014). Berikut merupakan rumus dalam menghitung IHK dengan indeks *Laspeyres*.

$$IHK_n = \frac{\sum_{i=1}^k \frac{P_{ni}}{P_{(n-1)i}} P_{(n-1)i} \cdot Q_{oi}}{\sum_{i=1}^k P_{oi} \cdot Q_{oi}} \quad (2.26)$$

Dimana,

IHK_n : IHK pada bulan ke- n (n : bulan Januari, Februari, ..., Desember)

$\frac{P_{ni}}{P_{(n-1)i}}$: Relatif harga komoditas i pada bulan ke- n

P_{ni} : Harga komoditas i pada bulan ke- n

$P_{(n-1)i}$: Harga komoditas i pada bulan ke- $(n - 1)$

$P_{(n-1)i} \cdot Q_{oi}$: Nilai konsumsi komoditas i pada bulan ke- $(n - 1)$

$P_{oi} \cdot Q_{oi}$: Nilai komoditas i pada tahun dasar

K : Jumlah barang atau jasa yang masuk dalam paket komoditas

Fungsi angka indeks sebagai petunjuk kondisi perekonomian secara umum (Saputra, Setiawan, & Mahatma, 2012). Dengan demikian, IHK sering digunakan untuk mengukur inflasi suatu negara dan sebagai pertimbangan penyesuaian gaji, upah, uang pensiun, maupun kontrak lainnya.

BAB 3 METODOLOGI

3.1 Sumber Data

Data yang digunakan pada penelitian ini merupakan data Indeks Harga Konsumen yang berasal dari Badan Pusat Statistik Kota Surabaya tahun 2019-2023. Periode data pada bulan Januari 2019 hingga Juli 2023. Data pada bulan Januari 2019 hingga Desember 2022 digunakan sebagai data *in sample* dan data Januari hingga Juli 2023 digunakan sebagai data *out sample*.

3.2 Variabel Penelitian

Variabel prediktor dan variabel respon yang digunakan dalam penelitian ini sebagai berikut.

1. Variabel respon (Y_t) merupakan data IHK yang dijadikan sebagai data *actual* atau target
2. Variabel bebas (X) merupakan data IHK pada waktu sebelumnya ($Y_{t-1}, Y_{t-2}, \dots, Y_{t-k}$)

Tabel 3.1 Struktur Data

Tahun	Bulan	Indeks Harga Konsumen	Keterangan
2019	Januari	Y_1	<i>In sample</i>
	Februari	Y_2	<i>In sample</i>
	:	:	:
	Desember	Y_{12}	<i>In sample</i>
:	:	:	:
2022	Desember	Y_{72}	<i>In sample</i>
2023	Januari	Y_{73}	<i>Out sample</i>
	:	:	:
	Juli	Y_{79}	<i>Out sample</i>

3.3 Langkah Analisis

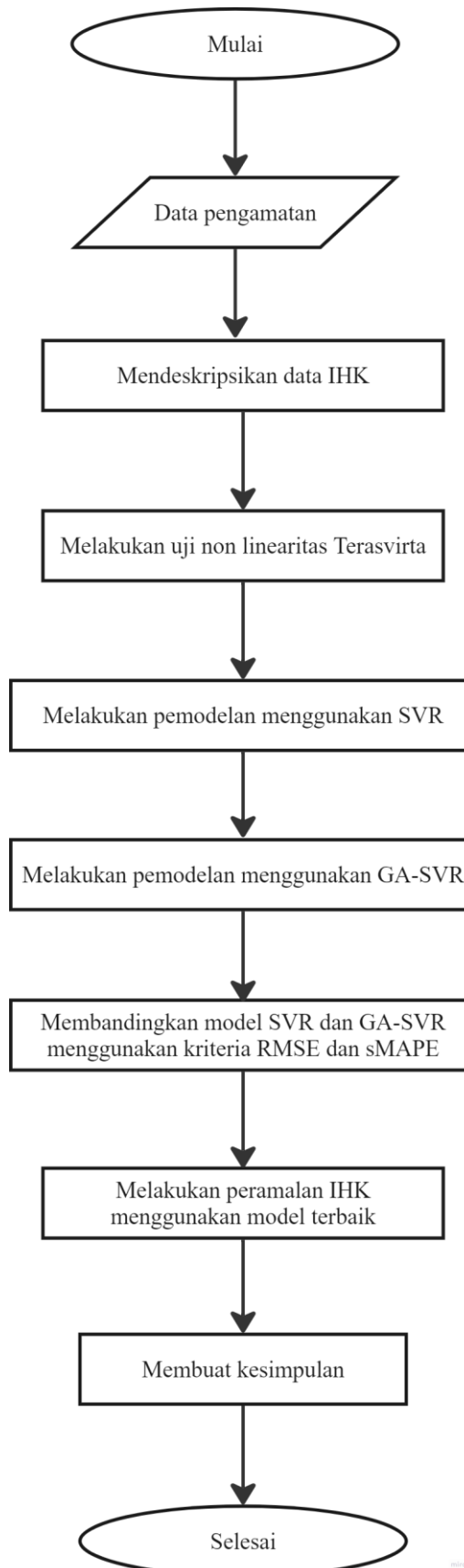
Berikut merupakan langkah analisis yang akan dilakukan dalam penelitian ini.

1. Melakukan analisis statistika deskriptif dan plot data IHK Kota Surabaya.
2. Melakukan uji nonlinearitas Terasvirta
 - a. Meregresikan Y_t sebagai variabel respon (y) dengan *lag-lag* signifikan Y_{t-k} berdasarkan plot-plot PACF sebagai variabel prediktor (x).
 - b. Meregresikan Y_t dengan $1, Y_{t-1}, Y_{t-2}, \dots, Y_{t-k}$ dengan menghitung residual serta jumlah kuadrat residual.
 - c. Meregresikan \hat{a}_t dengan $1, Y_{t-1}, Y_{t-2}, \dots, Y_{t-k}$ dan m prediktor tambahan yang merupakan suku kubik atau kuadrat hasil pendekatan ekspansi deret Taylor.
 - d. Menghitung nilai F dan membandingkannya dengan p -value.
3. Menganalisis dengan metode *Support Vector Regression*
 - a. Membagi data menjadi data *in sample* dan *out sample*.
 - b. Menyusun model SVR untuk meramalkan IHK dengan variabel respon (y) adalah IHK aktual (Y_t) dan variabel prediktor (x) adalah *lag-lag* IHK (Y_{t-k}) dari hasil penguraian model yang memiliki parameter signifikan.
 - c. Menentukan *range* nilai parameter pada *kernel* untuk optimasi *hyperplane* pada data *in sample* hingga diperoleh yang optimum menggunakan *grid search*.
 - d. Melakukan pemodelan SVR berdasarkan *range kernel* yang telah ditentukan.

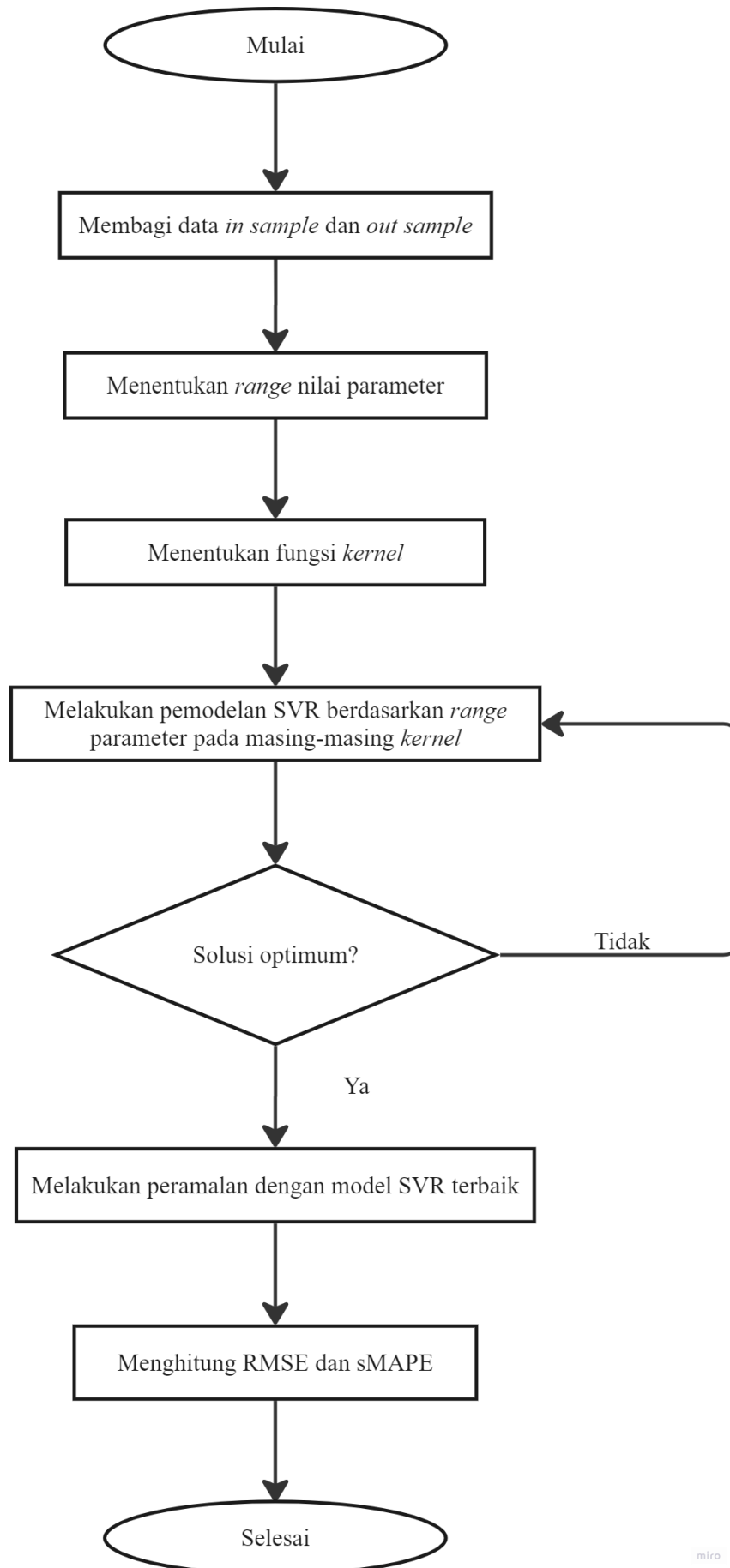
- e. Melakukan pemeriksaan solusi dari model SVR dimana model optimum diperoleh ketika nilai RMSE dan sMAPE minimum.
 - f. Melakukan peramalan menggunakan model SVR.
 - g. Melakukan evaluasi model menggunakan kriteria RMSE untuk *in sample* dan sMAPE untuk *out sample* serta membandingkan performa masing-masing model.
 - h. Pemilihan model terbaik SVR berdasarkan RMSE dan sMAPE terkecil.
4. Menganalisis dengan metode *Genetic Algorithm-Support Vector Regression*
 - a. Membagi data *in sample* dan *out sample*.
 - b. Menentukan kombinasi nilai parameter C, γ , dan ε untuk *kernel* linear, *polynomial*, dan RBF yang diperoleh dari pemodelan SVR pada langkah 3.
 - c. Menggunakan nilai parameter C, γ , dan ε yang didapatkan dari metode SVR sebelumnya sebagai *good initial value*.
 - d. Menyusun kromosom dan inialisasi parameter yang diperlukan dalam optimasi GA dengan membangkitkan populasi awal sebanyak 100, maksimum iterasi sebanyak 20, peluang *crossover* sebesar 0,8 dan peluang mutasi sebesar 0,01.
 - e. Menentukan *fitness function* yang digunakan untuk menghitung nilai *fitness* dalam penelitian ini, yaitu nilai sMAPE.
 - f. Melakukan proses seleksi kromosom dari sejumlah induk yang berasal dari populasi dengan seleksi *roulette wheel*.
 - g. Melakukan proses pindah silang pada individu yang terpilih sebagai orang tua dan mendapatkan individu baru apabila nilai bilangan *random* yang dibangkitkan kurang dari probabilitas pindah silang.
 - h. Melakukan proses mutasi pada individu apabila nilai bilangan *random* yang dibangkitkan kurang dari nilai probabilitas mutasi.
 - i. Melakukan proses *elitism*.
 - j. Melakukan pergantian populasi lama dengan generasi baru dengan cara memilih sejumlah kromosom dengan nilai *fitness* terbaik yang telah melalui proses seleksi, pindah silang, dan *elitism*.
 - k. Melakukan pengecekan terhadap solusi yang telah didapatkan dengan kriteria apabila nilai *fitness* terbaik telah konvergen.
 - l. Melakukan estimasi parameter dengan metode GA.
 - m. Memasukkan parameter yang didapat dari GA ke dalam algoritma SVR.
 - n. Mengevaluasi model berdasarkan nilai RMSE dan sMAPE.
 5. Membandingkan kedua metode, yaitu SVR dan GA-SVR menggunakan kriteria RMSE dan sMAPE.
 6. Memilih model terbaik berdasarkan nilai RMSE dan sMAPE terkecil.
 7. Melakukan peramalan IHK enam bulan selanjutnya berdasarkan model terbaik mulai Agustus 2023 hingga Januari 2024.
 8. Membuat kesimpulan dan saran berdasarkan hasil analisis yang didapatkan.

3.4 Diagram Alir

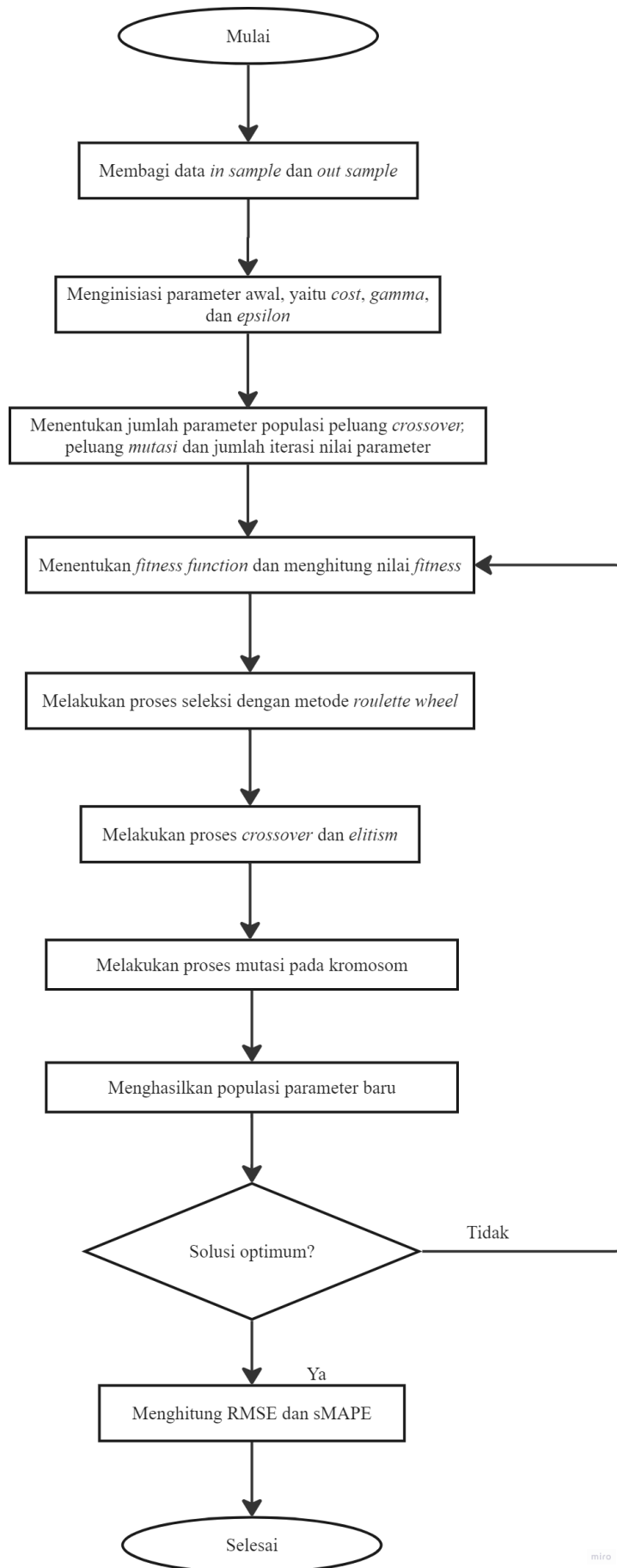
Berikut merupakan diagram alir dalam penelitian ini.



Gambar 3.1 Diagram Alir Penelitian



Gambar 3.2 Diagram Alir Tahapan SVR



Gambar 3.3 Diagram Alir Tahapan GA-SVR

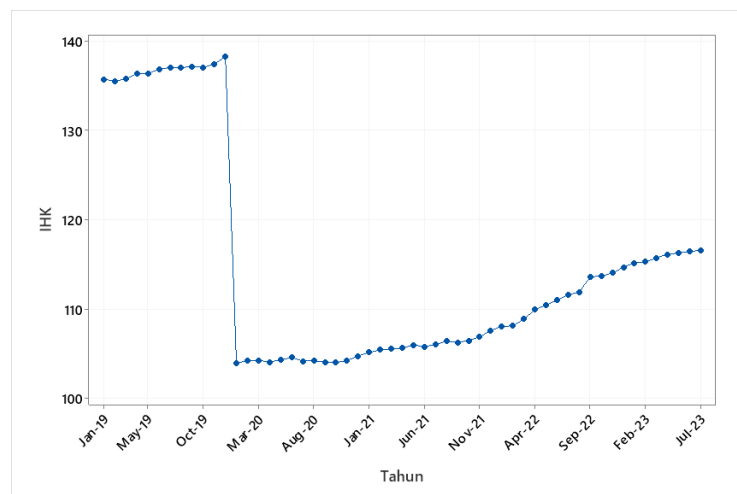
(Halaman ini sengaja dikosongkan)

BAB 4 HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan mengenai hasil analisis dan pembahasan dari prosedur pembentukan model *forecasting* IHK yang dilakukan di Kota Surabaya dengan metode *Genetic Algorithm-Support Vector Regression*. Data yang digunakan dibagi menjadi dua, yaitu data *in sample* dan data *out sample*. Setelah mendapatkan model *forecast* dari metode tersebut maka setiap model akan dievaluasi berdasarkan nilai RMSE dan sMAPE serta dilakukan perbandingan berdasarkan hasil optimasi menggunakan *Genetic Algorithm* untuk memperoleh model peramalan terbaik yang akan digunakan untuk *forecasting* IHK di Kota Surabaya selama enam bulan kedepan.

4.1 Karakteristik IHK Kota Surabaya

Indeks Harga Konsumen (IHK) Kota Surabaya menunjukkan sejumlah karakteristik statistik yang memberikan wawasan penting tentang perilaku harga di kota ini. Dengan nilai *mean* sebesar 114,77, data ini mencerminkan tingkat harga rata-rata di Surabaya dalam periode yang diukur. Nilai rata-rata ini memberikan gambaran umum tentang harga barang dan jasa di kota, yang dapat digunakan sebagai indikator ekonomi yang penting. Selain itu, nilai varians sebesar 152,24 menunjukkan bahwa terdapat penyebaran atau variasi harga yang cukup signifikan di Kota Surabaya. Varians yang tinggi ini mengindikasikan bahwa harga-harga barang dan jasa tidak seragam dan ada perbedaan yang cukup besar antara harga-harga tersebut. Lebih lanjut, nilai maksimum dan minimum masing-masing sebesar adalah 138,23 dan 103,92. Hal ini menunjukkan rentang harga yang luas. Nilai maksimum yang cukup tinggi dibandingkan dengan nilai minimum menunjukkan adanya beberapa barang atau jasa dengan harga yang jauh lebih tinggi dari rata-rata, sementara nilai minimum yang berada di bawah mean menunjukkan adanya barang atau jasa dengan harga yang lebih rendah dari rata-rata. Dalam melihat pola data dan pergerakan IHK dapat menggunakan *time series plot* yang ditunjukkan pada Gambar 4.1.



Gambar 4.1 Time Series Plot IHK

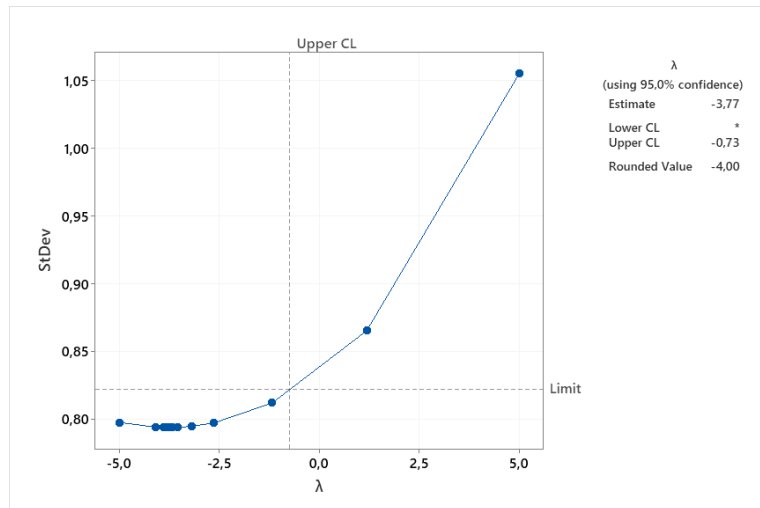
Berdasarkan Gambar 4.1 merupakan *time series plot* IHK Kota Surabaya dari Januari 2019 hingga Juli 2023. Pada awal periode, dari Januari 2019 hingga sekitar Oktober 2019, nilai IHK relatif stabil, berada di kisaran sekitar 137 hingga 138. Stabilitas ini menunjukkan bahwa harga-harga barang dan jasa di Kota Surabaya tidak mengalami perubahan yang signifikan selama periode tersebut. Stabilitas IHK ini mungkin disebabkan oleh kondisi ekonomi yang seimbang,

dimana permintaan dan penawaran barang dan jasa saling menyesuaikan dengan baik tanpa adanya tekanan inflasi yang besar. Namun, pada sekitar Oktober 2019 hingga Maret 2020, terjadi penurunan tajam dalam nilai IHK, yang turun dari sekitar 137 hingga menuju 105. Penurunan yang signifikan ini mencerminkan perubahan besar dalam kondisi ekonomi atau kebijakan yang mempengaruhi harga konsumen secara drastis. Penurunan ini bertepatan dengan awal munculnya pandemi *Covid-19* yang membawa dampak besar pada ekonomi global dan lokal. Pembatasan sosial, penutupan bisnis, dan gangguan pada rantai pasokan global menyebabkan penurunan permintaan dan harga barang serta jasa.

Periode pasca terjadi pandemi *Covid-19* pada nilai IHK tetap stabil di level yang lebih rendah sekitar 105 hingga awal tahun 2022. Periode stabil ini berlangsung cukup lama, menandakan bahwa setelah penurunan tajam, harga konsumen tidak mengalami banyak fluktuasi dan tetap berada di level yang baru dan lebih rendah. Stabilitas ini menunjukkan periode adaptasi dan penyesuaian dari perekonomian terhadap kondisi baru yang dibawa oleh pandemi. Selama periode ini, masyarakat dan bisnis mungkin telah beradaptasi dengan kondisi baru, dan kebijakan stimulus ekonomi mungkin telah membantu menstabilkan harga di level yang lebih rendah. Memasuki tahun 2022, mulai terlihat tren kenaikan dalam nilai IHK. Mulai dari sekitar Maret 2022, nilai IHK mulai meningkat secara bertahap dan berkelanjutan hingga Juli 2023, dari sekitar 105 menuju 114. Tren ini menunjukkan adanya peningkatan harga konsumen yang mungkin dipengaruhi oleh berbagai faktor seperti inflasi, peningkatan biaya produksi, atau peningkatan permintaan barang dan jasa, serta pemulihan ekonomi pasca pandemi *Covid-19*. Peningkatan ini mencerminkan pemulihan ekonomi di mana permintaan konsumen mulai pulih, bisnis mulai beroperasi kembali dengan kapasitas penuh, dan aktivitas ekonomi mulai kembali normal. Faktor-faktor lain seperti kenaikan harga bahan baku, gangguan rantai pasokan, dan kebijakan moneter juga dapat berkontribusi pada IHK.

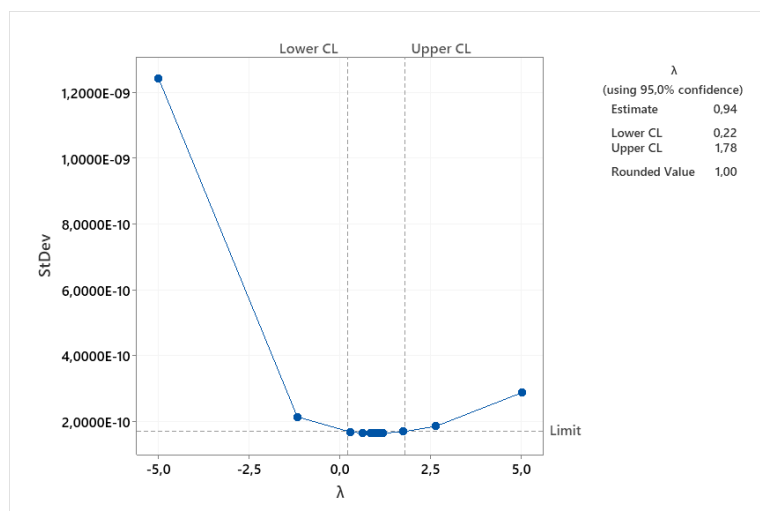
Secara keseluruhan, grafik ini memberikan gambaran yang jelas tentang fluktuasi nilai IHK selama periode lebih dari empat tahun. Penurunan tajam pada akhir 2019 dan awal 2020 diikuti oleh periode stabil dan kemudian kenaikan bertahap dari awal 2022 hingga pertengahan 2023, menunjukkan dinamika yang kompleks dalam harga konsumen selama periode tersebut. Perubahan signifikan ini dapat diakibatkan oleh berbagai faktor seperti kebijakan ekonomi, kejadian global pandemi *Covid-19*, serta perubahan dalam permintaan dan penawaran. Selama periode ini, berbagai kebijakan ekonomi seperti stimulus fiskal dan moneter mungkin telah diberlakukan untuk mengatasi dampak ekonomi dari pandemi. Selain itu, perubahan dalam perilaku konsumen dan bisnis, adaptasi terhadap norma baru, dan perubahan dalam rantai pasokan global juga turut mempengaruhi dinamika IHK. Grafik ini menunjukkan betapa kompleksnya interaksi antara berbagai faktor ekonomi dan bagaimana mereka dapat mempengaruhi harga konsumen selama periode yang diperpanjang.

Berdasarkan Gambar 4.1 diketahui bahwa IHK Kota Surabaya memiliki pola yang cenderung naik dari tahun ke tahun setelah terjadi penurunan signifikan pada tahun pandemi *Covid-19*. Meskipun secara visual data cenderung tidak stasioner, perlu dilakukan analisis untuk memeriksa apakah data benar-benar tidak stasioner dalam mean maupun varians. Stasioneritas data dalam varians dapat dilakukan dengan transformasi *Box-Cox*. Berdasarkan Gambar 4.2 diketahui bahwa data IHK Kota Surabaya belum stasioner terhadap varians. Hal ini ditunjukkan dengan nilai *Lower CL* dan *Upper CL* yang belum melewati angka satu atau pada *rounded value* yang belum bernilai satu. Dengan *rounded value* sebesar $-4,0$ maka data akan ditransformasikan dengan rumus $Y(\lambda) = (Y^\lambda - 1)/\lambda$ untuk menstabilkan varians.



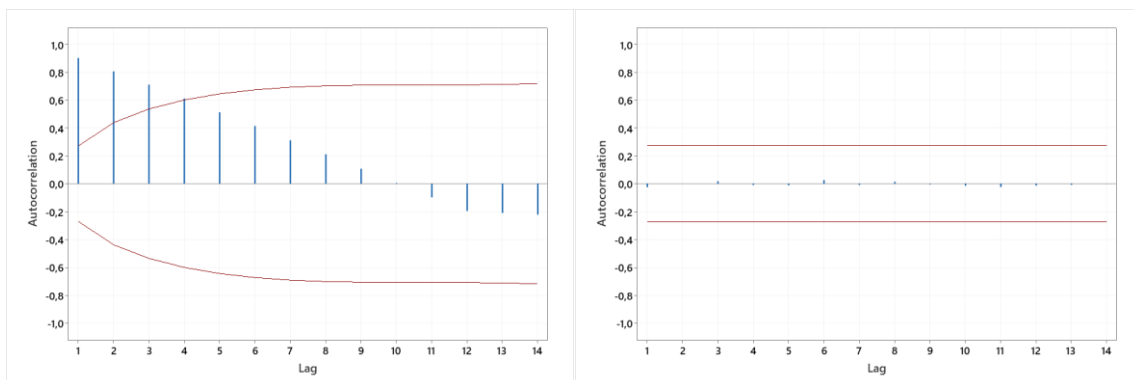
Gambar 4.2 Box-Cox Plot IHK

Setelah dilakukan transformasi maka dilakukan analisis Box Cox kembali untuk mengetahui apakah data telah stasioner terhadap varians. Pada Gambar 4.3 hasil transformasi menunjukkan bahwa *Box-Cox plot* data IHK Kota Surabaya memiliki nilai *Lower CL* dan *Upper CL* sudah melewati angka satu dan nilai *rounded value* bernilai satu sehingga data hasil transformasi sudah stasioner terhadap varians.



Gambar 4.3 Transformasi *Box-Cox* IHK

Pada Gambar 4.4 (a) menunjukkan plot ACF membentuk pola *dies down* atau bergerak turun lambat. Hal ini menunjukkan bahwa data memiliki komponen tren yang signifikan dimana mengindikasikan bahwa data tidak stasioner.

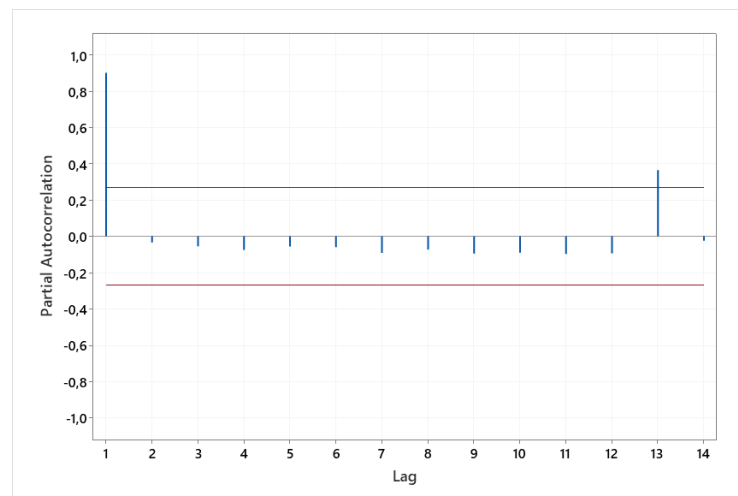


Gambar 4.4 (a) Plot ACF Sebelum *Differencing* dan **(b)** Plot ACF *Differencing* Orde 1

Berdasarkan Gambar 4.4 (a) diperoleh informasi bahwa data transformasi secara visual belum stasioner terhadap mean yang ditunjukkan bahwa plot yang tidak berfluktuasi di sekitar garis *mean*. Plot ACF tersebut memiliki pola *dies down* sehingga dapat dikatakan data belum stasioner terhadap *mean* sehingga perlu dilakukan *differencing* pada *lag* satu dengan hasil plot seperti pada gambar 4.4 (b).

4.2 Pengujian Nonlinearitas Terasvirta

Pada penelitian ini, sebelum melakukan pemodelan SVR, terlebih dahulu dilakukan pendeteksian nonlinearitas pada data IHK Kota Surabaya. Pendeteksian nonlinearitas pada data IHK menggunakan uji Terasvirta. Berdasarkan plot PACF pada Gambar 4.5 bahwa *lag* satu dan *lag* tiga belas memiliki nilai yang signifikan keluar batas yang ditentukan. Oleh karena itu, *lag-lag* tersebut dimasukkan sebagai variabel independen dalam uji Terasvirta.



Gambar 4.5 Plot PACF Data IHK Kota Surabaya

Pengujian nonlinearitas pada data IHK Kota Surabaya dilakukan untuk mengetahui apakah variabel IHK Kota Surabaya memiliki hubungan linear atau tidak secara signifikan terhadap lag data. Adapun hipotesis dari uji Terasvirta ditunjukkan sebagai berikut.

H_0 : Parameter pada regresi polinomial (δ) bersifat linier ($\delta_1 = \delta_2 = \delta_3 = \delta_4 = \delta_5 = \delta_6$)

H_0 : Salah satu parameter tidak linier ($\delta_i \neq \delta_j$)

Tabel 4.1 Hasil Uji Terasvirta

Variabel	F_{hitung}	$p-value$
IHK	7,5306	0,001716

Tabel 4.1 memberikan informasi bahwa hasil dari uji Terasvirta menunjukkan bahwa nilai $p-value$ sebesar 0,001716 sehingga nilai $p-value$ lebih kecil daripada taraf signifikansi ($\alpha = 0,05$) yang artinya tolak H_0 . Hasil analisis menunjukkan bahwa data IHK Kota Surabaya memiliki pola nonlinear sehingga dapat disimpulkan bahwa data IHK Kota Surabaya merupakan data nonlinear. Dengan demikian, dalam melakukan *forecasting* IHK Kota Surabaya dapat dilanjutkan dengan menggunakan SVR karena terbukti bersifat nonlinear.

4.3 Pemodelan IHK Kota Surabaya Menggunakan *Support Vector Regression*

Pada penelitian ini, metode *Support Vector Regression* digunakan dalam melakukan pemodelan untuk IHK Kota Surabaya. Data IHK Kota Surabaya dibagi menjadi data *in sample* dan *out sample*. Data *in sample* pada bulan Januari 2019 hingga Desember 2022 dan data *out sample* pada bulan Januari 2023 hingga Juli 2023.

4.3.1 Penentuan Data *Input* dan Data *Output*

Langkah awal yang dilakukan adalah mengonversi data IHK ke dalam bentuk *time lag* dalam *input* pemodelan SVR. Penentuan banyaknya *time lag* berdasarkan plot PACF pada seluruh *lag* yang signifikan. Berdasarkan Gambar 4.5 plot PACF menunjukkan bahwa *lag* yang signifikan adalah *lag* 1 dan *lag* 13. Dengan demikian data *output* SVR, yaitu y_{t-1} dan y_{t-13} artinya IHK pada bulan ini dipengaruhi oleh IHK pada satu bulan lalu dan tiga belas bulan lalu. Selanjutnya, data *output* dan *input* yang dihasilkan akan digunakan dalam pemodelan SVR.

4.3.2 Penentuan Parameter Optimal

Langkah selanjutnya yang dilakukan adalah penentuan parameter optimal. Pada penelitian ini digunakan fungsi *kernel* linear, *polynomial*, dan RBF dalam mendapatkan parameter terbaik berdasarkan nilai RMSE terkecil. Tahapan awal metode *grid search* adalah menentukan rentang nilai parameter. Pada penelitian-penelitian sebelumnya, *range* parameter yang sering digunakan adalah $C = \{10^{-5}, 10^{-4}, 10^{-3}, \dots, 10^4, 10^5\}$ atau $C = \{2^{-5}, 2^{-3}, \dots, 2^{13}, 2^{15}\}$; parameter $\gamma = \{2^{-15}, 2^{-13}, \dots, 2^3, 2^5\}$; dan parameter $\varepsilon = \{0,01; 0,1; 0,5; 1\}$. Pada penelitian ini, sebelum dilakukan optimasi GA akan menentukan *range* parameter yang ditunjukkan pada Tabel 4.2

Tabel 4.2 Range Parameter

Parameter	Range
ε	0,01; 0,1
γ	$2^{-5}, 2^{-3}, 2^0, 2^3, 2^5$
C	$2^{-5}, 2^0, 2^5$

Pada penentuan parameter terbaik, dilakukan metode *grid search* atau mencoba kombinasi dari beberapa nilai parameter pada range tertentu. Tabel 4.6 merupakan rangkuman hasil pencarian parameter C , γ , dan ε yang optimum pada masing-masing *input*. Pada *kernel* linear, dilakukan percobaan parameter C , yaitu $2^{-5}, \dots, 2^0, \dots, 2^5$. *Kernel polynomial* dilakukan percobaan range parameter C pada $2^{-5}, \dots, 2^0, \dots, 2^5$, serta *degree* dengan nilai 1, 2, dan 3 dan *scale*, yaitu $2^0, \dots, 2^5$. *Kernel RBF*, range parameter γ dan C yang telah dicobakan mulai dari $2^{-5}, \dots, 2^0, \dots, 2^5$. Pada penentuan parameter terbaik, dilakukan metode *grid search* atau mencoba kombinasi dari beberapa nilai parameter pada range tertentu. Rincian kombinasi nilai parameter yang terbentuk dari masing-masing kernel beserta nilai RMSE seperti yang dituliskan berikut.

Tabel 4.3 Kombinasi Nilai Parameter *Kernel* Linear

C	RMSE
2^{-5}	3,7248115
2^{-4}	1,8940256
2^{-3}	0,6293221
2^{-2}	0,5862847
2^{-1}	0,5527760
2^0	0,6770471
2^1	0,6775000
2^2	0,6786428
2^3	0,6771847
2^4	0,6773829
2^5	0,6772933

Tabel 4.4 Kombinasi Nilai Parameter *Kernel Polynomial*

<i>C</i>	<i>degree</i>	<i>scale</i>	RMSE
2^{-5}	1	2^0	3,7248115
2^{-5}	1	2^1	1,8940256
2^{-5}	1	2^2	0,6293221
2^{-5}	1	2^3	0,5862847
2^{-5}	1	2^4	0,5527760
2^{-5}	1	2^5	0,6770473
2^{-5}	2	2^0	2,0912332
2^{-5}	2	2^1	1,4955213
2^{-5}	2	2^2	1,1564303
2^{-5}	2	2^3	0,8437329
2^{-5}	2	2^4	0,5338889
2^{-5}	2	2^5	0,5455299
2^{-5}	3	2^0	0,6784156
2^{-5}	3	2^1	5,9294039
2^{-5}	3	2^2	6,1247501
2^{-5}	3	2^3	6,3749974
2^{-5}	3	2^4	6,5680794
2^{-5}	3	2^5	6,7554718
.	.	.	.
.	.	.	.
.	.	.	.
2^{-1}	2	2^0	0,7937722
2^{-1}	2	2^1	0,5322847
2^{-1}	2	2^2	0,5330358
2^{-1}	2	2^3	0,5339550
2^{-1}	2	2^4	0,5328990
2^{-1}	2	2^5	0,5335436
.	.	.	.
.	.	.	.
.	.	.	.
2^3	1	2^0	0,6782650
2^3	1	2^1	0,6777817
2^3	1	2^2	0,6777274
2^3	1	2^3	0,6782232
2^3	1	2^4	0,6781126
2^3	1	2^5	0,6760704
2^3	2	2^0	0,5302452
2^3	2	2^1	0,5329269
2^3	2	2^2	0,5330168
2^3	2	2^3	0,5334267
2^3	2	2^4	0,5314263
2^3	2	2^5	0,5357417
2^3	3	2^0	7,3589314
2^3	3	2^1	7,9225719

2^3	3	2^2	8,5854978
2^3	3	2^3	8,4796330
2^3	3	2^4	9,1213892
2^3	3	2^5	11,9808907
2^4	1	2^0	0,6777865
2^4	1	2^1	0,6782146
2^4	1	2^2	0,6777790
2^4	1	2^3	0,6777265
.	.	.	.
.	.	.	.
.	.	.	.

Tabel 4.5 Kombinasi Nilai Parameter Kernel RBF

C	γ	RMSE
2^{-5}	2^{-5}	7,4581958
2^{-5}	2^{-4}	7,4170713
2^{-5}	2^{-3}	7,4853712
2^{-5}	2^{-2}	7,6457065
2^{-5}	2^{-1}	7,7369758
2^{-5}	2^0	7,7547711
2^{-5}	2^1	7,7211445
2^{-5}	2^2	7,7107983
2^{-5}	2^3	7,6737645
2^{-5}	2^4	7,6871032
2^{-5}	2^5	7,6569873
2^{-5}	2^{-5}	7,2395983
2^{-5}	2^{-4}	7,1995612
2^{-5}	2^{-3}	7,2739599
.	.	.
.	.	.
.	.	.
2^3	2^{-5}	1,4728398
2^3	2^{-4}	2,3215877
2^3	2^{-3}	3,3903787
2^3	2^{-2}	4,0417249
2^3	2^{-1}	4,7752259
2^3	2^0	5,0493219
2^3	2^1	5,4755221
2^3	2^2	5,7078343
2^3	2^3	5,8500927
2^3	2^4	5,8826993
2^3	2^5	6,1093206
.	.	.
.	.	.
.	.	.
2^5	2^{-5}	0,9653101
2^5	2^{-4}	1,8869498

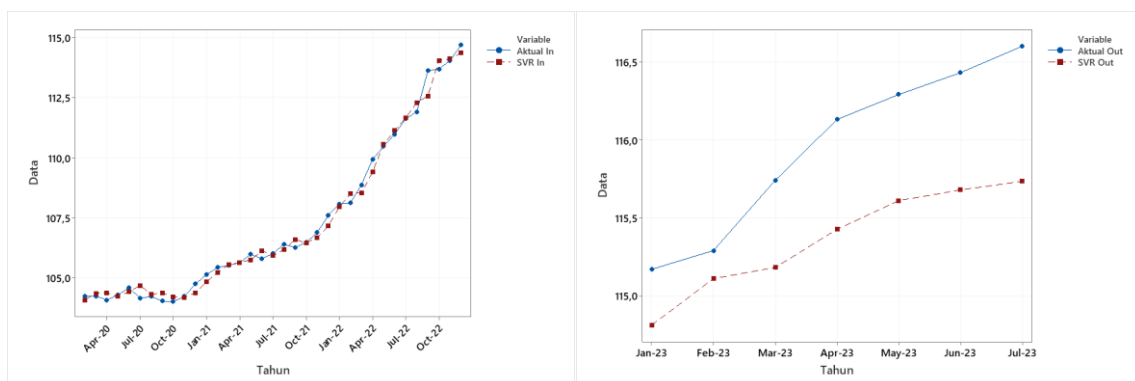
2^5	2^{-3}	3,0297227
2^5	2^{-2}	4,0936578
2^5	2^{-1}	4,4866090
2^5	2^0	4,9850172
2^5	2^1	5,5771476
2^5	2^2	5,8320382
2^5	2^3	5,8154086
2^5	2^4	5,8968528
2^5	2^5	6,0903575

Berdasarkan Tabel 4.3 hingga Tabel 4.5, diketahui kombinasi-kombinasi nilai parameter yang terbentuk dari *range* yang telah ditetapkan sebelumnya beserta nilai RMSE untuk masing-masing *kernel* linear, *polynomial*, dan RBF. Kombinasi nilai parameter terbaik dipilih berdasarkan nilai RMSE terkecil pada data validasi dalam *forecasting* IHK Kota Surabaya, dimana nilai RMSE yang diperoleh pada kombinasi parameter terbaik untuk masing-masing *kernel* adalah sebagai berikut.

Tabel 4.6 Nilai RMSE Kombinasi Parameter Terbaik

<i>Kernel</i>	Parameter					RMSE <i>in Sample</i>	RMSE <i>out Sample</i>	sMAPE <i>out Sample</i>
	C	γ	ϵ	<i>scale</i>	<i>degree</i>			
<i>Linear</i>	2^{-1}	-	0,1			0,3553293	0,2040144	0,1461814
<i>Polynomial</i>	2^5	-	0,1	2^4	2^1	0,7561149	0,3803943	0,2671095
RBF	2^5	2^{-5}	0,1			0,3161861	0,626140	0,5054331

Diketahui pada Tabel 4.6, kombinasi parameter terbaik adalah menggunakan fungsi *kernel* RBF dengan nilai kombinasi parameter C sebesar 2^5 , γ sebesar 2^{-5} dan ϵ sebesar 0,1. Nilai RMSE *kernel* RBF yang dihasilkan pada data validasi, yaitu 0,5054331 dimana nilai RMSE *in sample* tersebut lebih kecil dari *kernel* lainnya, yaitu *kernel* linear dengan nilai RMSE *in sample* sebesar 0,3553293 dan *kernel polynomial* dengan nilai RMSE *in sample* sebesar 0,7561149. Oleh karena itu, *forecasting* IHK Kota Surabaya dengan metode *Support Vector Regression* dilakukan menggunakan *kernel Radial Basis Function* dengan kombinasi nilai parameter $C = 2^5$, $\gamma = 2^{-5}$, dan $\epsilon = 0,1$.



Gambar 4.6 Time Series Plot Model SVR Terbaik pada Data (a) *In Sample* dan (b) *Out Sample*

Berdasarkan Gambar 4.6 (a) menunjukkan bahwa model SVR data *in sample* mampu mengikuti pola data aktual dengan cukup baik, meskipun terdapat beberapa deviasi yang menunjukkan bahwa model ini belum sepenuhnya akurat. Pola hasil ramalan tidak selalu sesuai dengan pola data aktual, terutama pada beberapa periode tertentu, yang mengindikasikan bahwa model ini masih memerlukan penyempurnaan lebih lanjut. Secara umum, hasil ramalan dari

$input\ f(y_{t-1}, y_{t-13})$ memiliki akurasi yang cukup baik dalam meramalkan IHK Kota Surabaya dan dapat menggambarkan data aktual dengan cukup baik. Pada Gambar 4.6 (b) menunjukkan bahwa data *out sample* dengan hasil peramalan menggunakan model SVR memiliki performa yang baik dalam mengikuti pola data aktual IHK Kota Surabaya. Hasil peramalan terlihat cukup akurat, terutama di awal dan akhir periode. Namun, ada beberapa perbedaan yang cukup signifikan pada bulan-bulan tertentu, seperti bulan Mei dan Juni, dimana model SVR kurang tepat dalam menangkap fluktuasi data aktual. Meskipun demikian, secara visual, model SVR memberikan hasil peramalan yang cukup baik sehingga dapat disimpulkan bahwa model SVR $input\ f(y_{t-1}, y_{t-13})$ merupakan model yang cukup andal meramalkan IHK Kota Surabaya.

4.4 Pemodelan IHK Kota Surabaya Menggunakan *Genetic Algorithm-Support Vector Regression*

Pada sub bab sebelumnya, telah diperoleh parameter SVR yang menghasilkan nilai RMSE dan sMAPE yang minimum menggunakan metode *grid search*. Agar memperoleh nilai *error* yang lebih optimum, dapat dilakukan optimasi pada parameter SVR. Salah satu algoritma yang dapat digunakan adalah metode *Genetic Algorithm* (GA). Pada *Genetic Algorithm* akan menggunakan model SVR terbaik yang terbentuk pada langkah sebelumnya, kemudian model tersebut akan dioptimalkan parameternya menggunakan *Genetic Algorithm*. Penggunaan *Genetic Algorithm* dimaksudkan untuk mengoptimalkan parameter SVR sehingga memiliki nilai RMSE dan sMAPE yang lebih kecil. Pada analisis ini akan dilakukan optimasi menggunakan *input SVR* seperti pada Tabel 4.6. Langkah awal yang dilakukan adalah mendefinisikan atau menginisialisasi kromosom, yaitu jumlah populasi pada variabel yang digunakan sebanyak 100 dengan batas iterasi sebanyak 20 dengan peluang terjadinya pindah silang. Nilai inisial kromosom akan menggunakan nilai parameter dari model SVR. Dimana setiap kromosom yang dibangkitkan memiliki tiga gen. Tiga gen ini merupakan parameter dari model SVR yang terpilih dengan menggunakan fungsi kernel RBF, yaitu C , γ , dan ϵ . Ilustrasi *input* yang digunakan $f(y_{t-1}, y_{t-13})$ dimana nilai parameter yang didapatkan dari SVR adalah $C = 2^5$, $\gamma = 2^{-5}$, dan $\epsilon = 0,1$. Berikut merupakan ilustrasi kromosom dengan tiga gen yang ditunjukkan pada Gambar 4.7.

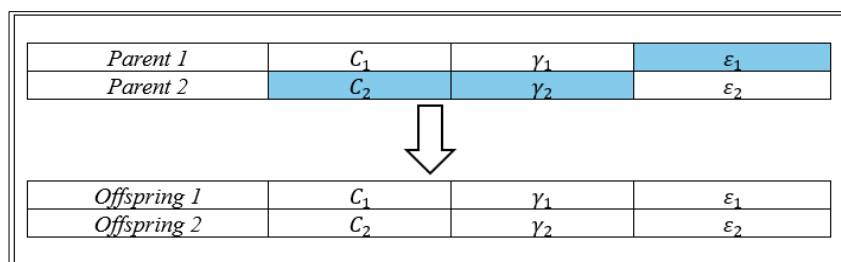
Parameter	Cost	γ	ϵ
Kromosom	2^5	2^{-5}	0,1

Gambar 4.7 Ilustrasi Kromosom Parameter SVR Tiga Gen

Salah satu contoh kromosom dengan tiga gen yang dibentuk dari model SVR dengan variasi *trial and error* ditunjukkan oleh Gambar 4.7. Nilai parameter yang telah diperoleh dari SVR akan dijadikan *initial value* pada GA-SVR kemudian akan *generate* sebanyak jumlah populasi sehingga terbentuk 100 kromosom dengan nilai yang bervariasi. Kromosom yang terbentuk akan menjalani proses seleksi *roulette wheel*, *crossover*, mutasi dan *elitism* sehingga diperoleh hasil yang konvergen dengan nilai RMSE dan sMAPE yang lebih kecil. Kromosom yang baik ditentukan dari nilai *fitness* sehingga perlu ditentukan terlebih dahulu. Nilai *fitness* yang digunakan pada penelitian ini adalah nilai kesalahan peramalan, yaitu sMAPE yang lebih kecil. Nilai *fitness* merupakan acuan dalam tahapan *Genetic Algorithm* seperti proses seleksi, pindah silang (*crossover*), mutasi dan *elitism* sehingga didapatkan nilai *fitness* terbaik. Tujuan objektif pada penelitian ini adalah meminimumkan nilai RMSE dan sMape pada model SVR. Jika nilai sMAPE semakin kecil, maka kromosom dikatakan baik dan dapat bertahan hidup. Sebelum melewati tahapan dalam GA akan dihitung terlebih dahulu nilai *fitness* berdasarkan kromosom-

kromosom yang terbentuk. Nilai *fitness* tersebut akan menjadi acuan dalam melakukan proses GA selanjutnya hingga diperoleh nilai *fitness* yang paling minimum.

Tahap selanjutnya adalah *selection* dimana pada penelitian ini menggunakan metode *roulette wheel*. Seleksi *roulette wheel* merupakan salah satu cara dalam menentukan orang tua kromosom mana yang akan melakukan proses pindah silang atau *crossover*. Orang tua kromosom yang terpilih dalam proses seleksi *roulette wheel* melibatkan nilai *fitness* relatif dan kumulatif. Setelah diperoleh nilai *fitness* relatif dan *fitness* kumulatif, selanjutnya membangkitkan bilangan *random* (0,1) sebanyak jumlah kromosom 100. Bilangan *random* ini digunakan sebagai pembanding setiap individu apakah layak menjadi individu baru atau tidak. Pada tahap ini, diperoleh nilai *fitness* pada masing-masing kromosom sebagai penentu apakah sebuah kromosom terpilih sebagai calon orang tua di generasi selanjutnya. Sebagai contoh, nilai bilangan *random* kromosom pertama lebih kecil dibandingkan dengan nilai *fitness* kumulatif kromosom kedua, sehingga kromosom kedua akan menjadi calon orang tua untuk generasi selanjutnya. Tahap selanjutnya setelah dilakukan proses seleksi kromosom induk (orang tua) adalah proses pindah silang (*crossover*). Kromosom yang terpilih sebagai calon orang tua akan diberikan bilangan *random* $U(0,1)$ dimana dua kromosom orang tua digabungkan untuk menghasilkan dan membentuk dua kromosom anak. Berikut ini adalah ilustrasi dari pindah silang pada kromosom.



Gambar 4.8 Ilustrasi Pindah Silang

Crossover melibatkan dua kromosom orang tua yang akan membentuk dua kromosom anak. Proses pindah silang terjadi apabila bilangan *random* bernilai lebih kecil dari probabilitas pindah silang ($P_c = 0,8$), maka orang tua yang terpilih dari individu satu dan individu dua akan dikawinkan sehingga menghasilkan keturunan individu baru. Tipe *crossover* yang digunakan salah satunya adalah *local arithmetic crossover* dimana menggunakan nilai bilangan real pada algoritma perhitungannya. Berikut ilustrasi proses pindah silang kromosom orang tua satu dan kromosom orang tua dua yang akan menghasilkan anak satu dan anak dua pada Gambar 4.9.

Orang Tua 1	1,0019	0,0478	0,456
Orang Tua 2	0,9977	0,0488	0,4763
↓			
Anak 1	0,999404	0,048394	0,468066
Anak 2	1,000196	0,048206	0,464234

Gambar 4.9 Ilustrasi Kromosom Melakukan Crossover

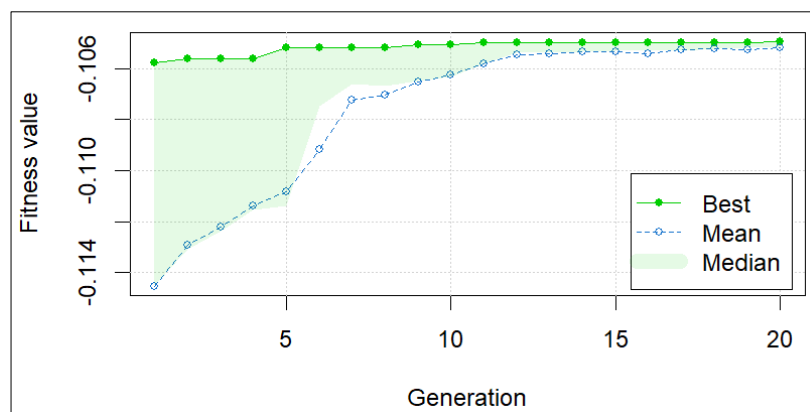
Gambar 4.9 menunjukkan ilustrasi kromosom orang tua dengan hasil *crossover*. Misalkan setelah bilangan *random* dibangkitkan diperoleh nilai $\alpha = 0,4055$ dan dihitung berdasarkan rumus perhitungan tipe *crossover* yang digunakan kemudian dari proses *crossover* akan menghasilkan anak satu dan anak dua. Proses selanjutnya adalah mutasi kromosom dengan probabilitas mutasi (P_m) sebesar 0,01. Proses memilih gen yang akan dilakukan mutasi dengan membangkitkan bilangan *random* antara nol sampai satu sebanyak gen dalam kromosom.

Apabila nilai bilangan *random* kurang dari probabilitas mutasi (P_m), maka gen pada kromosom tersebut akan dimutasi. Proses mutasi ini dilakukan dengan mengganti nilai gen tersebut dengan bilangan acak baru. Berikut ilustrasi bagaimana proses mutasi pada kromosom Gambar 4.10.

Sebelum dimutasi	Bil. Random	0,5161	0,3332	0,0019
	Kromosom	1,0233	0,0443	0,4112
Setelah dimutasi	Bil. Random	1,0234	0,0443	0,4315

Gambar 4.10 Ilustrasi Proses Mutasi

Proses mutasi dilakukan untuk mencegah terjadinya konvergensi yang prematur. Dimana pada umumnya terjadi ketika populasi dalam algoritma genetik terlalu homogen sehingga kehilangan unsur heterogenitas pada genetiknya. Gambar 4.10 menunjukkan bahwa gen ketiga adalah gen yang dilakukan mutasi karena pada gen tersebut memiliki bilangan random yang lebih kecil dari probabilitas mutasi (P_m) sebesar 0,01 sehingga gen dalam kromosom tersebut akan diganti dengan bilangan acak. GA akan *generate* generasi baru sebanyak 20 iterasi hingga diperoleh nilai parameter yang konvergen. Proses terakhir yang digunakan untuk mendapatkan nilai *fitness* terbaik adalah *elitism*. Proses *elitism* ini akan mempertahankan estimasi parameter yang menghasilkan nilai *fitness* terkecil untuk generasi selanjutnya. Jumlah kromosom terbaik yang dipertahankan untuk generasi selanjutnya adalah lima kromosom. Kromosom yang bertahan ke generasi selanjutnya adalah kromosom yang menghasilkan *fitness* terkecil. Lima kromosom dengan nilai *fitness* terbaik akan bertahan pada generasi kedua. Proses berikutnya generasi kedua juga akan melakukan proses *elitism* hingga generasi-generasi selanjutnya sampai kondisi *fitness* telah konvergen. Nilai parameter yang terbaik pada generasi sebelumnya akan disimpan pada proses *elitism* sehingga generasi selanjutnya akan memperoleh nilai parameter dengan nilai *fitness* terbaik. Pada Gambar 4.11 menunjukkan bahwa nilai *fitness* telah mencapai konvergensi pada iterasi/generasi ke-20 dalam GA.



Gambar 4.11 Nilai *Fitness* Konvergen

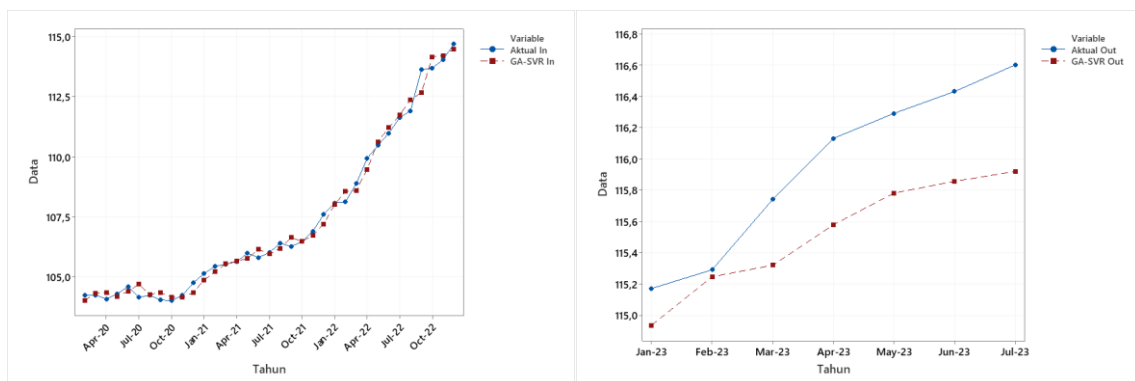
Berdasarkan Gambar 4.11 diperoleh nilai *fitness* dari serangkaian proses GA yang telah dilakukan. Jumlah kromosom terbaik yang dipertahankan untuk generasi selanjutnya adalah lima kromosom. Pada garis *best* menunjukkan nilai *fitness* terbaik yang dicapai setiap generasi meningkat dengan cepat di awal sekitar lima generasi pertama kemudian stabilitas di generasi selanjutnya. Hal ini menunjukkan bahwa solusi mendekati optimal di awal proses. Pada garis *mean* nilai *fitness* rata-rata dari seluruh populasi di setiap generasi. Garis ini menunjukkan naik ke atas menandakan bahwa secara keseluruhan rata-rata kualitas solusi di populasi meningkat

seiring waktu sehingga menunjukkan populasi bergerak menuju solusi yang lebih baik. Area median mewakili nilai *fitness* median dari populasi memberikan gambaran mengenai distribusi kualitas solusi dalam populasi mendekati garis terbaik. Secara keseluruhan, grafik menunjukkan konvergensi yang cepat di generasi awal dimana nilai terbaik, rata-rata, dan median mendekati satu sama lain. Setelah generasi ke-10, nilai *fitness* terbaik, rata-rata, dan median cenderung stabil mengindikasikan bahwa algoritma mencapai kondisi stabil dimana tidak ada perubahan signifikan lebih lanjut yang ditemukan. Peningkatan nilai *fitness* dari generasi ke generasi menunjukkan bahwa GA berhasil mengoptimalkan parameter SVR dalam meminimalkan kesalahan prediksi yang diukur menggunakan sMAPE. Parameter optimum yang didapatkan untuk *input* $f(y_{t-1}, y_{t-13})$ adalah $\varepsilon = 0,06756186$; $\gamma = 0,02830025$; dan $C = 29,84129$ sehingga dirangkum pada Tabel 4.7 berikut.

Tabel 4.7 Parameter Optimum GA-SVR

<i>Input</i>	<i>C</i>	γ	ε	RMSE <i>in sample</i>	RMSE <i>out sample</i>	sMAPE <i>out sample</i>
$f(y_{t-1}, y_{t-13})$	29,84129	0,02830025	0,06756186	0,3120703	0,4770836	0,3720165

Berdasarkan Tabel 4.7, optimasi menggunakan GA pada parameter SVR memberikan hasil nilai RMSE dan sMAPE yang lebih rendah dibandingkan dengan perolehan nilai sebelum dilakukannya optimasi yang ditunjukkan pada Tabel 4.6. Dengan adanya proses optimasi GA model SVR prediksi *input* $f(y_{t-1}, y_{t-13})$ terpilih sebagai model terbaik. Hal ini didasarkan pada nilai kriteria *error* terendah baik RMSE maupun sMAPE data *out sample* dan *in sample*. Dalam hal ini menunjukkan bahwa penggunaan GA untuk optimasi parameter SVR dapat menghasilkan akurasi hasil ramalan yang lebih baik daripada SVR sebelum optimasi. Berikut visualisasi kesesuaian dengan data aktual.



Gambar 4.12 Time Series Plot Model GA-SVR Terbaik Data (a) *In Sample* dan (b) *Out Sample*

Berdasarkan Gambar 4.12 (a) menunjukkan plot hasil peramalan IHK Kota Surabaya menggunakan GA-SVR dengan perbandingan plot data aktual IHK Kota Surabaya. Hasil analisis visual dari model *input* $f(y_{t-1}, y_{t-13})$ menunjukkan performa model yang baik dimana kesesuaian pola antara hasil ramalan dengan data aktual. Walaupun masih terdapat beberapa yang belum akurat sepenuhnya yang terlihat dari gambar pola data *in sample*. Model cukup berhasil dalam menunjukkan tren fluktuasi data aktual. Pada Gambar 4.12 (b) menunjukkan bahwa hasil peramalan IHK Kota Surabaya pada data *out sample* cukup baik dalam mengikuti pola naik data aktual secara keseluruhan. Pada bulan Januari 2023 prediksi model sedikit lebih rendah dibandingkan data aktual. Begitu pula pada bulan Maret 2023 hingga Juli 2023. Namun, dapat dilihat pada bulan Februari 2023 menunjukkan prediksi model yang mendekati nilai data aktual. Secara grafis, model GA-SVR memberikan hasil peramalan yang cukup baik walaupun

terdapat beberapa perbedaan nilai prediksi dengan nilai aktual. Pada GA-SVR ini analisis yang dihasilkan menunjukkan kemampuan dalam optimasi pola data yang lebih baik dengan hasil eror yang lebih rendah dibandingkan model SVR.

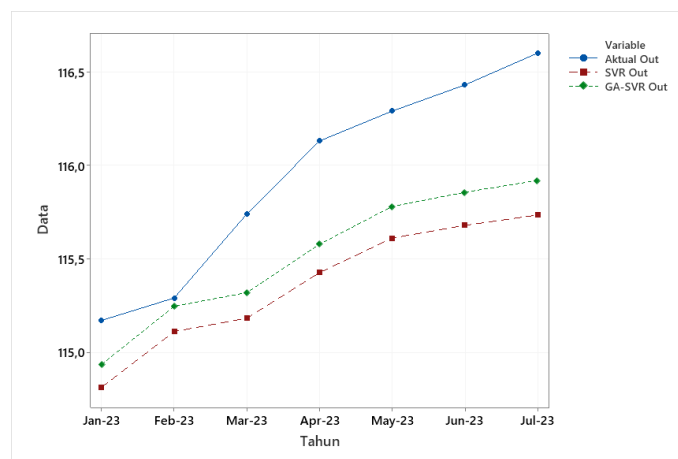
4.5 Forecasting IHK Kota Surabaya Menggunakan Model Terbaik

Pemodelan menggunakan SVR dan GA-SVR menghasilkan perbandingan nilai RMSE dan sMAPE pada masing-masing metode. Hal ini digunakan untuk mendapatkan model paling baik dalam meramalkan IHK Kota Surabaya selama enam bulan mendatang. Tabel 4.8 menunjukkan perbandingan performa dari model SVR dan GA-SVR

Tabel 4.8 Pemilihan Model Terbaik *Forecasting* IHK Kota Surabaya

Metode	Model	RMSE in sample	RMSE out sample	sMAPE out sample
SVR	$f(y_{t-1}, y_{t-13})$	0,3161861	0,626140	0,5054331
GA-SVR	$f(y_{t-1}, y_{t-13})$	0,3120703	0,4770836	0,3720165

Berdasarkan Tabel 4.8 menunjukkan bahwa model terbaik dalam meramalkan IHK Kota Surabaya adalah model dengan menggunakan metode GA-SVR. Model ini memiliki ketepatan akurasi lebih tinggi dibandingkan dengan SVR tanpa optimasi. Optimasi GA yang telah dilakukan pada parameter SVR menghasilkan nilai parameter yang lebih optimum ditandai dengan nilai sMAPE yang lebih rendah dibandingkan model SVR. Dari penelitian ini dapat dinyatakan bahwa penggunaan GA yang digunakan untuk optimasi parameter SVR dapat menghasilkan akurasi hasil ramalan yang lebih baik daripada SVR sebelum dioptimasi. Model juga tetap memiliki performansi yang baik meskipun terdapat kejadian pandemi *Covid-19*. Perbandingan secara visual pada data *out sample* terdapat pada Gambar 4.13.



Gambar 4.13 Perbandingan Visual Data *Out Sample* IHK Kota Surabaya

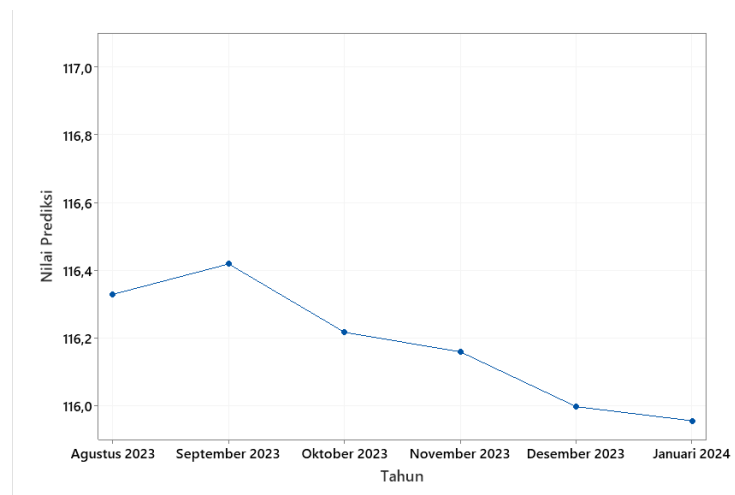
Berdasarkan Gambar 4.13 dapat diketahui bahwa perolehan hasil prediksi GA-SVR pada *input* $f(y_{t-1}, y_{t-13})$ dapat menunjukkan data aktual dengan baik. Dimana prediksi GA-SVR mendekati nilai data aktual dibandingkan model SVR. Adanya keadaan tren menurun pada model GA-SVR dengan perbedaan nilai prediksi yang lebih rendah pada bulan Januari 2023, Maret 2023 hingga Juli 2023. Namun, pada bulan Februari 2023 nilai prediksi lebih mendekati model GA-SVR dibandingkan SVR. Dengan demikian, hasil visualisasi lebih cenderung mendekati pada model GA-SVR dengan tingkat *error* atau kesalahan prediksi yang lebih rendah sehingga dapat dipilih karena keunggulan tersebut. Berdasarkan hasil analisis yang telah dilakukan Model terbaik GA-SVR dengan *input* $f(y_{t-1}, y_{t-13})$ dapat digunakan dalam

meramalkan IHK Kota Surabaya enam bulan kedepan mulai Agustus 2023 hingga Januari 2024. Tabel 4.9 merupakan hasil ramalan IHK Kota Surabaya enam bulan mendatang.

Tabel 4.9 Hasil *Forecast* IHK Kota Surabaya 6 Bulan

Bulan	IHK
Agustus 2023	116,3284
September 2023	116,4181
Oktober 2023	116,2165
November 2023	116,1587
Desember 2023	115,997
Januari 2024	115,9553

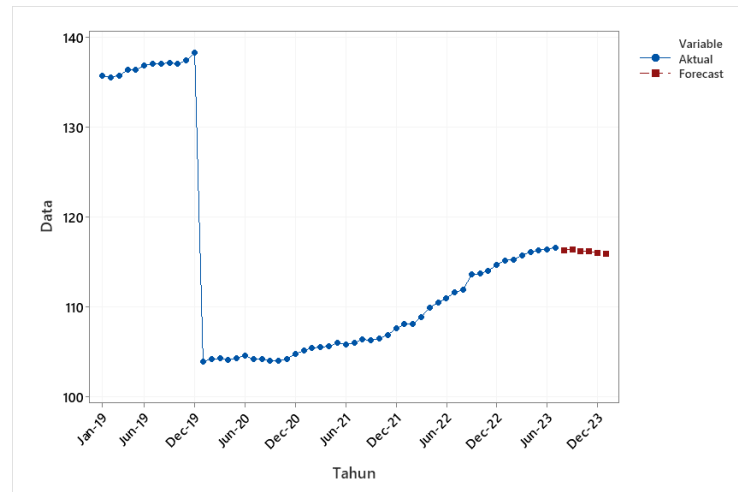
Tabel 4.9 menunjukkan bahwa hasil ramalan data IHK mulai bulan Agustus 2023 hingga enam bulan kedepan, yaitu sampai dengan Januari 2024 cenderung stabil pada kisaran nilai 116. Terjadi kenaikan pada nilai 116 dari bulan Agustus 2023 ke bulan September. Kemudian mulai bulan Oktober hingga Januari 2023 mengalami penurunan. Berikut merupakan visualisasi lebih jelas terkait adanya kenaikan ataupun penurunan IHK Kota Surabaya pada Gambar 4.14.



Gambar 4.14 Hasil *Forecasting* IHK 6 Bulan Mendatang

Berdasarkan Gambar 4.14 menunjukkan bahwa hasil *forecasting* IHK sedikit kenaikan pada bulan Agustus 2023 hingga September 2023, yaitu mencapai 116,41. Pada bulan September 2023, IHK mengalami sedikit kenaikan sebesar 0,007% dari bulan sebelumnya. Hal ini menunjukkan adanya sedikit kenaikan pada harga barang dan jasa, kemungkinan karena peningkatan permintaan atau faktor musiman. Namun, mulai bulan Oktober 2023 hingga Januari 2024, IHK menunjukkan tren penurunan yang konsisten. Penurunan tersebut dimulai dengan 0,173% dari bulan September hingga Oktober 2023, dilanjutkan penurunan 0,050% dari bulan Oktober ke November 2023, sebesar 0,139 dari bulan November hingga Desember 2023, dan 0,036% pada Desember 2023 menuju Januari 2024. Pada tren penurunan ini mencerminkan adanya deflasi ringan di Kota Surabaya, yang dapat disebabkan oleh penurunan permintaan konsumen, peningkatan pasokan barang dan jasa maupun faktor ekonomi lainnya. Meskipun penurunan bulanan ini relatif kecil, konsistensi penurunan selama beberapa bulan dapat menjadi tanda yang perlu diwaspadai bagi pemerintah Kota Surabaya. Deflasi berkelanjutan dapat mengindikasikan adanya masalah dalam perekonomian, seperti lemahnya permintaan domestik, yang berdampak negatif pada pertumbuhan ekonomi sehingga perlu untuk terus memantau tren guna memahami dampak ekonomi lokal dan tindakan berkelanjutan apabila diperlukan jika tren deflasi berlanjut berdasarkan hasil pada peramalan jangka waktu enam bulan.

Gambar 4.15 menunjukkan *time series plot* dari data aktual dan ramalan IHK Kota Surabaya jangka waktu Januari 2019 hingga Januari 2024. Garis biru mewakili data aktual dari variabel tersebut, yang menunjukkan tren yang terus meningkat selama beberapa bulan sebelumnya. Poin datanya berfluktuasi, namun secara keseluruhan trennya meningkat, yang menunjukkan kenaikan bertahap pada IHK di Surabaya. Garis merah mewakili hasil ramalan IHK Kota Surabaya, yang juga menunjukkan tren stabil nan menurun, mendekati dengan data aktual. Hal ini menunjukkan bahwa model perkiraan secara efektif menangkap dinamika dan pola yang mendasari data sehingga memberikan proyeksi nilai IHK di masa depan yang dapat diandalkan.



Gambar 4.15 Forecasting Plot Model terbaik

Secara grafis, analisis komprehensif mengenai tren IHK di Surabaya, baik dari segi data aktual historis maupun data ramalan enam bulan kedepan. Representasi visual tersebut memungkinkan identifikasi titik balik utama, seperti peningkatan tajam IHK pada pertengahan tahun 2020, serta tren kenaikan yang terus berlanjut pada tahun-tahun berikutnya. Dengan membandingkan garis data aktual dengan hasil ramalan dapat menilai keakuratan model peramalan dan mengidentifikasi potensi penyimpangan atau ketidaksesuaian antara nilai IHK yang diprediksi dan yang diamati. Dalam hal ini bermanfaat bagi para pengambil kebijakan, ekonom, dan pemangku kepentingan lainnya di wilayah Surabaya untuk mengambil keputusan yang tepat dan menerapkan langkah-langkah yang tepat untuk mengatasi perubahan indeks harga konsumen. Secara keseluruhan, *time series plot* memberikan visualisasi tren IHK di Surabaya yang komprehensif dan mendalam, sehingga memungkinkan pemahaman yang lebih mendalam mengenai dinamika perekonomian dan efektivitas model peramalan yang digunakan.

(Halaman ini sengaja dikosongkan)

BAB 5 KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisis peramalan IHK Kota Surabaya yang telah dilakukan, dapat ditarik kesimpulan dengan menjawab tujuan penelitian sebagai berikut.

1. Data IHK Kota Surabaya dari Januari 2019 hingga Juli 2023 stabil pada awal periode dimana dengan rentang nilai IHK sebesar 137 hingga 138. Pada sekitar Oktober 2019 hingga Maret 2020, terjadi penurunan tajam dalam IHK dari rentang nilai sekitar 137 hingga menuju 105. Hal ini memungkinkan terjadinya perubahan besar dalam ekonomi maupun kebijakan yang menyebabkan penurunan harga secara signifikan yang disebabkan oleh pengaruh *Covid-19*. Periode stabil setelah terjadi penurunan tajam, dari Maret 2020 hingga sekitar September 2021m IHK tetap stabil di sekitar nilai 105. Pada Oktober 2021 hingga Juli 2023, IHK menunjukkan tren kenaikan bertahap dari sekitar 105 menuju 114. Ini menunjukkan adanya kenaikan harga yang mungkin disebabkan oleh faktor inflasi maupun pemulihan ekonomi pasca terjadinya *Covid-19*.
2. Model terbaik untuk meramalkan IHK Kota Surabaya adalah menggunakan pemodelan GA-SVR *input* $f(y_{t-1}, y_{t-13})$. Penggunaan optimasi GA pada SVR menghasilkan model terbaik dengan nilai RMSE dan SMAPE data *in sample* maupun *out sample* paling kecil dibandingkan sebelum dioptimasi. Peramalan yang dihasilkan cukup akurat sehingga model terbaik ada pada GA-SVR dengan parameter terbaik $\varepsilon = 0,06756186$; $\gamma = 0,02830025$; dan $C = 29,84129$. Hasil analisis visual dari model *input* $f(y_{t-1}, y_{t-13})$ menunjukkan performa model yang baik dimana kesesuaian pola antara hasil ramalan dengan data aktual. Walaupun masih terdapat beberapa yang belum akurat sepenuhnya yang terlihat dari gambar pola data *in sample*. Pada data *out sample* cukup baik dalam mengikuti pola naik data aktual secara keseluruhan. Pada bulan Januari 2023 prediksi model sedikit lebih rendah dibandingkan data aktual. Begitu pula pada bulan Maret 2023 hingga Juli 2023. Namun, dapat dilihat pada bulan Februari 2023 menunjukkan prediksi model yang mendekati nilai data aktual.
3. Berdasarkan hasil peramalan diketahui bahwa IHK Kota Surabaya menunjukkan model perkiraan secara efektif menangkap dinamika dan pola yang mendasari data sehingga memberikan proyeksi nilai IHK di masa depan yang dapat diandalkan. Prakiraan tersebut memperkirakan IHK akan sedikit meningkat dalam beberapa bulan mendatang, yang diikuti dengan penurunan secara bertahap. Tren keseluruhan menunjukkan adanya fluktuasi nilai IHK, yang awalnya mengalami penurunan selama jangka waktu enam bulan. Secara grafis, analisis komprehensif mengenai tren IHK di Surabaya, baik dari segi data aktual historis maupun data ramalan enam bulan kedepan. Dengan membandingkan garis data aktual dengan hasil ramalan dapat menilai keakuratan model peramalan dan mengidentifikasi potensi penyimpangan atau ketidaksesuaian antara nilai IHK yang diprediksi dan yang diamati. Dalam hal ini bermanfaat bagi para pengambil kebijakan ekonomi, dan pemangku kepentingan lainnya di wilayah Surabaya untuk mengambil keputusan yang tepat dan menerapkan langkah-langkah yang tepat untuk mengatasi perubahan indeks harga konsumen. Secara keseluruhan, *time series plot* memberikan visualisasi tren IHK di Surabaya yang komprehensif dan mendalam, sehingga memungkinkan pemahaman yang lebih mendalam mengenai dinamika perekonomian dan efektivitas model peramalan yang digunakan.

5.2 Saran

Penelitian ini diharapkan menjadi lebih baik kedepannya sehingga peneliti merekomendasikan beberapa hal sebagai berikut.

1. Untuk penelitian selanjutnya disarankan menggunakan tambahan variabel-variabel lain yang sekiranya memiliki pengaruh signifikan terhadap IHK sehingga tidak hanya terbatas pada variabel IHK sebagai data utama yang digunakan. Contoh faktor-faktor variabel lain yang mempengaruhi IHK diantaranya adanya kenaikan pada nilai tukar dollar dan rupiah maupun variabel lain yang berkaitan dengan inflasi. Pada optimasi parameter disarankan untuk mencoba jenis optimasi lainnya selain GA untuk mendapatkan model yang lebih baik lagi.
2. Untuk BPS Kota Surabaya dapat menjadikan penelitian ini sebagai informasi mengenai kenaikan maupun penurunan IHK di masa mendatang. Apabila IHK mengalami kenaikan maka dapat dijadikan sebagai dasar awal dalam mengatasi inflasi kota setempat dengan mengatasi stabilitas seperti harga bahan pokok.

DAFTAR PUSTAKA

- Alwee, R., Shamsuddin, S. M., & Sallehuddin, R. (2013). Hybrid Support Vector Regression and Autoregressive Integrated Moving Average Models Improved by Particle Swarm Optimization for Property Crime Rates Forecasting with Economic Indicators. *The Scientific World Journal*, 1-11.
- Ayudhiah, M. P., Bahri, S., & Fitriyani, d. N. (2020). Peramalan Indeks Harga Konsumen Kota Mataram Menggunakan VARIMA. *Eig. Math*, 1, 1. doi:10.29303/emj.vli2.61
- Basak, Debasish, Pal, S., & Patranabis, a. D. (2007). Support Vector Regression. *Neural Information Processing-Letters and Reviews* , 203-224.
- BPS. (2022). *Badan Pusat Statistik*. Retrieved Desember 29, 2023, from www.bps.go.id
- BPS Kota Surabaya. (2023). Retrieved Januari 4, 2024, from www.surabayakota.bps.go.id
- Budiastuti, R. I. (2017). *Prediksi IHK Harian Menggunakan Support Vector Regression Berbasis Cloud Computing*. Institut Teknologi Sepuluh Nopember, Teknik Elektro. Surabaya: Repository ITS.
- Damanik, M. S., & Sinaga, S. M. (2014). Publikasi Analisis Indeks Harga Konsumen Kota Pematangsiantara.
- Dimashanti, A. R., & Sugiman. (2021). Peramalan IHK Kota Semarang Menggunakan SARIMA Berbantuan Software Minitab. *Prisma*, 4, 565-576.
- Drucker, Harris, Burges, C. J., & Linda Kaufman, A. S. (1996). *Support Vector Regression Machine*. West Long Branch: Bells Labs.
- Gen, M., & Cheng, R. (1997). *Genetic Algorithms and Engineering Design*. New York: John Wiley & Sons, Inc.
- Gorunescu, F. (2011). *Data Mining Concepts, Models and Techniques*. Verlag Berlin Heidelberg: Springer.
- Guo, Y. (2018). An Adaptive SVR for High-Frequency Stock Price Forecasting. 11397-11404.
- Hardle, W. W., Prasetyo, D. D., & Hefner, C. (2014). *Support Vector Machine with Evolutionary Model Selection for Default Prediction*. Oxford University Press.
- Hidayatullah, M. M., Cholissodin, I., & Perdana, r. S. (2018). Peramalan Kenaikan IHK/Inflasi Kota Malang menggunakan Metode Support Vector Regression dengan Chaotic Genetic Algorithm-Simulated Annealing (CGASA). *Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Hong, W. (2011). Forecasting Urban Traffic flow by SVR with Continuous ACO. *Applied Math*, 1282-1291.
- Hsu, c. W., Chang, C. C., & Lin, a. C. (2016). *A Practical Guide to Support Vector Classification*. Taipei: Department of Computer Science.

- Ismail, Z., & Irhamah. (2008). Solving the Vehicle Routing Problem with Stochastic Demands via Hybrid Genetic Algorithm-Tabu Search. *Journal of Mathematics and Statistics*, 161-167.
- Mankiw, N., Quah, E., & Wilson, P. (2012). *Pengantar Ekonomi Mikro Principles of Economics, An Asian Edition-Volume 1*. Salemba Empat.
- Meivita, M. P. (2014). Indeks Harga Konsumen 7 Kota di Jawa Timur Tahun 2014.
- Mukron, M. H., Susianti, I., Azzahra, F., Kumala, Y. N., Widiyana, F. r., & Haris, M. I. (2021). Peramalan IHK Indonesia Menggunakan ARIMA. *J.Stat. Ind. dan Komputasi*, 6, 20-25.
- Prasetyo, E. (2014). *Data Mining Konsep dan Aplikasi Menggunakan MATLAB*. Yogyakarta: Graha Ilmu.
- Rosdianawati, R., & Surjanto, S. D. (2023). Peramalan Inflasi Kota Kediri Berdasarkan Indeks Harga Konsumen Menggunakan Metode Exponential Smoothing. *Sains dan Seni ITS*.
- S. Yadav, a. S. (2016). Analysis of K-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification.
- Santosa, B. (2007). *Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis*. Yogyakarta: Graha Ilmu.
- Saputra, M. J., Setiawan, A., & Mahatma, T. (2012). Analisis Kointegrasi Data Runtun Waktu Indeks Harga Konsumen Beberapa Komoditas Barang Kota di Jawa Tengah. *Prosiding Seminar Nasional Penelitian, Pendidikan dan Penerapan MIPA*, 177.
- Schaffer, J. D., Caruana, R. A., Eshelman, L. J., & Das, R. (1989). *A Study of Control Parameters Affecting Online Performance of Genetic Algorithms For Function Optimization*. Proceedings of the Third International Conference on Genetic Algorithms.
- Setiawan, K. (2003). *Paradigma Sistem Cerdas: Jaringan Saraf Tiruan, Logika Fazi, Algoritma Genetik*. Malang: Banyumedia.
- Sharp, T. (den 4 March 2020). An Introduction to Support Vector Regression. *Towards Data Science*. Hämtat från <https://towardsdatascience.com/an-introduction-to-support-vector-regression-svr-a3ebc1672c2>
- Sivanandam, S., & Deepa, S. (2008). Introduction to Genetic Algorithm. *Verlag Berlin Heidelberg: Springer*.
- Smola, A. J., & Scholkopf, B. (2004). *A Tutorial on Support Vector Regression* (Vol. 14(3)). Statistics and Computing.
- Suarsih, S., & Achsani, N. A. (2016). Dampak Perubahan Nilai Tukar terhadap Indeks Harga Konsumen Bahan Makanan di Indonesia. *Jurnal Ekonomi dan Pembangunan Indonesia*, 1-14.
- Suprayogi, D., & Pardede, H. F. (2022). Support Vector Regression dalam Prediksi Penurunan Jumlah Kasus Penderita Covid-19 . *Journal of Information Technology and Computer Science* , 63-70.

- Terasvirta, T. L. (1993). *Power of The Neural Network Linierity Test* . USA: Blackwell Publishers.
- Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. New York: Springer.
- Windarto, & Wijaya. (2012). Aplikasi Penyusunan Jadwal dengan Algoritma Genetik pada Sekolah Menengah Kejuruan Budi Mulia Tangerang. *Prosiding Seminar Nasional Multidisiplin Ilmu*.
- Yuliani, W., Irhamah, & Prastyo, D. (2014). *Penaksiran Parameter Model ARIMA dengan Menggunakan Genetic Algorithm*. Surabaya: Institut Teknologi Sepuluh Nopember: Tugas Akhir.
- Zhang, D. L. (2011). Parameter Optimization for Support Vector Regression Based on Genetic Algorithm with Simplex Crossover Operator. *Journal of Information & Computational Science*, 911-920.
- Zhao, W., Tao, T., & Zio, E. (2015). System Reliability Prediction by Support Vector Regression with Analytic Selection and Genetic Algorithm Parameters Selection. *Applied Soft Computing*.

(Halaman ini sengaja dikosongkan)

LAMPIRAN

Lampiran 1. Data IHK Kota Surabaya

Yt	yt-1	yt-13	Yt	yt-1	yt-13
104,25	103,92	135,69	106,9	106,48	104,03
104,26	104,25	135,52	107,6	106,9	104,24
104,09	104,26	135,73	108,08	107,6	104,76
104,31	104,09	136,33	108,12	108,08	105,15
104,6	104,31	136,33	108,88	108,12	105,45
104,17	104,6	136,86	109,93	108,88	105,54
104,24	104,17	137,01	110,47	109,93	105,64
104,05	104,24	137,01	110,98	110,47	105,99
104,03	104,05	137,13	111,62	110,98	105,81
104,24	104,03	137,02	111,91	111,62	106,02
104,76	104,24	137,4	113,61	111,91	106,41
105,15	104,76	138,23	113,69	113,61	106,27
105,45	105,15	103,92	114,03	113,69	106,48
105,54	105,45	104,25	114,69	114,03	106,9
105,64	105,54	104,26	115,17	114,69	107,6
105,99	105,64	104,09	115,29	115,17	108,08
105,81	105,99	104,31	115,74	115,29	108,12
106,02	105,81	104,6	116,13	115,74	108,88
106,41	106,02	104,17	116,29	116,13	109,93
106,27	106,41	104,24	116,43	116,29	110,47
106,48	106,27	104,05	116,6	116,43	110,98

Lampiran 2. Syntax Uji Nonlinearitas Terasvirta

```
library(tseries)
library(readxl)
library(dplyr)
library(ggplot2)
library(lubridate)

data_ujinon = read_excel("D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/data_ihk.xlsx",sheet="Sheet4")
head(data_ujinon)

#Uji Terasvita
y = as.ts(data_ujinon$Yt);y
terasvirta.test(y,1,c("F"))
```

Lampiran 3. Output Uji Nonlinearitas Terasvirta

```
> #Uji Terasvirta
> y = as.ts(data_ujion$Yt);y
Time Series:
Start = 1
End = 42
Frequency = 1
 [1] 104.25 104.26 104.09 104.31 104.60 104.17 104.24 104.05 104.03 1
04.24 104.76 105.15
 [13] 105.45 105.54 105.64 105.99 105.81 106.02 106.41 106.27 106.48 1
06.90 107.60 108.08
 [25] 108.12 108.88 109.93 110.47 110.98 111.62 111.91 113.61 113.69 1
14.03 114.69 115.17
 [37] 115.29 115.74 116.13 116.29 116.43 116.60
> terasvirta.test(y,1,c("F"))

      Teraesvirta Neural Network Test

data: y
F = 7.5306, df1 = 2, df2 = 39, p-value = 0.001716
```

Lampiran 4. Syntax SVR

```
library(e1071)
library(tseries)
library(caret)
library(lattice)
library(kernlab)
library(readxl)
library(Metrics)

#Input Data
svr = read_xlsx("D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/data_ihk.xlsx",sheet="Sheet4")
svr
sum(is.na(svr))
as.ts(svr$Yt)

#Pembagian Data
train = svr[1:35,];train
train = data.frame(train);train

test = svr[36:42,];test
test = data.frame(test);test

#####
#####

# Pastikan data `train` memiliki cukup observasi
n_obs <- nrow(train)

# Tentukan parameter window yang sesuai
initial_window <- min(38, n_obs - 6) # Misalnya, gunakan minimum antara 38 dan
panjang data - horizon
```

```

horizon <- 6

# Cek jika pengaturan jendela valid
if (initial_window + horizon > n_obs) {
  stop("Initial window and horizon settings exceed the number of available
observations.")
}

# Atur train control dengan timeslice
tr_c1 <- trainControl(
  method = "timeslice",
  initialWindow = initial_window,
  horizon = horizon,
  fixedWindow = FALSE,
  skip = 0 # Jika ingin memindahkan jendela tanpa melewati data
)

# Grid search untuk parameter C
grid_linear <- expand.grid(C = 2^(-5:5))

# Latih model SVR Linear
model1 <- train(
  Yt ~ .,
  data = train,
  method = "svmLinear",
  trControl = tr_c1,
  tuneGrid = grid_linear,
  na.action = na.omit
)

# Tampilkan model
print(model1)

#####
#####

#Grid search hyperparameter
grid_poly <- expand.grid(C = 2^(-5:5),      # Cost parameter
  degree = 1:3,      # Degree of the polynomial
  scale = 2^(0:5))  # Scaling factor)

#Model SVR Polynomial
# Atur train control dengan timeslice
tr_c1 <- trainControl(
  method = "timeslice",
  initialWindow = initial_window,
  horizon = horizon,
  fixedWindow = FALSE,
  skip = 0 # Jika ingin memindahkan jendela tanpa melewati data
)

```

```

model2 = train(Yt~., data = train, method = "svmPoly",trControl=tr_c1,
              tuneGrid = grid_poly,na.action = na.omit); model2
model2$finalModel

#####
#####

#Grid search hyperparameter
grid_rbf <- expand.grid(C = 2^(-5:5),
                       sigma = 2^(-5:5))

#Model SVR RBF
# Atur train control dengan timeslice
tr_c1 <- trainControl(
  method = "timeslice",
  initialWindow = initial_window,
  horizon = horizon,
  fixedWindow = FALSE,
  skip = 0 # Jika ingin memindahkan jendela tanpa melewati data
)

model3 = train(Yt~., data = train, method = "svmRadial",trControl=tr_c1,
              tuneGrid = grid_rbf,na.action = na.omit); model3
model3$finalModel

#####
#####

#SVR (to be changed)
tuneResult = tune(svm, Yt ~.,data=train,kernel="linear",ranges=list(epsilon=0.1,
cost=2^-1))
tunedmodel = tuneResult$best.model;tunedmodel
forecast.in = predict(tunedmodel,train)
forecast.out = predict(tunedmodel,test)

forecast1 = (forecast.in)
forecast2 = (forecast.out)
testing = ((test[,1]))
training = ((train[,1]))
isi1 = testing-forecast2
isi2 = training-forecast1

rmse.in = sqrt(mean(isi2^2)) ;rmse.in
rmse.out = sqrt(mean(isi1^2)) ;rmse.out
smape.out = (sum(abs(testing-forecast2)/((abs(testing)+abs(forecast2))/2))
            /length(testing))*100; smape.out

write.table(forecast1)
write.table(forecast2)

```

```

write.table(forecast1,"D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/svr in fix 21.txt")
write.table(forecast2,"D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/svr out fix 21.txt")

#####
#####

tuneResult = tune(svm, Yt ~.,data=train,kernel="polynomial",ranges=list(epsilon=0.1,
cost=2^5, degree=2, scale=2^4))
tunedmodel = tuneResult$best.model;tunedmodel
forecast.in = predict(tunedmodel,train)
forecast.out = predict(tunedmodel,test)

forecast1 = (forecast.in)
forecast2 = (forecast.out)
testing = ((test[,1]))
training = ((train[,1]))
isi1 = testing-forecast2
isi2 = training-forecast1

rmse.in = sqrt(mean(isi2^2)) ;rmse.in
rmse.out = sqrt(mean(isi1^2)) ;rmse.out
smape.out = (sum(abs(testing-forecast2)/((abs(testing)+abs(forecast2))/2))
/length(testing))*100; smape.out

write.table(forecast1)
write.table(forecast2)

write.table(forecast1,"D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/svr in fix 22.txt")
write.table(forecast2,"D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/svr out fix 22.txt")

#####
#####

tuneResult = tune(svm, Yt ~.,data=train,kernel="radial",ranges=list(epsilon = 0.1, cost
=2^5, gamma =2^-5))
tunedmodel = tuneResult$best.model;tunedmodel
forecast.in = predict(tunedmodel,train)
forecast.out = predict(tunedmodel,test)

forecast1 = (forecast.in)
forecast2 = (forecast.out)
testing = ((test[,1]))
training = ((train[,1]))
isi1 = testing-forecast2

```

```

isi2 = training-forecast1

rmse.in = sqrt(mean(isi2^2)) ;rmse.in
rmse.out = sqrt(mean(isi1^2)) ;rmse.out
smape.out = (sum(abs(testing-forecast2)/((abs(testing)+abs(forecast2))/2))
            /length(testing))*100; smape.out

write.table(forecast1)
write.table(forecast2)

write.table(forecast1,"D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/svr in fix 23.txt")
write.table(forecast2,"D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/svr out fix 23.txt")

```

Lampiran 5. Output SVR

```

> # Tampilkan model
> print(model1)
Support Vector Machines with Linear Kernel

35 samples
 2 predictor

No pre-processing
Resampling: Rolling Forecasting Origin Resampling (6 held-out with no fixed window)
Summary of sample sizes: 29
Resampling results across tuning parameters:

   C          RMSE         Rsquared         MAE
0.03125  3.7248115  0.7980944  3.6687349
0.06250  1.8940256  0.8016647  1.8255317
0.12500  0.6293221  0.8035680  0.4117651
0.25000  0.5862847  0.8037007  0.3857903
0.50000  0.5527760  0.8043772  0.3992952
1.00000  0.6770471  0.8046174  0.6174549
2.00000  0.6775000  0.8046175  0.6179876
4.00000  0.6786428  0.8046178  0.6193287
8.00000  0.6771847  0.8046199  0.6175992
16.00000 0.6773829  0.8046167  0.6178551
32.00000 0.6772933  0.8046181  0.6177398

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was C = 0.5.

```

```

> model2 = train(Yt~., data = train, method = "svmPoly",trControl=tr_c1,
+ tuneGrid = grid_poly,na.action = na.omit); model2
Support Vector Machines with Polynomial Kernel

35 samples
 2 predictor

No pre-processing
Resampling: Rolling Forecasting Origin Resampling (6 held-out with no fixed window)
Summary of sample sizes: 29
Resampling results across tuning parameters:

```

C	degree	scale	RMSE	Rsquared	MAE
0.03125	1	1	3.7248115	0.7980944	3.6687348
0.03125	1	2	1.8940256	0.8016647	1.8255318
0.03125	1	4	0.6293221	0.8035680	0.4117651
0.03125	1	8	0.5862847	0.8037007	0.3857903
0.03125	1	16	0.5527760	0.8043772	0.3992952
0.03125	1	32	0.6770473	0.8046174	0.6174551
0.03125	2	1	2.0912332	0.7916241	1.6885055
0.03125	2	2	1.4955213	0.7885222	1.2792654
0.03125	2	4	1.1564303	0.7868927	1.0261239
0.03125	2	8	0.8437329	0.7837606	0.7728001
0.03125	2	16	0.5338889	0.7838504	0.4373836
0.03125	2	32	0.5455299	0.7770220	0.4522352
0.03125	3	1	0.6784156	0.7242575	0.6026903
0.03125	3	2	5.9294039	0.7833107	4.8523861
0.03125	3	4	6.1247501	0.7871457	5.0121935
0.03125	3	8	6.3749974	0.7840692	5.2195620
0.03125	3	16	6.5680794	0.7665825	5.4060866
0.03125	3	32	6.7554718	0.7456125	5.5801340

0.06250	1	1	1.8940257	0.8016647	1.8255318
0.06250	1	2	0.6293221	0.8035680	0.4117651
0.06250	1	4	0.5862846	0.8037007	0.3857903
0.06250	1	8	0.5527760	0.8043772	0.3992952
0.06250	1	16	0.6770473	0.8046174	0.6174551
0.06250	1	32	0.6775001	0.8046175	0.6179876
0.06250	2	1	1.6257558	0.7906935	1.3728047
0.06250	2	2	1.2743298	0.7885825	1.1152925
0.06250	2	4	0.8559525	0.7837894	0.7835310
0.06250	2	8	0.5349493	0.7834605	0.4384732
0.06250	2	16	0.5449983	0.7769264	0.4517587
0.06250	2	32	0.5358045	0.7800145	0.4414587
0.06250	3	1	0.8184904	0.6257386	0.6432320
0.06250	3	2	6.0405058	0.7850267	4.9476002
0.06250	3	4	6.2017893	0.7837019	5.0809352
0.06250	3	8	5.6022388	0.7804524	4.6155489
0.06250	3	16	6.7374328	0.7487433	5.5650062
0.06250	3	32	6.8533476	0.7433942	5.6600475
0.12500	1	1	0.6293221	0.8035680	0.4117651
0.12500	1	2	0.5862846	0.8037007	0.3857903
0.12500	1	4	0.5527104	0.8043845	0.3994299
0.12500	1	8	0.6770475	0.8046174	0.6174553
0.12500	1	16	0.6775001	0.8046175	0.6179876
0.12500	1	32	0.6786427	0.8046178	0.6193287
0.12500	2	1	1.4076636	0.7908794	1.2131578
0.12500	2	2	0.9420958	0.7863336	0.8562508
0.12500	2	4	0.5360543	0.7826200	0.4398722
0.12500	2	8	0.5366629	0.7789257	0.4424470
0.12500	2	16	0.5344471	0.7805476	0.4398736
0.12500	2	32	0.5341920	0.7806332	0.4395939

0.12500	3	1	2.2189951	0.4133576	1.7300299
0.12500	3	2	6.0442255	0.7833626	4.9567959
0.12500	3	4	6.0998982	0.7799264	5.0097094
0.12500	3	8	6.0372414	0.7707246	4.9771053
0.12500	3	16	6.7754578	0.7488730	5.5971671
0.12500	3	32	6.0819293	0.7533827	4.9744221
0.25000	1	1	0.5862847	0.8037007	0.3857903
0.25000	1	2	0.5527104	0.8043845	0.3994299
0.25000	1	4	0.6770476	0.8046174	0.6174554
0.25000	1	8	0.6781559	0.8046229	0.6187214
0.25000	1	16	0.6786427	0.8046178	0.6193286
0.25000	1	32	0.6771776	0.8046200	0.6175900
0.25000	2	1	1.0878212	0.7901839	0.9719016
0.25000	2	2	0.6354064	0.7855126	0.5658816
0.25000	2	4	0.5352759	0.7823526	0.4395650
0.25000	2	8	0.5344867	0.7805675	0.4399512
0.25000	2	16	0.5344442	0.7805489	0.4398751
0.25000	2	32	0.5331666	0.7808125	0.4381961
0.25000	3	1	4.2947822	0.7654385	3.5112757
0.25000	3	2	6.2463547	0.7803211	5.1252200
0.25000	3	4	6.2836253	0.7758059	5.1640749
0.25000	3	8	6.7841384	0.7534346	5.5980259
0.25000	3	16	6.5784153	0.7541128	5.4356881
0.25000	3	32	6.3144403	0.7619023	5.2166839
0.50000	1	1	0.5527104	0.8043845	0.3994298
0.50000	1	2	0.6770473	0.8046174	0.6174551
0.50000	1	4	0.6781559	0.8046229	0.6187214
0.50000	1	8	0.6786428	0.8046178	0.6193287
0.50000	1	16	0.6771782	0.8046200	0.6175908
0.50000	1	32	0.6773831	0.8046167	0.6178553

```

8.00000 1 1 0.6782650 0.8046236 0.6188445
8.00000 1 2 0.6777817 0.8046204 0.6182985
8.00000 1 4 0.6777274 0.8046201 0.6182365
8.00000 1 8 0.6782232 0.8046234 0.6187961
8.00000 1 16 0.6781126 0.8046228 0.6186701
8.00000 1 32 0.6760704 0.8046172 0.6163013
8.00000 2 1 0.5302452 0.7823075 0.4338361
8.00000 2 2 0.5329269 0.7809250 0.4378340
8.00000 2 4 0.5330168 0.7809643 0.4380042
8.00000 2 8 0.5334267 0.7809578 0.4387470
8.00000 2 16 0.5314263 0.7811530 0.4351004
8.00000 2 32 0.5357417 0.7794466 0.4416480
8.00000 3 1 7.3589314 0.7607978 6.0500123
8.00000 3 2 7.9225719 0.7502913 6.5166243
8.00000 3 4 8.5854978 0.7430192 7.0564473
8.00000 3 8 8.4796330 0.7442033 6.9710757
8.00000 3 16 9.1213892 0.7233052 7.4432730
8.00000 3 32 11.9808907 0.8110222 10.4577986
16.00000 1 1 0.6777865 0.8046204 0.6183040
16.00000 1 2 0.6782146 0.8046233 0.6187869
16.00000 1 4 0.6777790 0.8046205 0.6182938
16.00000 1 8 0.6777265 0.8046201 0.6182337
[ reached getOption("max.print") -- omitted 32 rows ]

RMSE was used to select the optimal model using the smallest value.
The final values used for the model were degree = 2, scale = 16 and C = 32.

```

```

> model2$finalModel
Support Vector Machine object of class "ksvm"

SV type: eps-svr (regression)
parameter : epsilon = 0.1 cost C = 32

Polynomial kernel function.
Hyperparameters : degree = 2 scale = 16 offset = 1

Number of Support Vectors : 14

Objective Function Value : -13.0933
Training error : 0.009627

```


Support Vector Machines with Radial Basis Function Kernel

35 samples
2 predictor

No pre-processing

Resampling: Rolling Forecasting Origin Resampling (6 held-out with no fixed window)

Summary of sample sizes: 29

Resampling results across tuning parameters:

C	sigma	RMSE	Rsquared	MAE
0.03125	0.03125	7.4581958	0.3551695491	7.3747052
0.03125	0.06250	7.4170713	0.7659445467	7.3282066
0.03125	0.12500	7.4853712	0.7977037239	7.3901718
0.03125	0.25000	7.6457065	0.8164186176	7.5503201
0.03125	0.50000	7.7369758	0.8242894939	7.6461167
0.03125	1.00000	7.7547711	0.8246036634	7.6674750
0.03125	2.00000	7.7211445	0.7905760536	7.6358005
0.03125	4.00000	7.7107983	0.6488459971	7.6270337
0.03125	8.00000	7.6737645	0.4977583931	7.5908528
0.03125	16.00000	7.6871032	0.4376642690	7.6051469
0.03125	32.00000	7.6569873	0.4319603725	7.5751973
0.06250	0.03125	7.2395983	0.4610980437	7.1540305
0.06250	0.06250	7.1995612	0.7608936463	7.1037549
0.06250	0.12500	7.2739599	0.7938854724	7.1656553
0.06250	0.25000	7.4623557	0.8133730206	7.3519764
0.06250	0.50000	7.6256856	0.8237748178	7.5228965

0.06250	1.00000	7.6621861	0.8246658677	7.5666768
0.06250	2.00000	7.5925074	0.7905767109	7.5010364
0.06250	4.00000	7.6439278	0.6488459971	7.5565812
0.06250	8.00000	7.5967661	0.4977583931	7.5114643
0.06250	16.00000	7.6341118	0.4376642690	7.5508841
0.06250	32.00000	7.6477699	0.4319603725	7.5656841
0.12500	0.03125	6.8754359	0.5500778360	6.7864469
0.12500	0.06250	6.7132042	0.7384911855	6.6058495
0.12500	0.12500	6.7853264	0.7858433171	6.6525555
0.12500	0.25000	6.9583517	0.8093480109	6.8125730
0.12500	0.50000	7.1886406	0.8230683789	7.0554133
0.12500	1.00000	7.2599755	0.8247244346	7.1429454
0.12500	2.00000	7.2352119	0.7905770769	7.1288778
0.12500	4.00000	7.2958244	0.6488459972	7.1980738
0.12500	8.00000	7.3398357	0.4977583931	7.2482348
0.12500	16.00000	7.4393257	0.4376642690	7.3524436
0.12500	32.00000	7.5290415	0.4319603725	7.4452466
0.25000	0.03125	6.1840631	0.7135234856	6.0893993
0.25000	0.06250	5.7888962	0.6807749450	5.6604273
0.25000	0.12500	5.7177399	0.7713308406	5.5306281
0.25000	0.25000	6.1054094	0.8020871501	5.8892983
0.25000	0.50000	6.4944254	0.8199006723	6.2974797
0.25000	1.00000	6.6800039	0.8258618660	6.5158679
0.25000	2.00000	6.7272559	0.7914571812	6.5886744
0.25000	4.00000	6.7742184	0.6488518762	6.6545012
0.25000	8.00000	6.8051527	0.4977583931	6.6987953
0.25000	16.00000	7.0212523	0.4376642690	6.9259661
0.25000	32.00000	7.1976375	0.4319603725	7.1090925

```
> model3$finalModel
Support Vector Machine object of class "ksvm"

SV type: eps-svr (regression)
parameter : epsilon = 0.1 cost C = 32

Gaussian Radial Basis kernel function.
Hyperparameter : sigma = 0.03125

Number of Support Vectors : 13

Objective Function Value : -22.263
Training error : 0.008985
```

```

Call:
best.tune(METHOD = svm, train.x = Yt ~ ., data = train, ranges = list(epsilon = 0.1,
  cost = 2^-1), kernel = "linear")

Parameters:
  SVM-Type:  eps-regression
  SVM-Kernel: linear
    cost: 0.5
    gamma: 0.5
    epsilon: 0.1

Number of Support Vectors: 10

> forecast.in = predict(tunedmodel,train)
> forecast.out = predict(tunedmodel,test)
> forecast1 = (forecast.in)
> forecast2 = (forecast.out)
> testing = ((test[,1]))
> training = ((train[,1]))
> isi1 = testing-forecast2
> isi2 = training-forecast1
> rmse.in = sqrt(mean(isi2^2)) ;rmse.in
[1] 0.3553293
> rmse.out = sqrt(mean(isi1^2)) ;rmse.out
[1] 0.2040144
> smape.out = (sum(abs(testing-forecast2)/((abs(testing)+abs(forecast2))/2))
+ /length(testing))*100; smape.out
[1] 0.1461814

```

```

Call:
best.tune(METHOD = svm, train.x = Yt ~ ., data = train, ranges = list(epsilon = 0.1,
  cost = 2^5, degree = 2, scale = 2^4), kernel = "polynomial")

Parameters:
  SVM-Type:  eps-regression
  SVM-Kernel: polynomial
    cost: 32
    degree: 2
    gamma: 0.5
    coef.0: 0
    epsilon: 0.1

Number of Support Vectors: 17

> forecast.in = predict(tunedmodel,train)
> forecast.out = predict(tunedmodel,test)
> forecast1 = (forecast.in)
> forecast2 = (forecast.out)
> testing = ((test[,1]))
> training = ((train[,1]))
> isi1 = testing-forecast2
> isi2 = training-forecast1
> rmse.in = sqrt(mean(isi2^2)) ;rmse.in
[1] 0.7561149
> rmse.out = sqrt(mean(isi1^2)) ;rmse.out
[1] 0.3803943
> smape.out = (sum(abs(testing-forecast2)/((abs(testing)+abs(forecast2))/2))
+ /length(testing))*100; smape.out
[1] 0.2671095

```

```

Call:
best.tune(METHOD = svm, train.x = Yt ~ ., data = train, ranges = list(epsilon = 0.1,
  cost = 2^5, gamma = 2^-5), kernel = "radial")

Parameters:
  SVM-Type:  eps-regression
  SVM-Kernel: radial
    cost:    32
    gamma:   0.03125
  epsilon:   0.1

Number of Support Vectors: 13

> forecast.in = predict(tunedmodel,train)
> forecast.out = predict(tunedmodel,test)
> forecast1 = (forecast.in)
> forecast2 = (forecast.out)
> testing = ((test[,1]))
> training = ((train[,1]))
> isi1 = testing-forecast2
> isi2 = training-forecast1
> rmse.in = sqrt(mean(isi2^2)) ;rmse.in
[1] 0.3161861
> rmse.out = sqrt(mean(isi1^2)) ;rmse.out
[1] 0.62614
> smape.out = (sum(abs(testing-forecast2)/((abs(testing)+abs(forecast2))/2))
+
  /length(testing))*100; smape.out
[1] 0.5054331

```

Lampiran 6. Syntax GA-SVR

```

library(e1071)
library(GA)
library(readxl)

data <- data.frame(read_xlsx("D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS
AKHIR PUTRI/Data/data_ihk.xlsx",sheet="Sheet4"))
train_data = data[1:35,]
train_data
test_data = data[-c(1:35),]
test_data

# Fitness function (to be maximized)
# Parameter vector x is: (cost, gamma, epsilon)
fitnessFunc <- function(x) {
  # Retrieve the SVM parameters
  cost <- x[1]
  gamma <- x[2]
  epsilon <- x[3]
  # Train
  model <- svm(Yt ~ ., data = train_data, cost = cost, gamma = gamma,
    epsilon = epsilon, type = "eps-regression", kernel = "radial")
  # Test
  rmse <- mean((predict(model, newdata = train_data) - train_data$Yt)^2)
  return (-rmse)
}

# Range of the parameter values to be tested

```

```

# Parameters are: (cost, gamma, epsilon)
theta_min <- c(cost = 2^4, gamma = 2^-6, epsilon = 0.01)
# Peubah dari SVR
theta_max <- c(cost = 2^5, gamma = 2^-5, epsilon = 0.1)

# Run the genetic algorithm
results <- ga(type = "real-valued", fitness = fitnessFunc,
             names = names(theta_min),
             lower = theta_min, upper = theta_max,
             popSize = 100, maxiter = 20, pcrossover = 0.8, pmutation = 0.01,
             monitor = plot)
summary(results)

modpred = svm(Yt ~ ., data=train_data, cost=29.84129, gamma=0.02830025,
             epsilon=0.06756186)
modpred
summary(modpred)

forecast.in = predict(modpred,train_data)
forecast.out = predict(modpred,test_data)
forecast1 = (forecast.in)
forecast2 = (forecast.out)

zt = (test_data[,1])
xt = (train_data[,1])
isi1 = xt-forecast1
isi2 = zt-forecast2
rmse.in = sqrt(mean(isi1^2))
rmse.out = sqrt(mean(isi2^2))
smape.out = sum(abs(zt-forecast2)/((abs(zt)+abs(forecast2))/2))/length(zt)*100

write.table(forecast1,"D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/prediksi in fix 2.txt")
write.table(forecast2,"D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data//prediksi out fix 2.txt")

rmse.in
rmse.out
smape.out

```

Lampiran 7. Output GA

```
> summary(results)
-- Genetic Algorithm -----

GA settings:
Type                = real-valued
Population size     = 100
Number of generations = 20
Elitism             = 5
Crossover probability = 0.8
Mutation probability = 0.01
Search domain =
  cost  gamma  epsilon
lower  16 0.015625  0.01
upper  32 0.031250  0.10

GA results:
Iterations          = 20
Fitness function value = -0.09730842
Solution =
  cost  gamma  epsilon
[1,] 29.84129 0.02830025 0.06756186
```

```
> rmse.in
[1] 0.3120703
> rmse.out
[1] 0.4770836
> smape.out
[1] 0.3720165
```

Lampiran 8. Syntax Forecast Model Terbaik

```
#cost=29.84129,gamma=0.02830025 ,epsilon=0.06756186

##### 1
#####
##### AGUSTUS
#####

#FORECAST BULAN AGUSTUS 2023
#Input Data
fore= read_xlsx("D:/Putri Herliani Nabilla/52. BISMILLAH TUGAS AKHIR
PUTRI/Data/data_ihk.xlsx",sheet="Data SVR")
datafore1=fore[,1:3];datafore1
n_datafore1=length(datafore1$y);n_datafore1
foretrain1=fore[1:41,1:3] ;foretrain1
foretrain1 = data.frame(foretrain1) ;foretrain1
foretest1=fore[42,1:3] ;foretest1
foretest1 = data.frame(foretest1) ;foretest1

#Range of parameter values FIX U/ AGUSTUS 2023
#Parameters are: (cost=29.84129,gamma=0.02830025,epsilon=0.06756186)
#to be changed U/ AGUSTUS 2023
foretrain1$y <- as.numeric(as.character(foretrain1$y))
```

```

modfore1=svm(y~.,data=foretrain1,cost=29.84129,gamma=0.02830025,epsilon=0.06
756186)
forecast.in1=predict(modfore1,foretrain1)
forecast.out1=predict(modfore1,foretest1)
forecastin1=forecast.in1
forecastout1=forecast.out1
zt1=(foretest1[,1])
xt1=(foretrain1[,1])
isi11=xt1-forecastin1
isi12=zt1-forecastout1
rmse.in1=sqrt(mean(isi11^2))
rmse.out1=sqrt(mean(isi12^2))
smape.out1=(sum(abs(zt1-
forecastout1)/((abs(zt1)+abs(forecastout1))/2)))/length(zt1))*100
rmse.in1
rmse.out1
smape.out1

#Hasil Forecast Bulan Agustus
forecast1=forecastout1;forecast1
print(paste("Forecast Bulan Agustus =", round(forecast1,digits = 2)))

##### 2
#####
##### SEPTEMBER
#####

#FORECAST BULAN SEPTEMBER 2023
#Input Data #membentuk data baru untuk forecast bulan september
datafore1
fore11=tail(datafore1$y,n=1);fore11
fore12=datafore1$y[n_datafore1-12];fore12
databarufore2 = data.frame(y = forecast1 , x1 = fore11, x2 = fore12 );databarufore2
datafore2 = rbind(datafore1 ,databarufore2 ) ;datafore2
datafore2 = data.frame(datafore2) ;datafore2
n_datafore2=length(datafore2$y);n_datafore2

#MEMBAGI DATA U/ SEPTEMBER 2023
foretrain2=datafore2[1:42,] ;foretrain2
foretrain2 = data.frame(foretrain2) ;foretrain2
foretest2=datafore2[43,] ;foretest2
foretest2 = data.frame(foretest2) ;foretest2

#Range of parameter values FIX U/ SEPTEMBER 2023
#Parameters are: (cost, gamma, epsilon)
#to be changed U/ SEPTEMBER 2023
modfore2=svm(y~.,data=foretrain2,cost=29.84129,gamma=0.02830025,epsilon=0.06
756186)
forecast.in2=predict(modfore2,foretrain2)
forecast.out2=predict(modfore2,foretest2)

```

```

forecastin2=forecast.in2
forecastout2=forecast.out2
zt2=(foretest2[,1])
xt2=(foretrain2[,1])
isi21=xt2-forecastin2
isi22=zt2-forecastout2
rmse.in2=sqrt(mean(isi21^2))
rmse.out2=sqrt(mean(isi22^2))
smape.out2=(sum(abs(zt2-
forecastout2)/((abs(zt2)+abs(forecastout2))/2))/length(zt2))*100
rmse.in2
rmse.out2
smape.out2

#Hasil Forecast Bulan September
forecast2=forecastout2;forecast2
print(paste("Forecast Bulan September =", round(forecast2,digits = 2)))

##### 3
#####
##### OKTOBER
#####

#FORECAST BULAN OKTOBER 2023
#Input Data #membentuk data baru untuk forecast bulan Oktober
datafore2
fore31=tail(datafore2$y,n=1);fore31
fore32=datafore2$y[n_datafore2-12];fore32
databarufore3 = data.frame(y = forecast2 , x1 = fore31 , x2 = fore32 );databarufore3
datafore3 = rbind(datafore2 ,databarufore3 ) ;datafore3
n_datafore3=length(datafore3$y);n_datafore3

#MEMBAGI DATA U/ OKTOBER 2023
foretrain3=datafore3[1:43,1:3] ;foretrain3
foretrain3 = data.frame(foretrain3) ;foretrain3
foretest3=tail(datafore3,n=1) ;foretest3
foretest3 = data.frame(foretest3) ;foretest3

#Range of parameter values FIX U/ OKTOBER 2023
#Parameters are: (cost, gamma, epsilon)
#to be changed U/ OKTOBER 2023
modfore3=svm(y~.,data=foretrain3,cost=29.84129,gamma=0.02830025,epsilon=0.06
756186)
forecast.in3=predict(modfore3,foretrain3)
forecast.out3=predict(modfore3,foretest3)
forecastin3=forecast.in3
forecastout3=forecast.out3
zt3=(foretest3[,1])
xt3=(foretrain3[,1])
isi31=xt3-forecastin3

```

```

isi32=zt3-forecastout3
rmse.in3=sqrt(mean(isi31^2))
rmse.out3=sqrt(mean(isi32^2))
smape.out3=(sum(abs(zt3-
forecastout3)/((abs(zt3)+abs(forecastout3))/2))/length(zt3))*100
rmse.in3
rmse.out3
smape.out3

#Hasil Forecast Bulan OKTOBER
forecast3=forecastout3;forecast3
print(paste("Forecast Bulan OKTOBER =", round(forecast3,digits = 2)))

##### 4
#####
##### NOVEMBER
#####

#FORECAST BULAN NOVEMBER 2023
#Input Data #membentuk data baru untuk forecast bulan November
datafore3
fore41=tail(datafore3$y,n=1);fore41
fore42=datafore3$y[n_datafore3-1];fore42
databarufore4 = data.frame(y = forecast3 , x1 = fore41 , x2 = fore42 );databarufore4
datafore4 = rbind(datafore3 ,databarufore4) ;datafore4
n_datafore4=length(datafore4$y);n_datafore4

#MEMBAGI DATA U/ NOVEMBER 2023
foretrain4=datafore4[1:44,1:3] ;foretrain4
foretrain4 = data.frame(foretrain4) ;foretrain4
foretest4=tail(datafore4,n=1) ;foretest4
foretest4 = data.frame(foretest4) ;foretest4

#Range of parameter values FIX U/ OKTOBER 2023
#Parameters are: (cost, gamma, epsilon)
#to be changed U/ NOVEMBER 2023
modfore4=svm(y~.,data=foretrain4,cost=29.84129,gamma=0.02830025,epsilon=0.06
756186)
forecast.in4=predict(modfore4,foretrain4)
forecast.out4=predict(modfore4,foretest4)
forecastin4=forecast.in4
forecastout4=forecast.out4
zt4=(foretest4[,1])
xt4=(foretrain4[,1])
isi41=xt4-forecastin4
isi42=zt4-forecastout4
rmse.in4=sqrt(mean(isi41^2))
rmse.out4=sqrt(mean(isi42^2))

```



```

smape.out4=(sum(abs(zt4-
forecastout4)/((abs(zt4)+abs(forecastout4))/2))/length(zt4))*100
rmse.in4
rmse.out4
smape.out4

#Hasil Forecast Bulan November
forecast4=forecastout4;forecast4
print(paste("Forecast Bulan November =", round(forecast4,digits = 2)))

##### 5
#####
##### DESEMBER
#####

#FORECAST BULAN DESEMBER 2023
#Input Data #membentuk data baru untuk forecast bulan Desember
datafore4
fore51=tail(datafore4$y,n=1);fore51
fore52=datafore4$y[n_datafore4-1];fore52
databarufore5 = data.frame(y = forecast4 , x1 = fore51 , x2 = fore52 );databarufore5
datafore5 = rbind(datafore4 ,databarufore5) ;datafore5
n_datafore5=length(datafore5$y);n_datafore5

#MEMBAGI DATA U/ DESEMBER 2023
foretrain5=datafore5[1:45,1:3] ;foretrain5
foretrain5 = data.frame(foretrain5) ;foretrain5
foretest5=tail(datafore5,n=1) ;foretest5
foretest5 = data.frame(foretest5) ;foretest5

#Range ofparameter values FIX U/ DESEMBER 2023
#Parameters are: (cost, gamma, epsilon)
#to be changed U/ DESEMBER 2023
modfore5=svm(y~.,data=foretrain5,cost=29.84129,gamma=0.02830025,epsilon=0.06
756186)
forecast.in5=predict(modfore5,foretrain5)
forecast.out5=predict(modfore5,foretest5)
forecastin5=forecast.in5
forecastout5=forecast.out5
zt5=(foretest5[,1])
xt5=(foretrain5[,1])
isi51=xt5-forecastin5
isi52=zt5-forecastout5
rmse.in5=sqrt(mean(isi51^2))
rmse.out5=sqrt(mean(isi52^2))
smape.out5=(sum(abs(zt5-
forecastout5)/((abs(zt5)+abs(forecastout5))/2))/length(zt5))*100
rmse.in5
rmse.out5
smape.out5

```

```

#Hasil Forecast Bulan Desember
forecast5=forecastout5;forecast5
print(paste("Forecast Bulan Desember =", round(forecast5,digits = 2)))

#####
#####
#####
#####
#FORECAST BULAN JANUARI 2024
#Input Data #membentuk data baru untuk forecast bulan JANUARI
datafore5
fore61=tail(datafore5$y,n=1);fore61
fore62=datafore5$y[n_datafore5-1];fore62
databarufore6 = data.frame(y = forecast5 , x1 = fore61 , x2 = fore62 );databarufore6
datafore6 = rbind(datafore5 ,databarufore6 ) ;datafore6
n_datafore6=length(datafore6$y);n_datafore6

#MEMBAGI DATA U/ JANUARI 2024
foretrain6=datafore6[1:46,1:3] ;foretrain6
foretrain6 = data.frame(foretrain6) ;foretrain6
foretest6=tail(datafore6,n=1) ;foretest6
foretest6 = data.frame(foretest6) ;foretest6

#Range ofparameter values FIX U/ JANUARI 2024
#Parameters are: (cost, gamma, epsilon)
#to be changed U/ JANUARI 2024
modfore6=svm(y~.,data=foretrain6,cost=29.84129,gamma=0.02830025,epsilon=0.06
756186)
forecast.in6=predict(modfore6,foretrain6)
forecast.out6=predict(modfore6,foretest6)
forecastin6=forecast.in6
forecastout6=forecast.out6
zt6=(foretest6[,1])
xt6=(foretrain6[,1])
isi61=xt6-forecastin6
isi62=zt6-forecastout6
rmse.in6=sqrt(mean(isi61^2))
rmse.out6=sqrt(mean(isi62^2))
smape.out6=(sum(abs(zt6-
forecastout6)/((abs(zt6)+abs(forecastout6))/2))/length(zt6))*100
rmse.in6
rmse.out6
smape.out6

#Hasil Forecast Bulan Januari
forecast6=forecastout6;forecast6
print(paste("Forecast Bulan Januari =", round(forecast6,digits = 2)))

```

6

JANUARI

```
##### 6 BULAN
#####
datafore6
fore71=tail(datafore6$y,n=1);fore71
fore72=datafore6$y[n_datafore6-12];fore72
databarufore7 = data.frame(y = forecast6 , x1 = fore71 , x2 = fore72 );databarufore7
datafore7 = rbind(datafore6 ,databarufore7 ) ;datafore7
n_datafore7=length(datafore7$y);n_datafore7

#HASIL DATA PERAMALAN 6 BULAN KEDEPAN
HASIL.FORE.6BULAN = data.frame(datafore7$y);HASIL.FORE.6BULAN
colnames(HASIL.FORE.6BULAN) ="Nilai_Prediksi"
HASIL.FORE.6BULAN
```

Lampiran 9. Output Forecast

```
> rmse.in1
[1] 0.3137862
> rmse.out1
[1] 0.2715919
> smape.out1
[1] 0.2331977
> #Hasil Forecast Bulan Agustus
> forecast1=forecastout1;forecast1
      1
116.3284
> print(paste("Forecast Bulan Agustus =", round(forecast1,digits = 2)))
[1] "Forecast Bulan Agustus = 116.33"
```

```
> rmse.in2
[1] 0.3116692
> rmse.out2
[1] 0.08969161
> smape.out2
[1] 0.07707236
> #Hasil Forecast Bulan September
> forecast2=forecastout2;forecast2
      43
116.4181
> print(paste("Forecast Bulan September =", round(forecast2,digits = 2)))
[1] "Forecast Bulan September = 116.42"
```

```
> rmse.in3
[1] 0.3077706
> rmse.out3
[1] 0.2015675
> smape.out3
[1] 0.1732911
> #Hasil Forecast Bulan OKTOBER
> forecast3=forecastout3;forecast3
      431
116.2165
> print(paste("Forecast Bulan OKTOBER =", round(forecast3,digits = 2)))
[1] "Forecast Bulan OKTOBER = 116.22"
```

```
> rmse.in4
[1] 0.3028172
> rmse.out4
[1] 0.05781149
> smape.out4
[1] 0.04975701
> #Hasil Forecast Bulan November
> forecast4=forecastout4;forecast4
  4311
116.1587
> print(paste("Forecast Bulan November =", round(forecast4,digits = 2)))
[1] "Forecast Bulan November = 116.16"
```

```
> rmse.in5
[1] 0.2981867
> rmse.out5
[1] 0.1617319
> smape.out5
[1] 0.1393306
> #Hasil Forecast Bulan Desember
> forecast5=forecastout5;forecast5
  43111
115.997
> print(paste("Forecast Bulan Desember =", round(forecast5,digits = 2)))
[1] "Forecast Bulan Desember = 116"
```

```
> rmse.in6
[1] 0.2970101
> rmse.out6
[1] 0.04164802
> smape.out6
[1] 0.03591085
> #Hasil Forecast Bulan Januari
> forecast6=forecastout6;forecast6
  431111
115.9553
> print(paste("Forecast Bulan Januari =", round(forecast6,digits = 2)))
[1] "Forecast Bulan Januari = 115.96"
```

Lampiran 10. Surat Pernyataan Data Sekunder

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini, mahasiswa Departemen Statistika FSAD ITS:

Nama : Putri Herliani Nabilla

NRP : 5003201014

menyatakan bahwa data yang digunakan dalam **Tugas Akhir/ Thesis** ini merupakan data sekunder yang diambil dari **penelitian / buku/ Tugas Akhir/ Thesis/ publikasi** lainnya yaitu:

Sumber : Laman surabayakota.bps.go.id

Keterangan : Indeks Harga Konsumen Januari 2019 – Juli 2023

Surat Pernyataan ini dibuat dengan sebenarnya. Apabila terdapat pemalsuan data maka saya siap menerima sanksi sesuai aturan yang berlaku.

Mengetahui
Pembimbing Tugas Akhir

Surabaya, 13 Juni 2024



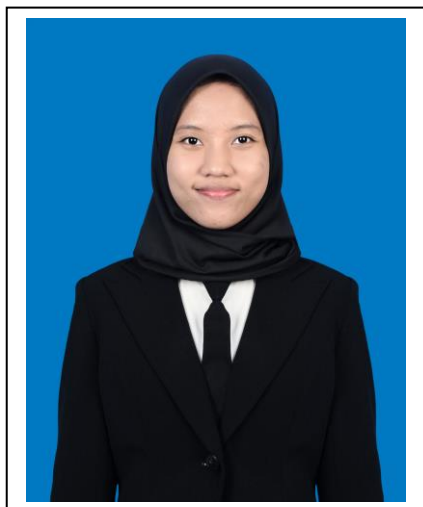
(Irhamah, S.Si., M.Si., Ph.D)
NIP. 19780406 200112 2 002



(Putri Herliani Nabilla)
NRP. 5003201014

(Halaman ini sengaja dikosongkan)

BIODATA PENULIS



Penulis dilahirkan di Sidoarjo, 25 Agustus 2002 memiliki nama lengkap Putri Herliani Nabilla. Pendidikan formal yang ditempuh penulis adalah SDN Pucang 3 Sidoarjo, SMPN 1 Sidoarjo, SMAN 1 Sidoarjo. Penulis diterima di Departemen Statistika pada tahun 2020. Semasa kuliah penulis aktif dalam beberapa kegiatan organisasi maupun kepanitiaan. Beberapa jabatan yang diemban diantaranya, Wakil Kepala II Departemen *Event* HIMASTA-ITS (Januari 2023 – Desember 2023), Staff Departemen *Event* HIMASTA-ITS (Februari 2022 – Desember 2022), Staff MUN ITS (Agustus 2021 – Agustus 2022). Sedangkan pada kepanitiaan penulis pernah menjabat diantaranya, Staff Divisi Acara pada webinar *Cracking Your Business* (2022), PIC Divisi Acara pada webinar *Power Up: Beta X*

Gamma (2022), *Organizing Committee* LKMM-TD VI HIMASTA ITS (2022), PIC Divisi Acara pada Sekolah *Event* HIMASTA-ITS (2022), Staff Divisi Acara pada *Public Awareness* HIMASTA-ITS (2022). Selain itu, penulis juga pernah mengikuti beberapa kegiatan MBKM, yaitu Studi Independen *Data Analyst* dan AI PT MariBelajar Indonesia Cerdas (Februari 2023 – Juni 2023), *Project Based Intern* di Kompas *Growth Center* sebagai *Course Analyst* (December 2022 – Januari 2023) dan Bank Muamalat sebagai *Business Intelligence Analyst* (Agustus 2023 – September 2023). Salah satu kepanitiaan besar dalam mengemban jabatan yaitu, *Vice Project Officer Event* III Pekan Raya Statistika (2022) dan *Project Officer Statistics Competition* (STATION) Pekan Raya Statistika (2023). Beberapa kegiatan lain penulis juga terlibat di luar lingkup ITS. Demikian biodata penulis, segala kritik dan saran yang membangun serta bagi pembaca yang ingin berdiskusi lebih lanjut dengan penulis mengenai segala sesuatu yang berkaitan dengan Tugas Akhir ini dapat disampaikan melalui nomor HP: 081238202433 dan email: putriherlianin@gmail.com.