



TUGAS AKHIR - IS4853

PENERAPAN METODE HYBRID GENERAL REGRESSION NEURAL NETWORK DAN FRUIT FLY OPTIMIZATION ALGORITHM (GRNN-FOA) UNTUK PERAMALAN NILAI AKTIVA BERSIH (NAB) REKSA DANA SAHAM

APPLICATION OF HYBRID GENERAL REGRESSION NEURAL NETWORK AND FLY OPTIMIZATION ALGORITHM (GRNN-FOA) METHODS FOR FORECASTING NET ASSETS VALUE (NAV) STOCK MUTUAL FUND

MUHAMMAD ARIF
NRP 05211540000119

Dosen Pembimbing
Wiwik Anggraeni, S.Si., M.Kom.

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2019



TUGAS AKHIR – IS4853

**PENERAPAN METODE HYBRID GENERAL
REGRESSION NEURAL NETWORK DAN
FRUIT FLY OPTIMIZATION ALGORITHM
(GRNN-FOA) UNTUK PERAMALAN NILAI
AKTIVA BERSIH (NAB) REKSA DANA
SAHAM**

MUHAMMAD ARIF
NRP 05211540000119

Dosen Pembimbing
Wiwik Anggraeni, S.Si., M.Kom.

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2019

Halaman ini sengaja dikosongkan



FINAL PROJECT – IS4853

***APPLICATION OF HYBRID GENERAL
REGRESSION NEURAL NETWORK AND FLY
OPTIMIZATION ALGORITHM (GRNN-FOA)
METHODS FOR FORECASTING NET ASSETS
VALUE (NAV) STOCK MUTUAL FUND***

**MUHAMMAD ARIF
NRP 05211540000119**

**Supervisor
Wiwik Anggraeni, S.Si., M.Kom.**

INFORMATION SYSTEMS DEPARTMENT

Faculty of Information and Communication Technology

Sepuluh Nopember Institut of Technology

Surabaya 2019

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

**PENERAPAN METODE HYBRID GENERAL REGRESSION
NEURAL NETWORK DAN FRUIT FLY OPTIMIZATION
ALGORITHM (GRNN-FOA) UNTUK PERAMALAN NILAI
AKTIVA BERSIH (NAB) REKSA DANA SAHAM**

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

MUHAMMAD ARIF

NRP. 05211540000119

Surabaya, Juli 2019

KEPALA

DEPARTEMEN SISTEM INFORMASI



Mahendrawathi ER, S.T., M.Sc., Ph.D.

NIP. 19761011 200604 2 001

Halaman ini sengaja dikosongkan

LEMBAR PERSETUJUAN

**PENERAPAN METODE HYBRID GENERAL REGRESSION
NEURAL NETWORK DAN FRUIT FLY OPTIMIZATION
ALGORITHM (GRNN-FOA) UNTUK PERAMALAN NILAI
AKTIVA BERSIH (NAB) REKSA DANA SAHAM**

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

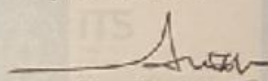
Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh :


MUHAMMAD ARIF
NRP. 05211540000119

Disetujui Tim Penguji : Tanggal Ujian : 8 Juli 2019
Periode Wisuda : September 2019

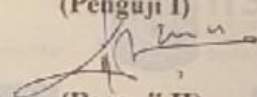
Wiwik Anggraeni, S.Si., M.Kom.


(Pembimbing I)

Ahmad Mukhlason, S.Kom., M.Sc., Ph.D.


(Penguji I)

Faizal Mahananto, S.Kom, M.Eng., Ph.D.


(Penguji II)



Halaman ini sengaja dikosongkan

**PENERAPAN METODE HYBRID GENERAL
REGRESSION NEURAL NETWORK DAN FRUIT FLY
OPTIMIZATION ALGORITHM (GRNN-FOA) UNTUK
PERAMALAN NILAI AKTIVA BERSIH (NAB) REKSA
DANA SAHAM**

Nama Mahasiswa : Muhammad Arif
NRP : 05211540000119
Departemen : Sistem Informasi
Dosen Pembimbing : Wiwik Anggraeni, S.Si., M.Kom.

ABSTRAK

Sejak tahun 2015, pemerintah melalui PT Bursa Efek Indonesia (BEI) telah berusaha meningkatkan kesadaran masyarakat terhadap dunia investasi. Salah satu kendala kurangnya kesadaran masyarakat terhadap investasi adalah kurangnya dana, waktu, serta pengetahuan terhadap investasi. Masyarakat yang memiliki karakteristik demikian sangat disarankan untuk melakukan investasi dalam bentuk reksa dana. Reksa dana merupakan salah satu bentuk investasi berupa wadah tempat berkumpulnya dana dari banyak investor. Investor yang ingin melakukan investasi jangka panjang (lebih dari lima tahun) dan mengharapkan hasil return berupa Nilai Aktiva Bersih per Unit Penyertaan (NAB/UP) besar maka sangat cocok untuk melakukan investasi dalam bentuk reksa dana saham. Hal ini dikarenakan reksa dana saham mengalokasikan minimal 80% dari investasinya ke pasar saham, yang mana pasar saham

merupakan salah satu bentuk investasi yang sangat cocok dengan karakter tersebut.

General Regression Neural Network (GRNN) merupakan metode peramalan yang memiliki kemampuan pemetaan non-linier yang kuat. Kekurangan dari GRNN adalah pemilihan parameter spread yang sulit. GRNN dengan parameter spread yang dioptimalkan mampu memberikan akurasi lebih tinggi daripada GRNN tanpa optimasi. Salah satu algoritma optimasi yang dapat digunakan untuk mencari parameter spread paling optimal adalah Fruit Fly Optimization Algorithm (FOA). Tugas akhir ini akan meramalkan NAB/UP menggunakan GRNN dengan parameter spread yang akan dioptimasi menggunakan FOA.

Metode peramalan GRNN yang dioptimasi menggunakan FOA mampu memberikan hasil peramalan dengan akurasi yang sangat baik yaitu MAPE 2,40% dan RMSE 56,53752789 pada peramalan periode kedepan. Parameter spread paling optimal yang dihasilkan dari FOA adalah 20,9004.

Kata kunci: Reksa Dana Saham, GRNN, FOA, Simas Saham Unggulan, NAB/UP

***APPLICATION OF HYBRID GENERAL REGRESSION
NEURAL NETWORK AND FLY OPTIMIZATION
ALGORITHM (GRNN-FOA) METHODS FOR
FORECASTING NET ASSETS VALUE (NAV) STOCK
MUTUAL FUND***

Name : Muhammad Arif
NRP : 05211540000119
Department : Information Systems
Supervisor : Wiwik Anggraeni, S.Si., M.Kom.

ABSTRACT

Since 2015, the government through the Indonesia Stock Exchange (IDX) has been trying to increase public awareness of the investment world. One obstacle to the lack of public awareness of investment is the lack of funds, time, and knowledge of investment. People who have such characteristics are strongly advised to invest in mutual funds. Mutual funds are one form of investment in the form of a place to gather funds from many investors. Investors who want to make a long-term investment (more than five years) and expect a return in the form of Net Asset Value per Participation Unit (NAV/PU) is very suitable for investing in the form of stock mutual funds. This is because stock mutual funds allocate a minimum of 80% of their investment to the stock market, which is where the stock market is a form of investment that is very suitable for that character.

General Regression Neural Network (GRNN) is a forecasting method that has strong non-linear mapping capabilities. The disadvantages of GRNN are the selection of difficult spread parameters. GRNN with optimized spread parameters can provide higher accuracy than GRNN without optimization. One of the optimization algorithms that can be used to find the most optimal spread parameters is Fruit Fly Optimization Algorithm (FOA). This final project will predict NAV / PU using GRNN with spread parameters that will be optimized using FOA.

The GRNN forecasting method optimized using FOA is able to provide forecasting results with excellent accuracy, namely MAPE 2.39% and RMSE 56.53752789 for the future forecasting period. The most optimal spread parameter generated from FOA is 20,9004.

Keywords: Equity Funds, GRNN, FOA, Simas Saham Unggulan, NAV / PU

KATA PENGANTAR

Asyhadu alla ilaaha illallah, Wa asyhadu anna Muhammadar Rasulullah. Astaghfirullah hal adzim, Astaghfirullah hal adzim, Astaghfirullah hal adzim. Alhamdulillah robbil 'alamin, segala puji bagi Allah SWT yang telah memberikan segala bentuk rahmat sehingga Penulis mampu menyelesaikan penelitian tugas akhir berjudul **“PENERAPAN METODE HYBRID GENERAL REGRESSION NEURAL NETWORK DAN FRUIT FLY OPTIMIZATION ALGORITHM (GRNN-FOA) UNTUK PERAMALAN NILAI AKTIVA BERSIH (NAB) REKSA DANA SAHAM”** yang merupakan salah satu syarat kelulusan dari Departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya.

Melalui tulisan ini Penulis ingin mengucapkan rasa terima kasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan segala rahmat bagi penulis sehingga penulis mampu menyelesaikan penelitian tugas akhir.
2. Muhammad SAW yang telah menjadi sosok panutan bagi penulis sehingga penulis mampu menyelesaikan penelitian tugas akhir dengan lancar.
3. Bapak dan Mak, Achmad Dahlan dan Ibu Siti Khotijah, yang selalu memberikan dukungan dalam berbagai bentuk selama Penulis menjalani kehidupan baik dari sisi akademik maupun non-akademik.

4. Ketiga adik Penulis, Nur Hayati, Nur Hidayati, Muhammad Irfan Syahputra, yang selalu menjadi motivasi terbesar Penulis untuk bisa menyelesaikan penelitian tugas akhir.
5. Almarhumah Mbah Suyatwi, Almarhum Mbah Amin, Almarhum Mbah Saekun, Mbah Kasmilah selaku kakek dan nenek Penulis yang selalu memberikan panutan dan kasih sayang kepada Penulis.
6. Ibu Renny Pradina Kusumawardani ST., MT. selaku osen wali yang selalu memberikan dukungan selama Penulis menjalani masa perkuliahan di Departemen Sistem Informasi ITS.
7. Ibu Wiwik Anggraeni, S.Si., M.Kom. selaku dosen pembimbing yang selalu memberikan motivasi serta bimbingan kepada Penulis selama pengerjaan tugas akhir.
8. Bapak Ahmad Muklason, S.Kom., M.Sc., Ph.D. serta Bapak Faizal Mahananto, S.Kom, M.Eng., Ph.D. selaku dosen penguji yang selalu memberikan masukan terhadap pengerjaan tugas akhir.
9. Seluruh tenaga pendidik serta karyawan Departemen Sistem Informasi ITS yang selalu membantu dalam setiap aktivitas perkuliahan.
10. Keluarga besar Komunitas YOT Surabaya, PT YOT Inspirasi Nusantara (YOT Pusat), Paguyuban KSE ITS, HMSI ITS, BEM ITS, Pemandu LKMM Phoenix yang telah memberikan “wadah” bagi Penulis untuk selalu mengembangkan diri.
11. Yayasan Karya Salemba Empat beserta para donatur, khususnya PT Indofood Sukses Makmur Tbk yang telah memberi kesempatan kepada Penulis untuk bergabung

sebagai keluarga besar Beasiswa Indofood Sukses Makmur (BISMA).

12. Sahabat-sahabat dekat penulis, Eric Syafril Rizal, Farih Fiddaroin Fadli, Umar Rizki Kusumo Widayu, yang selalu menemani penulis dalam menghabiskan masa muda.
13. Mba Fioriza Syahdana Yulmi yang selalu bersedia mendengarkan keluh kesah penulis dalam menjalani kehidupan.
14. Ardo Fachrizal Ilmy yang selalu sabar menemani penulis dan mengajarkan penulis tentang arti persahabatan.
15. Elsa Siffana Hedianti yang mau menjadi teman diskusi terkait pengerjaan tugas akhir
16. Penghuni Laboratorium Manajemen Sistem Informasi yang selalu bersedia menerima penulis hampir setiap malam dalam mengerjakan tugas akhir.
17. Lannister.

Semoga kita semua selalu diberikan kesehatan dan berada dalam lindungan Tuhan YME, dan semoga tulisan ini dapat bermanfaat bagi pembaca. Amin.

Surabaya, 1 Juli 2019

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

LEMBAR PENGESAHAN.....	Error! Bookmark not defined.
LEMBAR PERSETUJUAN.....	Error! Bookmark not defined.
ABSTRAK.....	ix
ABSTRACT.....	xi
KATA PENGANTAR	xiii
DAFTAR ISI.....	xvii
DAFTAR GAMBAR	xxiii
DAFTAR TABEL.....	xxv
DAFTAR KODE PROGRAM.....	xxvii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah	1
1.2. Perumusan Masalah.....	4
1.3. Batasan Masalah.....	5
1.4. Tujuan Penelitian.....	5
1.5. Manfaat Penelitian.....	6
1.6. Relevansi.....	6
BAB II TINJAUAN PUSTAKA.....	9
2.1. Studi Sebelumnya.....	9
2.2. Dasar Teori.....	13

2.2.1.	Konsep Umum Peramalan.....	13
2.2.2.	Reksa Dana.....	15
2.2.3.	Nilai Aktiva Bersih Per Unit Penyertaan (NAB/UP).....	17
2.2.4.	<i>Neural Network</i>	18
2.2.5.	<i>General Regression Neural Network (GRNN)</i>	19
2.2.6.	<i>Double Moving Average</i>	23
2.2.7.	<i>Double Exponential Smoothing Holt</i>	24
2.2.8.	<i>Double Exponential Smoothing Brown</i>	25
2.2.9.	<i>Fruit Fly Optimization Algorithm (FOA)</i>	27
2.2.7.	FOA Untuk Pemilihan Parameter Pada GRNN (GRNN-FOA).....	30
2.2.8.	Uji Akurasi Performa Model Peramalan	32
BAB III METODOLOGI PENELITIAN		35
3.1.	Diagram Metodologi	35
3.2.	Identifikasi Masalah	36
3.3.	Studi Literatur	36
3.4.	Pengumpulan Data	37
3.5.	Pre-Processing Data	37
3.6.	Parameter Tuning	37
3.7.	Peramalan Menggunakan GRNN.....	38
3.8.	Analisis Peramalan.....	38

3.9.	Penyusunan Tugas Akhir.....	39
BAB IV PERANCANGAN		41
4.1.	Pengumpulan Data.....	41
4.2.	<i>Pre-Processing</i> Data.....	42
4.3.	Parameter <i>Tuning</i>	42
4.4.	Proses Peramalan Data.....	44
4.5.	Analisis Performa Model.....	45
BAB V IMPLEMENTASI.....		47
5.1.	Pengumpulan Dan <i>Pre-processing</i> Data.....	47
5.2.	Parameter <i>Tuning</i>	49
5.2.1.	Inisialisasi Parameter FOA	50
5.2.2.	Inisialisasi Populasi FOA.....	51
5.2.3.	Evaluasi Populasi FOA	51
5.2.4.	Operasi Pemilihan Parameter	53
5.3.	Implementasi Model GRNN.....	58
5.3.1.	<i>Load</i> Data Aktual	58
5.3.2.	Peramalan menggunakan GRNN	61
5.4.	Analisis Performa Model.....	63
BAB VI HASIL DAN PEMBAHASAN		65
6.1.	Hasil Pengumpulan dan <i>Pre-processing</i> Data	65
6.2.	Hasil Parameter <i>Tuning</i>	66

6.2.1.	Skenario 100 Iterasi 10 Populasi	66
6.2.2.	Skenario 100 Iterasi 50 Populasi	68
6.2.3.	Skenario 100 Iterasi 100 Populasi	70
6.2.4.	Skenario 200 Iterasi 50 Populasi	72
6.2.5.	Skenario 200 Iterasi 100 Populasi	74
6.2.6.	Skenario 200 Iterasi 150 Populasi	76
6.2.7.	Skenario 200 Iterasi 200 Populasi	78
6.2.8.	Skenario 300 Iterasi 300 Populasi	80
6.2.9.	Skenario 400 Iterasi 400 Populasi	82
6.2.10.	Skenario 500 Iterasi 500 Populasi	85
6.2.11.	Perbandingan Seluruh Skenario	87
6.3.	Hasil Implementasi Model GRNN.....	89
6.3.1.	Peramalan Data <i>Training</i>	90
6.3.2.	Peramalan Data <i>Testing</i>	91
6.3.3.	Peramalan Periode Kedepan.....	92
6.3.4.	Pembuktian Gambar 6.12.....	97
BAB VII KESIMPULAN DAN SARAN		103
7.1.	Kesimpulan	103
7.2.	Saran	104
DAFTAR PUSTAKA.....		105
BIODATA PENULIS.....		109

LAMPIRAN A	111
LAMPIRAN B	113
LAMPIRAN C	115
LAMPIRAN D	117
LAMPIRAN E	119
LAMPIRAN F.....	121
LAMPIRAN G	123
LAMPIRAN H	125

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 2.1 Arsitektur GRNN	22
Gambar 2.2 Proses Iterasi Pencarian Makanan oleh Kawan Lalat Buah	29
Gambar 2.3 Flowchart Model GRNN-FOA.....	31
Gambar 3.1 Diagram Metodologi Penelitian Tugas Akhir	35
Gambar 5.1 Grafik NAB/UP Simas Saham Unggulan.....	48
Gambar 5.2 Hasil Visualisasi RMSE	56
Gambar 5.3 Hasil Visualisasi Posisi Lalat Buah	57
Gambar 6.1 RMSE 100 Iterasi 10 Populasi	68
Gambar 6.2 RMSE 100 Iterasi 50 Populasi	70
Gambar 6.3 RMSE 100 Iterasi 100 Populasi	72
Gambar 6.4 RMSE 200 Iterasi 50 Populasi	74
Gambar 6.5 RMSE 200 Iterasi 100 Populasi	76
Gambar 6.6 RMSE 200 Iterasi 150 Populasi	78
Gambar 6.7 RMSE 200 Iterasi 200 Populasi	80
Gambar 6.8 RMSE 300 Iterasi 300 Populasi	82
Gambar 6.9 RMSE 400 Iterasi 400 Populasi Iterasi Ke-202 .	84
Gambar 6.10 RMSE 400 Iterasi 400 Populasi Iterasi Ke-12 .	85
Gambar 6.11 RMSE 500 Iterasi 500 Populasi	87
Gambar 6.12 Hasil Peramalan Semua Data	96

Gambar 6.13 RMSE 1000 Iterasi 500 Populasi.....100
Gambar 6.14 Hasil Peramalan Semua Data (1000 Iterasi 100
Populasi).....101

DAFTAR TABEL

Tabel 2.1 Studi Sebelumnya.....	9
Tabel 2.2 Justifikasi nilai MAPE	33
Tabel 4.1 Parameter FOA.....	42
Tabel 6.1 Hasil Skenario 100 Iterasi 10 Populasi	67
Tabel 6.2 Hasil Skenario 100 Iterasi 50 Populasi	68
Tabel 6.3 Hasil Skenario 100 Iterasi 100 Populasi	70
Tabel 6.4 Hasil Skenario 200 Iterasi 50 Populasi	72
Tabel 6.5 Hasil Skenario 200 Iterasi 100 Populasi	74
Tabel 6.6 Hasil Skenario 200 Iterasi 150 Populasi	76
Tabel 6.7 Hasil Skenario 200 Iterasi 200 Populasi	78
Tabel 6.8 Hasil Skenario 300 Iterasi 300 Populasi	80
Tabel 6.9 Hasil Skenario 400 Iterasi 400 Populasi	82
Tabel 6.10 Hasil Skenario 500 Iterasi 500 Populasi	85
Tabel 6.11 Perbandingan Seluruh Skenario	87
Tabel 6.12 Hasil Peramalan Periode Kedepan	94
Tabel 6.13 Perbandingan Peramalan Periode Kedepan.....	94
Tabel 6.14 Hasil Skenario 1000 Iterasi 500 Populasi	98

Halaman ini sengaja dikosongkan

DAFTAR KODE PROGRAM

Kode Program 5.1 Pre-processing Data	49
Kode Program 5.2 Inisialisasi Parameter FOA	50
Kode Program 5.3 Inisialisasi Populasi FOA.....	51
Kode Program 5.4 Evaluasi Populasi FOA	52
Kode Program 5.5 Penghitungan Fitness Function	53
Kode Program 5.6 Pemilihan Parameter Terbaik.....	53
Kode Program 5.7 Operasi Pemilihan Parameter.....	54
Kode Program 5.8 Perbandingan Nilai RMSE	54
Kode Program 5.9 Visualisasi RMSE Dan Posisi Lalat Buah	55
Kode Program 5.10 Hasil Visualisasi RMSE Dan Posisi Lalat Buah	56
Kode Program 5.11 Load Data Peramalan Data Training.....	59
Kode Program 5.12 Load Data Peramalan Data Testing	59
Kode Program 5.13 Load Data Peramalan Periode Kedepan.	60
Kode Program 5.14 Peramalan Menggunakan GRNN	62
Kode Program 5.15 Penghitungan RMSE GRNN Menggunakan MATLAB	63

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

Pada bab ini akan dijelaskan mengenai Latar Belakang Masalah, Perumusan Masalah, Batasan Masalah, Tujuan Penelitian, Manfaat Penelitian, serta Relevansi tugas akhir dengan laboratorium.

1.1. Latar Belakang Masalah

Sejak tahun 2015, pemerintah melalui PT Bursa Efek Indonesia (BEI) telah berusaha meningkatkan kesadaran masyarakat terhadap dunia investasi. Diharapkan masyarakat Indonesia saat ini bukan hanya bergerak sebagai *saving society*, namun mampu bergerak sebagai *investing society*. Kesadaran masyarakat dalam berinvestasi ini sangat penting dikarenakan dengan berinvestasi, masyarakat diharapkan mampu melawan inflasi yang terjadi setiap tahun, menyiapkan kesiapan finansial masa depan, serta meraih target finansial yang telah ditentukan sebelumnya[1]. Salah satu kendala kurangnya kesadaran masyarakat terhadap investasi adalah kurangnya dana, waktu, serta pengetahuan terhadap investasi. Masyarakat yang memiliki karakteristik demikian sangat disarankan untuk melakukan investasi dalam bentuk reksa dana. Reksa dana merupakan salah satu bentuk investasi berupa wadah tempat berkumpulnya dana dari banyak investor. Dana tersebut nantinya akan dialokasikan oleh manajer investasi ke berbagai macam bentuk investasi lain seperti saham, obligasi, pasar uang, dan portofolio efek lainnya. Manajer investasi adalah orang yang memiliki

keahlian di bidang investasi dan bertugas untuk mengelola dana yang telah diberikan oleh investor reksa dana agar memberikan hasil yang menguntungkan bagi investor.

Investor yang ingin melakukan investasi jangka panjang (lebih dari lima tahun) dan mengharapkan hasil *return* berupa Nilai Aktiva Bersih per Unit Penyertaan (NAB/UP) besar maka sangat cocok untuk melakukan investasi dalam bentuk reksa dana saham. Hal ini dikarenakan reksa dana saham mengalokasikan minimal 80% dari investasinya ke pasar saham, yang mana pasar saham merupakan salah satu bentuk investasi yang sangat cocok dengan karakter tersebut. NAB/UP merupakan salah satu tolok ukur dalam memantau hasil kinerja reksa dana. NAB/UP merupakan akumulasi dari jumlah harga pasar portofolio dikurangi biaya operasional, dan dibagi dengan jumlah saham yang beredar[2]. Penghitungan NAB/UP dilakukan oleh Bank Kustodian setelah mendapatkan data dari Manajer Investasi, dan dapat dilihat pada surat kabar tertentu setiap harinya. NAB/UP bisa bergerak naik-turun setiap hari tergantung pada perubahan nilai portofolio efek dalam reksa dana. Peningkatan NAB/UP dapat menunjukkan meningkatnya nilai investasi yang dimiliki oleh investor reksa dana, begitupun sebaliknya. Investor perlu mengetahui berapa NAB/UP reksa dana ketika melakukan transaksi untuk memperkirakan jumlah unit dan keuntungan yang diperoleh. Apabila NAB/UP reksa dana masa depan lebih tinggi daripada masa sekarang, maka investor dapat menjual unit yang dimiliki di masa depan untuk mendapatkan keuntungan. Begitupun sebaliknya, apabila NAB/UP reksa dana masa depan lebih rendah daripada masa sekarang, maka investor investor dapat menjual unit yang dimiliki di masa sekarang

untuk menghindari kerugian. Oleh karena itu dibutuhkan adanya peramalan NAB/UP reksa dana saham. Tugas akhir ini akan menggunakan reksa dana Simas Saham Unggulan sebagai objek penelitian. Simas Saham Unggulan merupakan salah satu produk reksa dana saham dari Sinarmas Asset Management. Reksa dana saham Simas Saham Unggulan dipilih karena memiliki kinerja yang bagus dalam dunia reksa dana dalam lima tahun terakhir (Periode 3 Januari 2014 – 3 Januari 2019)[3]. Selain itu NAB/UP reksa dana Simas Saham Unggulan juga memiliki pola data yang fluktuatif pada setiap periode. Sehingga diperlukan peramalan terhadap NAB/UP reksa dana tersebut agar dapat dijadikan sebagai acuan bagi investor untuk mengetahui kapan waktu yang tepat untuk melakukan transaksi. Waktu transaksi yang tepat bagi investor adalah melakukan pembelian ketika NAB/UP reksa dana masa depan lebih tinggi daripada masa sekarang untuk mendapatkan keuntungan, serta melakukan penjualan ketika NAB/UP reksa dana masa depan lebih rendah daripada masa sekarang untuk menghindari kerugian.

Penelitian sebelumnya dilakukan oleh Tsul-Hua Hang dan Yungho Leu[4] mengenai peramalan reksa dana menggunakan metode *General Regression Neural Network – Fruit Fly Optimization Algorithm* (GRNN-FOA). Penelitian ini menunjukkan bahwa apabila dibandingkan dengan metode peramalan GRNN dan *Linear Regression*, kombinasi antara penggunaan metode GRNN dan FOA untuk melakukan peramalan reksa dana dapat memberikan hasil paling baik. Selain itu, penelitian sebelumnya juga dilakukan oleh Shi-Zhuan Han, Li-Hui Huang, Ying-Ying Zhou, dan Zong-Li Liu[5] membahas tentang perbandingan peramalan reksa dana

menggunakan metode GRNN yang dioptimasi menggunakan *Artificial Fish Swarm Algorithm* (AFSA), *Particle Swarm Optimization* (PSO), FOA, dan tanpa dilakukan optimasi. Penelitian ini menunjukkan bahwa apabila dibandingkan dengan model peramalan lain, GRNN yang dioptimasi menggunakan FOA merupakan model dengan akurasi peramalan paling tinggi.

GRNN dipilih sebagai metode peramalan NAB/UP reksa dana saham karena memiliki kemampuan pemetaan *non-linier* yang kuat. Data NAB/UP Simas Saham Unggulan menunjukkan grafik data non-linier. Kekurangan dari GRNN adalah pemilihan parameter *spread* yang sulit. Peneliti sebelumnya menggunakan algoritma genetika untuk mengoptimalkan parameter *spread* pada GRNN. GRNN dengan parameter *spread* yang dioptimalkan mampu memberikan akurasi lebih tinggi daripada GRNN tanpa optimasi. Sebagian besar peneliti memilih parameter *spread* berdasarkan pengalaman serta banyaknya eksperimen yang dilakukan[4].

Tujuan tugas akhir ini adalah membuat model peramalan GRNN yang selanjutnya dioptimasi menggunakan FOA. Model tersebut nantinya akan digunakan untuk meramalkan NAB/UP Simas Saham Unggulan. Model GRNN-FOA dipilih karena pada penelitian sebelumnya menunjukkan bahwa model tersebut memiliki akurasi peramalan lebih baik apabila dibandingkan dengan model lain[4], [5].

1.2. Perumusan Masalah

Berdasarkan uraian latar belakang yang telah dijelaskan, maka perumusan masalah dari tugas akhir ini, yaitu:

1. Bagaimana model GRNN-FOA terbaik yang dapat digunakan untuk meramalkan NAB/UP reksa dana saham?
2. Bagaimana hasil dan tingkat akurasi peramalan NAB/UP reksa dana saham menggunakan metode GRNN-FOA?

1.3. Batasan Masalah

Batasan masalah dalam tugas akhir ini, yaitu:

1. Data yang digunakan dalam tugas akhir ini adalah data NAB/UP reksa dana Simas Saham Unggulan.
2. Data yang digunakan adalah data harian pada periode 4 Januari 2016 sampai dengan 28 Desember 2018.
3. Variabel bebas yang berpengaruh pada peramalan ini adalah data *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan *exchange rate* IDR – USD (kurs jual dan kurs beli).
4. Perangkat lunak yang digunakan untuk melakukan peramalan dengan GRNN dan FOA adalah MATLAB.

1.4. Tujuan Penelitian

Berdasarkan perumusan masalah yang telah diuraikan, maka tujuan tugas akhir ini, yaitu:

1. Menemukan model terbaik untuk melakukan peramalan NAB/UP reksa dana saham dengan menggunakan metode GRNN-FOA.
2. Mengetahui hasil dan tingkat akurasi dalam peramalan NAB/UP reksa dana saham menggunakan metode GRNN-FOA.

1.5. Manfaat Penelitian

Tugas akhir ini diharapkan dapat memberikan manfaat antara lain:

1. Bagi Peneliti

Mampu memahami GRNN-FOA dalam melakukan peramalan NAB/UP reksa dana serta menghitung akurasi dari hasil peramalan tersebut.

2. Bagi Pendidikan

Menjadikan penelitian ini sebagai sumber masukan untuk penelitian terkait dengan peramalan reksa dana harian dengan menggunakan GRNN-FOA. Hasil penelitian tersebut nantinya dapat digunakan sebagai bahan pembelajaran serta sebagai acuan dalam pengembangan penelitian selanjutnya.

3. Bagi Investor dan Pelaku Bisnis Terkait

Memudahkan investor dan pelaku bisnis reksa dana di Indonesia dalam mengambil keputusan dengan melakukan peramalan yang akurat terhadap NAB/UP reksa dana.

1.6. Relevansi

Tugas akhir ini menggunakan metode *hybrid General Regression Neural Network – Fruit Fly Optimization Algorithm* (GRNN-FOA). Tugas akhir ini bertujuan untuk

memenuhi syarat kelulusan, serta sebagai bentuk implementasi disiplin ilmu yang telah didapatkan selama pendidikan akademik di Departemen Sistem Informasi ITS. Topik yang diangkat dalam penelitian tugas akhir ini adalah peramalan data. Topik tersebut secara garis besar memiliki relevansi dengan mata kuliah yang terdapat di Departemen Sistem Informasi ITS, bidang keilmuan terkait antara lain sebagai berikut:

- a. Statistika
- b. Penggalan Data dan Analitika Bisnis
- c. Teknik Peramalan
- d. Sistem Pendukung Keputusan

Topik ini relevan dengan bidang keilmuan *Business Analytic* pada sub bidang peramalan, dimana hal tersebut merupakan salah satu *roadmap* pada Laboratorium Rekayasa Data dan Intelegrasi Bisnis (RDIB) di Departemen Sistem Informasi, Institut Teknologi Sepuluh Nopember Surabaya.

Halaman ini sengaja dikosongkan

BAB II

TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan mengenai Studi Sebelumnya serta Dasar Teori terkait tugas akhir yang dikerjakan oleh Penulis.

2.1. Studi Sebelumnya

Pada sub bab ini akan dijelaskan mengenai studi sebelumnya yang digunakan untuk mendukung pengerjaan tugas akhir.

Tabel 2.1 Studi Sebelumnya

No	Studi Sebelumnya	
1.	Judul Penelitian	<i>Mixed chaotic FOA with GRNN to construction of a mutual fund forecasting model</i> [5]
	Tahun Penelitian	2018
	Nama Peneliti	Shi-Zhuan Han, Li-Hui Huang, Ying-Ying Zhou, Zong-Li Liu
	Penjelasan Singkat	Penelitian ini membahas tentang peramalan reksa dana yang ada di taiwan. Penelitian ini mencoba membandingkan akurasi peramalan reksa dana menggunakan metode GRNN yang tidak dilakukan optimasi dengan metode GRNN yang dioptimasi menggunakan AFSA, PSO, dan FOA. Setelah melakukan perbandingan

No	Studi Sebelumnya	
		terhadap hasil peramalan dari metode di atas, maka didapatkan hasil bahwa peramalan reksa dana menggunakan metode GRNN yang dioptimasi menggunakan FOA mendapatkan akurasi peramalan tertinggi daripada metode lainnya.
	Keterkaitan dengan Tugas Akhir	Penelitian ini menggunakan metode GRNN yang dioptimasi menggunakan banyak algoritma untuk melakukan peramalan reksa dana yang ada di Taiwan, salah satu algoritma terbaik yang dihasilkan adalah FOA. Tugas akhir ini menggunakan metode GRNN yang dioptimasi dengan FOA untuk melakukan peramalan reksa dana.
2.	Judul Penelitian	<i>A hybrid annual power load forecasting model based on generalized regression neural network with fruit fly optimization algorithm</i> [6]
	Tahun Penelitian	2013
	Nama Peneliti	Hong-ze Li, Sen Guo, Chun-jie Li, Jing-qi Sun
	Penjelasan Singkat	Penelitian ini membahas tentang penggunaan metode peramalan GRNN yang dioptimasi menggunakan FOA

No	Studi Sebelumnya
	<p>untuk melakukan peramalan terhadap beban daya listrik tahunan di Beijing, Cina. Peramalan beban daya listrik tahunan merupakan permasalahan <i>non-linear</i> karena grafik beban daya listrik menunjukkan karakteristik <i>non-linear</i>. GRNN telah terbukti dapat mengatasi permasalahan <i>non-linear</i> secara efektif, namun sayangnya metode peramalan GRNN sangat jarang digunakan untuk melakukan peramalan beban daya listrik tahunan. Bagaimanapun juga penentuan parameter persebaran yang cocok untuk digunakan dalam penerapan metode peramalan GRNN merupakan salah satu poin penting. Dalam penelitian ini metode hibrid GRNN-FOA digunakan untuk mengatasi masalah tersebut, dimana FOA mampu secara otomatis digunakan untuk menentukan nilai parameter persebaran yang cocok untuk model GRNN yang akan digunakan. Keefektivan model hibrid GRNN-FOA telah terbukti oleh simulasi dua eksperimen, yaitu model GRNN-FOA mampu mengungguli peramalan dengan menggunakan metode PSOGRNN (<i>GRNN model with particle swarm</i></p>

No	Studi Sebelumnya	
		<i>optimization</i>), SALSSVM (<i>support vector machine with simulated annealing algorithm</i>), dan OLS_LR (<i>ordinary least squares linear regression</i>).
	Keterkaitan dengan Tugas Akhir	Penelitian ini membahas tentang peramalan menggunakan metode GRNN yang dioptimasi menggunakan FOA. Tugas akhir ini akan menggunakan metode peramalan tersebut untuk melakukan peramalan pada reksa dana.
3.	Judul Penelitian	<i>A Mutual Fund Investment Method Using Fruit Fly Optimization Algorithm and Neural Network</i> [4]
	Tahun Penelitian	2014
	Nama Peneliti	Tsui-Hua Huang, Yungho Leu
	Penjelasan Singkat	Penelitian ini membahas tentang penggunaan metode peramalan GRNN yang dioptimasi menggunakan FOA untuk membentuk portofolio reksa dana yang menguntungkan untuk investor. Metode ini terdiri dari dua tahap. Pada tahap pertama, indeks DEA, Sharp dan Treynor reksa dana, serta <i>rates of return</i>

No	Studi Sebelumnya	
		(ROR) bulanan reksa dana digunakan untuk memilih portofolio reksa dana. Pada tahap kedua, model regresi linear, FOA dan GRNN digunakan untuk membentuk model peramalan pada portofolio reksa dana. Hasil dari penelitian ini adalah apabila dibandingkan dengan metode peramalan lain, metode peramalan GRNN-FOA merupakan metode terbaik untuk melakukan peramalan reksa dana yang ditandai dengan akurasi peramalan paling tinggi.
	Keterkaitan dengan Tugas Akhir	Penelitian ini membahas tentang peramalan reksa dana menggunakan metoder GRNN-FOA. Tugas akhir ini akan menggunakan metode GRNN-FOA untuk melakukan peramalan reksa dana.

2.2. Dasar Teori

Pada sub bab ini akan dijelaskan mengenai dasar teori yang digunakan untuk mendukung pengerjaan tugas akhir.

2.2.1. Konsep Umum Peramalan

Peramalan merupakan sebuah kegiatan teratur yang bertujuan untuk memprediksi kejadian masa depan dengan

menggunakan berbagai faktor yang tidak hanya bersifat metode ilmiah, namun juga mempertimbangkan hal-hal yang bersifat kualitatif, seperti perasaan, pengalaman seseorang, dan sifat-sifat kualitatif lainnya. Peramalan yang dibuat selalu diupayakan agar dapat meminimalkan pengaruh ketidakpastian. Dalam artian peramalan memiliki tujuan untuk mendapatkan hasil peramalan yang memiliki kesalahan peramalan (*forecast error*) seminimal mungkin yang bisa diukur dengan *Mean Squared Error* (MSE), *Mean Absolute Error* (MEA), dan sebagainya[7].

Metode peramalan dapat dibagi menjadi dua metode, yaitu[8]:

a. Metode kualitatif

Peramalan yang didasarkan pada data kualitatif masa lalu yang digabungkan dengan pemikiran dari penyusunnya.

b. Metode kuantitatif

Peramalan yang didasarkan pada data kuantitatif dari data sebelumnya dan variabel sebab akibat. Hasil peramalan bergantung pada metode yang akan digunakan, karena setiap metode memiliki hasil peramalan yang berbeda pula.

Peramalan secara kuantitatif ini dibagi lagi menjadi dua model, yaitu[8]:

a. Model deret waktu (*time series*)

Suatu variabel diramalkan berdasarkan nilai variabel itu sendiri pada periode sebelumnya.

b. Model kausal (*explanatory*)

Suatu variabel diramalkan berdasarkan pada nilai dari salah satu atau beberapa variabel lain yang berpengaruh. Model ini biasanya menggunakan analisis regresi dalam menentukan model yang tepat karena pencarian variabel yang signifikan akan mempengaruhi variabel terikat.

Tahapan yang harus dilakukan dalam melakukan peramalan adalah sebagai berikut[9]:

- a. Mendefinisikan tujuan
- b. Membuat diagram atau plot data
- c. Memilih model peramalan yang tepat
- d. Melakukan peramalan
- e. Menghitung kesalahan (*error*)
- f. Memilih metode peramalan dengan kesalahan paling kecil
- g. Melakukan verifikasi peramalan

2.2.2. Reksa Dana

Reksa dana merupakan salah satu bentuk investasi berupa wadah yang dapat dipergunakan sebagai penghimpun dana investor. Dana yang ada dalam wadah tersebut akan diinvestasikan dalam bentuk portofolio efek oleh Manajer Investasi[10].

Reksa dana dapat dibagi menjadi dua bentuk[10]:

- a. Reksa dana berbentuk perseroan

Perusahaan yang kegiatan usahanya menghimpun dana dengan menjual saham, selanjutnya dana tersebut diinvestasikan dalam berbagai jenis efek dan diperdagangkan di Pasar Modal dan Pasar Uang.

b. Reksa dana berbentuk Kontrak Investasi Kolektif (KIK)

Kontrak antara Manajer Investasi dan Bank Kustodian yang mengikat pemegang Unit Penyertaan. Manajer Investasi memiliki wewenang untuk mengelola portofolio investasi kolektif dan Bank Kustodian memiliki wewenang untuk melaksanakan penitipan kolektif.

Macam-macam reksa dana adalah sebagai berikut[10]:

a. Reksa dana pasar uang (*Money Market Funds*)

Reksa dana yang hanya melakukan investasi pada efek bersifat utang dengan jatuh tempo kurang dari satu tahun. Reksa dana ini mengutamakan investasi pada jenis-jenis efek di Pasar Uang dengan orientasi pendapatan jangka pendek.

b. Reksa dana pendapatan tetap (*Fixed Income Funds*)

Reksa dana yang melakukan investasi sekurang-kurangnya 80% dari aktivitya dalam bentuk efek bersifat utang. Reksa dana ini mengkhususkan pada efek yang memberikan pendapatan secara tetap.

c. Reksa dana saham (*Growth Funds*)

Reksa dana yang melakukan investasi sekurang-kurangnya 80% dari aktivitya dalam efek bersifat ekuitas (Contoh: saham). Reksa dana ini mengupayakan untuk memperoleh *capital gain* dalam jangka panjang. Beberapa variabel yang mempengaruhi pergerakan saham adalah variabel *open*, *high*, *low*, *close* Indeks Harga Saham Gabungan (IHSG), nilai kurs jual dan beli IDR-USD[11].

d. Reksa dana campuran (*Balanced Funds*)

Reksa dana yang melakukan investasi dalam efek bersifat ekuitas dan efek bersifat utang yang perbandingan tidak termasuk poin b dan c di atas. Reksa dana ini mengutamakan penganekaragaman jenis efek proporsi yang seimbang antara efek ekuitas dan efek utang.

2.2.3. Nilai Aktiva Bersih Per Unit Penyertaan (NAB/UP)

NAB/UP adalah harga wajar dari portofolio suatu reksa dana setelah dikurangi biaya operasional kemudian dibagi jumlah saham/unit penyertaan yang telah beredar (dimiliki investor) pada saat itu. Nilai ini berubah-ubah setiap harinya (hari transaksi Senin – Jumat). Nilai ini dipengaruhi oleh transaksi pembelian dan penjualan reksa dana oleh para investor, harga pasar dari aset reksa dana dan perubahan dana kelolaan (Asset Under Management – AUM)[2].

$$NAB/UP = \frac{NAB_1 - NAB_0 - \text{Biaya Operasional} + \text{Distribusi Pendapatan}}{\text{Jumlah Unit Penyertaan yang dimiliki}} \quad (2.1)$$

NAB_1 menyatakan harga reksa dana pada saat sekarang, NAB_0 menyatakan harga reksa dana pada saat awal, Distribusi Pendapatan bisa berasal dari dividen saham, dan sebagainya.

2.2.4. Neural Network

Jaringan Syaraf Tiruan (*Artificial Neural Network*) atau yang dikenal dengan istilah *Neural Network* (NN) merupakan sistem pemroses informasi yang memiliki karakteristik mirip dengan syaraf biologi, dimana dalam memproses informasi, otak manusia terdiri dari sejumlah neuron yang melakukan fungsi pemrosesan cukup kompleks. Pemrosesan informasi pada manusia bersifat adaptif, yang artinya hubungan antar neuron terjadi secara dinamis dan selalu memiliki kemampuan untuk mempelajari informasi-informasi yang belum diketahui sebelumnya[12].

Secara garis besar pada NN memiliki dua tahap pemrosesan informasi, yaitu tahap pelatihan dan tahap pengujian. Tahap pelatihan dimulai dengan memasukkan pola-pola belajar (data latih) ke dalam jaringan. Dengan menggunakan pola-pola ini, jaringan akan mengubah-ubah bobot yang menjadi penghubung antara *node*. Pada setiap iterasi dilakukan evaluasi terhadap output jaringan. Tahap ini berlangsung pada beberapa iterasi dan berhenti setelah jaringan menemukan bobot yang sesuai dimana nilai *error* yang diinginkan telah tercapai atau jumlah iterasi telah mencapai nilai maksimal yang ditetapkan. Selanjutnya bobot ini menjadi *knowledge base* (dasar pengetahuan) pada tahap pengenalan. Sedangkan pada tahap pengujian dilakukan pengujian terhadap suatu pola masukan yang belum pernah dilatihkan sebelumnya (*data testing*) dengan menggunakan bobot hasil tahap pelatihan.

Diharapkan bobot-bobot hasil pelatihan yang sudah menghasilkan error minimal juga akan memberikan *error* yang kecil pada tahap pengujian[13].

Arsitektur jaringan dalam NN merupakan susunan dari neuron-neuron dalam lapisan dan pola keterhubungannya dalam dan antar lapisan. Neuron-neuron yang berada dalam satu lapisan tertentu akan mempunyai pola keterhubungan yang sama. Sebuah NN biasanya terdiri atas lapisan-lapisan antara lain lapisan input (*input layer*) yaitu lapisan yang menerima masukan atau input dari jaringan luar, lapisan tersembunyi (*hidden layer*) yaitu lapisan yang terletak dalam satu atau beberapa lapisan dan tidak berhubungan langsung dengan keadaan diluar jaringan serta lapisan *output* (*output layer*) yaitu lapisan yang menghasilkan *output* dari jaringan. Berdasarkan jumlah *layer* yang dimiliki, NN dibedakan menjadi jaringan lapisan tunggal (*single layer*) dan jaringan multilapis (*multilayer*). Pada *single layer*, jaringan tersebut tidak mempunyai lapisan tersembunyi dari neuron atau hanya mempunyai satu lapisan bobot koneksi serta sekumpulan input neuron dihubungkan langsung dengan sekumpulan *output*. Sedangkan pada multilayer terdiri dari lapisan input, lapisan tersembunyi dan lapisan output. Lapisan tersembunyi terletak diantara lapisan input dan lapisan output. *Output* dari sebuah lapisan tersembunyi akan menjadi input bagi lapisan berikutnya. Jaringan ini minimal mempunyai satu lapisan tersembunyi[14].

2.2.5. General Regression Neural Network (GRNN)

GRNN adalah sejenis jaringan fungsi dasar radial (RBF) yang didasarkan pada teknik statistik standar yang disebut regresi

kernel. GRNN memiliki kinerja yang sangat baik pada kemampuan perkiraan dan kecepatan belajar, dan kecepatan tersebut adalah pembelajaran cepat dan konvergensi ke regresi yang optimal di permukaan karena jumlah data sampel menjadi sangat besar. Ketika jumlah data sampel kecil, GRNN masih memiliki hasil yang baik dalam peramalan[15]. GRNN memiliki kemampuan pemetaan non-linier yang kuat, kesederhanaan struktur jaringan dan toleransi kesalahan yang tinggi. Oleh sebab itu GRNN dapat secara efektif menyelesaikan masalah non-linear, dan telah secara luas diterapkan ke berbagai bidang termasuk pengenalan pola, peramalan beban jangka pendek, memodelkan dan memonitor proses batch, sistem kendali TWUSM, kimia obat, desulfurisasi batubara, peramalan nilai tukar, peramalan penjualan, peramalan kecepatan angin, dan sebagainya. Namun, kelemahan dari penerapan model GRNN adalah sangat sulit untuk memilih parameter penyebaran (*spread*) dengan benar[15].

Fungsi utama dari GRNN adalah untuk mengestimasi permukaan regresi linier maupun non-linier pada variabel bebas (*input vector*) $X = [x_1, x_2, \dots, x_n]^T$, diberikan variabel terikat (*output vector*) $Y = [y_1, y_2, \dots, y_k]^T$. Prosedur model GRNN bisa ditunjukkan sebagai berikut.

$$E[Y|X] = \frac{\int_{-\infty}^{\infty} Y f(Y, X) dX}{\int_{-\infty}^{\infty} f(Y, X) dX} \quad (2.2)$$

dimana X adalah input vektor dimensi, Y adalah nilai model GRNN, $E[Y|X]$ adalah nilai yang diharapkan dari output Y, diberikan vektor input X, $f(Y, X)$ adalah fungsi kepadatan probabilitas gabungan dari X dan Y.

GRNN diatur menggunakan empat lapisan: lapisan input (*input layer*), lapisan pola (*pattern layer*), lapisan penjumlahan (*summation layer*), dan lapisan keluaran (*output layer*). Lapisan input menerima informasi dan menyimpan input vektor X , yang jumlah neuronnya sama dengan dimensi input vektor. Kemudian, neuron input dari layer input memberi data ke lapisan pola. Lapisan pola memiliki trans-formasi non-linear dari ruang input ke ruang pola. Neuron di lapisan pola (juga disebut pola neuron) dapat mengingat hubungan antara neuron input dan respons yang tepat dari lapisan pola, dan jumlah neuron sama dengan jumlah variabel input saat itu. Pola fungsi Gaussian dapat disajikan sebagai berikut.

$$p_i = \exp \left[-\frac{(X-X_i)^T(X-X_i)}{2\sigma^2} \right] \quad (i = 1, 2, \dots, n) \quad (2.3)$$

dimana σ menunjukkan parameter pemulusan (*smoothing parameter*), X adalah variabel input dari jaringan, X_i adalah vektor pelatihan spesifik dari neuron i pada lapisan pola.

Lapisan penjumlahan memiliki dua penjumlahan, yaitu S_s dan S_w . Penjumlahan sederhana S_s menghitung jumlah aritmatika dari output lapisan pola, dan bobot interkoneksi sama dengan 1. Penjumlahan berbobot S_w menghitung penjumlahan berbobot dari keluaran lapisan pola, dan bobot interkoneksi w . Fungsi transfer dapat direpresentasikan sebagai berikut.

$$S_s = \sum_1^i p_i \quad (2.4)$$

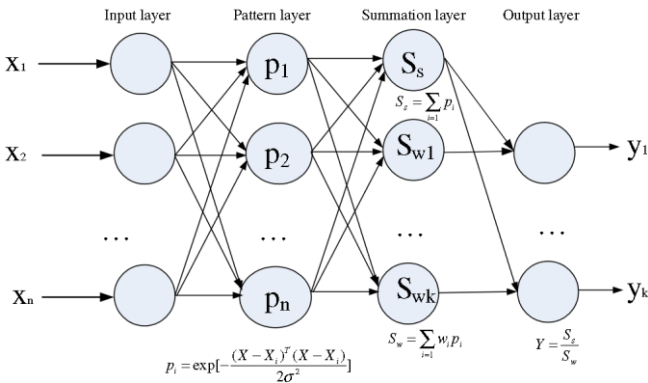
$$S_w = \sum_1^i w_i p_i \quad (2.5)$$

dimana w_i merupakan bobot dari jaringan pola i yang terhubung dengan lapisan penjumlahan.

Jumlah neuron pada layer output sama dengan dimensi k dari keluaran vektor Y . Setelah penjumlahan neuron pada lapisan penjumlahan dimasukkan ke dalam lapisan keluaran, keluaran Y dari model GRNN dapat dihitung sebagai berikut.

$$Y = S_s / S_w \quad (2.6)$$

Oleh karena itu, model GRNN hanya memiliki satu parameter σ yang perlu ditentukan, yang sangat penting dalam menggunakan GRNN untuk peramalan. Parameter σ (juga disebut '*spread*' dalam program Matlab) menentukan kemampuan generalisasi GRNN. Banyak peneliti memilih parameter berdasarkan pengetahuan sebelumnya atau pengalaman individu, yang mungkin tidak efisien untuk peramalan. Oleh karena itu, kita harus mengembangkan metode yang efisien secara otomatis untuk memilih parameter *spread* yang sesuai dalam model GRNN[15]. Arsitektur GRNN dapat dilihat pada Gambar 2.1.



Gambar 2.1 Arsitektur GRNN

Untuk mencapai tujuan ini, tugas akhir ini menggunakan algoritma optimisasi lalat buah (FOA) untuk secara otomatis menentukan nilai *spread parameter* dari model GRNN.

2.2.6. Double Moving Average

Metode peramalan *Double Moving Average* merupakan metode peramalan yang dilakukan dengan cara mengambil sekelompok nilai dari sebuah data dan mencari rata-rata dari data tersebut, misalnya adalah data rata-rata 3 mingguan, atau 2 bulanan. Perataan tersebut dilakukan sebanyak dua kali. Metode peramalan *Double Moving Average* sangat cocok digunakan untuk data yang dipengaruhi oleh tren dan untuk melakukan peramalan jangka pendek.

Caranya adalah dengan menghitung rata-rata bergerak menggunakan rumus di bawah ini.

$$M_t = \frac{Y_t + Y_{t-1} + \dots + Y_{t-n+1}}{n} \quad (2.7)$$

Y_t adalah data pada waktu ke t . Sedangkan n adalah jumlah data yang akan dilakukan peramalan.

Langkah selanjutnya adalah menghitung rata-rata bergerak kedua menggunakan rumus di bawah ini.

$$M'_t = \frac{M_t + M_{t-1} + \dots + M_{t-n+1}}{n} \quad (2.8)$$

M_t adalah rata-rata pertama pada waktu ke t . Sedangkan n adalah jumlah data yang akan dilakukan peramalan.

Setelah itu, hitung *intercept* (a) dan *slope* (b) dengan menggunakan rumus di bawah ini.

$$a_t = M_t + (M_t - M'_t) = 2M_t - M'_t \quad (2.9)$$

$$b_t = \frac{2}{n-1}(M_t - M'_t) \quad (2.10)$$

a_t adalah *intercept* pada waktu ke t , sedangkan b_t adalah *slope* (b) pada waktu ke t .

Langkah terakhir adalah melakukan perhitungan peramalan dengan menggunakan rumus di bawah ini.

$$Y_{t+1} = a_t + b_t \quad (2.11)$$

a_t adalah *intercept* pada waktu ke t , sedangkan b_t adalah *slope* (b) pada waktu ke t .

2.2.7. Double Exponential Smoothing Holt

Metode peramalan *Double Exponential Smoothing Holt* merupakan metode peramalan yang cocok digunakan untuk data yang memiliki tren, tidak memiliki pola data musiman, dan untuk melakukan peramalan jangka pendek. *Double Exponential Smoothing Holt* merupakan pengembangan dari *Single Exponential Smoothing* untuk bisa digunakan meramalkan data yang memiliki tren. Metode peramalan *Double Exponential Smoothing Holt* menggunakan 2 konstanta yang bernilai antara 0 sampai 1. Untuk melakukan peramalan menggunakan metode peramalan *Double Exponential Smoothing Holt*, yang pertama dilakukan adalah menghitung nilai level dapat menggunakan rumus di bawah ini.

$$L_t = \alpha Y_t + (1 - \alpha)(L_{t-1} + b_{t-1}) \quad (2.12)$$

α adalah konstanta yang digunakan untuk menghitung level pada *Double Exponential Smoothing Holt*. α memiliki nilai antara 0 sampai 1. Y_t adalah data aktual pada waktu ke t . L_{t-1}

adalah nilai level pada waktu $t-1$. b_{t-1} adalah nilai tren pada waktu $t-1$. Pada tahap awal, L_1 diinisiasi dengan nilai sama dengan Y_1 .

Langkah selanjutnya adalah menghitung nilai tren. Untuk menghitung nilai tren dapat menggunakan rumus di bawah ini.

$$b_t = \beta(L_t - L_{t-1}) + (1 - \beta)b_{t-1} \quad (2.13)$$

β adalah konstanta yang digunakan untuk menghitung tren pada *Double Exponential Smoothing Holt*. L_t adalah nilai level pada waktu ke t . L_{t-1} adalah nilai level pada waktu $t-1$. b_{t-1} adalah nilai tren pada waktu $t-1$. Pada tahap awal, b_1 diinisiasi dengan nilai sama dengan $Y_2 - Y_1$.

Langkah terakhir adalah menghitung nilai peramalan. Untuk menghitung nilai peramalan dapat menggunakan rumus di bawah ini.

$$F_{t+1} = L_t + b_t \quad (2.14)$$

L_t adalah nilai *level* pada waktu ke t , sedangkan b_t adalah *slope* (b) pada waktu ke t .

2.2.8. Double Exponential Smoothing Brown

Metode peramalan *Double Exponential Smoothing Brown* merupakan metode peramalan yang cocok digunakan untuk data yang memiliki tren, tidak memiliki pola data musiman, dan untuk melakukan peramalan jangka pendek. *Double Exponential Smoothing Brown* merupakan metode peramalan yang mirip dengan *Single Exponential Smoothing*. Perbedaannya terletak pada konstanta smoothing *Double Exponential Smoothing* yang diperoleh dari hasil “re-

smoothing” konstanta *Single Exponential Smoothing*. *Double Exponential Smoothing Brown* digunakan untuk data yang menunjukkan tren linier dari waktu ke waktu.

Untuk melakukan peramalan menggunakan metode peramalan *Double Exponential Smoothing Brown* yang pertama kali dilakukan adalah menghitung *Exponential Smoothing* pertama. Untuk menghitung *Exponential Smoothing* pertama dapat menggunakan rumus di bawah ini.

$$L_t = \alpha Y_t + (1 - \alpha)L_{t-1} \quad (2.15)$$

α adalah konstanta yang digunakan untuk menghitung level pada *Double Exponential Smoothing Brown*. α memiliki nilai antara 0 sampai 1. Y_t adalah data aktual pada waktu ke t . L_{t-1} adalah nilai level pada waktu $t-1$. Pada tahap awal, L_1 diinisiasi dengan nilai sama dengan Y_1 .

Langkah selanjutnya adalah menghitung nilai *Exponential Smoothing* kedua. Untuk menghitung *Exponential Smoothing* kedua dapat menggunakan rumus di bawah ini.

$$L'_t = \alpha L_t + (1 - \alpha)L'_{t-1} \quad (2.16)$$

α adalah konstanta yang digunakan untuk menghitung level pada *Double Exponential Smoothing Brown*. α memiliki nilai antara 0 sampai 1. L_t adalah nilai level pada waktu ke t . L'_{t-1} adalah nilai level pada waktu $t-1$.

Langkah selanjutnya adalah menghitung komponen A. Untuk menghitung komponen A dapat menggunakan rumus di bawah ini.

$$a_t = 2L_t - L'_t \quad (2.17)$$

L_t adalah nilai level pada waktu ke t . L'_t adalah nilai level pada waktu t .

Selanjutnya adalah menghitung komponen B. Untuk menghitung komponen B dapat menggunakan rumus di bawah ini.

$$b_t = \frac{\alpha}{1-\alpha}(L_t - L'_t) \quad (2.18)$$

a adalah konstanta yang digunakan untuk menghitung level pada *Double Exponential Smoothing Brown*. a memiliki nilai antara 0 sampai 1. L_t adalah nilai level pada waktu ke t . L'_t adalah nilai level pada waktu t .

Langkah terakhir adalah menghitung peramalan menggunakan metode *Double Exponential Smoothing Brown*. Untuk menghitung peramalan dapat menggunakan rumus di bawah ini.

$$F_{t+1} = a_t + b_t \quad (2.19)$$

a_t adalah nilai komponen A pada waktu ke t . b_t adalah nilai komponen B pada waktu ke t .

2.2.9. Fruit Fly Optimization Algorithm (FOA)

FOA adalah metode kecerdasan berkelompok baru, yang diusulkan oleh Pan[16], dan termasuk dalam jenis komputasi evolusi interaktif. FOA adalah metode baru untuk menemukan optimisasi global berdasarkan perilaku mencari makan lalat buah. Lalat buah adalah sejenis serangga, yang hidup di zona iklim sedang dan tropis dan makan buah busuk. Lalat buah lebih unggul dari spesies lain dalam penglihatan dan *osphresis* (penciuman). Proses pencarian makanan lalat buah adalah

sebagai berikut: pertama, mencium bau makanan oleh organ *osphresis*, dan terbang ke lokasi itu; kemudian, setelah mendekati lokasi makanan, penglihatan sensitif juga digunakan untuk menemukan makanan dan lokasi berkelompok lalat buah lainnya, dan ia terbang ke arah itu[16].

Menurut karakteristik penemuan makanan dari kawanan lalat buah, FOA dapat dibagi menjadi beberapa langkah seperti berikut:

1. Inisialisasi Parameter

Parameter utama dari FOA adalah jumlah maksimal iterasi (*maxgen*), ukuran populasi (*sizepop*), lokasi awal kawanan lalat buah (*X_axis*, *Y_axis*), dan jarak terbang acak (*FR*).

2. Inisialisasi Populasi

Berikan arah terbang secara acak dan jarak untuk pencarian makanan oleh individu lalat buah dengan menggunakan *osphresis*.

$$X_i = X_axis + \text{Random Value} \quad (2.20)$$

$$Y_i = Y_axis + \text{Random Value} \quad (2.21)$$

3. Evaluasi Populasi

Pada awalnya, jarak lokasi makanan ke titik awal (*Dist*) harus dihitung. Selanjutnya, nilai penilaian konsentrasi bau (*S*) harus dihitung, dan nilai tersebut adalah kebalikan dari jarak lokasi makanan ke titik awal (*Dist*).

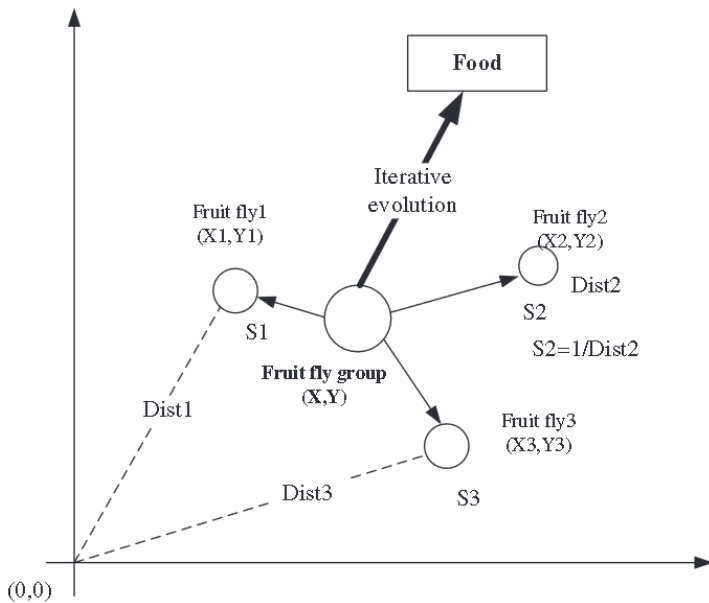
$$Dist_i = \sqrt{X_i^2 + Y_i^2} \quad (2.22)$$

$$S_i = \frac{1}{Dist_i} \quad (2.23)$$

Kemudian, hitung konsentrasi bau ($Smell_i$) individu lalat buah dengan mengganti nilai penilaian konsentrasi bau (S_i) ke fungsi penilaian konsentrasi bau ($Fitness Function$). Setelah itu, cari individu lalat buah dengan konsentrasi bau paling tinggi (nilai maksimal $Smell_i$) di antara kawan lalat buah.

$$Smell_i = Function(S_i) \quad (2.24)$$

$$[bestSmell \ bestIndex] = \max (Smell_i) \quad (2.25)$$



Gambar 2.2 Proses Iterasi Pencarian Makanan oleh Kawan Lalat Buah

4. Operasi Pemilihan

Pertahankan nilai konsentrasi bau maksimal, titik koordinat x dan y . Kemudian, kawan lalat buah akan terbang menuju

lokasi dengan nilai konsentrasi bau maksimal menggunakan penglihatan. Masukkan optimisasi berulang untuk mengulangi implementasi langkah 2–3. Ketika konsentrasi bau tidak lebih tinggi dari konsentrasi bau iterasi sebelumnya atau jumlah iterasi mencapai angka iterasi maksimal, iterasi akan berhenti.

$$Smellbest = bestSmell \quad (2.26)$$

$$X_axis = X(bestIndex) \quad (2.27)$$

$$Y_axis = Y(bestIndex) \quad (2.28)$$

2.2.7. FOA Untuk Pemilihan Parameter Pada GRNN (GRNN-FOA)

Memilih parameter ‘*spread*’ merupakan hal yang sangat penting. Dalam tugas akhir ini, FOA digunakan untuk memilih nilai parameter ‘*spread*’ yang sesuai dari model GRNN agar secara efektif mampu meningkatkan akurasi peramalan.

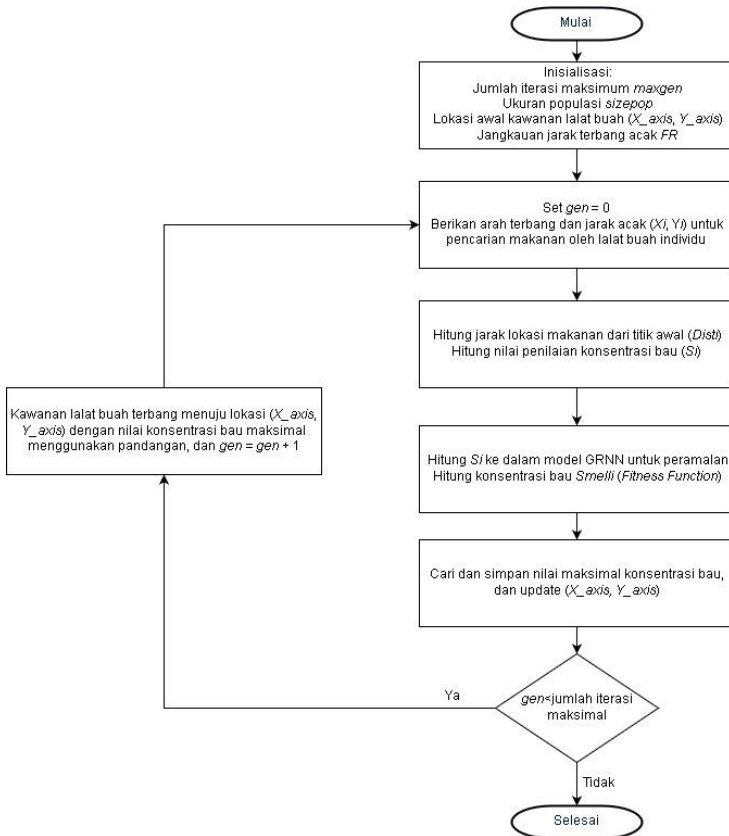
Rincian pemodelan GRNN-FOA dapat ditunjukkan sebagai berikut:

1. Inisialisasi Parameter

Jumlah iterasi maksimum *maxgen*, ukuran populasi *sizepop*, lokasi awal kawanan lalat buah (*X_axis*, *Y_axis*), dan rentang jarak terbang acak *FR* harus ditentukan pada awalnya.

2. Evolusi dimulai

Masukkan nilai *gen* = 0, dan berikan arah terbang acak serta jarak untuk penemuan makanan oleh individu lalat buah.



Gambar 2.3 Flowchart Model GRNN-FOA

3. Penghitungan Awal

Hitung jarak penerbangan $Dist_i$ dari pencarian makanan oleh lalat buah, dan kemudian hitung nilai penilaian konsentrasi bau S_i . Masukkan S_i ke dalam model GRNN sebagai paramater *spread* (σ) untuk peramalan. Berdasarkan hasil peramalan, hitunglah konsentrasi bau $Smelli_i$ (Fitness

Function). $Smell_i$ menggunakan *Root Mean Squared Error* (RMSE) yang mengukur deviasi antara nilai peramalan dan nilai aktual.

4. Generasi Keturunan

Masukkan hasil keturunan ke model GRNN dan hitung kembali nilai konsentrasi bau. Masukkan $gen = gen + 1$.

5. Iterasi Berhenti

Ketika gen mencapai jumlah iterasi maksimal, kriteria terpenuhi, dan nilai parameter GRNN didapatkan maka iterasi dihentikan. Apabila belum, kembali ke langkah ke-2.

2.2.8. Uji Akurasi Performa Model Peramalan

Evaluasi model diperlukan untuk mengukur ketepatan model yang ditemukan. Pengukuran akurasi dapat menggunakan beberapa metode. Metode yang digunakan untuk mengukur akurasi model peramalan dalam tugas akhir ini, yaitu:

a. *Root Mean Square Error* (RMSE)

RMSE merupakan nilai rata-rata dari jumlah kuadrat kesalahan. Juga dapat menyatakan ukuran besarnya kesalahan yang dihasilkan oleh suatu model peramalan. Nilai RMSE rendah menunjukkan bahwa hasil peramalan mendekati nilai aktual.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y'_i - y_i)^2} \quad (2.29)$$

dimana n menyatakan banyaknya data, y'_i menyatakan nilai aktual, y_i menyatakan nilai peramalan.

b. *Mean Absolute Percentage Error* (MAPE)

MAPE dihitung dengan menggunakan kesalahan absolut pada tiap periode dibagi dengan nilai observasi yang nyata untuk periode itu. Kemudian, merata-rata kesalahan persentase absolut tersebut. Pendekatan ini berguna ketika ukuran atau besar variabel ramalan itu penting dalam mengevaluasi ketepatan ramalan. MAPE mengindikasikan seberapa besar kesalahan dalam meramal yang dibandingkan dengan nilai nyata. Nilai MAPE rendah menunjukkan bahwa hasil peramalan mendekati nilai aktual.

$$MAPE = \left(\frac{1}{n} \sum \frac{|data\ aktual - data\ peramalan|}{|data\ aktual|} \right) * 100 \quad (2.30)$$

dimana n menyatakan banyaknya data yang dimiliki.

Adapun standar minimal untuk mengetahui kinerja model peramalan dapat dilihat pada tabel di bawah ini [17].

Tabel 2.2 Justifikasi nilai MAPE

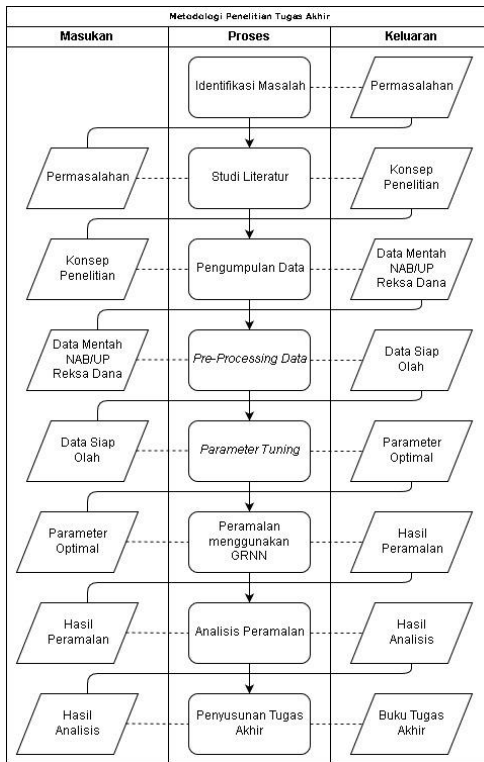
MAPE	Justifikasi
<10%	Kemampuan peramalan sangat baik
10 –20%	Kemampuan peramalan baik
20 –50%	Kemampuan peramalan layak/ memadai
>50%	Kemampuan peramalan buruk

Halaman ini sengaja dikosongkan

BAB III METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan mengenai Metodologi Penelitian terkait tugas akhir yang dikerjakan oleh Penulis.

3.1. Diagram Metodologi



Gambar 3.1 Diagram Metodologi Penelitian Tugas Akhir

Gambar 3.1 merupakan diagram metodologi penelitian tugas akhir yang akan digunakan dalam pengerjaan tugas akhir.

3.2. Identifikasi Masalah

Pada tahap ini akan dilakukan identifikasi mengenai permasalahan yang akan diselesaikan pada tugas akhir. Permasalahan tersebut adalah peramalan data NAB/UP reksa dana saham dalam bentuk harian. Studi kasus yang digunakan adalah reksa dana Simas Saham Unggulan. Data tersebut menunjukkan pola yang fluktuatif, *non-linear*, serta tidak stasioner. Variabel yang mempengaruhi NAB/UP reksa dana Simas Saham Unggulan adalah variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD.

3.3. Studi Literatur

Pada tahap ini akan dipelajari beberapa literatur yang mendukung pengerjaan tugas akhir, yaitu *paper* dari jurnal–jurnal lokal maupun internasional yang membahas mengenai topik terkait tugas akhir ini. Dari beberapa literatur yang ada, GRNN merupakan salah satu metode yang dapat digunakan untuk meramalkan data NAB/UP reksa dana saham. Metode ini dapat menghasilkan peramalan yang baik apabila penentuan parameter GRNN dapat dilakukan dengan tepat. Tahap ini akan menentukan metode penelitian yang digunakan dalam tugas akhir, yaitu penentuan parameter GRNN yang paling optimal menggunakan *Fruit Fly Optimization Algorithm* (FOA).

3.4. Pengumpulan Data

Data yang dikumpulkan adalah data NAB/UP reksa dana Simas Saham Unggulan dalam bentuk harian dan data variabel yang berpengaruh terhadap NAB/UP reksa dana saham. Terdapat beberapa variabel yang dapat mempengaruhi NAB/UP reksa dana saham, yaitu variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD. Variabel tersebut dipilih dikarenakan Simas Saham Unggulan mengalokasikan 99,13% dana yang dimiliki ke dalam pasar saham. Sedangkan pengaruh harga saham oleh variabel tersebut sangat besar.

3.5. Pre-Processing Data

Pada tahap ini akan dilakukan pemisahan data menjadi data *training* dan data *testing*. Data yang dibagi menjadi data *training* dan data *testing* adalah data yang akan digunakan untuk mencari model FOAGRNN dengan hasil peramalan paling optimal. Data yang akan digunakan untuk pencarian model GRNN yang optimal adalah data reksa dana Simas Saham Unggulan. Pembagian data *training* dan *testing* dalam tugas akhir ini adalah 75% data *training* dan 25% data *testing*.

3.6. Parameter Tuning

Pada tahapan ini akan dilakukan pencarian parameter GRNN paling optimal, yaitu nilai σ (*spread parameter*) menggunakan algoritma *Fruit Fly Optimization Algorithm* (FOA). Penentuan nilai σ akan mempengaruhi keakuratan peramalan menggunakan metode GRNN secara signifikan. Algoritma yang digunakan dalam FOA akan diinisiasi dengan beberapa

parameter yang ada dalam FOA, yaitu jumlah generasi dan jumlah populasi dari lalat buah. Nilai σ yang didapatkan dari algoritma FOA ini akan dimasukkan ke dalam model peramalan GRNN untuk digunakan pada tahap selanjutnya. Nilai σ optimal yang akan digunakan untuk melakukan peramalan menggunakan metode GRNN adalah nilai σ yang memiliki RMSE serta MAPE terkecil.

3.7. Peramalan Menggunakan GRNN

Pada tahap ini dilakukan peramalan NAB/UP reksa dana saham menggunakan metode GRNN. Peramalan dilakukan dengan masukan data *training* serta parameter σ yang telah ditetapkan pada tahap *pre-processing* data. Selanjutnya model peramalan akan divalidasi menggunakan data *testing*. Apabila menghasilkan performa yang baik, maka model tersebut yang akan digunakan untuk meramalkan masa mendatang. Peramalan masa mendatang yang dilakukan berupa peramalan harian untuk satu bulan kedepan yaitu bulan Januari 2019.

3.8. Analisis Peramalan

Setelah dilakukan peramalan, maka akan dilakukan uji akurasi performa model peramalan dengan menggunakan metode pengukuran performa hasil peramalan. Adanya pengukuran performa ini dapat memberitahukan tingkat keakuratan dari hasil peramalan yang telah dilakukan menggunakan model peramalan GRNN. Analisis performa ini nantinya akan menggunakan *Mean Absolute Percentage Error* (MAPE) dan *Root Mean Squared Error* (RMSE) untuk mengetahui akurasi dari peramalan setiap data. Analisis peramalan akan dilakukan

terhadap hasil peramalan data *training*, data *testing*, serta data peramalan satu bulan kedepan.

3.9. Penyusunan Tugas Akhir

Pada tahap ini akan dilakukan penyusunan laporan tugas akhir yang terdiri dari seluruh aktivitas dalam pengerjaan tugas akhir. Laporan tugas akhir akan disusun mengikuti format penulisan yang telah ditetapkan laboratorium Rekayasa Data dan Inteligensi Bisnis (RDIB) serta Departemen Sistem Informasi (DSI) ITS.

Format yang akan digunakan adalah sebagai berikut:

a. Bab I Pendahuluan

Pada bab ini akan dijelaskan mengenai Latar Belakang Masalah, Perumusan Masalah, Batasan Masalah, Tujuan Penelitian, Manfaat Penelitian, serta Relevansi tugas akhir dengan laboratorium.

b. Bab II Tinjauan Pustaka

Pada bab ini akan dijelaskan mengenai Studi Sebelumnya serta Dasar Teori terkait tugas akhir yang dikerjakan oleh Penulis.

c. Bab III Metodologi

Pada bab ini akan dijelaskan mengenai Metodologi Penelitian terkait tugas akhir yang dikerjakan oleh Penulis.

d. Bab IV Perancangan

Pada bab ini akan dijelaskan mengenai perancangan aktivitas yang akan digunakan untuk diimplementasikan dalam tugas akhir. Bab ini berisikan pengumpulan data, pre-processing

data, pembuatan model, proses peramalan data, dan analisis performa model.

e. Bab V Implementasi

Pada bab ini akan dijelaskan mengenai implementasi yang dilakukan pada tugas akhir untuk menjawab permasalahan yang telah ditentukan sebelumnya.

f. Bab VI Hasil dan Pembahasan

Pada bab ini akan dijelaskan mengenai hasil dan pembahasan dari implementasi yang dilakukan pada tugas akhir.

g. Bab VII Kesimpulan dan Saran

Pada bab ini akan dijelaskan mengenai kesimpulan dan saran yang ditujukan untuk penyempurnaan tugas akhir.

BAB IV

PERANCANGAN

Pada bab ini akan dijelaskan mengenai perancangan aktivitas yang akan digunakan untuk diimplementasikan dalam tugas akhir. Bab ini berisikan Pengumpulan Data, *Pre-processing* Data, Pembuatan Model, Proses Peramalan Data, dan Analisis Performa Model.

4.1. Pengumpulan Data

Data yang digunakan dalam pengerjaan tugas akhir ini adalah data NAB/UP reksa dana saham Simas Saham Unggulan, variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD. Periode data yang digunakan adalah data harian mulai 4 Januari 2016 sampai 28 Desember 2018. Data NAB/UP reksa dana Simas Saham Unggulan didapatkan dari situs *www.sinarmas-am.co.id*, variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG) didapatkan dari situs *finance.yahoo.com*, dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD didapatkan dari situs *bi.go.id*. Peramalan masa mendatang, yaitu peramalan NAB/UP untuk Bulan Januari 2019 akan dilakukan menggunakan model paling optimal yang didapat dari hasil *training* dan *testing*. Data variabel bebas yang akan digunakan pada peramalan masa mendatang diperoleh dari peramalan menggunakan metode *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*. Metode-metode tersebut dipilih karena cocok untuk digunakan

dalam peramalan data yang memiliki tren, tanpa memiliki musiman. Sama seperti pola data yang ada pada variabel bebas.

4.2. *Pre-Processing Data*

Data yang akan digunakan dibagi menjadi data *training* dan data *testing* dengan pembagian 75% dari total data sebagai data *training* dan 25% dari total data sebagai data *testing*. Dari total 724 data pada masing-masing variabel, maka diperoleh 543 data *training* dan 181 data *testing*. Pembagian *training* dan *testing* ini akan digunakan untuk mencari dan menguji model terbaik dalam melakukan peramalan kedepan.

4.3. *Parameter Tuning*

Parameter yang akan dicari untuk diterapkan ke dalam model GRNN adalah parameter σ (*spread parameter*). Pencarian parameter ini akan menggunakan algoritma optimasi *Fruit Fly Optimization Algorithm* (FOA). Algoritma optimasi FOA memiliki beberapa parameter yang dapat dijelaskan seperti tabel 4.1.

Tabel 4.1 Parameter FOA

Parameter dalam FOA	Keterangan
<i>Gen</i>	Generasi lalat buah
<i>Pop</i>	Populasi lalat buah
<i>X, Y</i>	Posisi lalat buah

Parameter dalam FOA	Keterangan
<i>Dist</i>	Jarak perpindahan posisi lalat buah
<i>Smell</i>	Nilai konsentrasi bau

Tabel 4.1 merupakan tabel berisi penjelasan dari istilah yang digunakan pada algoritma optimasi FOA. Dalam proses optimasi akan dicari satu konsentrasi lalat buah pada tiap populasi dalam tiap generasi. Generasi pada FOA merupakan sebutan untuk jumlah iterasi yang akan dilakukan oleh algoritma tersebut. Jumlah populasi (*pop*) adalah total populasi dalam tiap generasi. Iterasi yang terjadi nantinya adalah mencari satu nilai konsentrasi bau lalat buah dengan makanan terbaik pada tiap populasi dalam tiap generasi yang ditandai dengan *error* terkecil. Pada tugas akhir ini nilai konsentrasi bau yang dicari adalah nilai RMSE serta MAPE terkecil pada hasil peramalan. Inisiasi awal jumlah generasi (*gen*) dalam pengerjaan tugas akhir ini adalah 100 generasi serta jumlah populasi (*pop*) adalah 10 populasi[18].

Dalam pengerjaan tugas akhir ini akan dilakukan uji coba terhadap 10 skenario kombinasi generasi dan populasi, yaitu:

- a) Skenario 100 generasi dan 10 populasi
- b) Skenario 100 generasi dan 50 populasi
- c) Skenario 100 generasi dan 100 populasi
- d) Skenario 200 generasi dan 50 populasi

- e) Skenario 200 generasi dan 100 populasi
- f) Skenario 200 generasi dan 150 populasi
- g) Skenario 200 generasi dan 200 populasi
- h) Skenario 300 generasi dan 300 populasi
- i) Skenario 400 generasi dan 40 populasi
- j) Skenario 500 generasi dan 50 populasi

Pada tiap skenario akan dilakukan percobaan sebanyak 10 kali untuk mencari nilai parameter σ paling optimal. Hal ini dilakukan karena didapatkan bahwa hasil pada tiap percobaan dimungkinkan untuk mendapatkan nilai parameter σ yang berbeda. Perbedaan nilai yang didapatkan disebabkan oleh inisiasi posisi lalat buah (X, Y) bernilai acak menggunakan fungsi `rand()` pada MATLAB. Jumlah percobaan, kombinasi generasi dan populasi didapatkan dari hasil uji coba (*trial and error*).

4.4. Proses Peramalan Data

Proses peramalan data NAB/UP reksa dana saham akan menggunakan model GRNN yang memiliki satu buah parameter, yaitu σ (*spread parameter*). *Spread parameter* yang akan digunakan dalam tugas akhir ini akan dicari menggunakan algoritma optimasi FOA untuk mendapatkan parameter paling optimal. Parameter paling optimal dalam pengerjaan tugas akhir ini adalah parameter yang menghasilkan peramalan dengan akurasi paling tinggi, atau nilai RMSE serta MAPE paling rendah. Parameter yang didapatkan akan diuji pada data *training* untuk mengetahui akurasi peramalan terhadap data *training*. Setelah itu akan

diujikan pada data *testing* untuk mengetahui kehandalan model yang telah didapatkan. Apabila model memiliki akurasi yang baik pada data *testing*, maka model tersebut akan digunakan untuk melakukan peramalan data NAB/UP reksa dana saham Simas Saham Unggulan periode Januari 2019.

4.5. Analisis Performa Model

Analisis performa model akan dilakukan dengan cara menghitung nilai *Mean Absolute Percentage Error* (MAPE) dan *Root Mean Squared Error* (RMSE) terhadap hasil peramalan. Peramalan periode kedepan dapat dilakukan apabila model GRNN dari data *testing* telah memenuhi MAPE terbaik dari seluruh hasil uji coba. Peramalan data NAB/UP reksa dana saham juga akan dilakukan dengan menggunakan metode *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown* untuk mengetahui perbandingan hasil peramalan terhadap metode GRNN.

Halaman ini sengaja dikosongkan

BAB V

IMPLEMENTASI

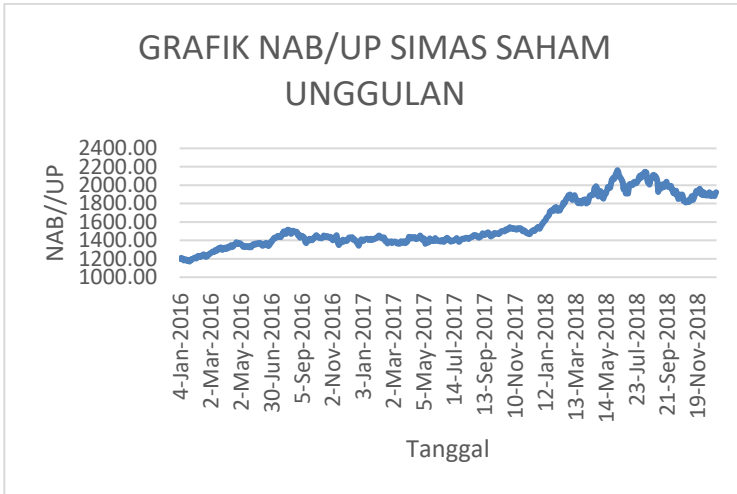
Pada bab ini akan dijelaskan mengenai implementasi yang dilakukan pada tugas akhir untuk menjawab permasalahan yang telah ditentukan sebelumnya.

5.1. Pengumpulan Dan *Pre-processing* Data

Data yang digunakan dalam pengerjaan tugas akhir ini adalah data NAB/UP reksa dana saham Simas Saham Unggulan, variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD. Periode data yang digunakan adalah data harian mulai 4 Januari 2016 sampai 28 Desember 2018. Data NAB/UP reksa dana Simas Saham Unggulan didapatkan dari situs www.sinarmas-am.co.id, variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG) didapatkan dari situs finance.yahoo.com, dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD didapatkan dari situs bi.go.id.

Data akan dibagi menjadi dua, yaitu data *training* dan data *testing*. Pembagian data *training* dan *testing* dalam tugas akhir ini adalah 75% data *training* dan 25% data *testing*. Dari total 724 data pada masing-masing variabel, maka diperoleh 543 data *training* dan 181 data *testing*. Untuk melakukan peramalan masa mendatang, akan digunakan hasil peramalan pada bulan sebelumnya. Untuk variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG) dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD akan didapatkan melalui peramalan menggunakan metode *Double*

Moving Average, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*. Grafik data NAB/UP reksa dana Simas Saham Unggulan dapat dilihat pada gambar di bawah ini.



Gambar 5.1 Grafik NAB/UP Simas Saham Unggulan

Dari Gambar 5.1 dapat dilihat bahwa data memiliki pola yang *non-linier*, memiliki tren dan tidak stasioner. Data menunjukkan NAB/UP reksa dana Simas Saham Unggulan bersifat fluktuatif, atau mengalami kenaikan dan penurunan dari tahun 2016 sampai dengan tahun 2018. Namun apabila dilihat secara jangka panjang, NAB/UP reksa dana Simas Saham Unggulan mengalami kenaikan dari tahun ke tahun. Seluruh data aktual (Data NAB/UP reksa dana Simas Saham Unggulan) yang digunakan dalam tugas akhir ini dapat dilihat pada LAMPIRAN A.

```

5 %load data
6 - load C:\Users\arifc\Documents\MATLAB\Examples\matlab\FOAGRNN\Training.txt;
7 - [row,col]=size(Training);
8 - row1=row/2;
9 - traindata1=Training(1:row1,1:col-1);
10 - traindata2=Training(row1+1:row,1:col-1);
11 - t1=Training(1:row1,col);
12 - t2=Training(row1+1:row,col);
13 - t1=t1';
14 - t2=t2';
15 - tr1=traindata1';
16 - tr2=traindata2';
17 - la=1;

```

Kode Program 5.1 Pre-processing Data

Kode Program 5.1 digunakan untuk melakukan *pre-processing* data. *Pre-processing* data dilakukan dengan cara memisahkan data *training* dan data *testing* yang akan digunakan untuk mencari model terbaik peramalan menggunakan GRNN. Baris ke 6 digunakan untuk melakukan inisialisasi *load* data yang akan digunakan untuk mencari parameter σ paling optimal. Data yang digunakan untuk mencari parameter σ pertama kali adalah data *training*. Baris ke 7 sampai 11 digunakan untuk menginisialisasi banyaknya baris dan kolom dalam data *training*. Pada pencarian parameter σ ini akan dilakukan *cross training* untuk menghindari *overfitting* atau pelatihan data berlebihan untuk mengurangi kesalahan dalam pelatihan data. *Overfitting* adalah keadaan di mana pengenalan data dilakukan secara berlebihan sehingga model yang terbentuk terlalu akurat pada data tersebut. Dampak dari *overfitting* adalah model yang ditemukan dari data *training* tidak dapat digunakan untuk melakukan peramalan terhadap data lain.

5.2. Parameter Tuning

Parameter σ (*spread parameter*) pada GRNN paling optimal akan dicari menggunakan algoritma optimasi FOA. Parameter σ paling optimal adalah yang memiliki MAPE serta RMSE

paling kecil pada hasil peramalan menggunakan metode GRNN. Pencarian parameter σ menggunakan algoritma optimasi FOA akan dijelaskan di bawah ini.

5.2.1. Inisialisasi Parameter FOA

Pada tahap ini akan dilakukan inisialisasi parameter yang ada di FOA untuk mencari parameter σ (*spread parameter*) GRNN. Konfigurasi parameter diinisiasi dengan menentukan jumlah generasi/iterasi (*gen*), jumlah populasi (*pop*), serta lokasi awal lalat buah. Pada tugas akhir ini akan dilakukan inisialisasi sebesar 100 pada jumlah iterasi dan 10 pada jumlah populasi. Angka ini diperoleh dari penelitian sebelumnya yang dilakukan oleh Pan [18] yaitu peramalan pada situs pelelangan menggunakan metode peramalan FOAGRNN. Parameter FOA (jumlah generasi/iterasi dan jumlah populasi) akan dilakukan penggantian angka dengan berbagai kombinasi untuk mendapatkan nilai parameter σ paling optimal yang ditandai dengan nilai RMSE serta MAPE paling kecil atau nilai MAPE kurang dari 10% (kemampuan peramalan sangat baik). Lokasi awal lalat buah akan diinisialisasi menggunakan angka acak.

```

18 %inisialisasi parameter FOA
19 - X_axis=rand();
20 - Y_axis=rand();
21 - maxgen=100;
22 - sizepop=10;

```

Kode Program 5.2 Inisialisasi Parameter FOA

Kode 5.2 merupakan kode program untuk melakukan inisialisasi parameter pada FOA. Baris 19 dan 20 digunakan untuk menginisialisasi lokasi awal lalat buah, yaitu menggunakan angka acak. Pada tugas akhir ini angka acak yang digunakan berasal dari fungsi *rand()* MATLAB yang

akan menghasilkan angka acak berupa bilangan *double* dengan interval 0 sampai 1. Baris ke 21 digunakan untuk menginisialisasi jumlah iterasi yang akan digunakan dalam optimasi FOA. Baris ke 22 digunakan untuk menginisialisasi jumlah populasi atau jumlah lalat buah yang akan disebar dalam satu kali iterasi.

5.2.2. Inisialisasi Populasi FOA

Pada tahap ini akan dilakukan pencarian parameter σ paling optimal oleh lalat buah dengan cara melakukan inisialisasi jarak terbang serta arah terbang lalat buah. Inisialisasi jarak terbang dan arah terbang dilakukan dengan cara menambahkan posisi awal lalat buah dengan angka acak menggunakan fungsi *rand()* pada MATLAB. Fungsi *rand()* pada MATLAB merupakan fungsi yang akan menghasilkan angka acak berupa bilangan *double* dengan interval 0 sampai 1.

```
28 - |      X(i)=X_axis+rand();
29 - |      Y(i)=Y_axis+rand();
```

Kode Program 5.3 Inisialisasi Populasi FOA

Kode 5.3 merupakan kode yang digunakan untuk melakukan inisialisasi populasi FOA. Baris 28 dan 29 digunakan untuk menginisialisasi perpindahan jarak dan arah lalat buah untuk mencari parameter σ paling optimal untuk digunakan dalam melakukan parameter menggunakan metode GRNN.

5.2.3. Evaluasi Populasi FOA

Pada tahap ini akan dilakukan evaluasi populasi FOA terhadap kinerja lalat buah dalam mencari parameter σ paling optimal. Evaluasi ini dilakukan pada tiap iterasi untuk memilih lokasi lalat buah dengan jarak terdekat dengan makanan. Evaluasi

populasi dilakukan dengan cara menghitung jarak lokasi makanan ke titik awal ($Dist$) dan menghitung konsentrasi bau makanan (S).

```

31 - |      Dist(i) = (X(i)^2+Y(i)^2)^0.5;
32 - |      S(i) = 1/Dist(i);
33 - |      g=0;
34 - |      p=S(i);

```

Kode Program 5.4 Evaluasi Populasi FOA

Kode Program 5.4 digunakan untuk melakukan evaluasi populasi FOA. Baris 31 digunakan untuk menghitung jarak lokasi makanan ke titik awal ($Dist_i$). $Dist_i$ dihitung dengan cara menjumlahkan x^2 dengan y^2 dan mengakarkan hasil dari penjumlahan tersebut. Baris 32 digunakan untuk menghitung konsentrasi bau makanan (S_i). S_i dihitung dengan cara membagikan angka 1 dengan bilangan hasil dari $Dist_i$. Hasil dari penghitungan S akan dijadikan sebagai parameter σ pada peramalan GRNN kedepan.

Kode Program 5.5 digunakan untuk melakukan penghitungan *fitness function*. Baris 35 sampai 53 merupakan kode program untuk melakukan *cross training* agar mampu mengurangi *overfitting*. Proses *overfitting* dilakukan dengan cara membagi data *training* menjadi dua bagian, dan mencari model terbaik dari kedua bagian data tersebut. Baris 42 digunakan untuk melakukan penghitungan konsentrasi bau ($Smell_i$). $Smell_i$ dilakukan dengan cara mengganti nilai S_i ke dalam *fitness function*. *Fitness function* yang digunakan dalam tugas akhir ini adalah penghitungan *Root Mean Squared Error* (RMSE).

```

35 -         if la == 1
36 -             net=newgrnn(tr1,t1,p);
37 -             yc=sim(net,tr2);
38 -             y=t2-yc;
39 -             for ii=1:row1
40 -                 g=g+y(ii)^2;
41 -             end
42 -             Smell(i)= (g/row1) ^0.5;
43 -             la=2;
44 -         else
45 -             net=newgrnn(tr2,t2,p);
46 -             yc=sim(net,tr1);
47 -             y=t1-yc;
48 -             for ii=1:row1
49 -                 g=g+y(ii)^2;
50 -             end
51 -             Smell(i)= (g/row1) ^0.5;
52 -             la=1;
53 -         end

```

Kode Program 5.5 Penghitungan Fitness Function

```

56 - [bestSmell, bestindex]=min(Smell);

```

Kode Program 5.6 Pemilihan Parameter Terbaik

Kode Program 5.6 digunakan untuk melakukan pemilihan parameter terbaik pada tiap iterasi. Baris ke 56 digunakan untuk menetapkan nilai RMSE terkecil sebagai *bestSmell* atau bau terbaik serta *bestIndex* atau index terbaik yang akan digunakan sebagai lokasi awal lalat buah untuk iterasi selanjutnya.

5.2.4. Operasi Pemilihan Parameter

Pada tahap ini akan dilakukan operasi pemilihan parameter. Parameter yang akan dicari pada tahap ini adalah parameter σ yang akan digunakan dalam melakukan peramalan GRNN. Setelah ditemukan RMSE terkecil pada tahap sebelumnya, maka posisi atau index dari nilai tersebut akan ditandai sebagai posisi bau terbaik oleh lalat buah. Posisi tersebut akan digunakan sebagai posisi awal lalat buah untuk melakukan

iterasi berikutnya. Lakukan perulangan terhadap langkah-langkah dalam melakukan *parameter tuning* sampai dengan batas akhir iterasi yang telah diinisialisasi sebelumnya.

```
57 - X_axis=X(bestindex);
58 - Y_axis=Y(bestindex);
59 - SpreadP=S(bestindex);
60 - RMSE=bestSmell;
```

Kode Program 5.7 Operasi Pemilihan Parameter

Kode Program 5.7 digunakan untuk melakukan operasi pemilihan parameter terbaik. Parameter yang dipilih dalam tahap ini adalah parameter σ yang akan digunakan dalam melakukan peramalan GRNN. Baris 57 dan 58 digunakan untuk menginisialisasi lokasi lalat buah dengan bau makanan terbaik yang nantinya ditetapkan sebagai lokasi awal lalat buah pada iterasi berikutnya. Baris 59 digunakan untuk menetapkan nilai parameter σ dengan bau terbaik. Baris 60 digunakan untuk menetapkan nilai RMSE terkecil pada suatu iterasi yang nantinya akan digunakan untuk membandingkan dengan RMSE pada iterasi berikutnya.

```
96 - [bestSmell, bestindex]=min(Smell);
97 - if bestSmell<RMSE
98 -     X_axis=X(bestindex);
99 -     Y_axis=Y(bestindex);
100 -     SpreadP=S(bestindex);
101 -     RMSE=bestSmell;
102 - end
```

Kode Program 5.8 Pembandingan Nilai RMSE

Kode Program 5.8 digunakan untuk melakukan pembandingan nilai RMSE pada tiap iterasi. Baris 97 sampai 102 digunakan untuk melakukan pembandingan nilai RMSE yang didapat dari suatu iterasi dengan iterasi sebelumnya. Apabila nilai RMSE

yang didapatkan pada suatu iterasi lebih kecil pada iterasi sebelumnya, maka nilai *bestSmell* dan *bestIndex* akan diganti sesuai dengan nilai yang terdapat pada iterasi sebelumnya. Namun apabila nilai RMSE suatu iterasi lebih besar daripada iterasi sebelumnya, maka nilai *bestSmell* dan *bestIndex* akan tetap diberikan nilai sesuai dengan yang ada dalam iterasi sebelumnya.

```

104 -      yy(gen)=RMSE;
105 -      Xbest(gen)=X_axis;
106 -      Ybest(gen)=Y_axis;
107
108 -      if bestSmell<0.01
109 -          break;
110 -      end

```

Kode Program 5.9 Visualisasi RMSE Dan Posisi Lalat Buah

Kode Program 5.9 digunakan untuk melakukan visualisasi terhadap nilai RMSE yang didapatkan sesuai dengan jumlah iterasi serta visualisasi terhadap perpindahan posisi lalat buah dalam mencari parameter σ paling optimal. Baris 104 digunakan untuk melakukan visualisasi RMSE sesuai dengan jumlah iterasi yang dilakukan. Baris 105 dan 106 digunakan untuk melakukan visualisasi terhadap perpindahan posisi lalat buah pada tiap iterasi. Baris 108 sampai 110 digunakan untuk mengurangi kemungkinan *overfitting* dengan cara membatasi RMSE yang didapat harus kurang dari 1% [18].

Kode Program 5.10 digunakan untuk menampilkan hasil visualisasi RMSE serta posisi lalat buah pada tiap iterasi. Baris 114 sampai 117 digunakan untuk menampilkan hasil visualisasi RMSE pada tiap iterasi. Baris 118 sampai 121 digunakan untuk menampilkan hasil visualisasi posisi lalat

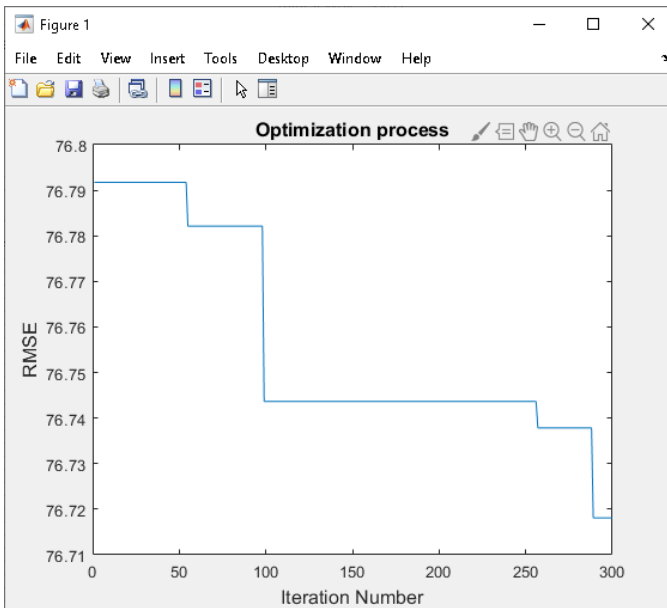
buah pada tiap iterasi. Hasil dari visualisasi RMSE dan posisi lalat buah dapat dilihat pada Gambar 5.2 dan Gambar 5.3.

```

114 - figure(1)
115 - plot(yy)
116 - title('Optimization process','fontsize',12)
117 - xlabel('Iteration Number','fontsize',12);ylabel('RMSE','fontsize',12);
118 - figure(2)
119 - plot(Xbest,Ybest,'b. ');
120 - title('Fruit fly flying route','fontsize',14)
121 - xlabel('X-axis','fontsize',12);ylabel('Y-axis','fontsize',12);

```

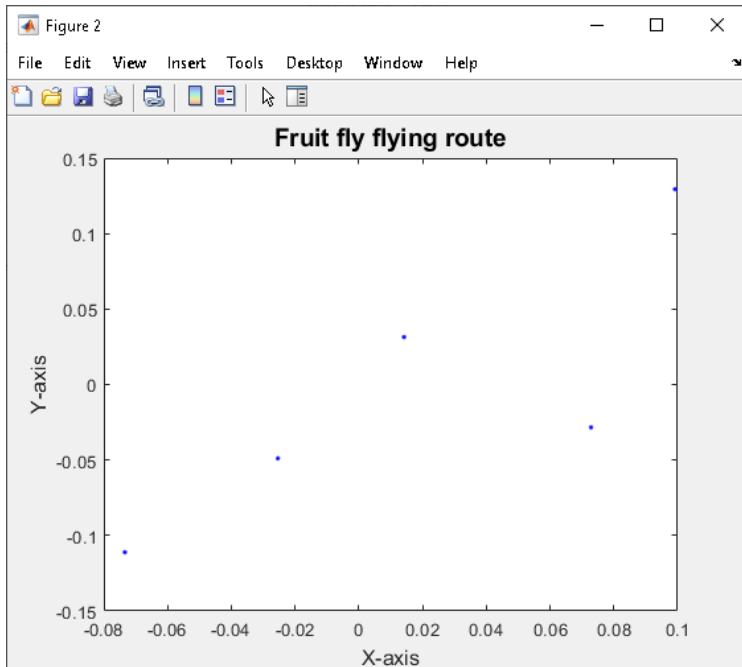
Kode Program 5.10 Hasil Visualisasi RMSE Dan Posisi Lalat Buah



Gambar 5.2 Hasil Visualisasi RMSE

Gambar 5.2 merupakan gambar hasil visualisasi RMSE pada salah satu percobaan pencarian parameter σ paling optimal. Gambar 5.2 menunjukkan bahwa pada percobaan tersebut dilakukan iterasi sebanyak 300 kali. Gambar 5.3 menunjukkan

bahwa pada 300 iterasi tersebut terjadi penurunan RMSE secara bertahap, yaitu pada iterasi ke-60, iterasi ke-100, iterasi ke-260, serta iterasi ke-280.



Gambar 5.3 Hasil Visualisasi Posisi Lalat Buah

Gambar 5.3 menunjukkan hasil visualisasi posisi lalat buah pada salah satu percobaan pencarian parameter σ paling optimal. Gambar 5.3 menunjukkan bahwa terdapat 4 titik dengan lokasi berbeda yang mengindikasikan adanya perpindahan posisi lalat buah sebanyak 4 kali. Perpindahan lokasi lalat buah ini terjadi karena adanya pergantian RMSE pada suatu iterasi dengan nilai RMSE pada iterasi lainnya

dikarenakan nilai RMSE suatu iterasi lebih kecil daripada nilai RMSE iterasi lain.

5.3. Implementasi Model GRNN

Setelah didapatkan parameter σ paling optimal yang ditandai dengan RMSE paling kecil dalam optimasi menggunakan FOA, maka parameter tersebut akan digunakan untuk melakukan peramalan menggunakan metode peramalan *Generalized Regression Neural Networks* (GRNN) pada data *training*, dan data *testing*. Tujuannya adalah untuk melakukan uji akurasi terhadap model yang telah didapatkan.

5.3.1. Load Data Aktual

Pada tahap ini akan dilakukan *load* atau masukan terhadap data aktual yang akan digunakan untuk implementasi model GRNN dengan parameter σ paling optimal. Data aktual yang akan digunakan dalam pada tahap ini adalah data NAB/UP reksa dana saham Simas Saham Unggulan, variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD. Data akan dibagi menjadi dua, yaitu data *training* dan data *testing*. Pembagian data *training* dan *testing* dalam tugas akhir ini adalah 75% data *training* dan 25% data *testing*. Dari total 724 data pada masing-masing variabel, maka diperoleh 543 data *training* dan 181 data *testing*.

Proses *load* data akan dibedakan menjadi 3 macam, yaitu *load* data untuk melakukan peramalan data *training*, *load* data untuk melakukan peramalan data *testing*, dan *load* untuk melakukan peramalan periode kedepan.


```

4   %forecast data training
5 - load C:\Users\arific\Documents\MATLAB\Examples\matlab\FOAGRNN\Training.txt;
6 - [row1,col1]=size(Training);
7 - trainData=Training(1:row1,1:col1-1);
8 - trainTarget=Training(1:row1,col1);
9 - testData=Training(1:row1,1:col1-1);

```

Kode Program 5.11 Load Data Peramalan Data Training

Kode Program 5.11 digunakan untuk melakukan *load* data yang akan digunakan dalam peramalan data *training*. Baris 5 digunakan untuk melakukan *load* data *training* yang terletak pada file *Training.txt*. Baris 6 digunakan untuk melakukan deklarasi jumlah baris dan kolom yang ada dalam file *Training.txt*. Baris 7 digunakan untuk memasukkan variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) pada data *training* ke dalam variabel *trainData*. Baris 8 digunakan untuk memasukkan variabel terikat (variabel NAB/UP reksa dana saham Simas Saham Unggulan) pada data *training* ke dalam variabel *trainTarget*. Baris 9 digunakan untuk memasukkan variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) pada data *training* ke dalam variabel *testData*.

```

11  %forecast data testing
12 - load C:\Users\arific\Documents\MATLAB\Examples\matlab\FOAGRNN\Training.txt;
13 - load C:\Users\arific\Documents\MATLAB\Examples\matlab\FOAGRNN\Testing.txt;
14 - [row1,col1]=size(Training);
15 - [row2,col2]=size(Testing);
16 - trainData=Training(1:row1,1:col1-1);
17 - trainTarget=Training(1:row1,col1);
18 - testData=Testing(1:row2,1:col2-1);
--

```

Kode Program 5.12 Load Data Peramalan Data Testing

Kode Program 5.12 digunakan untuk melakukan *load* data yang akan digunakan dalam peramalan data *testing*. Baris 12

digunakan untuk melakukan *load* data *training* yang terletak pada file *Training.txt*. Baris 13 digunakan untuk melakukan *load* data *testing* yang terletak pada file *Testing.txt*. Baris 14 digunakan untuk melakukan deklarasi jumlah baris dan kolom yang ada dalam file *Training.txt*. Baris 15 digunakan untuk melakukan deklarasi jumlah baris dan kolom yang ada dalam file *Testing.txt*. Baris 16 digunakan untuk memasukkan variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) pada data *training* ke dalam variabel *trainData*. Baris 17 digunakan untuk memasukkan variabel terikat (variabel NAB/UP reksa dana saham Simas Saham Unggulan) pada data *training* ke dalam variabel *trainTarget*. Baris 18 digunakan untuk memasukkan variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) pada data *testing* ke dalam variabel *testData*.

```

20     %forecast periode ke depan
21 -   load C:\Users\arifc\Documents\MATLAB\Examples\matlab\FOAGNN\All.txt;
22 -   load C:\Users\arifc\Documents\MATLAB\Examples\matlab\FOAGNN\Forecast.txt;
23 -   [row3,col3]=size(All);
24 -   [row4,col4]=size(Forecast);
25 -   trainData=All(1:row3,1:col3-1);
26 -   trainTarget=All(1:row3,col3);
27 -   testData=Forecast(1:row4,1:col4);

```

Kode Program 5.13 Load Data Peramalan Periode Kedepan

Kode Program 5.13 digunakan untuk melakukan *load* data yang akan digunakan dalam peramalan periode kedepan. Baris 21 digunakan untuk melakukan *load* semua data yang terletak pada file *All.txt*. File *All.txt* merupakan file yang berisi seluruh variabel yang ada pada data *training* maupun data *testing*. Baris 22 digunakan untuk melakukan *load* data yang terletak

pada file *Forecast.txt*. File *Forecast.txt* merupakan file berisi data yang akan digunakan untuk peramalan periode kedepan. Data tersebut adalah variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) yang didapatkan dari peramalan menggunakan metode *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*. Baris 23 digunakan untuk melakukan deklarasi jumlah baris dan kolom yang ada dalam file *All.txt*. Baris 24 digunakan untuk melakukan deklarasi jumlah baris dan kolom yang ada dalam file *Forecast.txt*. Baris 25 digunakan untuk memasukkan semua variabel bebas yang ada di data *training* dan *testing* ke dalam variabel *trainData*. Baris 26 digunakan untuk memasukkan variabel terikat (variabel NAB/UP reksa dana saham Simas Saham Unggulan) pada data *training* dan *testing* ke dalam variabel *trainTarget*. Baris 27 digunakan untuk memasukkan variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) yang didapatkan dari peramalan menggunakan metode *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*.

5.3.2. Peramalan menggunakan GRNN

Pada tahap ini akan dilakukan peramalan data menggunakan metode peramalan GRNN. Dalam implementasi GRNN, dibutuhkan adanya parameter σ paling optimal untuk meminimalkan kesalahan peramalan (*error*) yang terjadi. Parameter σ paling optimal pada tugas akhir ini akan dicari

menggunakan algoritma optimasi *Fruit Fly Optimization Algorithm* (FOA). Untuk mengukur tingkat akurasi peramalan yang dihasilkan dalam tugas akhir ini, digunakan metode pengukuran *Root Mean Square Error* (RMSE). Semakin kecil nilai RMSE yang didapat maka hasil peramalan akan semakin mendekati nilai aktual/nilai sebenarnya.

```

29 -   t1=trainTarget';
30 -   tr1=trainData';
31 -   tr2=testData';
32
33 -   net=newgrnn(tr1,t1,p);
34 -   yc=sim(net,tr2);
35
36 -   hasil=yc';

```

Kode Program 5.14 Peramalan Menggunakan GRNN

Kode Program 5.14 merupakan kode program yang digunakan untuk melakukan peramalan data menggunakan metode peramalan *Generalized Neural Network* (GRNN). Baris 29 digunakan untuk memasukkan data yang ada di variabel *trainTarget* ke dalam variabel *t1*. Baris 30 digunakan untuk memasukkan data yang ada di variabel *trainData* ke dalam variabel *tr1*. Baris 31 digunakan untuk memasukkan data yang ada di variabel *testdata* ke dalam variabel *tr2*. Variabel *t1*, *tr1*, dan *tr2* nantinya akan digunakan untuk melakukan peramalan menggunakan metode peramalan GRNN. Baris 33 digunakan untuk menggunakan fungsi GRNN yang ada di aplikasi MATLAB. Pada fungsi tersebut dibutuhkan tiga parameter, yaitu *tr1* adalah data training yang akan digunakan untuk mencari model peramalan GRNN terbaik, *t1* adalah data target yang akan digunakan untuk mencari model peramalan GRNN terbaik, dan *p* adalah parameter σ (*spread parameter*).

Nantinya parameter σ akan diberikan nilai sesuai dengan parameter σ paling optimal yang didapatkan dari optimasi menggunakan algoritma FOA. Baris 34 digunakan untuk menerapkan model GRNN yang didapatkan sebelumnya ke dalam data yang ingin dilakukan peramalan. Hasil dari tahap ini adalah peramalan data yang akan dimasukkan ke dalam variabel yc . Baris 36 digunakan untuk merubah posisi yc (hasil peramalan) dari Vektor X (Horizontal) ke Vektor Y (Vertikal) untuk mempermudah tahap berikutnya.

5.4. Analisis Performa Model

Pada tahap ini akan dilakukan analisis performa model GRNN yang didapat untuk melakukan peramalan GRNN. Model akan dianalisis dengan cara menghitung akurasi dari hasil peramalan yang didapat. Penghitungan akurasi peramalan akan dilakukan menggunakan 2 metode pengukuran, yaitu *Root Mean Squared Error* (RMSE) menggunakan MATLAB serta *Mean Absolute Percentage Error* (MAPE) menggunakan Microsoft Office Excel 365.

```

36 - |           net=newgrnn(tr1,t1,p);
37 - |           yc=sim(net,tr2);
38 - |           y=t2-yc;
39 - |          for ii=1:row1
40 - |           |           g=g+y(ii)^2;
41 - |           |           end
42 - |           |           Sme11(i)= (g/row1) ^0.5;

```

Kode Program 5.15 Penghitungan RMSE GRNN Menggunakan MATLAB

Kode Program 5.15 digunakan untuk melakukan penghitungan RMSE pada hasil peramalan GRNN menggunakan MATLAB. Baris 36 dan 37 digunakan untuk melakukan peramalan data menggunakan fungsi GRNN yang ada di MATLAB. Baris 38

digunakan untuk menghitung *error* hasil peramalan dengan cara variabel $t2$ (data aktual) dikurangi dengan variabel yc (data hasil peramalan) dan dimasukkan ke dalam variabel y . Baris 39 sampai 41 digunakan untuk mengambil nilai variabel y pada semua *index* dalam *array* agar bisa dilakukan penghitungan *Mean Squared Error* (MSE). Baris 42 digunakan untuk menghitung RMSE peramalan dengan cara mengakarkan nilai MSE yang didapat.

BAB VI

HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan mengenai hasil dan pembahasan dari implementasi penerapan metode FOAGRNN untuk melakukan peramalan data NAB/UP reksa dana saham Simas Saham Unggulan dengan menggunakan MATLAB.

6.1. Hasil Pengumpulan dan *Pre-processing* Data

Pengumpulan dan *pre-processing* data dilakukan terhadap data NAB/UP reksa dana saham Simas Saham Unggulan. Pengumpulan dan *pre-processing* data dilakukan dengan cara membagi variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) dan variabel terikat (NAB/UP reksa dana saham Simas Saham Unggulan) menjadi dua bagian data, yaitu data *training* dan data *testing*. Periode data yang digunakan adalah data harian mulai 4 Januari 2016 sampai 28 Desember 2018. Data variabel terikat didapatkan dari situs www.sinarmas-am.co.id, sedangkan data variabel bebas didapatkan dari situs finance.yahoo.com, dan bi.go.id. Data variabel bebas yang akan digunakan untuk melakukan peramalan periode kedepan didapatkan dari hasil peramalan menggunakan metode peramalan *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*.

Data variabel bebas dan variabel terikat periode 4 Januari 2016 sampai 28 Desember 2018 yang digunakan dalam tugas akhir

ini dapat dilihat di LAMPIRAN A. Data variabel bebas dan variabel terikat hasil dari peramalan menggunakan metode *Double Moving Average* dapat dilihat di LAMPIRAN B. Data variabel bebas dan variabel terikat hasil dari peramalan menggunakan metode *Double Exponential Smoothing Holt* dapat dilihat di LAMPIRAN C. Data variabel bebas dan variabel terikat hasil dari peramalan menggunakan metode *Double Exponential Smoothing Brown* dapat dilihat di LAMPIRAN D.

6.2. Hasil Parameter Tuning

Pencarian parameter σ (*spread parameter*) yang akan digunakan untuk melakukan peramalan menggunakan GRNN dicari menggunakan algoritma optimasi *Fruit Fly Optimization Algorithm* (FOA). Dalam pencarian parameter σ menggunakan FOA, dilakukan berbagai macam kombinasi antara jumlah iterasi dan jumlah populasi untuk mendapatkan parameter σ paling optimal. Parameter σ paling optimal ditandai dengan nilai RMSE paling kecil di antara semua kombinasi.

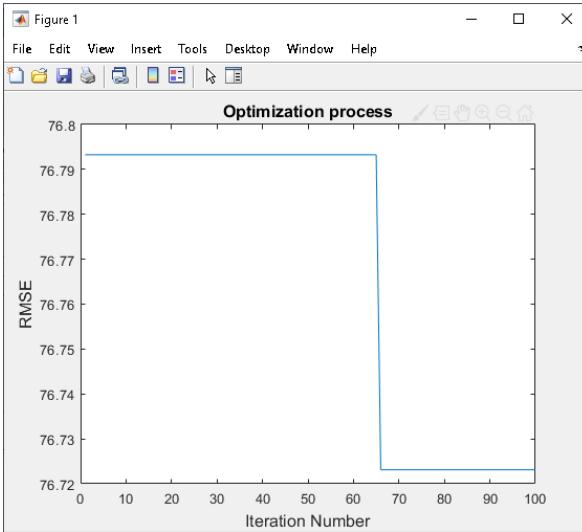
Dalam tiap kombinasi dilakukan 10 kali percobaan untuk mencari parameter σ paling optimal. Hal ini dikarenakan pada tiap percobaan selalu diinisiasi menggunakan angka acak. Sehingga hasil pencarian parameter σ dimungkinkan akan selalu berbeda pada tiap percobaan. Parameter σ yang akan dipilih pada tiap kombinasi adalah parameter σ dengan RMSE paling kecil di antara semua percobaan.

6.2.1. Skenario 100 Iterasi 10 Populasi

Tabel 6.1 Hasil Skenario 100 Iterasi 10 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	58	7,0194	76,7860
2	1	0,0998	76,7932
3	67	2,2322	76,7231
4	1	0,1294	76,7932
5	69	6,1199	76,7918
6	1	0,0952	76,7932
7	1	0,1057	76,7932
8	1	0,6202	76,7932
9	1	0,2687	76,7932
10	1	0,0842	76,7932

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 100 dan jumlah populasi sebesar 10. Tabel 6.1 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 100 iterasi dan 10 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-3 dengan RMSE sebesar 76,7231 dan parameter σ sebesar 25,2322.



Gambar 6.1 RMSE 100 Iterasi 10 Populasi

Gambar 6.1 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 100 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-67 dengan RMSE sebesar 76,7231 dan parameter σ sebesar 25,2322. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.

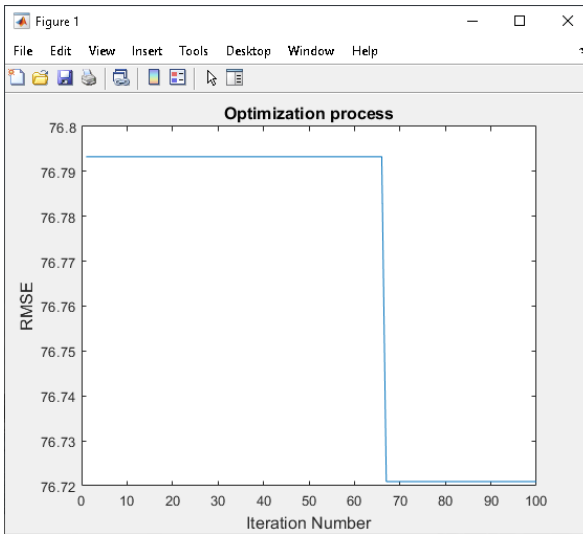
6.2.2. Skenario 100 Iterasi 50 Populasi

Tabel 6.2 Hasil Skenario 100 Iterasi 50 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	19	10,0391	76,7584

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
2	63	7,6943	76,7805
3	16	12,0174	76,7427
4	68	24,5686	76,7209
5	72	8,2284	76,7756
6	88	8,4360	76,7736
7	15	8,9509	76,7686
8	76	6,9955	76,7862
9	79	6,1899	76,7914
10	89	8,4619	76,7734

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 100 dan jumlah populasi sebesar 50. Tabel 6.2 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 100 iterasi dan 50 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-4 dengan RMSE sebesar 76,7209 dan parameter σ sebesar 24,5686.



Gambar 6.2 RMSE 100 Iterasi 50 Populasi

Gambar 6.2 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 100 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-68 dengan RMSE sebesar 76,7209 dan parameter σ sebesar 24,5686. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.

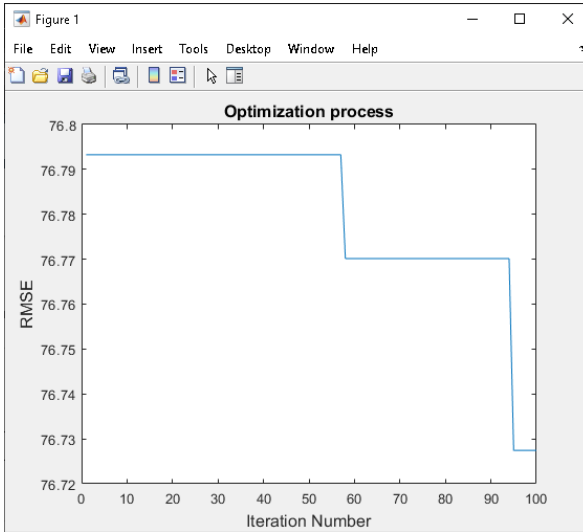
6.2.3. Skenario 100 Iterasi 100 Populasi

Tabel 6.3 Hasil Skenario 100 Iterasi 100 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	3	10,8593	76,7513

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
2	61	9,4163	76,7642
3	83	9,9734	76,7590
4	43	8,0978	76,7768
5	47	7,4596	76,7825
6	30	8,6931	76,7711
7	96	14,9257	76,7274
8	27	6,1375	76,7917
9	66	8,4619	76,7734
10	32	6,6428	76,7887

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 100 dan jumlah populasi sebesar 100. Tabel 6.3 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 100 iterasi dan 100 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-7 dengan RMSE sebesar 76,7274 dan parameter σ sebesar 14,9257.



Gambar 6.3 RMSE 100 Iterasi 100 Populasi

Gambar 6.3 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 100 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-96 dengan RMSE sebesar 76,7274 dan parameter σ sebesar 14,9257. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.

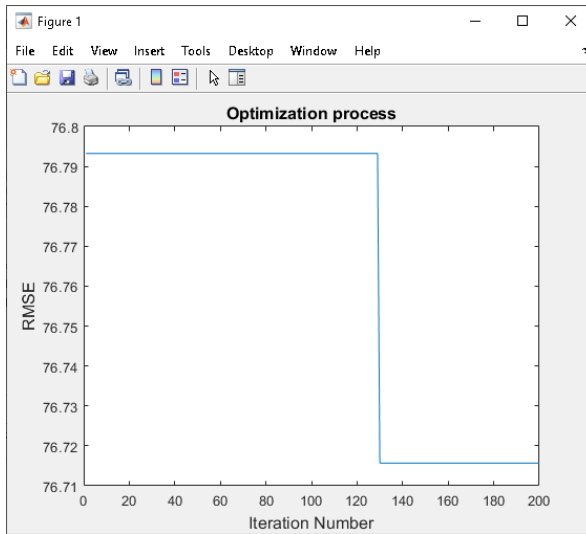
6.2.4. Skenario 200 Iterasi 50 Populasi

Tabel 6.4 Hasil Skenario 200 Iterasi 50 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	41	7.5843	76.7814

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
2	121	10.2774	76.7562
3	197	6.5572	76.7893
4	90	16.0173	76.7235
5	115	7.3090	76.7838
6	72	10.2150	76.7568
7	110	10.9767	76.7504
8	22	15.2157	76.7263
9	78	7.4820	76.7823
10	131	20.2972	76.7157

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 200 dan jumlah populasi sebesar 50. Tabel 6.4 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 200 iterasi dan 50 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-10 dengan RMSE sebesar 76,7157 dan parameter σ sebesar 20,2972.



Gambar 6.4 RMSE 200 Iterasi 50 Populasi

Gambar 6.4 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 200 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-131 dengan RMSE sebesar 76,7157 dan parameter σ sebesar 20,2972. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.

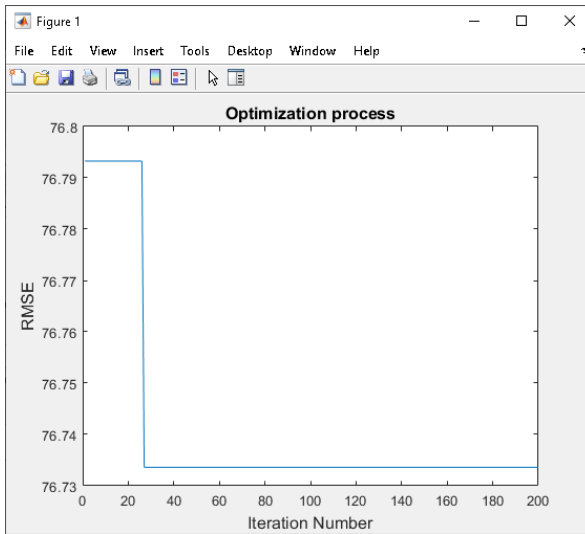
6.2.5. Skenario 200 Iterasi 100 Populasi

Tabel 6.5 Hasil Skenario 200 Iterasi 100 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	187	12,3217	76,7407

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
2	10	7,0345	76,7859
3	164	8,0093	76,7776
4	192	7,4827	76,7823
5	28	27,4733	76,7336
6	91	6,4541	76,7899
7	84	9,5161	76,7632
8	187	12,3352	76,7406
9	47	10,8778	76,7512
10	124	10,6173	76,7533

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 200 dan jumlah populasi sebesar 100. Tabel 6.5 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 200 iterasi dan 100 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-5 dengan RMSE sebesar 76,7336 dan parameter σ sebesar 27,4733.



Gambar 6.5 RMSE 200 Iterasi 100 Populasi

Gambar 6.5 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 200 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-28 dengan RMSE sebesar 76,7336 dan parameter σ sebesar 27,4733. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.

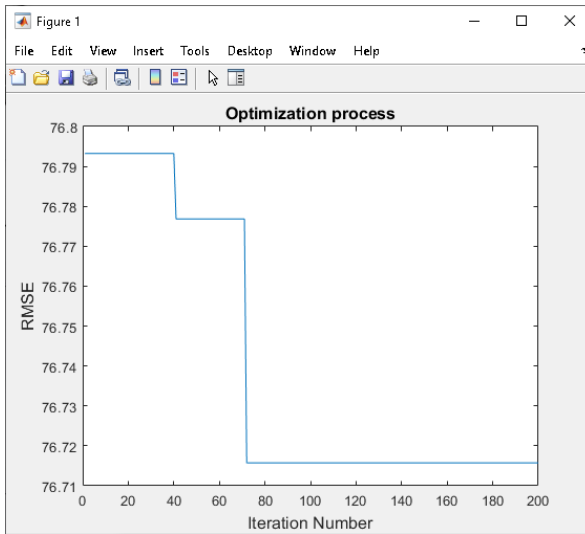
6.2.6. Skenario 200 Iterasi 150 Populasi

Tabel 6.6 Hasil Skenario 200 Iterasi 150 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	15	16,6926	76,7215

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
2	93	8,3257	76,7747
3	73	20,1160	76,7157
4	93	15,7385	76,7244
5	53	11,2818	76,7480
6	191	11,9668	76,7430
7	163	13,2614	76,7352
8	173	10,8017	76,7518
9	98	14,2314	76,7304
10	36	7,2390	76,7843

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 200 dan jumlah populasi sebesar 150. Tabel 6.6 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 200 iterasi dan 150 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-3 dengan RMSE sebesar 76,7157 dan parameter σ sebesar 20,1160.



Gambar 6.6 RMSE 200 Iterasi 150 Populasi

Gambar 6.6 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 200 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-73 dengan RMSE sebesar 76,7157 dan parameter σ sebesar 20,1160. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.

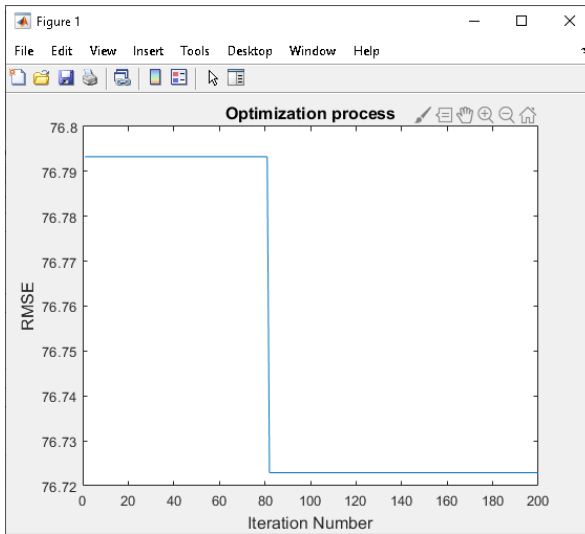
6.2.7. Skenario 200 Iterasi 200 Populasi

Tabel 6.7 Hasil Skenario 200 Iterasi 200 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	34	12.0594	76.7424

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
2	90	13.7173	76.7328
3	173	6.6684	76.7885
4	191	28.7604	76.7419
5	114	9.1086	76.7671
6	195	10.5484	76.7539
7	83	16.2010	76.7229
8	108	10.6331	76.7532
9	199	14.5889	76.7288
10	27	9.8043	76.7605

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 200 dan jumlah populasi sebesar 200. Tabel 6.7 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 200 iterasi dan 200 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-7 dengan RMSE sebesar 76,7229 dan parameter σ sebesar 16,2010.



Gambar 6.7 RMSE 200 Iterasi 200 Populasi

Gambar 6.7 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 200 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-83 dengan RMSE sebesar 76,7229 dan parameter σ sebesar 16,2010. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.

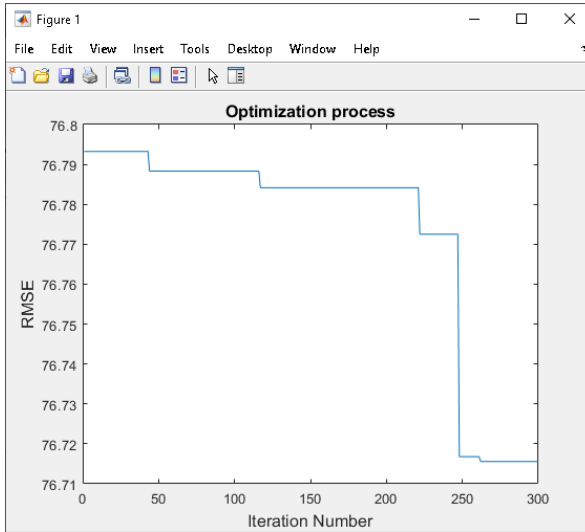
6.2.8. Skenario 300 Iterasi 300 Populasi

Tabel 6.8 Hasil Skenario 300 Iterasi 300 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	290	18,1423	76,7181

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
2	86	21,9635	76,7160
3	174	22,7240	76,7169
4	191	13,4406	76,7342
5	101	17,4372	76,7196
6	185	14,9290	76,7274
7	263	20,9004	76,7156
8	128	10,7094	76,7525
9	107	11,8688	76,7437
10	185	14,9290	76,7274

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 300 dan jumlah populasi sebesar 300. Tabel 6.8 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 300 iterasi dan 300 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-7 dengan RMSE sebesar 76,7156 dan parameter σ sebesar 20,9004.



Gambar 6.8 RMSE 300 Iterasi 300 Populasi

Gambar 6.8 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 300 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-263 dengan RMSE sebesar 76,7156 dan parameter σ sebesar 20,9004. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.

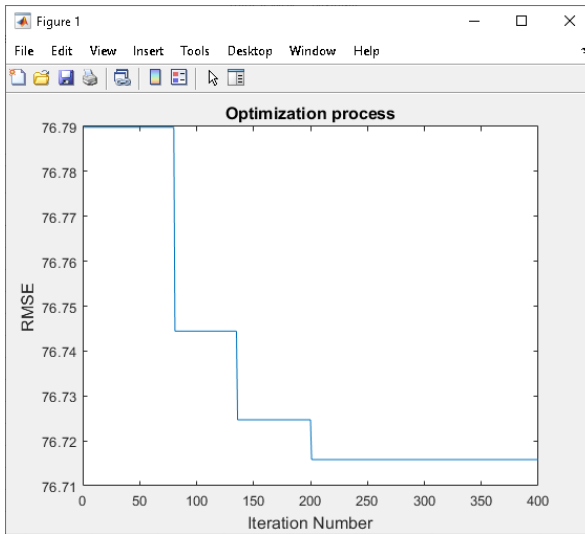
6.2.9. Skenario 400 Iterasi 400 Populasi

Tabel 6.9 Hasil Skenario 400 Iterasi 400 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	1	0,0961	76,7932

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
2	238	22,1599	76,7162
3	202	19,9669	76,7158
4	144	14,5733	76,7289
5	202	19,9669	76,7158
6	67	17,8949	76,7186
7	10	17,7057	76,7190
8	360	21,8542	76,7159
9	216	17,4177	76,7196
10	350	22,1898	76,7162

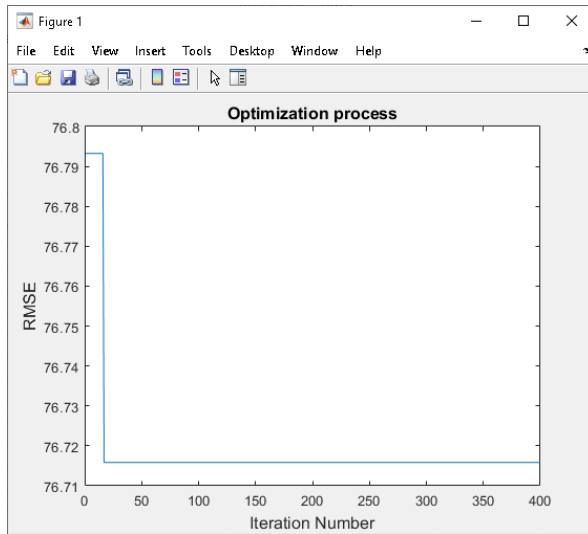
Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 400 dan jumlah populasi sebesar 400. Tabel 6.9 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 400 iterasi dan 400 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-3 dengan RMSE sebesar 76.7158, parameter σ sebesar 19,9669 serta percobaan ke-5 dengan RMSE sebesar 76,7158, parameter σ sebesar 21,6948.



Gambar 6.9 RMSE 400 Iterasi 400 Populasi Iterasi Ke-202

Gambar 6.9 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 400 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-202 dengan RMSE sebesar 76,7158, parameter σ sebesar 19,9669.

Gambar 6.10 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 400 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-12 dengan RMSE sebesar 76,7158, parameter σ sebesar 19,9669. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.



Gambar 6.10 RMSE 400 Iterasi 400 Populasi Iterasi Ke-12

6.2.10. Skenario 500 Iterasi 500 Populasi

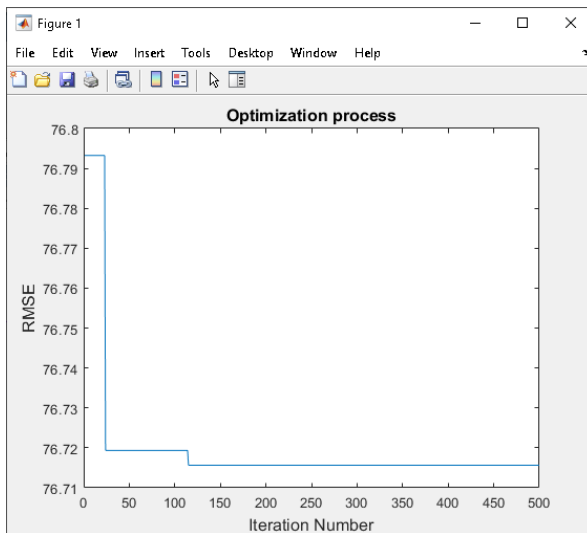
Tabel 6.10 Hasil Skenario 500 Iterasi 500 Populasi

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
1	345	12,1502	76,7394
2	105	22,7024	76,7168
3	105	22,7024	76,7168
4	35	14,8034	76,7279
5	112	21,3783	76,7157

Percobaan Ke	Iterasi Ke	Parameter σ	RMSE
6	138	22,4549	76,7165
7	275	16,2417	76,7228
8	55	22,2349	76,7163
9	275	16,2417	76,7228
10	35	14,8034	76,7279

Pada skenario ini dilakukan kombinasi antara jumlah iterasi sebesar 500 dan jumlah populasi sebesar 500. Tabel 6.10 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 500 iterasi dan 500 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan ke-5 dengan RMSE sebesar 76,7157 dan parameter σ sebesar 21,3783.

Gambar 6.11 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 500 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-112 dengan RMSE sebesar 76,7157 dan parameter σ sebesar 21,3783. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.



Gambar 6.11 RMSE 500 Iterasi 500 Populasi

6.2.11. Perbandingan Seluruh Skenario

Dari perbandingan seluruh skenario kombinasi iterasi dan populasi yang telah ditentukan sebelumnya, maka didapatkan hasil bahwa parameter σ paling optimal didapatkan dari skenario kombinasi antara iterasi sebesar 300 dan populasi 300. Skenario tersebut menghasilkan RMSE paling kecil di antara seluruh skenario yang dilakukan, yaitu sebesar 76,7156 dengan parameter σ sebesar 20,9004.

Tabel 6.11 Perbandingan Seluruh Skenario

Skenario	RMSE	Parameter σ
100 Iterasi 10 Populasi	76,7231	25,2322

Skenario	RMSE	Parameter σ
100 Iterasi 50 Populasi	76,7209	24,5686
100 Iterasi 100 Populasi	76,7274	14,9257
200 Iterasi 50 Populasi	76,7157	20,2972
200 Iterasi 100 Populasi	76,7336	27,4733
200 Iterasi 150 Populasi	76,7157	20,1160
200 Iterasi 200 Populasi	76,7229	16,2010
300 Iterasi 300 Populasi	76,7156	20,9004
400 Iterasi 400 Populasi	76,7158	19,9669 & 21,6948
500 Iterasi 500 Populasi	76,7157	21,3783

Tabel 6.1 merupakan tabel yang menunjukkan perbandingan hasil dari seluruh skenario yang telah dilakukan. Dari Tabel

6.1 dapat dilihat bahwa dari seluruh skenario, kombinasi antara iterasi sebesar 300 dan populasi sebesar 300 menghasilkan parameter σ paling optimal, yaitu sebesar 20.9004. Hal ini dapat dilihat dari RMSE paling kecil di antara semua skenario, yaitu sebesar 76,7156. Dari 10 skenario yang dilakukan, didapatkan bahwa semakin banyak generasi yang dideklarasikan, maka akan semakin banyak iterasi yang dilakukan untuk mencari parameter σ paling optimal. Namun banyaknya generasi tidak memiliki pengaruh yang signifikan terhadap besar kecilnya nilai RMSE yang dihasilkan. Dari 10 skenario kombinasi generasi dan populasi, serta 10 kali percobaan pada setiap skenario, didapatkan bahwa nilai *error* hasil peramalan yang ditunjukkan oleh nilai RMSE pada tiap percobaan tidak memiliki perubahan nilai yang signifikan. Hal ini dikarenakan jumlah data yang digunakan dalam 10 skenario memiliki jumlah yang sama, yang mana jumlah data dalam *neural network* akan digunakan sebagai jumlah *input layer*[14].

6.3. Hasil Implementasi Model GRNN

Parameter σ paling optimal telah ditemukan dengan menggunakan algoritma optimasi *Fruit Fly Optimization Algorithm* (FOA). Parameter paling optimal yang didapatkan dari penggunaan algoritma optimasi FOA memiliki RMSE paling kecil di antara semua skenario, yaitu 76,7156 dengan nilai parameter σ sebesar 20,9004. Nilai parameter σ ini selanjutnya akan digunakan untuk melakukan peramalan terhadap data *training*, data *testing*, dan peramalan periode kedepan. Hasil dari peramalan periode kedepan menggunakan metode peramalan *Generalized Regression Neural Network*

(GRNN) akan dibandingkan dengan peramalan periode kedepan menggunakan *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*.

6.3.1. Peramalan Data *Training*

Peramalan data *training* dilakukan dengan menggunakan metode peramalan *Generalized Regression Neural Network* (GRNN). Data *training* merupakan data yang diambil dari 75% data aktual, yaitu sebesar 543 dari total 724 data. Pada peramalan data *training* dilibatkan beberapa variabel, yaitu data variabel terikat (NAB/UP reksa dana saham Simas Saham Unggulan) serta variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) periode harian dengan rentang waktu 4 Januari 2016 sampai 26 Maret 2018. Peramalan data *training* dilakukan untuk menguji akurasi model GRNN terhadap data *training*. Model GRNN didapatkan dari penggunaan fungsi *newgrnn()* MATLAB terhadap data *training*. Peramalan ini menggunakan parameter σ paling optimal yang ditemukan melalui algoritma optimasi FOA, yaitu sebesar 20,9004.

Pada peramalan data *training*, hasil peramalan menunjukkan model GRNN yang ditemukan sangat cocok digunakan untuk melakukan peramalan. Hal itu dibuktikan dengan tingkat akurasi hasil peramalan yang diukur menggunakan *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE). Peramalan data *training* menggunakan metode peramalan GRNN dengan parameter σ sebesar 20,9004 memiliki nilai RMSE sebesar 8,415172346 dan MAPE sebesar

0,32%. MAPE sebesar 0,32% menunjukkan hasil peramalan yang sangat baik karena memiliki nilai kurang dari 10%. Hasil peramalan seluruh data *training* dan data *testing* dapat dilihat di LAMPIRAN F.

6.3.2. Peramalan Data *Testing*

Peramalan data *testing* dilakukan dengan menggunakan metode peramalan *Generalized Regression Neural Network* (GRNN). Data *training* merupakan data yang diambil dari 25% data aktual, yaitu sebesar 181 dari total 724 data. Pada peramalan data *testing* dilibatkan beberapa variabel, yaitu data variabel terikat (NAB/UP reksa dana saham Simas Saham Unggulan) serta variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) periode harian dengan rentang waktu 27 Maret 2018 sampai 28 Desember 2018. Peramalan data *testing* dilakukan untuk menguji akurasi model GRNN terhadap data *testing*. Model GRNN didapatkan dari penggunaan fungsi *newgrnn()* MATLAB terhadap data *training*. Peramalan ini menggunakan parameter σ paling optimal yang ditemukan melalui algoritma optimasi FOA, yaitu sebesar 20,9004.

Pada peramalan data *testing*, Hasil peramalan menunjukkan model GRNN yang ditemukan sangat cocok digunakan untuk melakukan peramalan terhadap data *testing*. Hal itu dibuktikan dengan tingkat akurasi hasil peramalan yang diukur menggunakan *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE). Peramalan data *testing* menggunakan metode peramalan GRNN dengan parameter σ sebesar 20,9004 memiliki nilai RMSE sebesar 249,3054604

dan MAPE sebesar 10,29%. MAPE sebesar 10,29% menunjukkan hasil peramalan yang baik karena memiliki nilai di antara 10% sampai dengan 20%. Hasil peramalan seluruh data *training* dan data *testing* dapat dilihat di LAMPIRAN F.

6.3.3. Peramalan Periode Kedepan

Peramalan periode kedepan dilakukan dengan menggunakan metode peramalan *Generalized Regression Neural Network* (GRNN). Peramalan periode kedepan akan menghasilkan hasil peramalan 1 bulan kedepan dengan periode harian dalam rentang waktu 2 Januari 2019 – 31 Januari 2019. Pada peramalan periode kedepan dilibatkan beberapa variabel, yaitu data variabel terikat (NAB/UP reksa dana saham Simas Saham Unggulan) serta variabel bebas (variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD) periode harian dengan rentang waktu 4 Januari 2016 sampai 28 Desember 2018. Peramalan periode kedepan ini akan melibatkan variabel bebas yang didapatkan melalui metode peramalan *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*. Peramalan periode kedepan dilakukan untuk mendapatkan hasil peramalan 1 bulan kedepan dengan menggunakan model GRNN. Model GRNN didapatkan dari penggunaan fungsi *newgrnn()* MATLAB terhadap data *training* dan *testing*. Peramalan ini menggunakan parameter σ paling optimal yang ditemukan melalui algoritma optimasi FOA, yaitu sebesar 20,9004. Hasil peramalan periode kedepan menggunakan metode peramalan GRNN akan dibandingkan dengan metode

peramalan *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*.

Tabel 6.2 merupakan tabel akurasi hasil peramalan periode kedepan menggunakan metode peramalan *Generalized Regression Neural Network* (GRNN) dengan variabel bebas yang didapatkan menggunakan metode peramalan *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*. Tabel 6.2 menunjukkan tingkat akurasi hasil peramalan yang diukur menggunakan *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE). Dari Tabel 6.2 dapat disimpulkan bahwa peramalan menggunakan GRNN dengan variabel bebas yang diramalkan menggunakan metode peramalan *Double Exponential Smoothing Holt* memiliki tingkat akurasi peramalan paling tinggi apabila dibandingkan dengan metode peramalan *Double Moving Average* dan *Double Exponential Smoothing Brown*, yang ditandai dengan nilai MAPE dan RMSE paling kecil. Penggunaan metode peramalan *Generalized Regression Neural Network* (GRNN) dengan variabel bebas yang didapatkan menggunakan metode peramalan *Double Exponential Smoothing Holt* sangat cocok apabila digunakan untuk melakukan peramalan NAB/UP reksa dana saham Simas Saham Unggulan. Hal ini dapat ditunjukkan oleh nilai RMSE sebesar 56,53752789 dan nilai MAPE sebesar 2,397% yang dapat dikategorikan sangat baik karena memiliki nilai MAPE kurang dari 10%. Hasil peramalan periode kedepan dapat seluruhnya dapat dilihat di LAMPIRAN G.

Tabel 6.12 Hasil Peramalan Periode Kedepan

Metode Peramalan	RMSE	MAPE
MA GRNN	56,54874716	2,40%
HOLT GRNN	56,53752789	2,40%
BROWN GRNN	58,13538437	2,44%

Tabel 6.13 Perbandingan Peramalan Periode Kedepan

Metode Peramalan	RMSE	MAPE
HOLT GRNN	56,53752789	2,40%
MA	60,2355043	2,63%
HOLT	60,72927405	2,59%
BROWN	48,02174048	2,04%

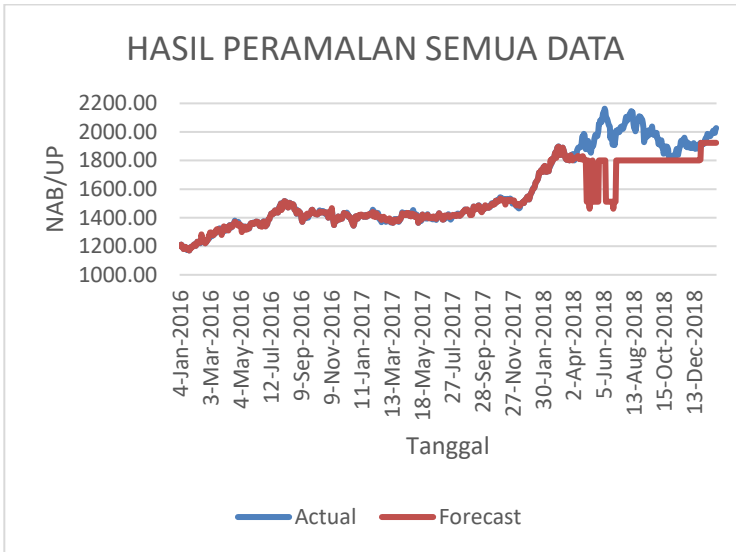
Tabel 6.3 merupakan tabel yang menunjukkan perbandingan akurasi hasil peramalan NAB/UP reksa dana saham Simas Saham Unggulan periode harian dalam rentang waktu 2 Januari 2019 – 31 Januari 2019 menggunakan metode peramalan *Generalized Regression Neural Network (GRNN)*, *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown*. Tabel 6.3

menunjukkan bahwa untuk melakukan peramalan periode kedepan metode peramalan *Double Exponential Smoothing Brown* memiliki tingkat akurasi paling tinggi, yaitu RMSE sebesar 48,02174048 dan MAPE sebesar 2,04%.

Namun peramalan menggunakan metode peramalan *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown* hanya mampu melakukan peramalan menggunakan 1 variabel saja, yaitu variabel NAB/UP reksa dana saham. Sehingga apabila diterapkan ke dalam tugas akhir ini, maka metode peramalan tersebut sangat kurang cocok apabila digunakan untuk meramalan NAB/UP reksa dana saham. NAB/UP pada tugas akhir sangat dipengaruhi oleh beberapa variabel lain, yaitu variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs jual dan kurs beli) IDR–USD.

Penggunaan metode peramalan *Generalized Regression Neural Network* (GRNN) terhadap peramalan NAB/UP reksa dana saham memang memiliki tingkat akurasi yang lebih rendah apabila dibandingkan dengan metode peramalan *Double Exponential Smoothing Brown*. Namun apabila diterapkan ke tugas akhir ini, metode peramalan *Generalized Regression Neural Network* (GRNN) sangat cocok digunakan untuk melakukan peramalan NAB/UP reksa dana saham. Metode peramalan GRNN sudah mampu melibatkan lebih dari 1 variabel untuk melakukan peramalan. Pada tugas akhir ini penggunaan metode peramalan GRNN melibatkan 6 variabel lain, yaitu variabel *open*, *close*, *high*, *low* pada Indeks Harga Saham Gabungan (IHSG), dan variabel *exchange rate* (kurs

jual dan kurs beli) IDR–USD. Tingkat akurasi yang dihasilkan oleh metode peramalan GRNN sudah mampu dikategorikan sebagai sangat baik karena memiliki nilai kurang dari 10%. Tabel 6.3 menunjukkan bahwa hasil peramalan menggunakan metode peramalan GRNN memiliki nilai RMSE sebesar 56,53752789 dan MAPE sebesar 2,40%.



Gambar 6.12 Hasil Peramalan Semua Data

Gambar 6.12 merupakan grafik hasil peramalan dari data *training*, *testing*, dan peramalan periode kedepan (1 bulan) menggunakan metode peramalan GRNN dengan parameter σ paling optimal yang dicari menggunakan algoritma optimasi FOA. Data 4 Januari 2016 sampai 26 Maret 2018 merupakan hasil peramalan data *training*. Peramalan data *training* menghasilkan nilai MAPE 0,32% dan RMSE 8,415172346. Data 27 Maret 2018 sampai 28 Desember 2018 merupakan

hasil peramalan data *testing*. Peramalan data *testing* menghasilkan nilai MAPE 10,29% dan RMSE 249,3054604. Data 2 Januari 2019 sampai 31 Januari 2019 merupakan hasil peramalan data periode kedepan. Peramalan periode kedepan menghasilkan nilai MAPE 2,40% dan RMSE 56,53752789. Dari Gambar 6.12 dapat dilihat bahwa grafik hasil peramalan data *training* hampir sama dengan data aktual *training*. Namun hal ini tidak terjadi pada data *testing* dan data peramalan periode kedepan. Hal ini disebabkan karena data *testing* dan data periode kedepan memiliki nilai yang jauh lebih besar daripada data *training*. Nilai data *testing* dan data periode kedepan yang lebih besar daripada data *training* mengakibatkan model GRNN yang dihasilkan dari data *training* tidak mampu mengenali nilai yang ada di dalam data *testing* dan data periode kedepan. Sehingga ketika melakukan peramalan terhadap data tersebut, tidak bisa menghasilkan nilai yang mendekati nilai aktual NAB/UP.

6.3.4. Pembuktian Gambar 6.12

Gambar 6.12 menunjukkan bahwa grafik hasil peramalan data *training* hampir sama dengan data aktual *training*. Namun hal ini tidak terjadi pada data *testing* dan data peramalan periode kedepan. Hal ini disebabkan karena data *testing* dan data periode kedepan memiliki nilai yang jauh lebih besar daripada data *training*. Nilai data *testing* dan data periode kedepan yang lebih besar daripada data *training* mengakibatkan model GRNN yang dihasilkan dari data *training* tidak mampu mengenali nilai yang ada di dalam data *testing* dan data periode kedepan. Sehingga ketika melakukan peramalan

terhadap data tersebut, tidak bisa menghasilkan nilai yang mendekati nilai aktual NAB/UP.

Untuk membuktikan kesimpulan tersebut maka dilakukan uji coba terhadap peramalan data NAB/UP Simas Saham Unggulan dengan cara menambah rentang nilai yang ada pada data aktual dengan membagi data menjadi 2 bagian dengan proporsi yang berbeda dari tahap perancangan, yaitu sebesar 642 data (4 Januari 2016 - 30 Agustus 2018) untuk data *training* dan 82 data (31 Agustus 2016 - 28 Desember 2018) untuk data *testing*. Pembagian data dengan proporsi tersebut dilakukan agar nilai maksimal dari data *training* yang akan digunakan dalam mencari model peramalan dapat menjangkau keseluruhan rentang nilai data untuk melakukan peramalan. Dalam uji coba ini dilakukan skenario kombinasi 1000 iterasi dan 500 populasi. Jumlah iterasi dan populasi ditentukan dengan cara *trial and error*. Uji coba dilakukan sebanyak 10 kali pada skenario kombinasi 1000 iterasi dan 500 populasi.

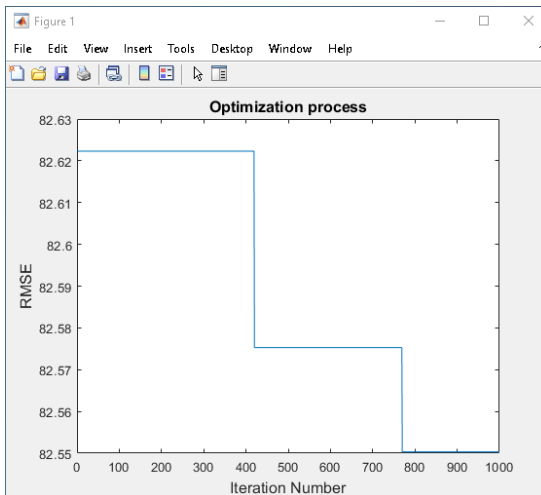
Tabel 6.14 Hasil Skenario 1000 Iterasi 500 Populasi

Percobaan Ke	Iterasi Ke	RMSE	Parameter σ
1	771	82.5503	70.7986
2	295	82.6168	36.8300
3	698	82.6037	41.1816
4	525	82.6071	40.0921
5	350	82.6007	42.1377

Percobaan Ke	Iterasi Ke	RMSE	Parameter σ
6	833	82.5525	64.2852
7	560	82.6049	40.7937
8	812	82.6113	38.6861
9	560	82.5613	89.8977
10	302	82.5509	74.8522

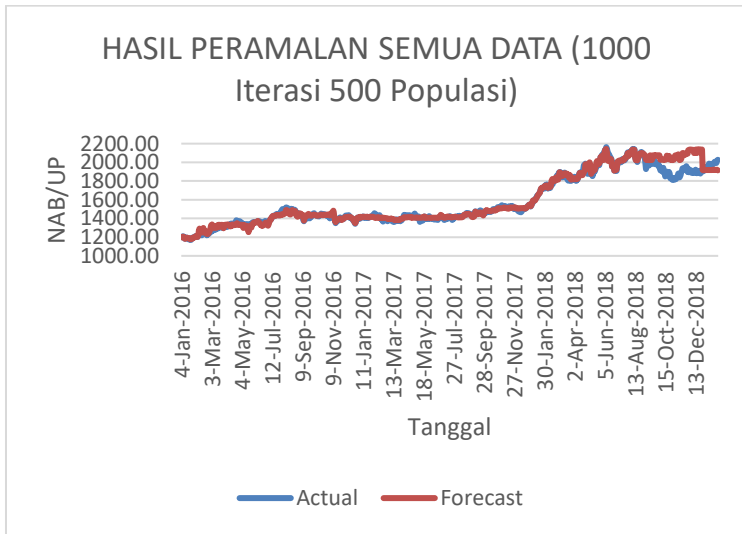
Tabel 6.14 merupakan tabel yang menunjukkan parameter σ paling optimal hasil dari 10 kali percobaan pada skenario 1000 iterasi dan 500 populasi. Dari 10 kali percobaan didapatkan hasil bahwa nilai RMSE paling kecil diperoleh dari percobaan pertama dengan RMSE sebesar 82.5503 dan parameter σ sebesar 70.7986.

Gambar 6.13 merupakan grafik yang menunjukkan perubahan RMSE pada tiap iterasi. Dari gambar tersebut dapat dilihat bahwa diantara 1000 iterasi yang ditetapkan, nilai σ paling optimal didapatkan pada iterasi ke-771 dengan RMSE sebesar 82.5503 dan parameter σ sebesar 70.7986. Seluruh hasil percobaan pada semua skenario dapat dilihat di LAMPIRAN E.



Gambar 6.13 RMSE 1000 Iterasi 500 Populasi

Gambar 6.14 merupakan grafik hasil peramalan dari data *training*, *testing*, dan peramalan periode kedepan (1 bulan) menggunakan metode peramalan GRNN dengan parameter σ sebesar 70.7986. Data 4 Januari 2016 sampai 30 Agustus 2018 merupakan hasil peramalan data *training*. Peramalan data *training* menghasilkan nilai MAPE 0,98% dan RMSE 20.29961515. Data 31 Agustus 2018 sampai 28 Desember 2018 merupakan hasil peramalan data *testing*. Peramalan data *testing* menghasilkan nilai MAPE 8,23% dan RMSE 170.3085833. Data 2 Januari 2019 sampai 31 Januari 2019 merupakan hasil peramalan data periode kedepan. Peramalan periode kedepan menghasilkan nilai MAPE 2,57% dan RMSE 60.88414903. Seluruh hasil peramalan dengan skenario 1000 iterasi dan 500 populasi dapat dilihat di LAMPIRAN H.



Gambar 6.14 Hasil Peramalan Semua Data (1000 Iterasi 100 Populasi)

Halaman ini sengaja dikosongkan

BAB VII

KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan mengenai kesimpulan yang didapatkan dari pengerjaan tugas akhir serta saran untuk pengembangan penelitian berikutnya.

7.1. Kesimpulan

Kesimpulan yang didapatkan dari pengerjaan tugas akhir ini adalah sebagai berikut:

1. Peramalan NAB/UP reksa dana saham menggunakan metode peramalan *Generalized Regression Neural Network* (GRNN) dengan parameter σ (*spread parameter*) yang dicari menggunakan algoritma optimasi *Fruit Fly Optimization Algorithm* (FOA) dapat menghasilkan peramalan yang sangat baik.
2. Apabila dibandingkan dengan metode peramalan *Double Moving Average* (MAPE 2,40%, RMSE 56,54874716) dan *Double Exponential Smoothing Brown* (MAPE 2,44%, RMSE 58,13538437), *Double Exponential Smoothing Holt* memiliki tingkat akurasi peramalan yang lebih baik untuk meramalkan variabel bebas pada peramalan reksa dana saham menggunakan GRNN, yaitu dengan MAPE sebesar 2,40% dan RMSE sebesar 56,53752789.
3. Parameter σ paling optimal untuk melakukan peramalan reksa dana saham adalah 20,9004 yang menghasilkan nilai MAPE sebesar 2,40% (sangat baik) dan RMSE sebesar 56,53752789 pada peramalan periode kedepan.

4. Parameter σ paling optimal yang dicari menggunakan FOA dapat ditemukan dengan melakukan kombinasi antara populasi sebesar 300 dan iterasi sebesar 300.
5. Jumlah generasi dan jumlah populasi lalat buah pada FOA bukan penentu utama parameter σ paling optimal. Namun semakin besar jumlah generasi dan jumlah populasi lalat buah, maka peluang mendapatkan parameter σ yang lebih optimal akan semakin besar.

7.2. Saran

Dari hasil pengerjaan tugas akhir yang telah dilakukan, terdapat beberapa hal yang dapat dilakukan untuk mengembangkan penelitian kedepan, yaitu:

1. Memperbanyak skenario kombinasi antara populasi dan iterasi pada FOA untuk mendapatkan parameter σ paling optimal pada metode peramalan GRNN.
2. Memperbanyak data *training* dan mengimplementasikan model GRNN yang ditemukan kepada data NAB/UP reksa dana saham lain untuk menguji akurasi model peramalan.
3. Melakukan perbandingan algoritma optimasi FOA dengan algoritma optimasi lain untuk mencari parameter σ paling optimal.
4. Melakukan perbandingan peramalan reksa dana saham menggunakan metode peramalan GRNN dengan metode peramalan lain.

DAFTAR PUSTAKA

- [1] “Yuk Nabung Saham,” 2019. [Online]. Available: yuknabungsaham.idx.co.id. [Accessed: 04-Jun-2019].

- [2] A. Sujatmiko, “Alternatif Investasi Melalui Reksa Dana,” *Perspektif*, vol. 3, no. 2. pp. 1–8, 1998.

- [3] H. K. Dewi, “Top 10 Reksadana Saham Terbaik Bareksa, YTD Untung Sampai 41 Persen,” 2018. [Online]. Available: <https://www.bareksa.com/id/text/2018/08/28/top-10-reksadana-saham-terbaik-bareksa-ytd-untung-sampai-41-persen/20157/news>. [Accessed: 26-Mar-2018].

- [4] T. H. Huang and Y. H. Leu, “A Mutual Fund Investment Method Using Fruit Fly Optimization Algorithm and Neural Network,” *Appl. Mech. Mater.*, vol. 571–572, pp. 318–325, 2014.

- [5] S. Z. Han, L. H. Huang, Y. Y. Zhou, and Z. L. Liu, “Mixed chaotic FOA with GRNN to construction of a mutual fund forecasting model,” *Cogn. Syst. Res.*, vol. 52, pp. 380–386, 2018.

- [6] H. Z. Li, S. Guo, C. J. Li, and J. Q. Sun, "A hybrid annual power load forecasting model based on generalized regression neural network with fruit fly optimization algorithm," *Knowledge-Based Syst.*, vol. 37, pp. 378–387, 2013.

- [7] H. Himawan, P. Studi, and T. Informatika, "EFEKTIFITAS PENGGUNAAN METODE EXPONENTIAL SMOOTHING," 2005.

- [8] V. E. Mcgee, "Forecasting: methods and applications / Spyros Makridakis , Steven C .," no. February, pp. 3–5, 2018.

- [9] N. Gnfc, I. Tower, and S. G. Road, "6 : FORECASTING TECHNIQUES Dr . Ravi Mahendra Gor ICFAI Business School."

- [10] M. Usman, *Bunga Rampai Reksa Dana*. Jakarta: Balai Pustaka, 2000.

- [11] E. S. HEDIANTI, "PERAMALAN HARGA SAHAM DENGAN MENGGUNAKAN METODE REGRESSION (SVR)," *Dep. Sist. Inf. Fak. Teknol. Inf. dan Komun. Inst. Teknol. Sepuluh Nop.*, 2019.

- [12] R. I. Agustin, *Peramalan Data Intermiten menggunakan Metode Autoregressive Integrated Moving Average dan Neural Network (ARIMA-NN)*. Surabaya: Institut Teknologi Sepuluh Nopember, 2018.
- [13] L. Fausett, *Fundamentals of Neural Networks; Architectures, Algorithms, and Applications*. .
- [14] J. Siang, *Jaringan Syaraf Tiruan dan Pemrogramannya menggunakan MATLAB*. Yogyakarta: Penerbit ANDI, 2005.
- [15] M. Amiri, H. Davande, A. Sadeghian, and S. Chartier, “Feedback associative memory based on a new hybrid model of generalized regression and self-feedback neural networks,” *Neural Networks*, vol. 23, no. 7, pp. 892–904, 2010.
- [16] W. Pan, “Knowledge-Based Systems A new Fruit Fly Optimization Algorithm : Taking the financial distress model as an example,” *Knowledge-Based Syst.*, vol. 26, pp. 69–74, 2012.
- [17] A. A. S, *PENERAPAN METODE SINGLE MOVING AVERAGEDAN EXPONENTIAL SMOOTHING DALAM PERAMALAN PERMINTAAN PRODUK MEUBEL JENIS*

*COFFEE TABLEPADA JAVA FURNITURE
KLATEN*. Surakarta: UNIVERSITAS SEBELAS
MARET, 2009.

- [18] W. Pan, “Fruit Fly Optimization Algorithm (Using MATLaab),” pp. 4–8, 2014.

BIODATA PENULIS



Penulis memiliki nama lengkap Muhammad Arif, dan memiliki nama panggilan Mamad. Penulis berasal dari kota legen, Tuban, Jawa Timur. Selain aktif di bidang akademik, penulis juga aktif di berbagai kegiatan non-akademik.

Selama menjalani perkuliahan, penulis pernah menduduki beberapa posisi strategis dalam organisasi, diantaranya adalah sebagai Komting kedua LANNISTER, *Steering Committee* Kaderisasi HMSI ITS 2016-2017, Pemandu LKMM ITS 2016-2018, Ketua Senat Beasiswa Indofood Sukses Makmur (BISMA) Camp 2, Presiden YOT Surabaya 2017-2018, Ketua Paguyuban KSE ITS 2018-2019, dan Community Coordinator YOT 2018-2019. Penulis pernah mengikuti berbagai pelatihan baik dari dalam, maupun luar kampus. Pelatihan yang pernah diikuti oleh Penulis diantaranya adalah SINERGI 2015, LKMM PRA-TD FTif 2015, LKMM TD HMSI ITS 2016, PP LKMM FTif 2016, Mahameru Youth Camp 2016, PGN Innovation Camp 2017, Indofood Leadership Camp 1 2018, Indofood Leadership Camp 2 2018, Google Gapura Digital 2018, Pelatihan Kepemimpinan 2019, dan lain sebagainya.

Apabila terdapat beberapa hal yang ingin dibagikan, Penulis dapat dihubungi melalui email mamad.arif96@gmail.com dan akun LinkedIn www.linkedin.com/in/mamadarif .

Halaman ini sengaja dikosongkan

LAMPIRAN A

Data NAB/UP Simas Saham Unggulan beserta variabel yang mempengaruhi NAB/UP reksa dana saham (variabel *close*, *open*, *high*, *low* IHSG, Kurs Jual, Kurs Beli IDR-USD) periode 4 Januari 2016 sampai 28 Desember 2018 dapat dilihat pada link <https://intip.in/LampiranATAMamad> .

Halaman ini sengaja dikosongkan

LAMPIRAN B

Data hasil peramalan NAB/UP Simas Saham Unggulan beserta variabel yang mempengaruhi NAB/UP reksa dana saham (variabel *close*, *open*, *high*, *low* IHSG, Kurs Jual, Kurs Beli IDR-USD) menggunakan metode *Double Moving Average* dapat dilihat pada link <https://intip.in/LampiranBTAMamad> .

Halaman ini sengaja dikosongkan

LAMPIRAN C

Data hasil peramalan NAB/UP Simas Saham Unggulan beserta variabel yang mempengaruhi NAB/UP reksa dana saham (variabel *close*, *open*, *high*, *low* IHSG, Kurs Jual, Kurs Beli IDR-USD) menggunakan metode *Double Exponential Smoothing Holt* dapat dilihat pada link <https://intip.in/LampiranCTAMamad> .

Halaman ini sengaja dikosongkan

LAMPIRAN D

Data hasil peramalan NAB/UP Simas Saham Unggulan beserta variabel yang mempengaruhi NAB/UP reksa dana saham (variabel *close*, *open*, *high*, *low* IHSG, Kurs Jual, Kurs Beli IDR-USD) menggunakan metode *Double Exponential Smoothing Brown* dapat dilihat pada link <https://intip.in/LampiranDTAMamad> .

Halaman ini sengaja dikosongkan

LAMPIRAN E

Hasil dari pencarian parameter σ menggunakan algoritma optimasi FOA pada seluruh skenario dapat dilihat pada link <https://intip.in/LampiranETAMamad> .

Halaman ini sengaja dikosongkan

LAMPIRAN F

Hasil dari peramalan data *training* dan data *testing* menggunakan metode peramalan *Generalized Regression Neural Network* (GRNN) dapat dilihat pada link <https://intip.in/LampiranFTAMamad> .

Halaman ini sengaja dikosongkan

LAMPIRAN G

Hasil dari peramalan periode kedepan menggunakan metode peramalan *Generalized Regression Neural Network* (GRNN) dengan variabel bebas dari hasil peramalan *Double Moving Average*, *Double Exponential Smoothing Holt*, dan *Double Exponential Smoothing Brown* dapat dilihat pada link <https://intip.in/LampiranGTAMamad> .

Halaman ini sengaja dikosongkan

LAMPIRAN H

Hasil peramalan seluruh data menggunakan metode peramalan *Generalized Regression Neural Network* (GRNN) dengan parameter σ sebesar 70.7986 yang didapat dari skenario 1000 iterasi dan 500 populasi dapat dilihat pada link <https://intip.in/LampiranHTAMamad> .

17-126