

SKRIPSI

**OPTIMALISASI ESTIMASI DURASI *DRY DOCKING* KAPAL
MENGUNAKAN MODEL
ARTIFICIAL NEURAL NETWORK (ANN)
Studi Kasus: PT. Industri Kapal Indonesia (Persero)**

Disusun dan diajukan oleh:

**MUWAFFAQ AHNAF SHIDDIQ
D071191022**



**PROGRAM STUDI TEKNIK INDUSTRI
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
GOWA
2023**

SKRIPSI

**OPTIMALISASI ESTIMASI DURASI *DRY DOCKING* KAPAL
MENGUNAKAN MODEL
ARTIFICIAL NEURAL NETWORK (ANN)
Studi Kasus: PT. Industri Kapal Indonesia (Persero)**

Disusun dan diajukan oleh:

**MUWAFFAQ AHNAF SHIDDIQ
D071191022**



**PROGRAM STUDI TEKNIK INDUSTRI
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
GOWA
2023**

LEMBAR PENGESAHAN SKRIPSI
OPTIMALISASI ESTIMASI DURASI *DRY-DOCKING* KAPAL
MENGGUNAKAN *ARTIFICIAL NEURAL NETWORK*
(Studi Kasus: PT. Industri Kapal Indonesia)

Disusun dan diajukan oleh

MUWAFFAQ AHNAF SHIDDIQ
D071191022

Telah dipertahankan dihadapan Panitia Ujian yang dibentuk dalam rangka penyelesaian Studi Program Sarjana Program Studi Teknik Industri Fakultas Teknik Universitas Hasanuddin Pada tanggal 15 November 2023 dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,



Dr. Ir. Rosmalina Hanafi, M.Eng.
NIP. 19660128 199103 2 003



Dwi Handayani, S.T., M.T.
NIP. 19950902 202208 6 001

Ketua Program Studi Teknik Industri
Fakultas Teknik Universitas Hasanuddin



Ir. Kifayah Amar, S.T., M.Sc., Ph.D., IPU
NIP. 19740621 200604 2 001

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Muwaffaq Ahnaf Shiddiq
NIM : D071191022
Program Studi : Teknik Industri
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

Optimalisasi Estiamsi Durasi *Dry-Docking* Kapal Menggunakan *Artificial Neural Network* (Studi Kasus: PT. Industri Kapal Indonesia)

Adalah karya tulisan saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain dan bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Semua Informasi yang ditulis dalam skripsi yang berasal dari penulis lain telah diberi penghargaan, yakni dengan mengutip sumber dan tahun penerbitannya. Oleh karena itu semua tulisan dalam skripsi ini sepenuhnya menjadi tanggung jawab penulis. Apabila ada pihak manapun yang merasa ada kesamaan judul dan atau hasil temuan dalam skripsi ini, maka penulis siap untuk diklarifikasi dan mempertanggungjawabkan segala resiko.

Segala data dan informasi yang diperoleh selama proses pembuatan skripsi, yang akan dipublikasi oleh Penulis di masa depan harus mendapat persetujuan dari Dosen Pembimbing.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan isi skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 15 November 2023

Yang Menyatakan Tanda Tangan,



Muwaffaq Ahnaf Shiddiq

ABSTRAK

MUWAFFAQ AHNAF SHIDDIQ. *Pengoptimalan Estimasi Durasi Dry-Docking Kapal Menggunakan Model Artificial Neural Network (Studi Kasus: PT. Industri Kapal Indonesia)* (dibimbing oleh Rosmalina Hanafi dan Dwi Handayani)

Salah satu fasilitas operasional penunjang kapal agar dapat tetap berlayar dengan kondisi yang layak adalah galangan kapal (*shipyard*). Peningkatan jumlah kapal skala nasional juga turut berdampak pada meningkatnya jumlah kapal yang masuk ke PT. IKI (Persero) pada kurun waktu tiga tahun belakangan dimulai sejak 2021 hingga 2023 di bulan Agustus. Peningkatan jumlah volume kapal di PT. IKI belum diimbangi dengan optimalisasi model estimasi durasi *dry-docking* kapal terhadap durasi penyelesaian sebenarnya. Berdasarkan data yang menampilkan jumlah kapal yang masuk di galangan PT. IKI, setidaknya terdapat sekitar 137 keterlambatan proyek kapal atau sekitar 86%. Salah satu cara dalam membuat model alternatif yang mampu mengestimasi durasi *dry-docking* kapal dengan lebih optimal, yaitu dengan memanfaatkan *data mining*. Upaya mengimplementasikan *data mining* dalam membuat model estimasi durasi *dry-docking* alternatif lainnya ini menggunakan model *Artificial Neural Network* (ANN) sebagai alat pemodelan komputasi yang kompleks. Tujuan penelitian ini adalah membuat model *Artificial Neural Network* untuk mengestimasi durasi *dry-docking* kapal melalui pengujian dan penilaian performa model ANN. Hasil penelitian ini berupa model ANN untuk mengestimasi durasi *dry-docking* dilakukan dengan dua simulasi, yaitu simulasi awal dan simulasi optimasi parameter model. Simulasi awal bertujuan untuk mengetahui apakah model dapat diimplementasikan konsepnya dengan menggunakan nilai parameter yang sama untuk setiap jenis model satu sampai sepuluh *neuron*. Sementara untuk simulasi optimasi parameter bertujuan untuk menemukan nilai kombinasi parameter model yang paling optimal dengan menggunakan seluruh nilai parameter dalam ketetapan jangkauannya. Penilaian performa model *Artificial Neural Network* (ANN) berdasarkan nilai parameter evaluasi model ANN dengan MSE (%) dan MAPE. Pertama, model ANN lima *neuron* merupakan model yang performanya paling baik pada simulasi awal dengan nilai MSE (%) sebesar 6% dan MAPE sebesar 28%. Kedua, model ANN dengan dua *neuron* merupakan model dengan performa paling optimal pada simulasi optimasi parameter dengan nilai MSE (%) sebesar 8% dan MAPE sebesar 30%. Kedua model tersebut dapat dikategorikan sebagai model yang layak untuk melakukan estimasi berdasarkan nilai MAPE dibawah 50%. Penilaian performa model ANN tersebut juga dapat dinilai berdasarkan selisih antara nilai estimasi kedua model (model ANN dan model perusahaan) terhadap nilai aktual/realisasi *dry-docking* kapal.

Kata Kunci: *Artificial Neural Network, Dry-Docking, Estimasi, Data Mining*

ABSTRACT

MUWAFFAQ AHNAF SHIDDIQ. *Optimizing Ship Dry-Docking Duration Estimation Using an Artificial Neural Network Model (Case Study: PT. Industri Kapal Indonesia) (supervised by Rosmalina Hanafi and Dwi Handayani)*

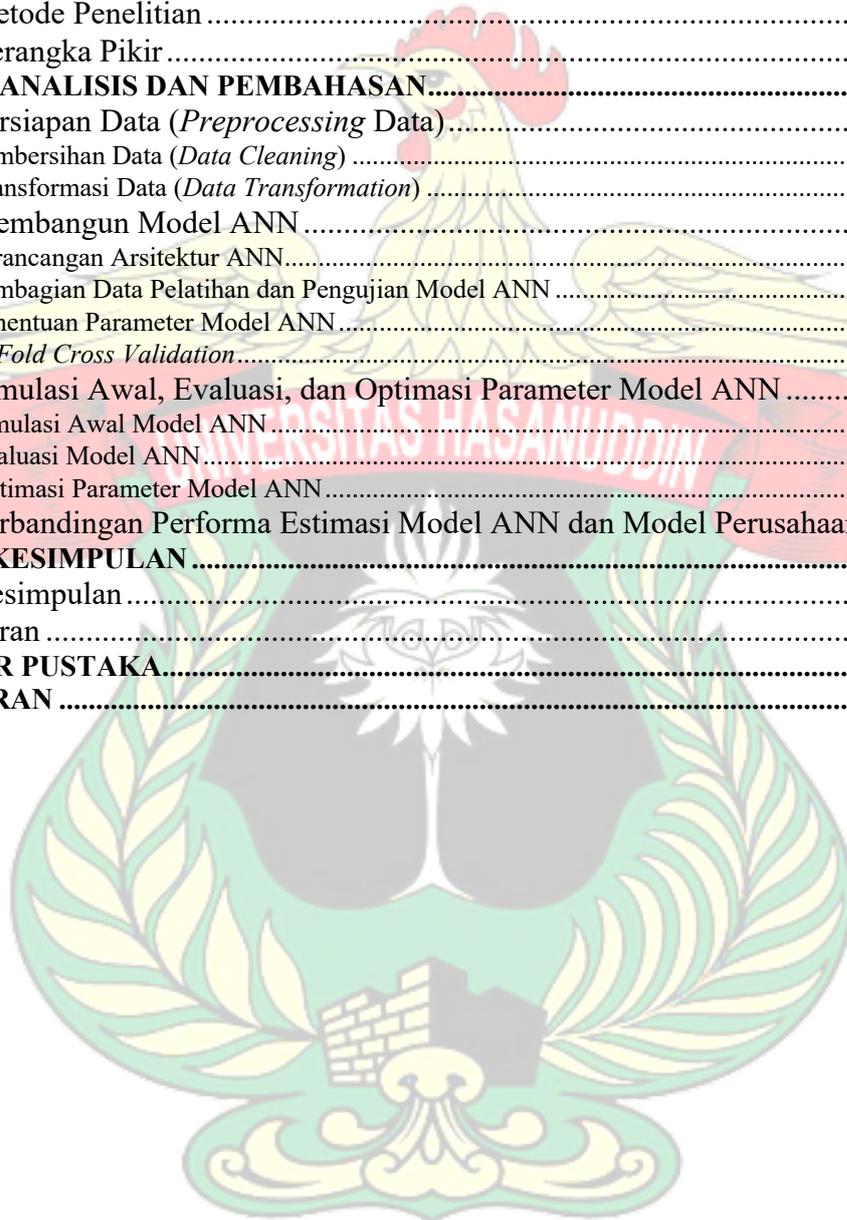
One of the operational facilities to support ships so they can continue sailing in proper condition is a shipyard. The increase in the number of national scale vessels also has an impact on the increase in the number of vessels entering PT. IKI (Persero) in the last three years starting from 2021 to 2023 in August. Increase in the number of ship volumes at PT. IKI has not been balanced by optimizing the model for estimating the ship's dry-docking duration against the actual completion duration. Based on data showing the number of ships entering the PT. IKI, there were at least 137 ship project delays or around 86%. One way to create an alternative model that is able to estimate the dry-docking duration of a ship more optimally is by utilizing data mining. This effort to implement data mining in creating an alternative dry-docking duration estimation model uses an Artificial Neural Network (ANN) model as a complex computational modeling tool. The aim of this research is to create an Artificial Neural Network model to estimate the dry-docking duration of ships through testing and assessing the performance of the ANN model. The results of this research are an ANN model to estimate the duration of dry-docking carried out with two simulations, namely the initial simulation and the model parameter optimization simulation. The initial simulation aims to find out whether the concept of the model can be implemented using the same parameter values for each type of model from one to ten neurons. Meanwhile, parameter optimization simulation aims to find the most optimal combination of model parameters by using all parameter values within its specified range. Artificial Neural Network (ANN) model performance assessment based on ANN model evaluation parameter values with MSE (%) and MAPE. First, the five neuron ANN model was the model that performed best in the initial simulation with an MSE (%) value of 6% and a MAPE of 28%. Second, the ANN model with two neurons is the model with the most optimal performance in parameter optimization simulations with an MSE (%) value of 8% and MAPE of 30%. These two models can be categorized as appropriate models for estimating based on MAPE values below 50%. The performance assessment of the ANN model can also be assessed based on the difference between the estimated value of the two models (the ANN model and the company model) to the actual/realized value of ship dry-docking.

Keyword: Artificial Neural Network, Dry-Docking, Estimation, Data Mining

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
DAFTAR ISI	iii
DAFTAR GAMBAR	v
DAFTAR TABEL	i
DAFTAR RUMUS	i
DAFTAR LAMPIRAN	ii
KATA PENGANTAR	iii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	4
1.3 Tujuan Penelitian.....	4
1.4 Manfaat Penelitian.....	5
1.5 Batasan Masalah.....	5
BAB II TINJAUAN PUSTAKA	6
2.1 <i>Data Mining</i>	6
2.1.1 Tahap Penerapan <i>Knowledge Discovery in Database (KDD)</i>	7
2.1.2 Proses Standar <i>Data Mining</i> antar Industri (CRISP-DM).....	8
2.1.3 Tipe Pekerjaan <i>Data Mining</i>	10
2.2 Jaringan Saraf Tiruan (JST)/ <i>Artificial Neural Network (ANN)</i>	12
2.2.1 Jenis-jenis Arsitektur <i>Artificial Neural Network</i>	13
2.2.2 Fungsi Aktivasi <i>Artificial Neural Network</i>	14
2.3 <i>Machine Learning</i>	18
2.4 Algoritma.....	18
2.3.1 Struktur Dasar Urut (<i>Sequence</i>).....	19
2.3.2 Struktur Dasar Pemilihan (<i>Selection</i>).....	19
2.3.3 Struktur Dasar Pengulangan (<i>Iteration</i>).....	20
2.3.4 <i>Algoritma Backpropagation</i>	21
2.5 <i>Cross Validation</i>	24
2.6 <i>Hyperparameter</i> Model ANN.....	25
2.5.1 Jumlah <i>Neuron/Nodes</i> pada <i>Hidden Layer</i>	25
2.5.2 Jumlah <i>Epochs</i>	26
2.5.3 Nilai <i>Learning Rate</i>	26
2.5.4 Nilai Momentum.....	26
2.7 Parameter Evaluasi Performa Model terhadap Kesalahan.....	27
2.6.1 <i>Mean Square Error (MSE)</i>	27
2.6.2 <i>Mean Absolute Percentage Error (MAPE)</i>	28
2.8 <i>Software Data Mining</i>	28
2.7.1 <i>Rapid Miner Studio 2</i>	29
2.7.2 WEKA.....	29
2.7.3 <i>R Studio</i>	30
2.7.4 MATLAB.....	30
2.9 <i>Dry-Docking</i> Kapal.....	31
2.8.1 Pekerjaan Perawatan Rutin Kapal pada <i>Dry-Docking</i>	31
2.10 Penelitian Terdahulu.....	33
BAB III METODOLOGI PENELITIAN	40
3.1 Lokasi Penelitian.....	40
3.2 Populasi dan Sampel Penelitian.....	40
3.2.1 Populasi.....	40
3.2.2 Sampel.....	41

3.3	Variabel Penelitian.....	41
3.3.1	Variabel Independen	41
3.3.2	Variabel Dependen.....	43
3.4	Jenis Data.....	44
3.5	Metode Pengumpulan Data.....	44
3.5.1	Studi Kepustakaan/Kajian Literatur	44
3.5.2	Wawancara.....	45
3.5.3	Studi Dokumen	45
3.6	Metode Penelitian	46
3.7	Kerangka Pikir	49
BAB IV ANALISIS DAN PEMBAHASAN.....		51
4.1	Persiapan Data (<i>Preprocessing Data</i>).....	51
4.1.1	Pembersihan Data (<i>Data Cleaning</i>)	51
4.1.2	Transformasi Data (<i>Data Transformation</i>)	52
4.2	Membangun Model ANN.....	54
4.2.1	Perancangan Arsitektur ANN.....	54
4.2.2	Pembagian Data Pelatihan dan Pengujian Model ANN	55
4.2.3	Penentuan Parameter Model ANN	56
4.2.4	<i>K-Fold Cross Validation</i>	57
4.3	Simulasi Awal, Evaluasi, dan Optimasi Parameter Model ANN	58
4.3.1	Simulasi Awal Model ANN	58
4.3.2	Evaluasi Model ANN.....	58
4.3.3	Optimasi Parameter Model ANN	62
4.4	Perbandingan Performa Estimasi Model ANN dan Model Perusahaan	65
BAB V KESIMPULAN.....		68
5.1	Kesimpulan	68
5.2	Saran	69
DAFTAR PUSTAKA.....		70
LAMPIRAN		73

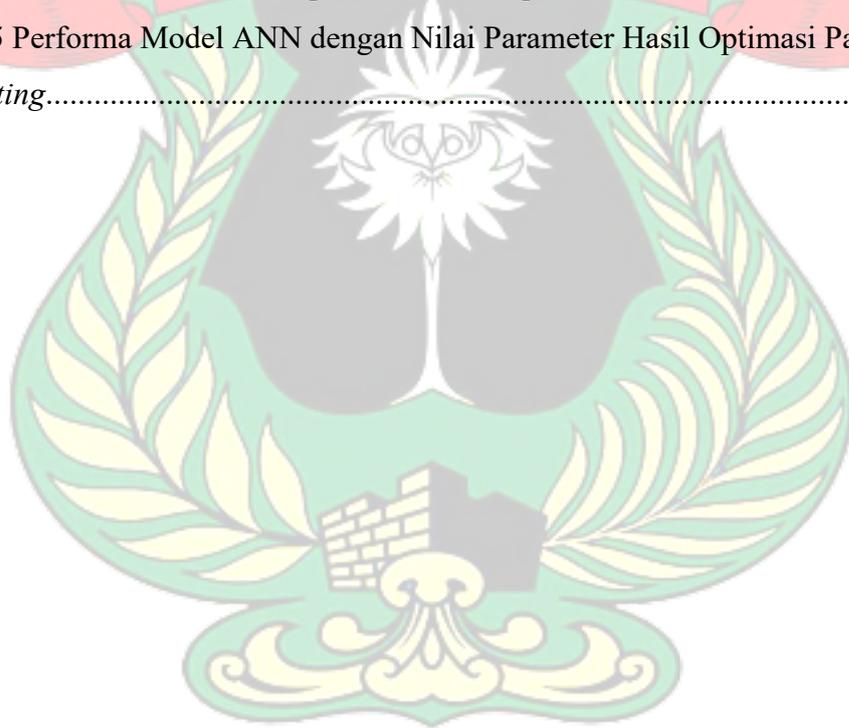


DAFTAR GAMBAR

Gambar 1 Jumlah Kapal Berdasarkan Periode Waktu.....	3
Gambar 2. Proses Penerapan <i>Knowledge Discovery in Database</i>	7
Gambar 3. Siklus CRISP-DM (Febian, D., dkk, 2017).....	9
Gambar 4. Konsep Dasar <i>Artificial Neural Network</i> (Hukubune, A. J., 2022)	12
Gambar 5. Arsitektur <i>Single-Layer Neural Network</i>	13
Gambar 6. Arsitektur <i>Multi-Layer Neural Network</i>	13
Gambar 7. Arsitektur <i>Recurrent Network</i>	14
Gambar 8. <i>Identity Function</i>	15
Gambar 9. <i>Binary Step Function</i>	15
Gambar 10. <i>Sigmoid Function</i>	16
Gambar 11. <i>Bipolar Sigmoid/Hyperbolic Tangent</i>	16
Gambar 12. <i>ReLU Function</i>	17
Gambar 13. <i>Leaky ReLU Function</i>	17
Gambar 14. Ilustrasi Penerapan <i>K-Fold Cross Validation</i>	25
Gambar 15 Lokasi Penelitian.....	40
Gambar 16 Alur Penelitian.....	46
Gambar 17 Kerangka Pikir.....	49
Gambar 18 Model ANN Tujuh Neuron Fase <i>Training</i> pada Simulasi Awal.....	60
Gambar 19 Model ANN Lima Neuron Fase <i>Testing</i> pada Simulasi Awal.....	61
Gambar 20 Model ANN Dua Neuron Hasil Simulasi Optimasi Parameter Fase <i>Testing</i>	64
Gambar 21 Grafik Perbandingan Estimasi Durasi <i>Dry Docking</i> Model ANN 5 <i>Neuron</i> dan Model Perusahaan	66
Gambar 22 Grafik Perbandingan Estimasi Durasi <i>Dry Docking</i> Model ANN 2 dan Model Perusahaan	66

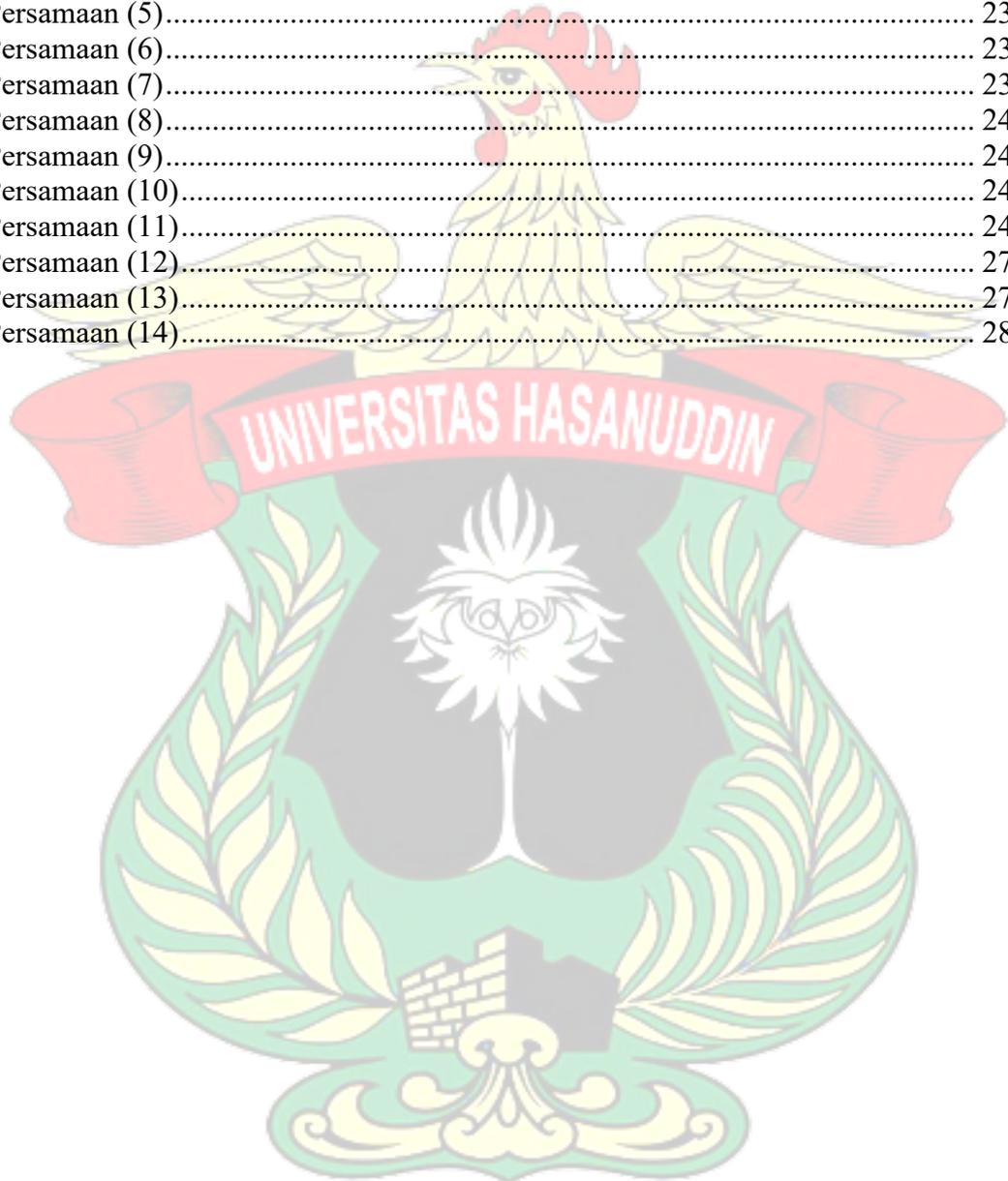
DAFTAR TABEL

Tabel 1 Deskripsi Interpretasi MAPE	28
Tabel 3. <i>Record</i> Kapal yang Dieliminasi	52
Tabel 4 Hasil Encoding Variabel Tipe Kapal	53
Tabel 5 Tabel Perbandingan Tingkat Validitas.....	54
Tabel 6 Rincian Perancangan Arsitektur ANN.....	55
Tabel 7 Contoh Rincian Pembagian <i>Dataset</i> Pada Fase <i>Training</i>	56
Tabel 8 Contoh Rincian Pembagian <i>Dataset</i> Pada Fase <i>Testing</i>	56
Tabel 9 Rincian Nilai Awal Parameter Model ANN	57
Tabel 10 Tabel Rincian <i>K-Fold Cross Validation</i>	57
Tabel 11 Rincian Nilai Parameter Model ANN Simulasi Awal	58
Tabel 12 Performa Model ANN Simulasi Awal pada Fase <i>Training</i>	59
Tabel 13 Performa Model ANN Simulasi Awal pada Fase <i>Testing</i>	60
Tabel 14 Performa Model ANN pada Simulasi Optimasi Parameter	62
Tabel 15 Performa Model ANN dengan Nilai Parameter Hasil Optimasi Parameter fase <i>Testing</i>	63



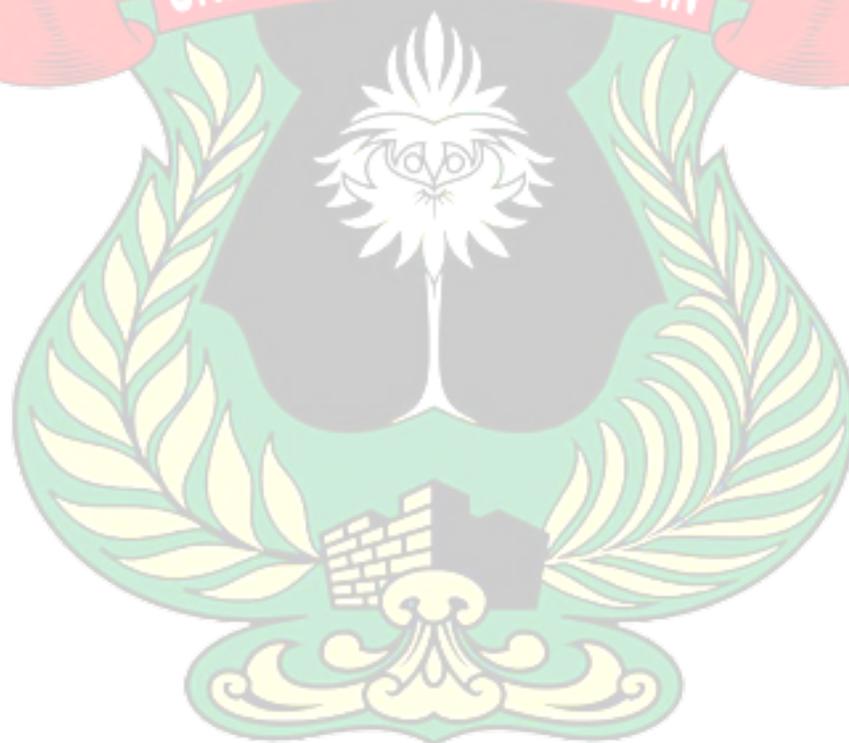
DAFTAR RUMUS

Persamaan (1).....	22
Persamaan (2).....	22
Persamaan (3).....	22
Persamaan (4).....	23
Persamaan (5).....	23
Persamaan (6).....	23
Persamaan (7).....	23
Persamaan (8).....	24
Persamaan (9).....	24
Persamaan (10).....	24
Persamaan (11).....	24
Persamaan (12).....	27
Persamaan (13).....	27
Persamaan (14).....	28



DAFTAR LAMPIRAN

Lampiran 1 Surat Pengantar PT. IKI ke Unit Kerja Bersangkutan.....	73
Lampiran 2 Surat Balasan Persetujuan Penelitian PT. IKI ke Pihak Kampus	74
Lampiran 3 Dokumen <i>Docking Record</i> Kapal PT. IKI Periode 2021-2023	75
Lampiran 4 Dokumen Kapal Masuk Fasilitas <i>Dock</i> PT. IKI	76
Lampiran 5 Hasil Konversi Data Fisik <i>Docking Record</i> ke dalam Format Excel	77
Lampiran 6 Contoh Dokumen <i>Repair List</i> Kapal Milik PT. IKI	80
Lampiran 7 Survei dan Wawancara terkait Proses <i>Dry-Docking</i>	84
Lampiran 8 Pengecekan Berkas <i>Docking Record</i> di PT. IKI.....	85
Lampiran 9 Data Model ANN pada Proses <i>Training</i> Simulasi Awal.....	86
Lampiran 10 Data Model ANN pada Proses <i>Testing</i> Simulasi Awal	116
Lampiran 11 Data Model ANN pada Proses <i>Testing</i> Hasil Simulasi Optimasi Parameter.....	121



KATA PENGANTAR

Puji syukur kita panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Optimalisasi Estimasi Durasi *Dry Docking* Kapal Menggunakan Model *Artificial Neural Network* (ANN) Studi Kasus: PT. Industri Kapal Indonesia (Persero)”. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Departemen Teknik Industri Fakultas Teknik Universitas Hasanuddin. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikannya. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah SWT. Sebagai pemberi rahmat dan pengabul doa - doa penulis hingga penulis mampu menyelesaikan tugas akhir ini.
2. Kedua orang tua, Ayah Dudung Abdulrahman dan Ibu Yaba yang telah memberikan dukungan moral dan material serta doa yang tiada hentinya untuk penulis sejak lahir hingga detik ini.
3. Adik saya Mutmainnah Annisa Shidqiyyah yang selalu mendukung dan memotivasi penulis untuk terus berjuang tanpa rasa pamrih.
4. Ibu Ir. Kifayah Amar, ST., M.Sc., Ph.D., IPU selaku Ketua Departemen Teknik Industri Fakultas Teknik Universitas Hasanuddin.
5. Ir. Rosmalina Hanafi, M.Eng., Ph.D. selaku Dosen Pembimbing I dan Ir. Dwi Handayani, S.T., M.T., IPM selaku Dosen Pembimbing II pada tugas akhir ini, karena telah bersedia memberikan kesempatan, waktu, tenaga dan pikiran selama membimbing penulis.
6. Ir. Nurfaidah Tahir, S.T., M.T., IPM selaku dosen pengampu mata kuliah *data mining* yang selalu bersedia meluangkan waktu untuk menjadi tempat berkonsultasi penulis selama mengerjakan tugas akhir, serta memberikan banyak masukan dan pandangan kepada penulis.
7. Seluruh staf dan dosen pengajar di Departemen Teknik Industri Fakultas Teknik Universitas Hasanuddin yang dengan sabar dan penuh perhatian dalam mencerdaskan anak bangsa.

8. Pihak PT. Industri Kapal Indonesia (Persero) yang terlibat, yaitu tim *quality control*, tim *planner*, dan tim *marketing* yang secara langsung terlibat dalam proses penelitian penulis. Serta seluruh unit kerja PT. IKI secara umum yang telah memberikan kesempatan bagi penulis untuk melakukan penelitian.
9. *A living source of dopamine-oxytocin for me; My fav. creature in this universe; Anisa Putri Atria; thank you for your presence that make me better & keep doing my best in this long-life journey.*
10. Kawan – kawan seperjuangan di *Houzetik* yang menjadi salah satu tempat berproses, berpulang, dan bercerita bagi penulis. Terima kasih untuk tarru, eki, niko, qalbi, accul, ipul, aaron, dimas, farid, haedir, dan mufti untuk setiap momen dan kebaikan yang selalu menyertai.
11. HEURIZTIC19 yang turut serta dalam mewarnai perjalanan hidup penulis selama berkuliah di teknik industri dan juga fase-fase lainnya dikemudian hari.
12. Keluarga besar HMTI FT–UH yang telah menjadi wadah untuk berproses. Terkhusus untuk pengurus BE HMTI FT-UH Periode 2021/2022 yang telah mewadahi penulis untuk berproses menjadi pribadi yang lebih baik.
13. *Thanks for me, my self, and I- terima kasih untuk terus bertahan hingga detik ini.*
14. Serta semua pihak yang turut membantu saya namun tidak dapat ditulis satu per satu.

Akhir kata penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu. Sekian dan terima kasih.

Gowa, 15 November 2023

Penulis

BAB I PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan negara kepulauan yang tercatat oleh Badan Pusat Statistik (BPS) pada tahun 2022 memiliki 17.001 pulau. Komposisi pulau tersebut dapat dibagi menjadi enam pulau besar dan 16.995 lainnya tersebar di 38 provinsi di Indonesia. Sementara itu, wilayah perairan dan lautan yang memisahkan pulau-pulau tersebut memiliki luas wilayah sekitar 6,32 juta km² atau setara 62% dari luas total wilayah Indonesia.

Kondisi geografis yang dimiliki Indonesia tersebut memiliki banyak tantangan dalam menghubungkan wilayah daratan yang terpisah oleh perairan atau lautan. Hal tersebut sejalan dengan pernyataan Firdhitama, W. (2018) dalam mendukung mobilisasi masyarakat dan aspek-aspek penunjang perputaran ekonomi bangsa dibutuhkan peranan transportasi penyebrangan wilayah perairan dan lautan sejenis perahu atau kapal. Ketersediaan fasilitas transportasi air sebagai salah satu tumpuan masyarakat perlu didukung dengan manajemen yang baik dan meminimalisasi kendala apapun.

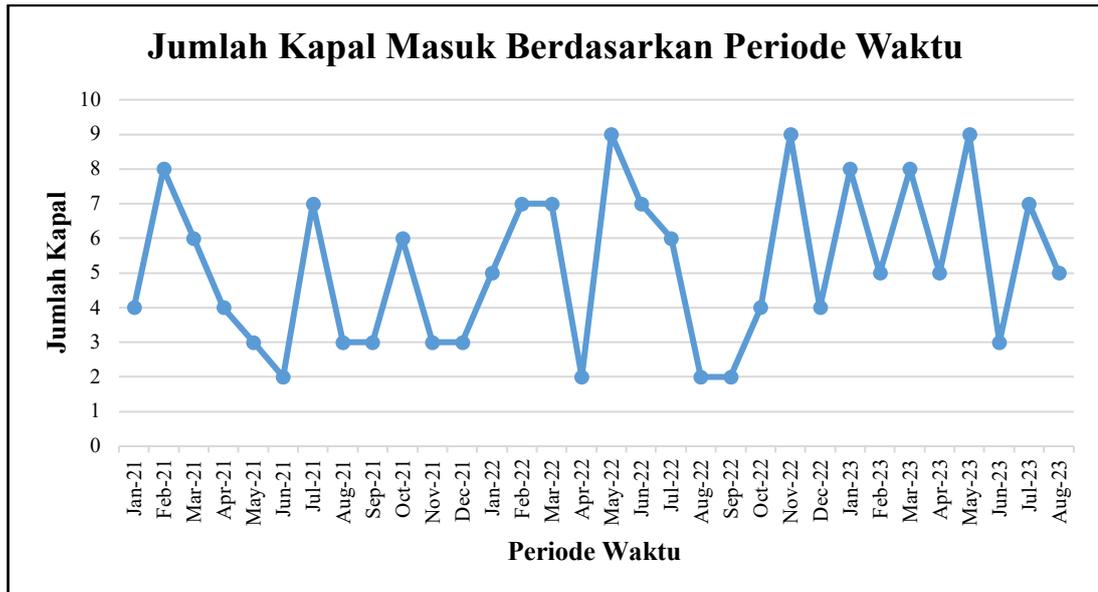
Berdasarkan laporan Kementerian Perhubungan tahun 2021, jumlah armada kapal laut yang berada di Indonesia tercatat sebanyak 72.313 (37.772 Nasional) unit dengan kenaikan sebesar 13,9% dari tahun sebelumnya yaitu sebanyak 63.390 (33.770 Nasional) unit. Menurut Novita, R. (2012) kenaikan angka pada konteks ini merupakan akibat dari penerapan asas *cabotage* yang pada dasarnya mendorong penggunaan transportasi nasional dalam memenuhi kebutuhan perekonomian bangsa. Sehingga secara langsung hal ini mewajibkan kapal untuk tetap berada pada kondisi layak untuk berlayar.

Salah satu fasilitas operasional penunjang kapal agar dapat tetap berlayar dengan kondisi yang layak adalah galangan kapal (*shipyard*). Galangan kapal menurut Daniel, Y. dalam Al-Bait M. A. (2022) merupakan sebuah tempat yang memiliki beberapa fungsi utama seperti proses pembangunan kapal (*new building*), perbaikan kapal (*ship repair*), dan pemeliharaan kapal (*ship maintenance*). Masing-masing penerapan dari ketiga fungsi ini kemudian menjadi tanggung

jawab pihak galangan kapal dengan tetap berada pada regulasi hukum yang berlaku. Hal ini sebagai bentuk pemenuhan terhadap standar yang disepakati baik secara nasional atau internasional.

Pelayanan perbaikan kapal (*ship repair*) dan pemeliharaan kapal (*ship maintenance*) merupakan dua fungsi utama galangan kapal yang menentukan suatu kapal dapat kembali berlayar dengan layak. Menurut Awal, Z. I. dan Abdullah, A. (2021) perbaikan dan perawatan merupakan proses rekayasa yang kompleks dalam melibatkan sumber daya manusia, material, serta alam dengan dampak terhadap lingkungan dan ekonomi pada masyarakat. Pelaksanaan perbaikan dan perawatan terbagi menjadi dua bagian, yaitu pada saat berada di dalam *dock (docking time)* dan ketika berada di sisi dermaga (*berthing time*). Tentunya dengan peningkatan volume jumlah kapal, maka dibutuhkan manajemen yang tepat dalam mengestimasi durasi perbaikan dan pemeliharaan kapal untuk menghasilkan proses bisnis yang optimal.

Saat ini terdapat sekitar 250 perusahaan galangan kapal di Indonesia yang masih beroperasi, 43% di Provinsi Kepulauan Riau, 39% di Pulau Jawa, dan 18% sisanya berada di luar kedua wilayah sebelumnya (Prakoso, L. Y. & Apriyani, 2017). PT. Industri Kapal Indonesia (Persero) adalah salah satu perusahaan Badan Usaha Milik Negara yang bergerak dibidang pembangunan kapal baru dan perawatan kapal – kapal yang berbahan dasar baja, aluminium maupun kapal yang terbuat dari kayu, serta merupakan galangan kapal terbesar di kawasan timur Indonesia.



Gambar 1 Jumlah Kapal Berdasarkan Periode Waktu

(Sumber: Data Sekunder dari PT. Industri Kapal Indoensia Tahun 2023)

Gambar 1 di atas menampilkan data jumlah kapal yang masuk ke PT. IKI (Persero) pada kurun waktu tiga tahun belakangan. Berdasarkan grafik di atas pada tahun 2021 terdapat 52 kapal yang masuk di galangan PT. IKI dan pada tahun 2022 terdapat 60 kapal yang masuk disepanjang tahun. Hingga pada periode bulan Agustus 2023 tercatat sudah ada 54 kapal yang masuk fasilitas galangan PT. IKI. Angka tersebut diproyeksikan akan terus bertambah hingga akhir tahun ini.

Berdasarkan hasil observasi awal oleh peneliti, peningkatan jumlah volume kapal di PT. IKI belum diimbangi dengan optimalisasi model estimasi durasi *dry-docking* kapal terhadap durasi penyelesaian sebenarnya. Data yang menampilkan jumlah kapal yang masuk di galangan PT. IKI sebelumnya, setidaknya terdapat sekitar 137 keterlambatan proyek kapal atau sekitar 86%. Keterlambatan yang terjadi di luar kontrak estimasi merugikan PT. IKI dengan harus membayar penalti berdasarkan kesepakatan dengan *owner* kapal. Pengurangan nilai penalti memang masih bisa ditempuh dengan membuat kesepakatan baru antara kedua belah pihak, namun dalam jangka panjang akan berpengaruh terhadap reputasi perusahaan.

Salah satu cara dalam membuat model alternatif yang mampu mengestimasi durasi *dry-docking* kapal dengan lebih optimal, yaitu dengan memanfaatkan *data mining*. Dokumen-dokumen kapal berupa *docking record*, *repair list*, serta *satisfaction note* yang tadinya sebagai berkas kelengkapan administrasi dan

rujukan jadwal kerja dimanfaatkan untuk membuat model estimasi alternatif melalui implementasi *data mining*.

Upaya mengimplementasikan *data mining* dalam membuat model estimasi durasi *dry-docking* menggunakan model *Artificial Neural Network* (ANN) atau Jaringan Saraf Tiruan (JST) sebagai alat pemodelan komputasi yang kompleks. Model *data mining* yang terinspirasi oleh konsep jaringan saraf manusia ini digunakan dengan mempertimbangkan kelebihan model dalam menangkap dan memahami pola-pola yang kompleks, terutama dalam hubungan antar variabel independen dan dependen yang saling terkait. Penerapan *data mining* melalui model ANN dalam mengatasi kompleksitas hubungan tersebut diharapkan dapat membantu perusahaan membuat model estimasi durasi *dry-docking* kapal yang lebih optimal.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat dirumuskan permasalahan yang terjadi, yaitu sebagai berikut.

- a. Bagaimana menentukan model estimasi durasi *dry-docking* kapal yang optimal dengan menggunakan *Artificial Neural Network*?
- b. Bagaimana hasil uji dan performa model *Artificial Neural Network* dalam mengestimasi durasi *dry docking* kapal?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah di atas, maka dapat ditetapkan tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut:

- a. Mengidentifikasi variabel-variabel yang mempengaruhi model ANN dalam proses estimasi durasi *dry-docking* kapal.
- b. Membangun model *Artificial Neural Network* untuk mengestimasi durasi *dry-docking* kapal sebagai analisis implementasi *data mining*.
- c. Menguji dan menilai performa model *Artificial Neural Network* dalam mengestimasi durasi *dry-docking* kapal.

1.4 Manfaat Penelitian

Manfaat yang diharapkan dapat diperoleh dari penelitian ini adalah:

a. Bagi Perusahaan (PT. Industri Kapal Indonesia)

Perusahaan dapat menambah sekaligus membuka peluang kerjasama dengan pihak lembaga pendidikan tinggi, dan menambah pertimbangan model alternatif lainnya dalam melakukan estimasi durasi *dry-docking* kapal.

b. Bagi Perguruan Tinggi

Hasil penelitian ini dapat digunakan sebagai rujukan dan referensi keperluan akademik, baik bagi pengembangan keilmuan Teknik Industri secara khusus, dan keilmuan lainnya secara umum.

c. Bagi Penulis

Penelitian ini sebagai pemenuhan syarat administrasi kelulusan program sarjana S-1, serta sebagai penyaluran implementasi wawasan dan keilmuan yang diperoleh selama kuliah di Departemen Teknik Industri Universitas Hasanuddin.

1.5 Batasan Masalah

Terdapat beberapa batasan masalah yang ditetapkan oleh penulis dalam mengangkat penelitian ini. Berikut adalah beberapa batasan yang ditetapkan:

a. Penelitian dilaksanakan di PT. Industri Kapal Indonesia (Persero)

b. Data yang digunakan dalam penelitian ini menggunakan data *docking record* dan *repair list* setiap kapal pada rentang periode 2021 hingga bulan Agustus 2023 milik PT. Industri Kapal Indonesia (Persero).

c. Penelitian ini berfokus pada kegiatan *dry-docking* kapal dan melibatkan pekerjaan *dry-docking* berupa perawatan rutin kapal.

d. Model *Artificial Neural Network* yang digunakan pada penelitian ini menggunakan integrasi algoritma *backpropagation* untuk mengoptimalkan kinerja model dalam mengestimasi durasi *dry-docking* kapal.

BAB II TINJAUAN PUSTAKA

2.1 *Data Mining*

Peningkatan pemanfaatan teknologi dalam dunia bisnis kian masif dan berhasil merekam berbagai informasi sesuai dengan kebutuhan organisasi. Perekaman seperti data transaksi kemudian disimpan ke dalam penampungan/gudang data (*data warehouse*). Hal ini memberikan konsekuensi logis pada pasar yang dengan menuntut pelaku bisnis untuk terus bersaing pada tingkatan yang lebih kompleks (*competitive pressure*). Sehingga menurut Muflihah, Khailil dkk (2018) dibutuhkan cara bersaing dengan pemanfaatan temuan pengetahuan di dalam *database* atau lebih dikenal dengan istilah *Knowledge Discovery in Database* (KDD).

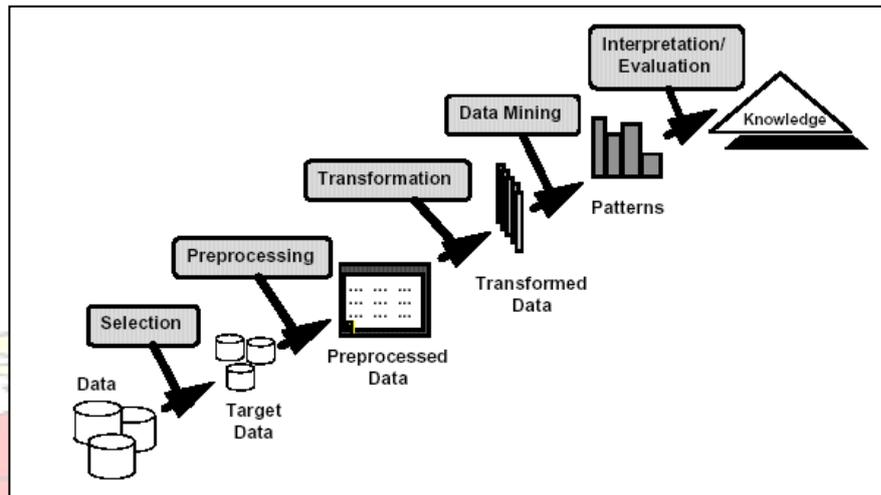
Berdasarkan penjabaran cakupan terkait dengan *Knowledge Discovery in Database* sebelumnya, maka terlihat bahwa terdapat perbedaan secara konsep mendasar antara KDD dan *data mining*. Menurut Han, J. dkk (2012) *data mining* merupakan proses atau cara untuk menemukan pola dan pengetahuan yang berasal dari sejumlah besar data. Selain itu, definisi *data mining* menurut Suntoro (2019) merupakan proses mendapatkan informasi baru yang berguna dengan mengekstraksi data dari sumber yang besar secara kuantitas dan digunakan untuk membantu pengambilan keputusan.

Jika kedua definisi *data mining* di atas dielaborasi dan ditinjau dalam konteks bisnis, maka proses *data mining* merupakan proses analisis dari sejumlah besar data yang hasilnya diinterpretasi sebagai pengetahuan dan digunakan untuk peningkatan keuntungan, minimasi pengeluaran, hingga optimalisasi sistem di dalam suatu perusahaan atau organisasi.

Pada dasarnya *data mining* menurut Steinbach dan Kumar dalam Muflihah, Khailil dkk (2018) merupakan disiplin ilmu yang memanfaatkan disiplin ilmu lainnya, seperti penggunaan metode statistik, matematika, *Artificial Intelligent* (AI), *database system*, dan *machine learning*.

2.1.1 Tahap Penerapan *Knowledge Discovery in Database* (KDD)

Implementasi *Knowledge Discovery in Database* (KDD) menurut Han dan Kamber dalam Arta, I., dkk (2019) terdiri atas tahapan-tahapan penting. Tahapan-tahapan tersebut direpresentasikan melalui Gambar 2 berikut.



Gambar 2. Proses Penerapan *Knowledge Discovery in Database*
(Sumber: Arta, I., dkk, 2019)

a. *Data Collection*

Proses KDD tentunya dimulai dengan proses pengumpulan data yang dibutuhkan. *Database* tidak hanya menjadi satu-satunya sumber data yang digunakan, melainkan juga memungkinkan untuk mengumpulkan data dari berbagai sumber relevan lainnya baik secara langsung maupun tidak.

b. *Data Cleaning & Data Integration*

Pada tahapan *data cleaning* dan *data integration* data akan diproses dengan pemeriksaan terhadap duplikasi data, inkonsistensi data, validasi data, dan melakukan penambahan data dengan informasi lain yang dibutuhkan pada proses KDD.

c. *Data Selection*

Seleksi data (*Data Selection*) merupakan lanjutan tahapan setelah proses *cleaning* dan *integration* dengan berfokus pada pemilihan data yang dianggap relevan dan memenuhi syarat untuk digunakan dalam proses analisis. Umumnya pada proses ini, data yang telah dipisahkan akan disimpan ke dalam *separate database* yang berbeda dari *main database* utama (*induk*).

d. *Data Transformation*

Proses transformasi data dilakukan untuk mengubah format tipe data ke dalam bentuk tertentu sesuai dengan rancangan implementasi metode *data mining* yang akan digunakan pada tahap berikutnya. Contohnya penyesuaian terhadap kebutuhan fitur dan label.

e. *Data Mining*

Sesuai dengan namanya, *data mining* merupakan proses dalam KDD yang melakukan ekstraksi atau penggalian informasi menggunakan teknik tertentu atau memanfaatkan algoritma *data mining*. Proses ini bertujuan untuk menemukan pola, relasi, atau informasi tersembunyi yang tidak diketahui sebelumnya.

f. *Evaluation & Interpretation*

Model *data mining* yang sebelumnya diterapkan kemudian akan dievaluasi untuk mengetahui apakah telah optimal dalam menjalankan fungsinya. Tentunya proses ini memungkinkan untuk model dapat disesuaikan kembali dan hasil akhirnya akan diterjemahkan.

g. *Knowledge Representation*

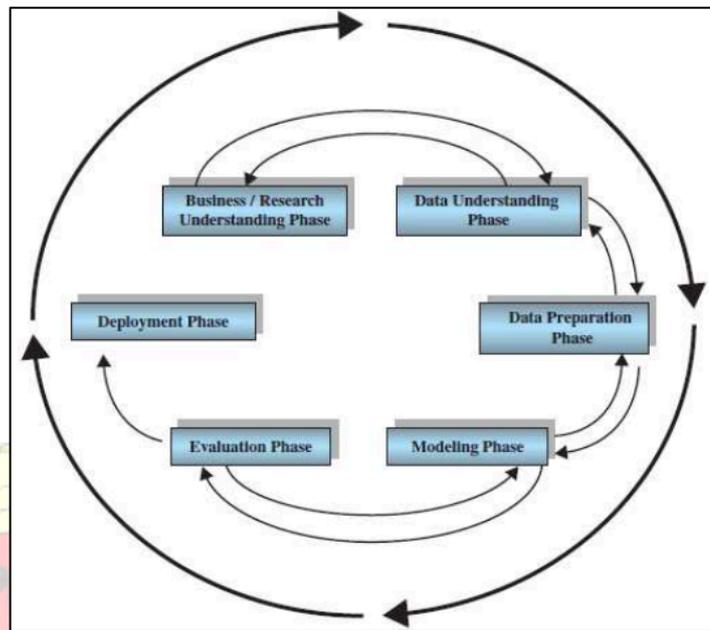
Tahapan ini berfokus pada menampilkan informasi dan pengetahuan yang telah ditemukan sebelumnya dalam konteks yang mudah dipahami oleh manusia. Umumnya proses ini memiliki beberapa opsi cara penyampaian, seperti bentuk laporan, grafik, diagram, dan bentuk lainnya.

2.1.2 Proses Standar *Data Mining* antar Industri (CRISP-DM)

Proses standar data mining antar industri atau yang lebih dikenal sebagai *Cross-Industry Standard Process for Data Mining* (CRISP-DM) merupakan standarisasi terhadap serangkaian proses *data mining* sebagai strategi pemecahan masalah secara umum seperti bisnis atau secara khusus pada konteks penelitian akademik. Menurut Larose dalam Febian, D. dkk (2017) CRISP-DM dikembangkan sejak tahun 1966 oleh beberapa analis dari berbagai industri seperti Daimler Chrysler, SPSS, dan NCR.

CRISP-DM sebagai standar yang menjembatani kesesuaian antar pelaku industri yang menggunakan *data mining*, maka dalam penerapannya terdapat

enam fase untuk satu siklusnya. Keenam fase tersebut ditunjukkan melalui Gambar 3 berikut:



Gambar 3. Siklus CRISP-DM
(Sumber: Febian, D., dkk, 2017)

a. *Business/Research Understanding*

Fase pemahaman bisnis merupakan penetapan implementasi dari kerangka *data mining* terkait dengan mekanisme bisnis secara utuh. Kerangka tersebut dapat berupa tujuan bisnis, proses bisnis, serta situasi dan kondisi bisnis.

b. *Data Understanding*

Fase pemahaman data mencakup kesesuaian antara data yang dibutuhkan dengan rancangan model *data mining*. Fase ini mempelajari dan mengidentifikasi data apa saja yang mampu memberikan kontribusi terhadap pengembangan model *data mining* untuk mencapai tujuan.

c. *Data Preparation*

Fase persiapan data berfokus pada format kumpulan data yang sebelumnya telah diidentifikasi dan diambil. Format data yang dimaksud disini seperti identifikasi konsistensi data, mengatasi *missing value* dan *outlier*, serta transformasi data jika dibutuhkan untuk memastikan dapat digunakan oleh model yang dibangun.

d. *Data Modelling*

Fase pemodelan data adalah proses membuat model *data mining* yang sebelumnya telah ditetapkan dengan menggunakan berbagai *tools data mining*, algoritma *data mining*, sekaligus menentukan parameter pada model.

e. *Evaluation*

Fase evaluasi setelah pemodelan data sebelumnya dimaksudkan untuk melihat performa model yang digunakan dalam menyelesaikan masalah, kemudian melakukan penyesuaian ulang pada model jika tujuan sebelumnya belum tercapai melalui hasil interpretasi.

f. *Deployment*

Fase penggunaan model merupakan lanjutan fase evaluasi model *data mining* yang dianggap telah sesuai untuk menyelesaikan permasalahan. Pada fase ini juga model *data mining* yang sebelumnya digunakan dapat dipertimbangkan kembali terhadap kasus lainnya. Selain itu, laporan (*procedural report*) hasil penerapan model *data mining* juga masuk dalam cakupan fase ini.

Keenam fase ini saling berhubungan melalui satu siklus. Optimal tidaknya suatu fase tergantung bagaimana eksekusi fase sebelumnya dijalankan. Pada proses CRISP-DM dimungkinkan untuk melakukan iterasi balik untuk mengatasi permasalahan yang muncul pada suatu fase dengan meninjau fase sebelumnya.

2.1.3 Tipe Pekerjaan *Data Mining*

Data mining sebagai domain dari berbagai disiplin ilmu dapat melakukan berbagai tipe pekerjaan atau fungsi yang dapat dijalankan. Menurut Larose dalam Rerung, R. (2018), secara umum terdapat enam tipe pekerjaan *data mining* yang dapat dilakukan, yaitu:

a. Deskripsi (*Description*)

Tipe deskripsi merupakan tipe pekerjaan *data mining* yang memiliki tujuan untuk menggambarkan pola data atau *trend* dari sekumpulan besar data.

b. Estimasi (*Estimation*)

Estimasi dibuat berdasarkan data lengkap yang terdiri dari *target variable* dan *predictor variable*. *Target variable* merupakan variabel yang diestimasi.

Predictor variable merupakan variabel yang menjadi pertimbangan dalam mengestimasi *target variable*.

c. Kluster (*Clusteritation*)

Tipe kluster merupakan pengelompokkan objek yang sama ke dalam grup atau kelas. Kluster bertujuan untuk mengelompokkan seluruh data ke dalam grup atau kelas yang relatif homogen, dimana persamaan data di luar grup atau kelas diminimalkan. Umumnya, kluster dilakukan sebagai tahap awal dari sebuah proyek dan hasil yang didapat digunakan sebagai input untuk metode lain.

d. Asosiasi (*Association*)

Asosiasi dalam *data mining* adalah tipe pekerjaan dengan tujuan untuk menemukan hubungan antar variabel, lebih detailnya adalah untuk menemukan aturan- aturan yang mengkuantifikasikan hubungan antar dua atau lebih variabel. Aturan asosiasi yang telah dihasilkan dapat digunakan untuk kepentingan perusahaan.

e. Klasifikasi (*Classification*)

Klasifikasi merupakan salah satu tipe pekerjaan dalam *data mining* yang bertujuan untuk mempartisi data hingga terbagi ke dalam kelas-kelas dengan berdasar pada data historis. Terdapat dua tipe variabel dalam klasifikasi, yaitu *target variable* dan *predictor variable*. *Target variable* merupakan variabel yang berbentuk kategorikal dan berisikan kategori-kategori. Kategori-kategori tersebut menjadi kelas-kelas dalam klasifikasi. Sedangkan *predictor variable* merupakan variabel yang menjadi dasar pertimbangan dalam mengklasifikasi data ke dalam kelas-kelas.

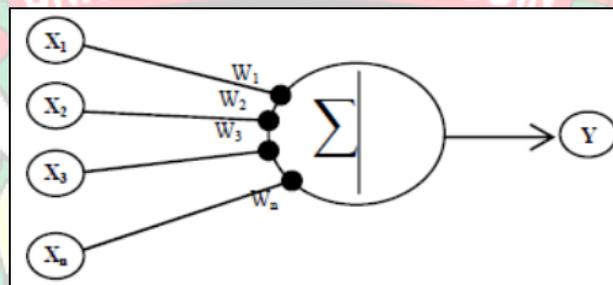
f. Prediksi (*Prediction*)

Serupa dengan klasifikasi dan estimasi, perbedaannya terletak pada jangka waktu. Prediksi bertujuan untuk meramalkan suatu kejadian di masa mendatang. Contohnya adalah penggunaan *data mining* untuk memprediksi persentase kenaikan tingkat kematian selama satu tahun ke depan.

2.2 Jaringan Saraf Tiruan (JST)/*Artificial Neural Network* (ANN)

Artificial Neural Network (ANN) merupakan model *machine learning* yang pertama kali diperkenalkan oleh McCulloch & Pitts pada tahun 1943. *Artificial Neural Network* (ANN) menurut El-Shahat dalam Sutoyo, Edi dan Fadlurrahman, M. A. (2020) merupakan struktur komputasi yang didasarkan oleh sistem jaringan saraf biologis dengan serangkaian proses melalui *neuron*.

Berdasarkan mekanisme sistem kerja saraf manusia sebagai inspirasi lahirnya *Artificial Neural Network* (ANN), maka algoritma ini juga mengadopsi cara kerja yang serupa dalam penerapannya. Sehingga karakteristik ANN terdiri atas bagian yang berfungsi menerima sinyal elemen input (X_n), relasi (terdapat nilai pembobotan (W_n) untuk setiap hubungan *neuron*) menuju suatu fungsi aktivasi ($\Sigma(f(x))$), dan *output* (Y). Berikut konsep dasar ANN yang dituangkan melalui Gambar 4 berikut.



Gambar 4. Konsep Dasar *Artificial Neural Network* (Hukubune, A. J., 2022)
(Sumber: Hukubune, A. J., 2022)

Struktur *Artificial Neural Network* menurut Hukubune, A. J. (2022) pada dasarnya memiliki tiga lapisan, yaitu:

- Lapisan input atau berupa masukan (*buffer*) merupakan bagian lapisan *Neural Network* yang berfungsi sebagai penerima nilai variabel independen yang ditetapkan, dan memiliki jumlah *neuron* berdasarkan jumlah fitur yang digunakan.
- Lapisan tersembunyi (*hidden layer*) merupakan lapisan yang berfungsi untuk menghubungkan lapisan *input* dan *output* dengan melalui relasi/pembobotan pada nilai input yang diberikan, serta melalui fungsi aktivasi yang ditetapkan di dalam aksitektur jaringannya.
- Lapisan *output* atau lapisan keluaran merupakan lapisan yang memberikan hasil akhir dari serangkaian proses pada lapisan sebelumnya. Lapisan *output*

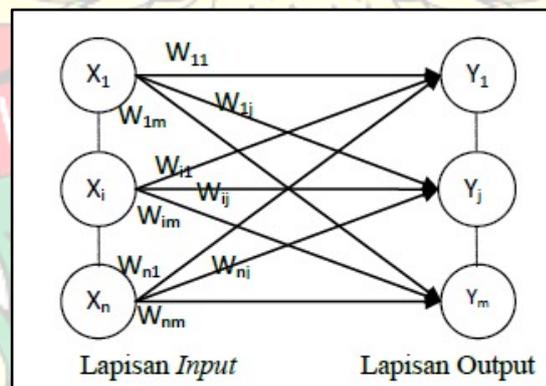
memiliki jumlah *neuron* yang disesuaikan dengan tujuan penerapannya (klasifikasi, estimasi, prediksi, dan lainnya.)

2.2.1 Jenis-jenis Arsitektur *Artificial Neural Network*

Menurut Aprilia, D., dkk (2013) berdasarkan struktur yang dimiliki ANN, maka secara umum dapat dibagi menjadi tiga, yaitu:

a. *Single-Layer Neural Network* (Jaringan Saraf Tunggal)

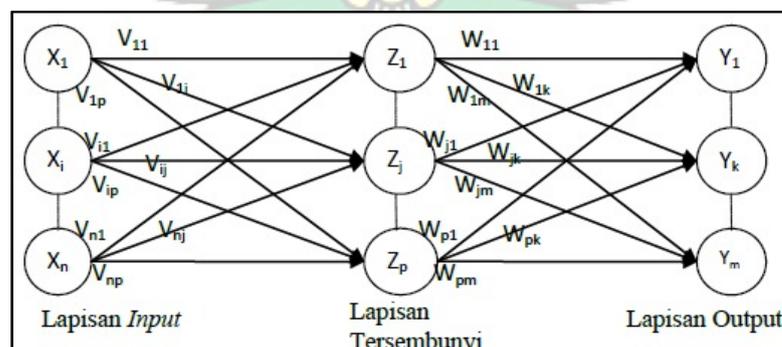
Jenis *Neural Network* ini merupakan jenis *feed-forward* yang hanya terdiri atas *input layer* dan *output layer*. *Input* data langsung menuju ke nilai *output* melalui pembobotan pada sinapsis koneksinya. Arsitektur ANN *single layer* ditampilkan melalui Gambar 5 berikut.



Gambar 5. Arsitektur *Single-Layer Neural Network*

b. *Multi-Layer Perceptron Neural Network* (Jaringan Lapis Banyak)

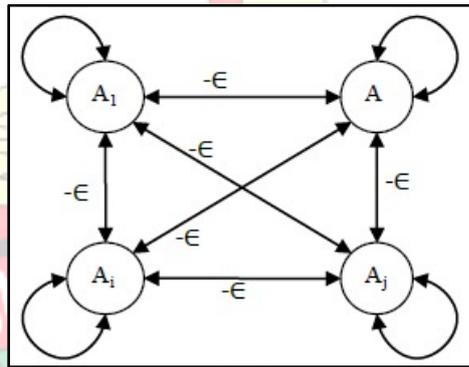
Berbeda dengan jenis sebelumnya, jenis *multi-layer* ini melibatkan banyak *layer* yang saling terhubung secara *feedforward*. Secara sederhana, model pada jenis ini memiliki *hidden layer* diantara *input layer* dan *output layer*. Model ini melakukan perhitungan sesuai *input layer*, pembobotan pada setiap koneksi, dan menjalankan fungsi aktivasi. Secara konsep dituangkan melalui Gambar 6.



Gambar 6. Arsitektur *Multi-Layer Neural Network*

c. *Recurrent Network* (Jaringan Lapisan Kompetitif)

Lapisan berbeda dari lapisan tunggal dan jamak, lapisan ini antara *neuron* satu dengan yang lainnya saling berhubungan dan mempunyai peraturan bobot sudah ditetapkan dan tidak mempunyai proses pelatihan. Pada lapisan jaringan ini mempunyai minimal satu *feedback loop* yang terdiri dari masing-masing *neuron* untuk memberikan kembali *output*-nya sebagai input pada *neuron* lain. Berikut Gambar 7 yang menampilkan konsep *recurrent network* pada ANN.



Gambar 7. Arsitektur *Recurrent Network*

2.2.2 Fungsi Aktivasi *Artificial Neural Network*

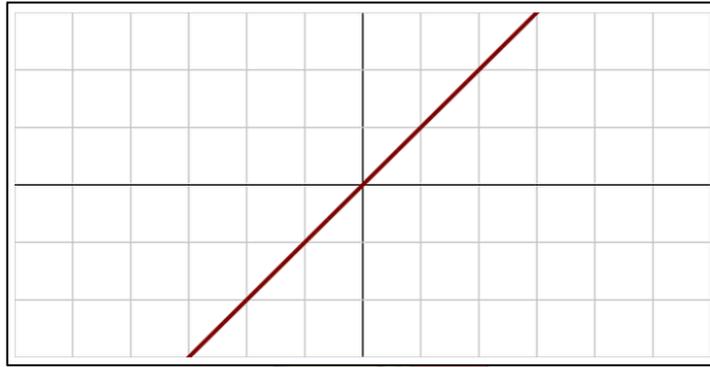
Pengembangan Menurut Julpan, dkk (2015), fungsi aktivasi merupakan suatu fungsi yang digunakan untuk mengaktifkan atau tidak mengaktifkan neuron di dalam arsitektur ANN. Berikut beberapa jenis fungsi aktivasi yang umumnya digunakan dalam ANN:

a. *Identity Function*

Fungsi aktivasi identitas (*identity function*) atau fungsi aktivasi linear merupakan jenis fungsi aktivasi yang memiliki mekanisme yang mengembalikan nilai *output* menjadi negatif sesuai dengan nilai inputnya. Jika didefinisikan secara matematis, maka dapat melalui fungsi berikut:

$$f(x) = x$$

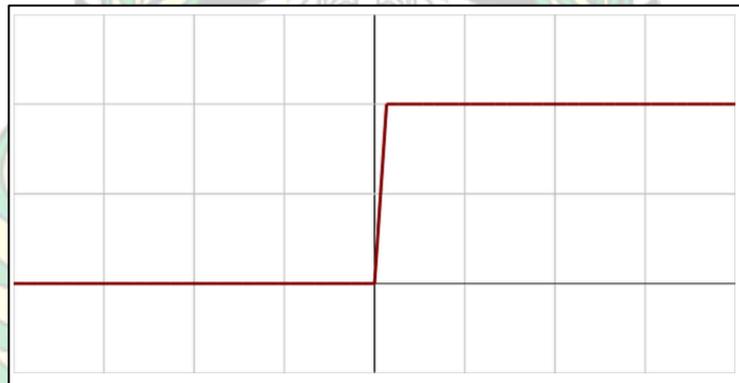
Berikut visualisasi pada gambar 8 terkait fungsi aktivasi identitas melalui dua koordinat hasil fungsi (x, y):

Gambar 8. *Identity Function*b. *Binary Step*

Binary step function merupakan salah satu jenis fungsi aktivasi yang mekanisme kerjanya dengan mengembalikan nilai menjadi 1 jika $x \geq 0$, dan mengembalikan nilai menjadi 0 jika $x < 0$. Sehingga *output* dari fungsi aktivasi ini hanya akan menghasilkan nilai 0 atau 1.

$$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ 1, & \text{for } x \geq 0 \end{cases}$$

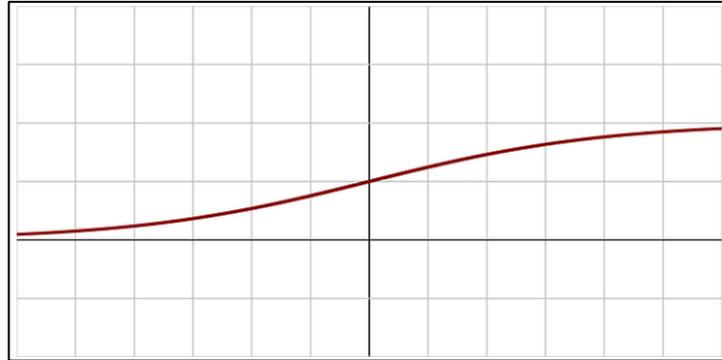
Jika divisualisasikan, maka fungsi aktivasi *binary step* akan menampilkan representasi fungsi melalui gambar 9 berikut:

Gambar 9. *Binary Step Function*c. *Sigmoid (Logistic Function)*

Fungsi aktivasi *sigmoid* atau dikenal juga dengan *logistic function* memiliki mekanisme menghasilkan nilai fungsi berada pada rentang $[0, 1]$, untuk menghasilkan nilai tersebut digunakan rumus sebagai berikut:

$$f(x) = \frac{1}{1 + \exp(-x)}$$

Secara visual fungsi *sigmoid* memiliki kurva yang berubah secara eskponensial hingga titik tertentu mengalami saturasi dan menjadi linear hingga seterusnya. Fungsinya ditampilkan melalui gambar 10 berikut.



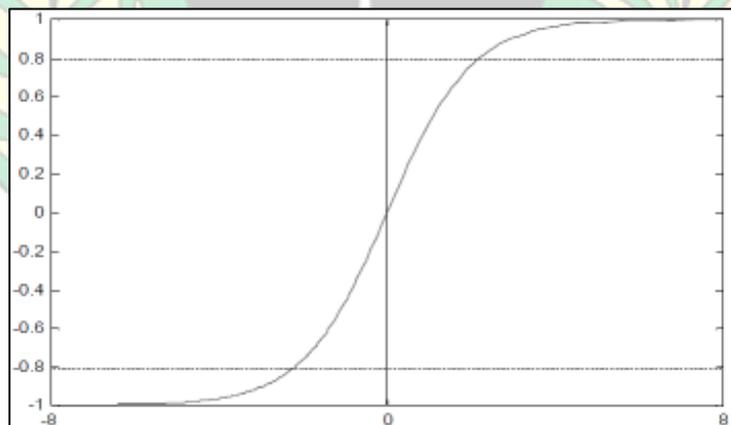
Gambar 10. *Sigmoid Function*

d. *Bipolar Sigmoid/Hyperbolic Tangent*

Fungsi aktivasi *bipolar sigmoid* atau disebut juga fungsi aktivasi *TanH* merupakan bentuk pengembangan dari fungsi aktivasi *sigmoid* dengan nilai *output* berada pada rentang $[-1,1]$, fungsi ini secara matematis dapat didefinisikan sebagai berikut:

$$f(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$$

Sehingga kurva yang dimiliki oleh fungsi aktivasi ini juga mengalami perubahan layaknya fungsi aktivasi *sigmoid* dengan perbedaan rentang nilai *output* melalui Gambar 11 berikut:



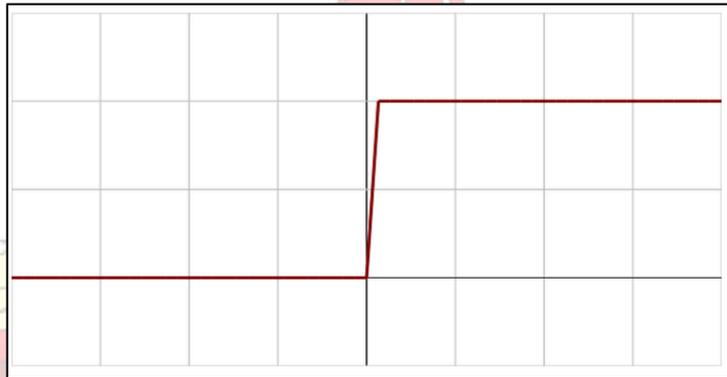
Gambar 11. *Bipolar Sigmoid/Hyperbolic Tangent*

e. *Rectified Linear Unit (ReLU)*

Fungsi aktivasi *Rectified Linear Unit (ReLU)* memiliki mekanisme matematis sebagai berikut:

$$f(x) = \begin{cases} 0, & \text{for } x \leq 0 \\ 1, & \text{for } x > 0 \end{cases}$$

Kurva fungsi aktivasi ReLU divisualisasikan melalui Gambar 12 berikut:



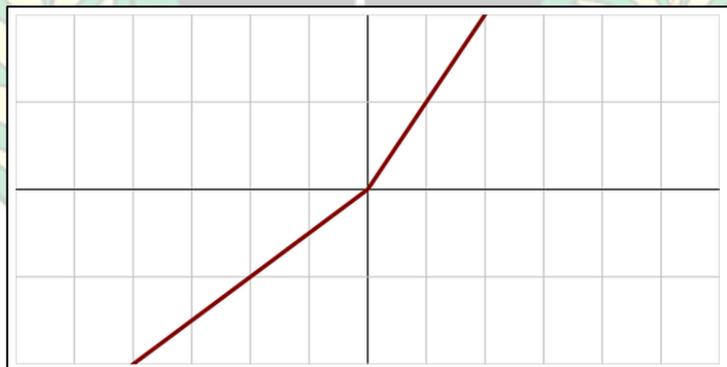
Gambar 12. *ReLU Function*

f. *Leaky Rectified Linear Unit (Leaky ReLU)*

Jenis fungsi ini merupakan bentuk pengembangan dari fungsi ReLU yang memiliki toleransi sedikit nilai gradient positif ketika unit tidak aktif. Sehingga secara matematis fungsi ini memiliki mekanisme sebagai berikut:

$$f(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.01x, & \text{otherwise} \end{cases}$$

Sementara itu, visualisasi kurva Leaky ReLU melalui Gambar 13 berikut.



Gambar 13. *Leaky ReLU Function*

2.3 Machine Learning

Machine learning menurut Sari, P. (2022) merupakan salah satu bagian disiplin ilmu dari kecerdasan buatan (*Artificial Intelligence*). Pendekatan *machine learning* dapat digunakan untuk menyelesaikan permasalahan kompleks di berbagai bidang yang tidak dapat diselesaikan dengan program konvensional, misalnya seperti mendeteksi penyakit di bidang kesehatan, program komputer untuk mengajar dibidang pendidikan, pengenalan wajah manusia, pencarian informasi, analisis pasar, dan sebagainya. *Machine learning* dapat dikategorikan ke dalam dua kelompok, yaitu *supervised learning* dan *unsupervised learning* (Azis, A. I. & Santoso, B., 2020).

Kategori *supervised learning* merupakan jenis pembelajaran mesin di mana model diberikan data yang sudah dilabeli (data latih) yang berisi *input* dan *output* yang sesuai. Tujuan dari *supervised learning* adalah untuk menghasilkan model atau algoritma yang dapat memetakan *input* ke *output* atau label yang benar. Sementara itu, kategori *unsupervised learning* merupakan jenis pembelajaran mesin di mana model diberikan data yang tidak memiliki label atau keluaran yang benar. Tujuan dari *unsupervised learning* adalah untuk mengungkap struktur dalam data, seperti pola, grup, atau hubungan antar data (Azis, A. I. & Santoso, B., 2020).

2.4 Algoritma

Algoritma merupakan serangkaian langkah penyelesaian masalah yang logis dan disusun secara sistematis. Batasan serangkaian langkah yang logis dan sistematis tersebut dimaksudkan untuk memastikan benar tidaknya *output* pemecahan masalah (Rambe P., 2021). Sementara itu menurut Gun Gun Maulana (2017), algoritma merupakan suatu metode efektif yang diekspresikan sebagai rangkaian terbatas melalui kumpulan perintah untuk menyelesaikan masalah. Berdasarkan kedua definisi tersebut dapat disimpulkan bahwa algoritma merupakan serangkaian instruksi yang disusun secara sistematis dan logis untuk menyelesaikan suatu masalah.

Struktur dasar algoritma menurut Gun Gun Maulana (2017) terdiri dari tiga bagian, yaitu struktur sekuensial (*sequential*) yang dimaksudkan pada langkah-langkah berurut seperti langkah pertama hingga akhir, struktur seleksi (*branching/selection*) yang menyatakan pemilihan langkah yang didasarkan oleh suatu kondisi atau *decision making*, dan struktur perulangan (*iteration*) yang memungkinkan suatu algoritma melakukan perintah dengan jumlah sesuai dengan instruksi yang diberikan. Berikut merupakan ketiga jenis algoritma menurut Gun Gun Maulana (2017):

2.4.1 Struktur Dasar Urut (*Sequence*)

Struktur urut merupakan struktur algoritma yang mengerjakan jenis program melalui perintah beruntun atau baris per baris mulai dari awal hingga akhir. Pada struktur urut ini tidak terdapat lompatan atau pengulangan dalam perintahnya. Karakteristik yang dimiliki struktur urut pada suatu algoritma adalah sebagai berikut:

- a. Setiap perintah dikerjakan satu per satu sebanyak satu kali.
- b. Eksekusi perintah dilakukan secara berurutan.
- c. Perintah terakhir merupakan akhir dari algoritma.

Perubahan urutan dapat memberikan *output* yang berbeda.

2.4.2 Struktur Dasar Pemilihan (*Selection*)

Struktur pemilihan adalah jenis struktur algoritma yang pada dasarnya tetap dikerjakan secara berurut, namun terdapat kondisi pengujian untuk menentukan opsi proses apa yang akan dijalankan. Kondisi pengujian ini yang kemudian secara umum membagi struktur pemilihan menjadi tiga jenis, yaitu:

- a. Struktur Pemilihan Satu Kondisi

Struktur pemilihan dengan satu kondisi hanya memiliki satu kriteria kondisi dalam perintahnya. Sehingga skema proses bergantung pada terpenuhi atau tidaknya kondisi tersebut. Misalnya jika kondisi terpenuhi maka proses berikutnya akan dieksekusi, namun jika tidak terpenuhi maka proses berikutnya akan dilewati atau tidak dieksekusi.

b. Struktur Pemilihan Dua Kondisi

Struktur pemilihan dengan dua kondisi memberikan skema algoritma untuk mengeksekusi salah satu diantara dua proses berdasarkan kondisi pertama atau kondisi kedua yang terpenuhi.

c. Struktur Pemilihan Tiga Kondisi atau Lebih

Struktur pemilihan dengan tiga kondisi atau lebih memungkinkan algoritma pemilihan menetapkan beberapa kondisi untuk diterapkan di dalam strukturnya dan mengeksekusi salah satu proses berdasarkan mana kondisi yang terpenuhi diantara beberapa opsi.

2.4.3 Struktur Dasar Pengulangan (*Iteration*)

Struktur pengulangan merupakan jenis algoritma dengan struktur yang memiliki perintah untuk melakukan perintah yang sama berdasarkan syarat perintah yang diberikan, baik secara kondisi maupun jumlah pengulangannya (*iterasi*). Berdasarkan perintah yang diberikan, maka jenis struktur pengulangan dapat dibagi menjadi dua seperti berikut:

a. Pengulangan dengan Kondisi (*While Loop*)

Pengulangan berdasarkan kondisi pada suatu struktur algoritma akan memungkinkannya untuk menjalankan serangkaian proses selama kondisi tertentu terpenuhi. Skema dasarnya adalah jika kondisi tertentu terpenuhi (*true*), kemudian dievaluasi, dan perintah yang ditetapkan akan dieksekusi. Proses ini kemudian akan terus berulang hingga kondisi menjadi tidak terpenuhi lagi (*false*).

b. Pengulangan dengan Jumlah Iterasi yang Ditentukan (*For Loop*)

Struktur pengulangan dengan jenis ini memungkinkan algoritma yang dirancang untuk mengotomatisasi eksekusi sebanyak berulang kali sebanyak perintah yang diberikan. Umumnya, perancang algoritma pengulangan jenis ini akan menetapkan jangkauan nilai tertentu untuk menghentikan proses iterasi.

2.4.4 *Algoritma Backpropagation*

Backpropagation Algorithm atau diterjemahkan sebagai algoritma propagasi balik merupakan hasil buah pikir Werbos, kemudian disebar luaskan oleh Rumelhart dan McClelland melalui penerapannya di dalam ANN. Algoritma ini masuk dalam jenis algoritma pengulangan (*iteration*) yang menitik beratkan pada kondisi yang harus terpenuhi (*while loop*). Algoritma *backpropagation* merupakan bagian dari domain *supervised learning* dan didesain pada *feed forward multi-layers* ANN. Menurut Anwar dalam Hukubune, Ataline J. (2022) algoritma *backpropagation* merupakan algoritma yang termasuk dalam kategori *supervised learning*, dan diimplementasikan di dalam ANN untuk melakukan pencarian nilai pembobotan paling optimal dalam menghasilkan nilai *error* seminimal mungkin berdasarkan pembelajaran yang diberikan.

Sejalan dengan pernyataan sebelumnya terkait algoritma *backpropagation*, Putra, Hasdi dan Walmi, Nabilah U., (2020) penggunaan *backpropagation* pada ANN bertujuan untuk meminimalkan *error* terhadap jaringan yang menghasilkan keluaran (*output*).

Minimasi *error* pada hasil yang dikeluarkan oleh ANN dengan algoritma *backpropagation* merupakan *output* mekanisme kerja algoritma ini. Sementara itu, gambaran umum terkait dengan cara kerja *backpropagation* adalah melalui proses iteratif pada kumpulan data latih (*training dataset*), kemudian membandingkan hasil antara keluaran model terhadap data uji (*testing dataset*). Pada setiap proses iterasi model ANN, bobot yang digunakan pada relasi (*synapsis*) akan terus dimodifikasi untuk mencapai nilai minimum dari *Mean Square Error* (MSE) antara nilai yang diprediksi dari model dan nilai sebenarnya (*ground truth*). Proses modifikasi bobot pada setiap iterasi ini dilakukan dengan skema arah mundur, mulai dari *output layer* hingga *layer* pertama dari *hidden layer*. Skema arah mundur inilah yang kemudian mendasari istilah *backpropagation* (Julpan, dkk., 2015).

Terdapat tiga tahapan pada penerapan algoritma *backpropagation* antara lain *feedforward* (umpan maju), *backpropagation* (umpan mundur), dan modifikasi bobot serta bias. Langkah-langkah dalam penerapan algoritma *backpropagation* adalah sebagai berikut:

a. Langkah 0:

- 1) Menetapkan target *error*, *learning rate*, maksimum *epochs* dan jumlah *neuron* pada *hidden layer*.
- 2) Kondisi berhenti ketika MSE/MAPE (parameter evaluasi *error*) atau *epochs* berada pada nilai maksimum simulasi.

b. *Feedforward*

- 1) Langkah 1: Setiap unit input ($X, i = 1, 2, 3, \dots, n$) menerima sinyal input X_i dan meneruskan sinyal ini kepada semua unit pada *hidden layer*.
- 2) Langkah 2: Masing-masing unit *hidden layer* akan menjumlahkan sinyal input yang terboboti termasuk biasnya dan dirumuskan sebagai berikut:

$$Z_{net_j} = V_{0j} + \sum X_i V_{ij} \quad \text{Persamaan (1)}$$

Dimana:

Z_{net} = jumlah sinyal *neuron* input *layer* yang masuk menuju *neuron hidden layer*

V_{0j} = bobot bias *hidden layer*

X_i = unit input *layer*

V_{ij} = Bobot sinyal dari *neuron* input *layer* yang menuju *neuron hidden layer*

Selanjutnya digunakan fungsi aktivasi untuk menghitung sinyal *output* menggunakan jenis fungsi aktivasi yang dipilih dalam metode.

- 3) Langkah 3: Tiap-tiap unit di *output layer* ($Y_k, k = 1, 2, 3, \dots, m$) menjumlahkan sinyal-sinyal masuk yang berbobot sebagai berikut:

$$Y_{net_k} = W_{0k} + \sum Z_j W_{jk} \quad \text{Persamaan (2)}$$

Dimana

Y_{net_k} = jumlah sinyal yang masuk menuju *neuron output layer*

W_{0k} = bobot bias *output layer*

W_{jk} = bobot sinyal dari *neuron hidden layer* yang masuk ke *neuron output layer*

c. *Backpropagation*

- 1) Langkah 4: Hitung faktor kesalahan (δ) pada *output* jaringan Y_k ($k = 1, 2, \dots, m$)

$$\delta_k = (t_k - Y_k) f'(Y_{net_k}) = (t_k - Y_k) Y_k (1 - Y_k) \quad (3)$$

Dimana:

δ_k = Faktor kesalahan *output* jaringan

t_k = Target yang dicapai

Selanjutnya, hitung koreksi bobot (digunakan untuk memperbaiki nilai W_{jk}) persamaan yang digunakan sebagai berikut:

$$\Delta W_{jk} = \alpha \delta_k Z_j \quad (4)$$

Dimana:

ΔW_{jk} = suku perubahan bobot

α = konstanta laju pelatihan (*learning rate*)

Kemudian dilakukan perhitungan nilai korelasi bias (digunakan untuk memperbaiki nilai W_{0k}) dengan menggunakan persamaan berikut ini:

$$\Delta W_{0k} = \alpha \delta_k \quad (5)$$

Dimana:

ΔW_{0k} = Suku perubahan bobot bias pada *output layer*

- 2) Langkah 5: Hitung faktor kesalahan sinyal yang masuk menuju *neuron* pada *hidden layer* berdasarkan faktor kesalahan disetiap *neuron* pada *hidden layer* dengan *output layer*:

$$\delta_{net_j} = \sum_{k=1}^m \delta_k W_{jk} \quad (6)$$

Dimana:

δ_{net_j} = Faktor kesalahan sinyal yang masuk ke *neuron* pada *hidden layer*

Selanjutnya, kalikan nilai ini dengan turunan fungsi aktivasinya untuk mencari faktor kesalahan setiap *neuron* pada *hidden layer* melalui persamaan berikut:

$$\delta_j = \delta_{net_j} f'(Z_{net_j}) = \delta_{net_j} Z_j (1 - Z_j) \quad (7)$$

Dimana:

δ_j = faktor kesalahan pada *hidden layer*

Selanjutnya, hitung perubahan bobot yang digunakan untuk memperbaiki V_{ij} menggunakan persamaan berikut ini:

$$\Delta V_{ij} = \alpha \delta_j X_i \quad (8)$$

Dimana:

ΔV_{ij} = suku perubahan bobot sinyal *neuron* input *layer* yang menuju ke *neuron hidden layer*

Hitung suku perubahan bobot bias yang digunakan untuk memperbaiki V_{0j} sebagai berikut ini:

$$\Delta V_{0j} = \alpha \delta_j \quad (9)$$

Dimana:

ΔV_{0j} = suku perubahan bobot bias pada *hidden layer*

d. Perubahan Bobot dan Bias

- 1) Langkah 6: Tiap unit pada *output layer* ($Y_k, k = 1, 2, 3, \dots, m$) memperbaiki bias dan bobotnya dengan menggunakan persamaan sebagai berikut:

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta W_{jk} \quad (10)$$

Setiap unit di *hidden layer* ($Z_j, j = 1, 2, 3, \dots, p$) melakukan perubahan bobot dan bias yang berasal dari tiap unit di *input layer* ($X_i, i = 1, 2, \dots, n$) menggunakan persamaan sebagai berikut:

$$V_{ij}(\text{baru}) = V_{ij}(\text{lama}) + \Delta V_{ij} \quad \text{Persamaan (11)}$$

- 2) Langkah 7: Periksa kondisi berhenti. Jika kondisi berhenti telah terpenuhi, maka pelatihan jaringan dapat diberhentikan.

2.5 Cross Validation

Cross validation atau validasi silang merupakan proses validasi untuk menilai dan menguji performa suatu model *machine learning*. Menurut Kepemilikan dan Bermotor dalam Purnama, J. (2019), *cross validation* adalah bentuk uji standar yang diterapkan untuk memprediksi tingkat kesalahan (*error rate*). Gambaran mekanisme kerja dari *cross validation* ini sendiri dimulai dengan membagi data latih (*training dataset*) secara acak ke dalam beberapa bagian dengan proporsi yang sama (*stratified*), kemudian tingkat kesalahannya dikalkulasi untuk setiap bagiannya, dan dihitung rata-rata seluruh tingkat kesalahan yang diperoleh sebelumnya.

Salah satu teknik *cross validation* adalah *K-Fold Cross Validation*. Penggunaan *K-Fold Cross Validation* menurut Nurhayati, dkk dalam Mardiana, L. dkk (2022) melalui pembagian *dataset* menjadi k bagian dengan proporsi jumlah

yang sama pada masing-masing bagiannya. Kemudian model *machine learning* akan melakukan iterasi sebanyak k pengulangan menggunakan salah satu bagian *subset* sebagai (*testing subset*) dan sisanya sebagai data latih (*training subset*). Pengulangan dengan pilihan *subset* sebagai data latih dan data uji ini dilakukan secara acak.

K-Fold Cross Validation untuk menganalisis stabilitas suatu model *machine learning* diterapkan melalui penggunaan $K - 1$ bagian sebagai data latih (*training dataset*), dan menggunakan satu bagian sisanya sebagai data uji atau sebagai bagian yang mengaproksimasi kinerja pada *testing* data (Putra, J., 2020). Berikut ilustrasi konsep implementasi *K-Fold Cross Validation* melalui Gambar 14 berikut.



Gambar 14. Ilustrasi Penerapan *K-Fold Cross Validation*
(Sumber: Putra, J., 2020)

2.6 Hyperparameter Model ANN

Menurut Michelucci (2018), *hyperparameter* merupakan parameter yang terdapat dalam suatu model dan telah ditentukan sebelum proses *training* dilakukan. Kombinasi parameter yang optimal akan menghasilkan model yang baik. Kombinasi parameter yang optimal dapat dicari dengan menggunakan *hypertuning* atau proses *optimize parameter*, *hyperparameter* yang umum digunakan, yaitu:

2.6.1 Jumlah Neuron/Nodes pada Hidden Layer

Jumlah *neuron/nodes* pada *hidden layer* akan mempengaruhi proses *training* dan *testing* suatu model. Jumlah *neuron* tersebut dapat menyebabkan model mengalami kondisi *underfitting* atau *overfitting*. Kondisi *underfitting*

merepresentasikan performa model yang buruk pada saat *training* dan *testing*. Sementara untuk kondisi *overfitting* merupakan kondisi performa model yang baik pada fase *training* dan pada fase *testing* justru sebaliknya. Berikut beberapa cara untuk menentukan jumlah *neuron/nodes* pada *hidden layer* (Heaton, 2017):

- a. Jumlah *neuron/nodes* pada *hidden layer* berada di antara jumlah *neuron/nodes* pada *input layer* dan *output layer*.
- b. Jumlah *neuron/nodes* pada *hidden layer* bernilai $2/3$ dari jumlah *nodes* pada *input layer* ditambah *output layer*.
- c. Jumlah *neuron/nodes* pada *hidden layer* kurang dari dua kali jumlah *nodes* pada *input layer*.

2.6.2 Jumlah Epochs

Menurut Viajayalakshmi dan Venkatachalapathy (2019) jumlah *epochs* atau iterasi merupakan perulangan selama proses *training* yang memberikan input dari jaringan dan juga memperbarui bobot jaringan. Jumlah *epochs* atau iterasi pada suatu model umumnya dapat dilakukan dengan eksperimen (proses *trial-error*) atau menggunakan teknik validasi silang (*cross validation*). Proses pelatihan tersebut berlangsung pada rangkaian model *Artificial Neural Network* sampai ke awal lagi (per siklus).

2.6.3 Nilai Learning Rate

Learning rate menurut (Aina, 2018) atau tingkat pembelajaran merupakan faktor yang digunakan pada parameter model dengan tujuan mengontrol sejauh mana model mengubah parameter lainnya (misalnya bobot atau *error*) dalam setiap *epochs* selama proses pelatihan. Salah satu pendekatan yang umum digunakan untuk menentukan nilai *learning rate* yang sesuai adalah dengan melakukan eksperimen dan melihat performa model pada set validasi. Nilai *learning rate* yang umumnya digunakan, yaitu berada pada rentang 0,0001 sampai 1 agar model mencapai kondisi konvergen.

2.6.4 Nilai Momentum

Momentum menurut Srivastava, dkk (2014) merupakan salah satu parameter model ANN yang melibatkan teknik optimasi stokastik seperti *Stochastic Gradient Descent* (SGD). Momentum sebagai parameter berfungsi

dalam mengontrol sejauh mana model mempertimbangkan arah dan kecepatan perubahan gradien selama proses pelatihan. Penerapan momentum ini juga berguna untuk mempercepat model melintasi lembah minimum dalam fungsi kerugian (*loss function*) dan mempercepat konvergensi. Nilai momentum umumnya berada pada rentang 0 hingga 1.

2.7 Parameter Evaluasi Performa Model terhadap Kesalahan

Parameter evaluasi performa model digunakan untuk menilai seberapa baik performa dalam mengestimasi nilai *output* terhadap nilai aktualnya. Ukuran kesalahan tersebut dievaluasi dengan dua parameter, yaitu *Mean Square Error* (MSE) dan *Mean Absolute Percentage Error* (MAPE).

2.7.1 Mean Square Error (MSE)

Mean Square Error (MSE) adalah salah satu parameter evaluasi yang umum digunakan dalam statistic dan *machine learning* untuk mengukur seberapa baik performa model berdasarkan nilai *output* dan nilai aktualnya. MSE mengukur rata-rata kuadrat dari selisih nilai yang diprediksi atau diestimasi model dengan nilai sebenarnya dalam *dataset*. Semakin kecil nilai MSE, maka semakin baik performa model. Berikut persamaan yang digunakan dalam menghitung MSE:

$$\text{MSE} = \frac{1}{n} \sum_{t=1}^n (X_t - y_t)^2 \quad (12)$$

Dimana:

n = jumlah data

y_t = nilai *output* (estimasi/prediksi) pada periode ke- t

X_t = nilai data actual atau observasi pada periode ke- t

Nilai MSE dapat dihitung dengan format persentase (%) untuk memudahkan penilaian performa terhadap rata-rata data aktual. Persentase tersebut mengukur seberapa besar kesalahan model relatif terhadap rata-rata sebenarnya. Semakin besar persentase kesalahan, maka semakin besar kesalahan model relatif terhadap rata-rata data sebenarnya. Berikut persamaan yang dapat digunakan untuk menghitung nilai MSE dalam persen (%) terhadap rata-rata data aktual.

$$\text{MSE} (\%) = \left(\frac{\text{MSE}}{\text{Actual Average}^2} \right) \times 100\% \quad (13)$$

Dimana:

Actual Average = rata-rata data aktual/sebenarnya pada *dataset*

2.7.2 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) merupakan parameter atau metrik yang digunakan untuk mengukur rata-rata persentase kesalahan absolut antara nilai aktual/sebenarnya dan nilai yang diestimasi/diprediksi. Berikut persamaan yang digunakan untuk menghitung MAPE

$$\text{MAPE} = \frac{1}{n} \sum_{t=1}^n \left| \frac{x_t - y_t}{x_t} \right| \times 100\% \quad (14)$$

Dimana:

n = jumlah data

y_t = nilai *output* (estimasi/prediksi) pada periode ke- t

X_t = nilai data aktual atau observasi pada periode ke- t

Interpretasi dari nilai MAPE suatu model dapat dikategorikan sebagai berikut berdasarkan *range*-nya:

Tabel 1 Deskripsi Interpretasi MAPE

MAPE	Deskripsi Interpretasi
<10%	Kemampuan Estimasi/Prediksi Sangat Baik
10% – 20%	Kemampuan Estimasi/Prediksi Baik
20% – 50%	Kemampuan Estimasi/Prediksi Layak/Memadai
> 50%	Kemampuan Estimasi/Prediksi Buruk

2.8 Software Data Mining

Data mining secara garis besar merupakan proses menemukan pola, informasi, dan pengetahuan yang berharga dari kumpulan data dengan kuantitas yang besar atau kompleks. Tujuan utama dari *data mining* ini sendiri adalah untuk mengidentifikasi hubungan yang tidak terlihat secara langsung (*hidden*) dan menghasilkan wawasan (*knowledge*) yang dapat digunakan untuk mendukung pengambilan keputusan.

Implementasi proses *data mining* perlu ditunjang dengan kecakapan yang baik untuk setiap prosesnya. Sehingga untuk menunjang proses ini dapat berjalan dengan lebih baik terdapat opsi menggunakan berbagai *software data mining*.

Beberapa diantara *software data mining* adalah sebagai berikut:

2.8.1 *Rapid Miner Studio 2*

RapidMiner merupakan aplikasi yang bersifat terbuka (*open source*). *RapidMiner* adalah sebuah solusi untuk melakukan analisis terhadap data mining, text mining dan analisis prediksi. *RapidMiner* menggunakan berbagai teknik deskriptif dan prediksi dalam memberikan wawasan kepada pengguna sehingga dapat membuat keputusan yang paling baik. *RapidMiner* memiliki kurang lebih 500 operator *data mining*, termasuk operator untuk *input*, *output*, data *preprocessing* dan visualisasi (Aprilia, D., 2013).

Tampilan (*User Interface*) yang disediakan oleh *RapidMiner* saat ini memiliki keuntungan berupa kemudahan bagi penggunanya. Fitur tampilan yang dimiliki *RapidMiner* ini disebut sebagai *Prespective*. Terdapat tiga *Prespective* yang disediakan oleh *RapidMiner*, yaitu *Welcome Prespective*, *Design Prespective*, dan *Result Prespective*. Ketiga jenis tampilan ini memiliki perannya masing-masing. Pertama, *Welcome Prespective* berfungsi sebagai tampilan awal yang pertama kali dilihat oleh pengguna ketika mengoperasikan aplikasi ini. Kedua, *Design Prespective* yang merupakan tampilan lembar kerja bagi para pengguna dalam merancang, membangun model, menganalisis, dan mengelola data. Ketiga, *Result Prespective* sebagai tampilan aplikasi yang memberikan hasil kerja dari *Design Prespective* (Prasetyo, V. R. dkk, 2021).

Penelitian ini menggunakan aplikasi *RapidMiner* dalam merancang dan membangun model *Artificial Neural Network* (ANN) berdasarkan fungsi dan fitur yang dimiliki oleh aplikasi ini.

2.8.2 WEKA

WEKA adalah sebuah paket *tools machine learning* praktis. WEKA merupakan singkatan dari *Waikato Environment for Knowledge Analysis*, yang dibuat di Universitas Waikato, New Zealand untuk penelitian, pendidikan dan berbagai aplikasi. WEKA mampu menyelesaikan masalah-masalah *data mining* di dunia nyata, khususnya klasifikasi yang mendasari pendekatan-pendekatan machine learning. Perangkat lunak ini ditulis dalam hirarki *class Java* dengan

metode berorientasi objek dan dapat berjalan hampir di semua *platform* (Pujiono, S. dkk, 2013).

WEKA mudah digunakan dan diterapkan pada beberapa tingkatan yang berbeda. Tersedia implementasi algoritma-algoritma pembelajaran *state-of-the-art* yang dapat diterapkan pada *dataset* dari *command line*. WEKA mengandung *tools* untuk *pre-processing data*, klasifikasi, regresi, *clustering*, aturan asosiasi, dan visualisasi. *User* dapat melakukan *preprocess* pada data, memasukkannya dalam sebuah skema pembelajaran, dan menganalisa *classifier* yang dihasilkan dan performansinya semua itu tanpa menulis kode program sama sekali (Pujiono, S. dkk, 2013).

2.8.3 R Studio

Menurut Permana, dkk dalam Yanti, H. A. (2021) R adalah bahasa pemrograman *open-source* yang berhubungan dengan komputasi dan pengolahan data untuk statistika dan yang berhubungan dengan penampilan grafik menggunakan *tools* yang disediakan oleh paket-paketnya yang sangat berguna di dalam penelitian dan industri. Versi awal dari R dibuat pada tahun 1992 di Universitas Auckland, New Zealand oleh Ross Ihaka dan Robert Gentleman. Sementara itu untuk penggunaan aplikasi R *Studio* pengolahan data linear dan nonlinear untuk identifikasi model, uji statistik, klasifikasi, analisis dan visualisasi. Kelebihan R lainnya yaitu plot grafik berkualitas penampilan simulasi dapat berupa plot diagram batang, grafik, kurva, *wordcloud* dan lainnya, termasuk simbol matematika dan rumus jika diperlukan (Budiarto, W. dan Rachmawati, R., 2013).

2.8.4 MATLAB

MATLAB atau Matrix Laboratory merupakan perangkat lunak yang digunakan untuk pemrograman, analisis, serta komputasi teknis dan matematis berbasis matriks. MATLAB pertama dirilis pada tahun 1970 oleh Cleve Moler. Awalnya MATLAB digunakan untuk menyelesaikan permasalahan yang berhubungan tentang persamaan aljabar linear. Dan hingga saat ini sistem pada MATLAB semakin berkembang dalam segi fungsi dan performa komputasinya

Penggunaan MATLAB dalam ilmu Matematika digunakan sebagai alat pendukung pembelajaran pemrograman matematika. Sedangkan dalam bidang lain, MATLAB dipilih sebagai alat perhitungan, analisis matematika dan pengembangan untuk penelitian. MATLAB menyediakan kotak kakas (*toolbox*) yang dapat digunakan untuk aplikasi-aplikasi khusus, seperti logika *fuzzy*, simulasi, optimasi, dan pengolahan citra digital, dan berbagai teknologi lainnya (Febrianti T., dan Erwin, H., 2021).

2.9 Dry-Docking Kapal

Dry-docking merupakan jenis aktivitas *docking* yang umumnya dilakukan pada berbagai jenis fasilitas *dock* (*graving dock*, *slipway dock*, dan lainnya) dengan melibatkan kapal secara penuh tidak berada di dalam perairan atau berada pada kondisi tidak basah (*dry*). Proses pemindahan kapal dalam *dry-docking* ini bertujuan untuk memungkinkan akses ke bagian bawah kapal yang biasanya terendam dalam air untuk kemudian kapal dapat dilakukan pengerjaan sesuai kebutuhan *owner* kapal. Umumnya dilakukan pengerjaan pemeliharaan kapal pada *dry-docking* milik galangan yang berupa perawatan rutin kapal (*routine maintenance*) atau perawatan korektif kapal (*corrective maintenance*) (Novita, R., 2012).

Proses *dry-docking* kapal menurut Wulan dalam Novita, R. (2012) merupakan aktivitas kompleks yang melibatkan pihak galangan dan pihak *owner* kapal. Umumnya proses *docking* diawali dengan penyusunan daftar pemeliharaan kapal (*repair list*) hasil survei pihak galangan kapal atas permintaan *owner* kapal. Kemudian akan dilakukan estimasi waktu *docking* berdasarkan hasil analisis galangan kapal dan kesepakatan antara kedua belah pihak, sehingga jenis dan jumlah pengerjaan selama *dry-docking* memiliki peranan besar untuk mempengaruhi durasi *dry-docking*.

2.9.1 Pekerjaan Perawatan Rutin Kapal pada *Dry-Docking*

Perawatan rutin kapal (*routine maintenance*) adalah serangkaian tindakan pemeliharaan yang dilakukan secara berkala dan terjadwal pada kapal untuk menjaga kapal dalam kondisi yang baik, aman, dan efisien. Tujuan utama

perawatan rutin adalah mencegah kerusakan yang dapat terjadi akibat limitasi komponen fungsional kapal secara *time based* atau *condition based*.

Serangkaian pekerjaan pada perawatan rutin kapal menurut Ahammad Abdullah (2021) meliputi beberapa pekerjaan sebagai berikut:

1) Pengerjaan Pelat (*Plate Work*)

Plate work merujuk pada semua pekerjaan yang melibatkan perbaikan atau penggantian pelat baja pada lambung kapal. Hal ini mungkin mencakup pengelasan, penggantian pelat yang rusak, atau pemasangan pelat tambahan untuk memperkuat struktur kapal.

2) Penyemprotan Pasir ke Permukaan Kapal (*Sand Blasting/Sweepblasting*)

Sand blasting adalah proses penghilangan karat, cat lama, dan kotoran lainnya dari permukaan baja kapal dengan menggunakan aliran pasir atau media abrasif lainnya. Proses ini mempersiapkan permukaan kapal untuk pengerjaan pelapisan anti-karat atau pengecatan ulang.

3) Pengerjaan Pengecatan (*Painting Work*)

Pekerjaan pengecatan melibatkan pengecatan ulang permukaan kapal setelah proses sand blasting atau sebagai bagian dari perawatan rutin. Tujuannya adalah melindungi permukaan kapal dari korosi dan lingkungan laut.

4) Pembersihan Tangki (*Tank Cleaning*)

Tangki kapal, seperti tangki kargo atau tangki bahan bakar, perlu dibersihkan secara berkala untuk menghindari akumulasi kotoran, residu, dan korosi di dalamnya. Proses pembersihan tangki ini mungkin melibatkan pengosongan tangki, pembersihan mekanis, dan pembersihan kimia.

5) Pembersihan Lambung (*Hull Cleaning*)

Pembersihan lambung kapal melibatkan penghilangan kotoran, alga, dan organisme laut lainnya yang menempel pada bagian bawah kapal. Ini dapat membantu mempertahankan efisiensi hidrodinamika kapal dan mencegah pertumbuhan yang tidak diinginkan.

6) Pengerjaan Katup (*Valve Work*)

Pekerjaan katup melibatkan perawatan dan perbaikan katup-katup yang ada pada kapal, termasuk katup-katup yang mengatur aliran cairan atau gas di

dalam sistem kapal. Pekerjaan ini bisa mencakup pelumasan, penggantian, atau perbaikan katup yang rusak.

2.10 Penelitian Terdahulu

Tabel 2 Penelitian Terdahulu Terkait dengan Tugas Akhir Ini

No.	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
1.	Abdullah, A., Rahman, H., Uddin, Md. Imran, Mia, Md. Jobayer (2022)	<i>A General Method to Develop Mathematical Model for Estimating Ship Repairing Man-Hour</i>	<i>Multiple Linear Regression</i>	Berdasarkan hasil penelitian yang diberikan, peneliti memberikan klaim bahwa penerapan metode <i>multiple linear regression</i> merupakan model yang tepat dalam memprediksi estimasi durasi waktu perbaikan kapal dengan menyesuaikan variabel independen kepada masing-masing galangan.
2.	Zagan, R., Paprocka, I., Manea, Mihaela-Greti, Manea, E. (2021)	<i>Estimation of Ship Repair Time Using the Genetic Algorithm</i>	<i>Data Mining</i> dengan Menggunakan <i>Genetic Algorithm</i>	Berdasarkan hasil perbandingan yang dilakukan peneliti menggunakan <i>multiple linear regression</i> ($R^2 = 0,42$) dan <i>Genetic Algorithm</i> ($R^2 = 0,23$). Estimasi tersebut menggunakan beberapa <i>features</i> , seperti ukuran kapa, lingkup pekerjaan, jumlah tenaga kerja, ketersediaan sumber daya.

No.	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
3.	Abdullah, Ahammad (2021)	<i>Development of A Mathematical Model for Ship Repair Time Estimation</i>	<i>Multiple Linear Regression</i>	Berdasarkan hasil menggunakan model matematis <i>multiple linear regression</i> diperoleh hasil $R^2 = 0,63$ dengan penyebab <i>error</i> pada penelitian ini, yaitu absennya beberapa variabel independen yang diasumsikan memiliki kontribusi signifikan dalam estimasi <i>repair time</i> kapal.
4.	Dev, A., & Makaraksha, Saha (2018)	<i>Dry-Docking Time and Labour</i>	<i>Linear Regression</i>	Berdasarkan hasil penelitian yang dilakukan terhadap 586 kapal dengan menggunakan <i>linear regression</i> menemukan hubungan linear terhadap waktu <i>dry docking</i> dengan tiga variabel independen, yaitu <i>ship's deadweight</i> , <i>ship's age</i> , dan <i>ship's type</i> .
5.	Surjadari, I., Dhini, A., Rachman, A., & Novita Riara (2015)	<i>Estimation of Dry-Docking Duration Using a Numerical Ant Colony Decision Tree</i>	<i>Numerical Ant Colony Decision Tree (Combines CART and Ant Colony Optimization)</i>	Pada penelitian ini hasil estimasi terhadap 100 sampel menggunakan <i>Ant ACDT</i> kemudian dibandingkan dengan CART melalui lima pengukuran performa. Berikut kelima akurasi estimasi terhadap error, yaitu $CC = 0.2519$, $MAE = 1.3636$, $RMSE = 1,9462$, $RAE = 27.68\%$, $RRSE = 33.26\%$

No.	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
6.	Firstdhitama, W., Suastika, I. K., & Ma'ruf, B. (2018)	Perencanaan Penjadwalan Reparasi Kapal Ferry dengan Menggunakan Metode Flash	Flash	Meningkatkan efisiensi waktu docking kapal ferry sebesar 32,4% dari 37 hari menjadi 24 hari.
7.	Hananti, H. dan Sari, K. (2021)	Perbandingan metode <i>Support Vector Machine</i> (SVM) dan <i>Artificial Neural Network</i> (ANN) pada klasifikasi gizi balita Puskesmas Salissingan	<i>Support Vector Machine</i> (SVM) dan <i>Artificial Neural Network</i> (ANN)	Tingkat akurasi model SVM sebesar 94,46% (<i>precision</i> = 46,08%, <i>recall</i> = 50,59%, <i>AUC</i> = 0,900) dan model ANN sebesar 94,82% (<i>precision</i> = 51%, <i>recall</i> = 51,09%, <i>AUC</i> = 0,910). Terlihat bahwa untuk dua metode <i>data mining</i> tersebut model ANN memiliki performa yang lebih baik dibanding model SVM.
8.	Beşikçi, E. dkk (2016)	<i>An Artificial Neural Network Based Decision Support System for Energy Efficient Ship Operations</i>	<i>Multiple Regression & Artificial Neural Network</i> (ANN)	Parameter pembanding evaluasi model yang digunakan, yaitu MSE dan RMSE. Pada <i>multiple regression</i> nilai MSE sebesar 0,038 dan RMSE sebesar 0,196. Model ANN dalam penelitian tersebut menghasilkan nilai MSE sebesar 0,037 dan RMSE sebesar 0,193.

No.	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
9.	Putra, H. dan Walmi, N. (2020)	Prediksi Produksi Padi Menggunakan ANN dengan Algoritma <i>Backpropagation</i>	<i>Artificial Neural Network (ANN)</i> dengan Algoritma <i>Backpropagation</i>	Hasil tingkat akurasi prediksi produksi padi pada 19 kabupaten di Sumatera Barat sebesar 88,14%. Tingkat akurasi tersebut diperoleh dengan menggunakan data historis produksi padi dari daerah tersebut dari tahun 2015 hingga 2022.
10.	Çelik, U., & Başarır, C. (2017)	<i>The Prediction of Precious Metal Prices Via Artificial Neural Network by Using Rapidminer</i>	<i>Artificial Neural Network (ANN)</i>	Proses penelitian yang melibatkan 180 total data harga tiap logam mulia dari tahun 2017 hingga tahun 2022. Hasil penelitian ini menunjukkan akurasi sebesar 92,3% untuk emas, 91,2% untuk perak dan 89,5% untuk platinum.

Berikut beberapa penelitian terdahulu yang digunakan sebagai referensi penelitian ini. Referensi yang digunakan berfokus pada topik seputar penelitian terkait dengan estimasi durasi *dry-docking* kapal, durasi reparasi kapal, atau konsep model matematis yang digagas untuk menghitung durasi pekerjaan kapal, dipaparkan untuk mengetahui perkembangan topik penelitian seputar *dry-docking* dan metode *Artificial Neural Network (ANN)*.

Pada penelitian yang dilakukan oleh Surjadari, I. dkk (2015) digunakan pendekatan *data mining* dengan metode *Numerical Ant Colony Decision Tree (Combines CART and Ant Colony Optimization)* untuk mengestimasi durasi *dry-docking* kapal. Performa model pada penelitian tersebut diukur dengan parameter

evaluasi RAE (*Root Average Error*) dengan nilai sebesar 27,68% dan RRSE (*Root Relative Square Error*) dengan nilai sebesar 33,26%. Penelitian tersebut mengklaim bahwa model tersebut sudah cukup layak melaukan estimasi durasi *dry-docking*, namun masih membutuhkan pengembangan model atau penerapan metode pengoptimalan lainnya.

Penelitian yang dilakukan oleh Zagan, R. dkk (2021) juga menggunakan pendekatan *data mining* dengan metode Algoritma Genetika (*Genetic Algorithm*) untuk mengestimasi waktu perbaikan kapal. Penelitian tersebut hanya berfokus pada jenis kapal *tanker*, *bulk carrier*, dan *container ship*. Performa algoritma tersebut dinilai berdasarkan nilai R^2 sebesar 0,23. Hasil performa model tersebut diperoleh dengan variabel independen yang terdiri dari seperti ukuran kapal, lingkup pekerjaan, jumlah tenaga kerja, ketersediaan sumber daya.

Pada penelitian yang dilakukan oleh Abdullah, A. dkk (2022) dikemukakan hasil penelitian untuk menggunakan metode umum dalam mengembangkan model matematis terhadap perhitungan estimasi *man-hour* proses perbaikan kapal. Penelitian tersebut berfokus pada perkiraan *man-hour* perbaikan kapal dengan model *multiple linear regression*. Model ini memberikan keleluasaan untuk masing-masing galangan menetapkan variabel bebas apa saja yang mempengaruhi estimasi *man-hour* proses perbaikan kapal.

Selain itu, pada penelitian milik Dev. A. & Makaraksha, S. (2018) terkait dengan durasi *dry-docking* dan *labour* dengan menggunakan metode *linear regression*. Penelitian ini memberikan output variabel-variabel bebas yang mempengaruhi durasi *dry-docking* dan *labour* seperti *ship's deadweight*, *ship's age*, *ship's type*, dan *repair ship's works*. Keempat variabel tersebut yang diklaim merupakan variabel bebas yang paling mempengaruhi nilai durasi *dry-docking* dan *labour*.

Penelitian yang dilakukan oleh Abdullah, A. (2021) memberikan variabel-variabel berdasarkan jenis pekerjaan *dry-docking* yang paling mempengaruhi durasi *dry-docking* kapal. Pengembangan model matematis menggunakan metode *multiple linear regression* memberikan *output* kombinasi jenis pekerjaan *dry-docking* yang paling optimum, yaitu pengerjaan pelat (*plate work*), penyemprotan pasir ke permukaan kapal (*sandblasting*), pengerjaan pengecatan (*painting work*),

pembersihan tangki (*tank cleaning*), pembersihan lambung (*hull cleaning*), dan pengerjaan katup (*valve work*).

Pada penelitian milik Firstdhitama, W. dkk (2018) yang menggunakan metode *Flash* untuk perencanaan reparasi khusus kapal *ferry* untuk efisiensi waktu *docking*. Hasil penelitian dengan metode *Flash* ini berhasil meningkatkan efisiensi waktu *docking* kapal *ferry* sebesar 32,4% dari 37 hari menjadi 24 hari.

Pada penelitian milik Hananti, H. dan Sari, K. (2021) terkait dengan perbandingan metode *Support Vector Machine* (SVM) dan *Artificial Neural Network* (ANN) pada klasifikasi gizi balita Puskesmas Salisingan, diperoleh hasil perbandingan tingkat akurasi model SVM sebesar 94,46% (*precision* = 46,08%, *recall* = 50,59%, *AUC* = 0,900) dan model ANN sebesar 94,82% (*precision* = 51%, *recall* = 51,09%, *AUC* = 0,910). Terlihat bahwa untuk dua metode *data mining* tersebut model ANN memiliki performa yang lebih baik dibanding model SVM walaupun secara angka tidak sangat signifikan selisih tingkat akurasi keduanya.

Pada penelitian milik Beşikçi, E. dkk (2016) terkait sistem pendukung keputusan (*Decision Support System*, DSS) terhadap efisiensi energi pengoperasian kapal berbasis model ANN. Penelitian tersebut menggunakan pembandingan metode *Multiple Regression* terhadap performa model ANN. Parameter pembandingan evaluasi model yang digunakan, yaitu MSE dan RMSE. Pada *multiple regression* nilai MSE sebesar 0,038 dan RMSE sebesar 0,196. Model ANN dalam penelitian tersebut menghasilkan nilai MSE sebesar 0,037 dan RMSE sebesar 0,193. Berdasarkan perbandingan nilai tersebut, kesimpulan yang sekaligus digunakan sebagai klaim adalah metode ANN memiliki performa yang baik terhadap tujuan penelitian dan lebih unggul dibanding metode pembandingnya.

Pada penelitian Putra, H. dan Walmi, N. (2020) terkait prediksi produksi padi menggunakan ANN dengan algoritma *backpropagation*, diperoleh hasil tingkat akurasi prediksi produksi padi pada 19 kabupaten di Sumatera Barat sebesar 88,14%. Tingkat akurasi tersebut diperoleh dengan menggunakan data historis produksi padi dari daerah tersebut dari tahun 2015 hingga 2022. Persentase

akurasi tersebut diinterpretasikan sebagai model prediksi yang cukup baik dalam melakukan prediksi.

Penelitian dengan menggunakan ANN sebagai model prediksi juga dilakukan oleh Çelik, U., & Başarır, C. (2017) terkait dengan prediksi harga logam mulia dengan aplikasi *RapidMiner*. Proses penelitian yang melibatkan 180 total data harga tiap logam mulia dari tahun 2017 hingga tahun 2022. Hasil penelitian ini menunjukkan akurasi sebesar 92,3% untuk emas, 91,2% untuk perak dan 89,5% untuk platinum.

Berdasarkan rujukan referensi sebelumnya yang menggunakan model *Artificial Neural Network* (ANN), maka pada penelitian ini berusaha mengembangkan model ANN tersebut dengan lebih optimal melalui penerapan algoritma *backpropagation* dan teknik *K-Fold Cross Validation* (validasi silang) untuk mengestimasi durasi *dry-docking* kapal di PT. Industri Kapal Indonesia.

