

TESIS
ANALISIS PERBANDINGAN ARIMA, SARIMA, GRU DAN LSTM
UNTUK PERAMALAN PENJUALAN MOBIL TOYOTA



disusun oleh

FARIZ ZAKARIA

22.55.2323

Konsentrasi : Business Intelligence

FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2025

TESIS
ANALISIS PERBANDINGAN ARIMA, SARIMA, GRU DAN LSTM
UNTUK PERAMALAN PENJUALAN MOBIL TOYOTA

ANALYSIS OF COMPARISON OF ARIMA, SARIMA, GRU AND LSTM
FOR FORECASTING TOYOTA CAR SALES

Diajukan untuk memenuhi salah satu syarat mencapai derajat Pascasarjana
Program Studi S2 Pendidikan Jarak Jauh Teknik Informatika



disusun oleh

FARIZ ZAKARIA

22.55.2323

Konsentrasi : Business Intelligence

FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA

2025

HALAMAN PERSETUJUAN

**ANALISIS PERBANDINGAN ARIMA, SARIMA, GRU DAN LSTM
UNTUK PERAMALAN PENJUALAN MOBIL TOYOTA**

**ANALYSIS OF COMPARISON OF ARIMA, SARIMA, GRU AND LSTM
FOR FORECASTING TOYOTA CAR SALES**

yang disusun dan diajukan oleh

Fariz Zakaria

22.55.2323

telah disetujui oleh Dosen Pembimbing Tesis
pada tanggal 04 November 2025

Dosen Pembimbing,



Prof. Dr. Ema Cahni, S.Si., M.Kom.

NIK. 190302037

HALAMAN PENGESAHAN

**ANALISIS PERBANDINGAN ARIMA, SARIMA, GRU DAN LSTM
UNTUK PERAMALAN PENJUALAN MOBIL TOYOTA**

**ANALYSIS OF COMPARISON OF ARIMA, SARIMA, GRU AND LSTM
FOR FORECASTING TOYOTA CAR SALES**

yang disusun dan diajukan oleh

Fariz Zakaria

22.55.2323

Telah dipertahankan di depan Dewan Penguji
pada tanggal 04 November 2025

Susunan Dewan Penguji

Nama Penguji

Tanda Tangan

Alva Hendi Muhammad, S.T., M.Eng., Ph.D.
NIK. 190302493



Dhani Ariatmoko, S. Kom., M.Kom., Ph.D.
NIK. 190302197



Prof. Dr. Ema Utami, S.Si., M.Kom.
NIK. 190302037



Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer
Tanggal 04 November 2025

DEKAN FAKULTAS ILMU KOMPUTER



Prof. Dr. Kusri, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Fariz Zakaria
NIM : 22.55.2323

Menyatakan bahwa Tesis dengan judul berikut:

Analisis Perbandingan ARIMA, SARIMA, GRU, dan LSTM Untuk Peramalan Penjualan Mobil Toyota

Dosen Pembimbing: Prof. Dr. Ema Utami, S.Si., M.Kom.

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya.
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Dosen Pembimbing.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini.
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta.
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi.

Yogyakarta, 04 November 2025

Yang Menyatakan,



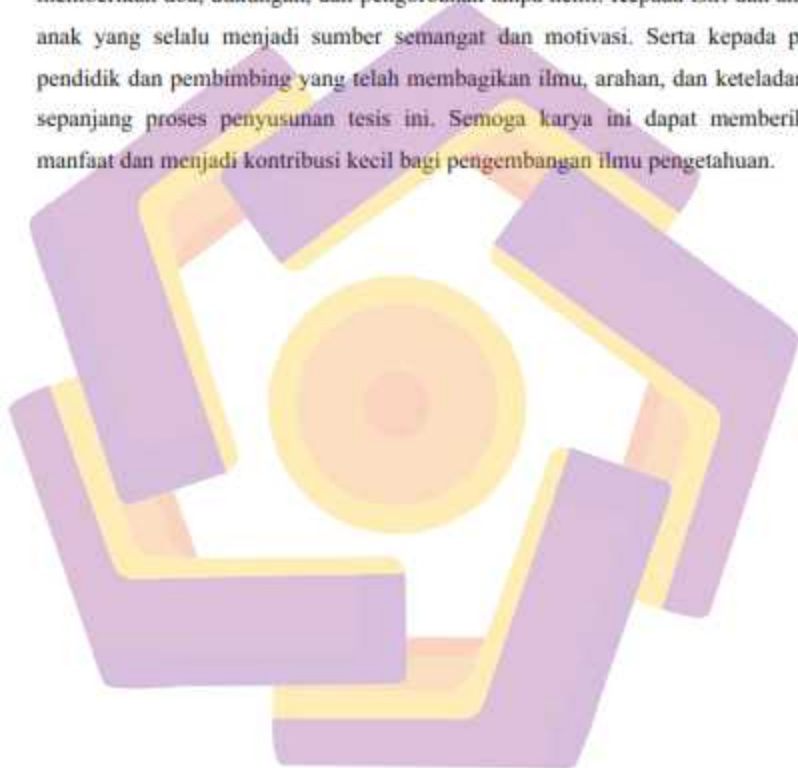
10000
METRAI
TEMPEL
0004AMAK10728402

Fariz Zakaria

HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahim

Tesis ini saya persembahkan kepada orang tua tercinta yang telah memberikan doa, dukungan, dan pengorbanan tanpa henti. Kepada istri dan anak-anak yang selalu menjadi sumber semangat dan motivasi. Serta kepada para pendidik dan pembimbing yang telah membagikan ilmu, arahan, dan keteladanan sepanjang proses penyusunan tesis ini. Semoga karya ini dapat memberikan manfaat dan menjadi kontribusi kecil bagi pengembangan ilmu pengetahuan.



KATA PENGANTAR

Puji syukur ke hadirat Allah Subhanahu wa Ta'ala atas segala limpahan rahmat, taufik, dan hidayah-Nya sehingga penulis dapat menyelesaikan tesis ini yang merupakan syarat dalam menyelesaikan jenjang Pendidikan S2 Pascasarjana Fakultas Ilmu Komputer Universitas AMIKOM Yogyakarta yang berjudul Analisis Perbandingan ARIMA, SARIMA, GRU, dan LSTM Untuk Peramalan Penjualan Mobil Toyota dengan baik. Pada kesempatan ini Penulis ingin menyampaikan ucapan terima kasih kepada:

1. Orang tua tercinta, Ibu Nurlaela dan Bapak Imamudin yang senantiasa memberikan doa, dukungan, dan dorongan moral selama proses studi dan penyusunan tesis ini.
2. Istri tercinta, Defriyanti Hasan dan anak-anak atas kesabaran, pengertian, dan motivasi yang tidak pernah berhenti hingga karya ini dapat terselesaikan.
3. Bapak Prof. Dr. M. Suyanto, M.M., selaku Rektor Universitas AMIKOM Yogyakarta.
4. Ibu Prof. Dr. Kusriani, M.Kom., selaku Dekan Fakultas Ilmu Komputer Universitas AMIKOM Yogyakarta.
5. Ibu Prof. Dr. Ema Utami, S.Si., M.Kom., selaku dosen pembimbing utama yang telah memberikan arahan, bimbingan, masukan, dan waktu yang sangat berharga selama proses penelitian berlangsung.
6. Bapak Alva Hendi Muhammad, S.T., M.Eng., Ph.D dan Bapak Dhani Ariatanto, S.Kom., M.Kom., Ph.D selaku tim dosen Penguji yang telah memberikan kritik, saran, dan evaluasi yang konstruktif demi penyempurnaan laporan tesis ini.
7. Segenap Dosen dan staff Universitas AMIKOM Yogyakarta yang telah memberikan ilmu, pengetahuan, motivasi, dan pengalaman yang berharga selama perkuliahan.

Akhir kata, penulis menyadari bahwa tesis ini masih memiliki keterbatasan dan jauh dari kesempurnaan, oleh karena itu kritik dan saran yang membangun sangat diharapkan demi penyempurnaan karya ini di masa mendatang. Semoga tesis ini dapat memberikan kontribusi positif bagi pengembangan ilmu pengetahuan serta bermanfaat bagi para pembaca dan pihak-pihak yang berkepentingan.

Yogyakarta, 04 November 2025

Penulis



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN KEASLIAN TESIS	iv
HALAMAN PERSEMBAHAN	v
KATA PENGANTAR	vi
DAFTAR ISI	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiii
DAFTAR LAMBANG DAN SINGKATAN	xiv
DAFTAR ISTILAH	xvi
INTISARI	xvii
ABSTRACT	xviii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	6
BAB 2 TINJAUAN PUSTAKA	7
2.1 Tinjauan Pustaka	7
2.2 Keaslian Penelitian	12
2.3 Landasan Teori	19
2.3.1 <i>Time Series</i> dan Prediksi Data	19

2.3.2	<i>Autoregressive Integrated Moving Average (ARIMA)</i>	20
2.3.3	<i>Seasonal Autoregressive Integrated Moving Average (SARIMA)</i>	22
2.3.4	<i>Gated Recurrent Unit (GRU)</i>	23
2.3.5	<i>Long Short-Term Memory (LSTM)</i>	25
2.3.6	Optimasi Algoritma	27
2.3.7	Evaluasi Kinerja Model	31
BAB 3	METODE PENELITIAN	33
3.1	Jenis, Sifat, dan Pendekatan Penelitian	33
3.2	Metode Pengumpulan Data	34
3.3	Metode Analisis Data	35
3.4	Alur Penelitian	37
BAB 4	HASIL PENELITIAN DAN PEMBAHASAN	43
4.1	Pengumpulan Data	43
4.2	Preprocessing Data	44
4.3	Eksplorasi Data	44
4.3.1	Visualisasi Data Awal	45
4.3.2	Cek Stasioneritas	47
4.3.3	Seasonal Decomposition	49
4.3.4	Boxplot	51
4.4	Pembagian Data	52
4.5	Pembuatan Model	52
4.5.1	<i>Autoregressive Integrated Moving Average (ARIMA)</i>	53
4.5.2	<i>Seasonal Autoregressive Integrated Moving Average (SARIMA)</i>	58
4.5.3	<i>Gated Recurrent Unit (GRU)</i>	62
4.5.4	<i>Long Short-Term Memory (LSTM)</i>	65
4.6	Optimasi Model	68
4.6.1	Optimasi Model ARIMA	68
4.6.2	Optimasi Model SARIMA	70
4.6.3	Optimasi Model GRU	73
4.6.4	Optimasi Model LSTM	75
4.7	Perbandingan Evaluasi Model	78

4.7.1	Evaluasi Model ARIMA	78
4.7.2	Evaluasi Model SARIMA	79
4.7.3	Evaluasi Model GRU	79
4.7.4	Evaluasi Model LSTM	80
4.7.5	Ringkasan Perbandingan	80
4.8	Interpretasi Hasil	83
4.8.1	Ringkasan Evaluasi Numerik	83
4.8.2	Analisis Performa Model Sebelum dan Sesudah Optimasi	85
4.8.2	Interpretasi Pola Trend Musiman	94
4.8.3	Implikasi Praktis	96
4.8.4	Keterbatasan dan Rekomendasi	98
BAB 5	PENUTUP	101
5.1	Kesimpulan	101
5.2	Saran	102

DAFTAR TABEL

Tabel 2.1 Matriks literatur review dan posisi penelitian	12
Tabel 4.1 Statistik Deskriptif Dataset	46
Tabel 4.2 Hasil ADF Test	49
Tabel 4.3 Hasil ADF Test ARIMA	53
Tabel 4.4 Metrik Evaluasi model ARIMA	57
Tabel 4.5 Metrik Evaluasi model SARIMA	61
Tabel 4.6 Metrik Evaluasi model GRU	64
Tabel 4.7 Metrik Evaluasi model LSTM	67
Tabel 4.8 Hasil Optimasi Model ARIMA	69
Tabel 4.9 Hasil Optimasi Model SARIMA	70
Tabel 4.10 Hasil Optimasi Model GRU	74
Tabel 4.11 Ringkasan Perbandingan Evaluasi Model	80



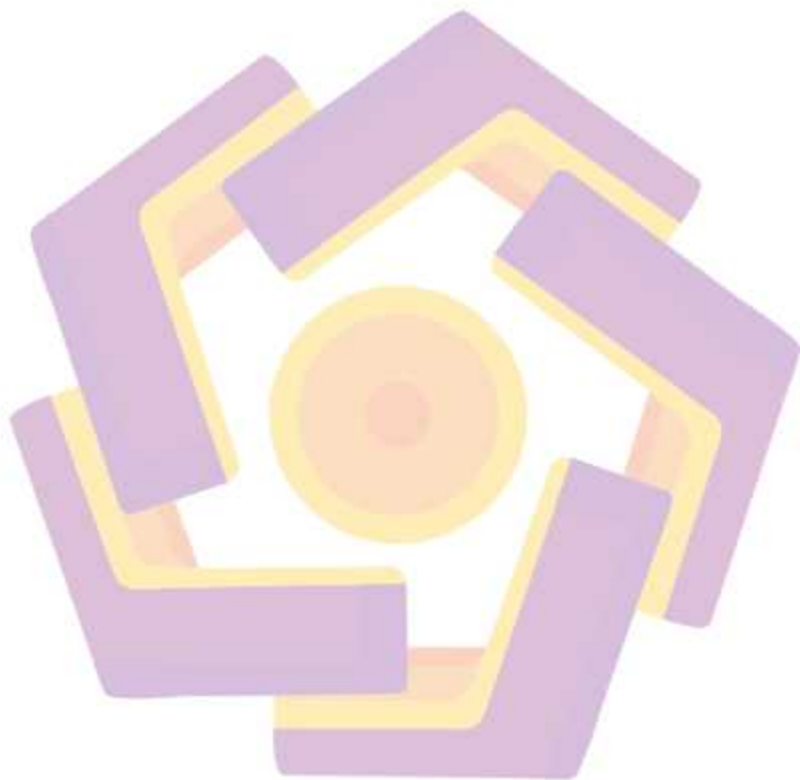
DAFTAR GAMBAR

Gambar 3.1 Alur Penelitian	42
Gambar 4.1 Informasi Dataset	43
Gambar 4.2 Gambaran Dataset	44
Gambar 4.3 Grafik penjualan	45
Gambar 4.4 Plot Rolling mean dan Rolling std deviasi	48
Gambar 4.5 Seasonal Decomposition Dataset	50
Gambar 4.6 Boxplot	51
Gambar 4.7 Grafik ACF Plot	54
Gambar 4.8 Grafik PACF Plot	55
Gambar 4.9 Grafik Perbandingan ARIMA Sebelum dan Setelah Optimasi	86
Gambar 4.10 Grafik Perbandingan SARIMA Sebelum dan Setelah Optimasi	88
Gambar 4.11 Grafik Perbandingan GRU Sebelum dan Setelah Optimasi	90
Gambar 4.12 Grafik perbandingan LSTM Sebelum dan Setelah Optimasi	92


DAFTAR LAMPIRAN

Lampiran 1 Dataset

109

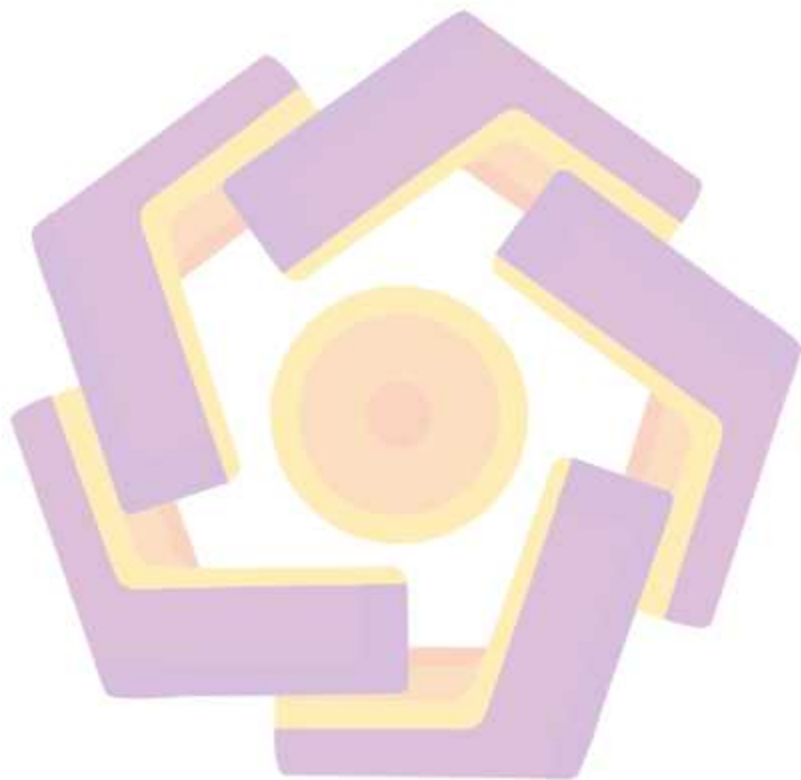


DAFTAR LAMBANG DAN SINGKATAN



ϕ_p	Tingkat aggressive
σ	menyatakan fungsi sigmoid logistik
Σ	Sigma
C_t	Cell State
h_t	Hidden state
f_t	Forget gate
i_t	Input gate
o_t	Output gate
r_t	Gerbang Reset
z_t	Gerbang Update
h'_t	kandidat hidden layer
ACF	Autocorrelation Function
ADF	Augmented Dickey-Fuller
AIC	Akaike Information Criterion
ARIMA	Autoregressive Integrated Moving Average
BIC	Bayesian Information Criterion
BiLSTM	Bidirectional Long Short-Term Memory
CNN	Convolutional Neural Network
GA	Genetic Algorithm
GRU	Gated Recurrent Unit
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MSE	Mean Squared Error
PACF	Partial Autocorrelation Function
PSO	Particle Swarm Optimization
RMSE	Root Mean Squared Error

RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SARIMA	Seasonal Autoregressive Integrated Moving Average



DAFTAR ISTILAH

Autokorelasi	Hubungan data dengan lag waktunya
Deep Learning	Metode pembelajaran menggunakan jaringan saraf
Differencing	Proses membuat data menjadi stasioner
Dropout	Teknik mematikan neuron secara acak saat pelatihan
Early Stopping	Menghentikan pelatihan saat performa tidak meningkat
Forecasting	Proses peramalan data masa depan
Grid Search	Metode pencarian kombinasi parameter terbaik secara sistematis
Hyperparameter	Parameter yang ditentukan sebelum proses pelatihan model
Lag Plot	Grafik hubungan antar waktu
Learning Rate	Laju pembaruan bobot saat pelatihan model
Model Prediksi	Model untuk meramalkan data
Non-linear	Pola tidak membentuk garis lurus
Normalisasi	Proses penskalaan data ke rentang tertentu
Optimasi	Proses mencari parameter terbaik
Overfitting	Kondisi model terlalu menyesuaikan data latih
Regularisasi	Teknik untuk mengurangi kompleksitas model
Residual	Selisih nilai aktual dan prediksi
Stasioneritas	Kondisi data deret waktu dengan rata-rata dan varians konstan
Time Series	Data yang dikumpulkan secara berurutan berdasarkan waktu
White Noise	Residual acak tanpa pola yang signifikan
Windowing	Proses membentuk data sekuensial berdasarkan periode waktu

INTISARI

Penelitian berjudul "Perbandingan Kinerja ARIMA, SARIMA, GRU, dan LSTM untuk Peramalan Penjualan Mobil Toyota di Indonesia" ini bertujuan untuk mengevaluasi dan membandingkan efektivitas empat model peramalan deret waktu dalam memprediksi penjualan bulanan mobil Toyota. Variabel penelitian dibatasi pada data penjualan historis tanpa melibatkan faktor eksternal, dengan metode analisis meliputi prapemrosesan data, eksplorasi data, pembangunan model, serta optimasi hiperparameter menggunakan pendekatan grid search dan teknik regularisasi.

Hasil penelitian menunjukkan bahwa model statistik ARIMA dan SARIMA pada konfigurasi awal belum mampu menghasilkan prediksi yang akurat, ditandai dengan nilai kesalahan yang tinggi, khususnya SARIMA dengan MAPE melebihi 100%. Setelah optimasi, kinerja keduanya meningkat signifikan, di mana ARIMA mencapai MAPE 15,64% dan SARIMA 18,38%. Di sisi lain, model deep learning menunjukkan performa yang lebih unggul. GRU pada tahap awal menghasilkan MAPE 17,36% dan meningkat tajam menjadi 7,66% setelah penerapan dropout dan penyesuaian window size. LSTM juga mengalami peningkatan dari MAPE 19,24% menjadi 15,76% melalui optimasi jumlah neuron, learning rate, dan dropout.

Secara keseluruhan, penelitian ini menyimpulkan bahwa GRU dan LSTM lebih efektif dibandingkan ARIMA dan SARIMA dalam menangkap pola non-linear serta dinamika kompleks pada data penjualan mobil Toyota. Optimasi hiperparameter terbukti berperan penting dalam meningkatkan akurasi model. Penelitian selanjutnya direkomendasikan untuk memasukkan variabel eksternal dan menguji model berbasis arsitektur hybrid guna memperoleh performa prediksi yang lebih komprehensif.

Kata kunci: peramalan deret waktu, ARIMA, SARIMA, GRU, LSTM.

ABSTRACT

This study, entitled "Comparison of ARIMA, SARIMA, GRU, and LSTM Performance for Toyota Car Sales Forecasting in Indonesia," aims to evaluate and compare the effectiveness of four time series forecasting models in predicting monthly Toyota car sales. The research variables are limited to historical sales data without involving external factors, with analysis methods including data preprocessing, data exploration, model building, and hyperparameter optimization using the grid search approach and regularization techniques.

The results show that the ARIMA and SARIMA statistical models in their initial configurations were unable to produce accurate predictions, as indicated by high error values, particularly SARIMA with a MAPE exceeding 100%. After optimization, the performance of both models improved significantly, with ARIMA achieving a MAPE of 15.64% and SARIMA achieving a MAPE of 18.38%. On the other hand, deep learning models showed superior performance. GRU initially produced a MAPE of 17.36% and increased sharply to 7.66% after applying dropout and adjusting the window size. LSTM also improved from a MAPE of 19.24% to 15.76% through optimization of the number of neurons, learning rate, and dropout.

Overall, this study concludes that GRU and LSTM are more effective than ARIMA and SARIMA in capturing non-linear patterns and complex dynamics in Toyota car sales data. Hyperparameter optimization has proven to play an important role in improving model accuracy. Further research is recommended to include external variables and test hybrid architecture-based models to obtain more comprehensive prediction performance.

Keyword: *time series forecasting, ARIMA, SARIMA, GRU, LSTM.*

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Dalam sektor otomotif, prediksi penjualan memegang peran krusial dalam menentukan langkah-langkah strategis, termasuk perencanaan produksi dan manajemen rantai pasok. Sebagai salah satu perusahaan otomotif terkemuka di dunia, Toyota memiliki catatan penjualan yang mencerminkan pola musiman serta variasi yang dipengaruhi oleh dinamika ekonomi dan kondisi pasar. Dengan semakin banyaknya data historis yang tersedia, metode analisis berbasis data menjadi semakin vital untuk meningkatkan ketepatan prediksi dan meminimalkan potensi risiko dalam operasional bisnis.

Prediksi data time series memiliki tantangan tersendiri, terutama dalam menangani masalah stasioneritas, overfitting, dan penyesuaian model terhadap pola non-linear. Data time series sering kali memiliki komponen musiman dan tren jangka panjang yang tidak dapat diabaikan dalam analisis prediksi (Hamiane et al., 2024). Stasioneritas adalah persyaratan mendasar untuk banyak model deret waktu, termasuk ARIMA dan SARIMA. Deret waktu yang stasioner memiliki rata-rata dan varians yang konstan dari waktu ke waktu, yang sangat penting untuk peramalan yang akurat. Data yang tidak stasioner dapat menyebabkan prediksi model yang tidak dapat diandalkan. Teknik seperti pembedaan (differencing) sering digunakan untuk mengubah data nonstasioner menjadi bentuk yang stasioner, seperti yang terlihat pada penerapan SARIMA untuk prediksi arus lalu lintas (Kumar &

Vanajakshi, 2015). Memastikan stasioneritas sangat penting untuk penerapan model ARIMA yang efektif (Arumugam & Natarajan, 2023; Rizvi, 2024). Overfitting terjadi ketika sebuah model terlalu kompleks dan menangkap noise dalam data daripada pola yang mendasarinya. Hal ini dapat mengakibatkan kinerja prediksi yang buruk pada data baru. Dalam peramalan deret waktu, memilih model dengan terlalu banyak parameter dapat menyebabkan overfitting. Penggunaan metrik evaluasi seperti AIC dan BIC membantu dalam memilih model parsimoni yang menyeimbangkan antara kompleksitas dan kecocokan (Arumugam & Natarajan, 2023; Ghide et al., 2022). Overfitting adalah masalah umum dalam model pembelajaran mesin, termasuk yang digunakan untuk peramalan deret waktu.

Meskipun model statistik seperti ARIMA dan SARIMA telah banyak digunakan dalam analisis deret waktu, keduanya memiliki keterbatasan fundamental. Kedua model ini hanya efektif untuk data yang bersifat linear dan stasioner, sehingga kurang mampu menangkap hubungan non-linear dan dinamika kompleks yang sering muncul dalam data penjualan otomotif. Dalam praktiknya, pola penjualan kendaraan tidak hanya dipengaruhi oleh faktor musiman, tetapi juga oleh fluktuasi pasar, perilaku konsumen, dan kondisi ekonomi yang bersifat non-linear dan tidak dapat dijelaskan sepenuhnya oleh pendekatan statistik tradisional. Hal ini menyebabkan model seperti ARIMA dan SARIMA cenderung menghasilkan prediksi yang kurang akurat ketika diterapkan pada data dengan karakteristik non-linear atau volatilitas tinggi (Arumugam & Natarajan, 2023; Khaira et al., 2019).

Untuk mengatasi keterbatasan tersebut, pendekatan berbasis pembelajaran mesin (machine learning) mulai banyak digunakan dalam peramalan deret waktu. Algoritma modern seperti Gated Recurrent Unit (GRU) dan Long Short-Term Memory (LSTM) menawarkan kemampuan yang lebih baik dalam menangkap pola data yang kompleks, non-linear, serta dependensi jangka panjang yang tidak dapat diakomodasi oleh model statistik konvensional. GRU memiliki struktur sederhana dengan dua gerbang utama yang efisien dalam menangkap pola jangka pendek, sedangkan LSTM memiliki tiga gerbang utama yang memungkinkannya menyimpan informasi lebih lama dan menghindari masalah vanishing gradient. Kedua model ini merupakan bagian dari keluarga Recurrent Neural Network (RNN) yang dirancang khusus untuk memproses data sekuensial dan temporal, menjadikannya lebih adaptif terhadap variasi pola penjualan yang berubah seiring waktu.

Dengan mempertimbangkan kekuatan dan kelemahan masing-masing pendekatan, penelitian ini berfokus pada perbandingan kinerja model tradisional (ARIMA dan SARIMA) dengan model deep learning (GRU dan LSTM) dalam memprediksi penjualan mobil Toyota di Indonesia. Penelitian ini tidak hanya mengevaluasi akurasi prediksi, tetapi juga mengimplementasikan teknik optimasi hyperparameter dan regularisasi untuk meminimalkan overfitting serta meningkatkan kemampuan generalisasi model. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi empiris terhadap pengembangan model prediksi penjualan berbasis data yang lebih presisi dan adaptif, sekaligus menjawab

permasalahan utama dalam peramalan data deret waktu pada industri otomotif modern.

1.2 Rumusan Masalah

Berikut merupakan rumusan masalah dalam penelitian ini:

1. Bagaimana kinerja dari algoritma pembelajaran mesin SARIMA, ARIMA, GRU, dan LSTM saling berbanding dalam konteks peramalan penjualan mobil toyota sebelum dan sesudah optimasi?
2. Algoritma mana yang menunjukkan hasil paling optimal dalam hal akurasi dan ketepatan untuk peramalan penjualan mobil toyota?

1.3 Batasan Masalah

Berikut merupakan batasan masalah dalam penelitian ini:

1. Penelitian ini berfokus pada evaluasi kinerja empat model prediktif utama, yaitu ARIMA, SARIMA, GRU, dan LSTM, dalam konteks analisis data penjualan mobil Toyota. Model prediktif lainnya, baik dari kategori statistik maupun pembelajaran mesin, tidak akan dibahas dalam penelitian ini.
2. Data yang digunakan terbatas pada data historis penjualan mobil Toyota yang diambil dari periode Januari 2011 hingga Mei 2023. Periode ini dipilih untuk memastikan ketersediaan data yang cukup untuk analisis tren dan pola musiman yang signifikan. Data di luar periode ini tidak akan dipertimbangkan.

3. Penelitian ini hanya akan menganalisis data penjualan bulanan dari Toyota. Data lain seperti penjualan harian atau mingguan, serta data dari produsen otomotif lain, tidak termasuk dalam cakupan penelitian ini.
4. Optimasi parameter ARIMA dan SARIMA menggunakan metode hyperparameter tuning menggunakan teknik grid search, model GRU menggunakan penerapan teknik regularisasi, sedangkan LSTM menggunakan hyperparameter tuning.
5. Evaluasi kinerja model akan dilakukan menggunakan metrik akurasi prediksi seperti Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), dan Root Mean Squared Error (RMSE).

1.4 Tujuan Penelitian

Tujuan utama penelitian ini adalah untuk mengevaluasi dan membandingkan kinerja empat algoritma peramalan deret waktu, yaitu SARIMA, ARIMA, GRU, dan LSTM, dalam memprediksi penjualan mobil Toyota. Analisis dilakukan dengan menilai performa masing-masing model sebelum dan sesudah proses optimasi, sehingga dapat terlihat bagaimana penyesuaian parameter memengaruhi tingkat akurasi prediksi.

Selain itu, penelitian ini juga bertujuan untuk mengidentifikasi algoritma yang mampu memberikan hasil paling optimal berdasarkan indikator evaluasi seperti MAE, MAPE, MSE, dan RMSE. Dengan demikian, penelitian ini diharapkan dapat memberikan rekomendasi model peramalan yang paling efektif

dan akurat untuk digunakan dalam analisis penjualan mobil Toyota pada periode berikutnya.

1.5 Manfaat Penelitian

Berikut merupakan manfaat penelitian ini:

1. **Manfaat Teoritis:** Penelitian ini bertujuan menambah pengetahuan mengenai kinerja berbagai algoritma machine learning untuk prediksi data deret waktu, khususnya dalam perbandingan antara model statistik tradisional dan deep learning.
2. **Manfaat Praktis:** Hasil dari penelitian ini diharapkan membantu profesional dalam memilih algoritma yang akurat untuk memprediksi data deret waktu di berbagai sektor, seperti ekonomi, cuaca, dan bisnis.
3. **Referensi Penelitian Lanjutan:** Studi ini dapat menjadi dasar bagi penelitian lanjutan yang ingin mengeksplorasi atau memperbaiki model prediksi untuk data yang lebih kompleks dan beragam, serta memfokuskan pada kombinasi model atau pendekatan inovatif dalam analisis data deret waktu.

BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Kajian terhadap hasil penelitian sebelumnya menjadi landasan utama dalam mendukung penelitian yang akan dilakukan. Selain itu, kajian ini berfungsi sebagai bahan perbandingan untuk menilai kelebihan dan kekurangan penelitian terdahulu sekaligus memperkuat dasar argumen penelitian. Oleh karena itu, peneliti mengacu pada sejumlah penelitian yang relevan dengan topik yang diangkat, yaitu perbandingan kinerja algoritma pembelajaran mesin untuk prediksi.

Penelitian ini mengacu pada berbagai studi sebelumnya yang mengevaluasi kinerja model prediktif dalam analisis data deret waktu. Tinjauan pustaka bertujuan untuk menyoroti benang merah antara penelitian-penelitian terdahulu dengan fokus penelitian saat ini, yaitu evaluasi kinerja ARIMA, SARIMA, GRU, dan LSTM dalam memprediksi penjualan mobil Toyota.

(Arumugam & Natarajan, 2023) dalam studinya mengidentifikasi beberapa tantangan kunci dalam pemodelan deret waktu, seperti masalah stasioneritas, parsimoni, dan risiko overfitting. Penelitian ini membandingkan efektivitas model ARIMA dan SARIMA menggunakan enam dataset berbeda, dan hasilnya mengungkapkan bahwa SARIMA cenderung lebih baik dalam mengakomodasi pola musiman dibandingkan ARIMA pada mayoritas kasus. Evaluasi kinerja model dilakukan dengan metrik seperti MSE, RMSE, MAPE, dan Theil's U-statistics,

yang menunjukkan bahwa pemilihan parameter yang tepat dapat meningkatkan akurasi prediksi secara signifikan.

Penelitian yang dilakukan oleh (Rizvi, 2024) menunjukkan bahwa ARIMA telah digunakan secara luas dalam berbagai bidang, termasuk kesehatan, meteorologi, keuangan, dan manajemen rantai pasok. Model ini terbukti efektif dalam menangkap pola tren dan musiman, meskipun dalam beberapa kasus, model hybrid atau kombinasi dengan metode lain dapat memberikan hasil yang lebih akurat. Studi ini memperluas penggunaan ARIMA dengan mengeksplorasi aplikasinya dalam konteks peramalan deret waktu, sambil mengakui tantangan dan keterbatasan yang melekat pada model ini.

(Febiola et al., 2024) dalam studi mereka membandingkan kinerja ARIMA dan SARIMA dalam konteks peramalan jumlah penumpang bandara. Temuan mereka menunjukkan bahwa SARIMA unggul dalam menangkap pola musiman, yang relevan dengan studi ini mengingat data penjualan mobil Toyota juga menunjukkan pola musiman. Dengan demikian, penelitian ini memperluas penggunaan SARIMA dalam konteks yang berbeda, yakni penjualan mobil.

Penelitian yang dilakukan oleh (Ridwan et al., 2023) mengevaluasi GRU dan ARIMA dalam data deret waktu berfrekuensi tinggi. Hasil penelitian mereka menunjukkan keunggulan GRU dalam menangani pola data yang kompleks. Penelitian ini mengambil inspirasi dari hasil tersebut untuk menguji apakah keunggulan GRU dalam data frekuensi tinggi juga berlaku dalam data penjualan bulanan mobil.

(Ariyanti & Tristyanti Yusnitasari, 2023) menyoroti efektivitas SARIMA dalam memprediksi harga minyak mentah yang juga memiliki pola musiman. Temuan ini memperkuat relevansi SARIMA untuk data musiman, yang menjadi dasar penggunaan SARIMA dalam penelitian ini untuk data penjualan mobil.

(Nusaiba Yulisa et al., 2023) mengaplikasikan GRU dalam memprediksi nilai ekspor migas. Hasil penelitian mereka menunjukkan bahwa GRU efektif dalam menangani data deret waktu yang kompleks. Penelitian ini berupaya mengevaluasi keefektifan GRU dalam konteks penjualan mobil, memperluas aplikasi GRU ke domain baru.

(Arisena, 2020) dalam penelitiannya menggunakan ARIMA untuk memprediksi penjualan mobil Toyota. Penelitian ini menemukan bahwa ARIMA mampu memberikan hasil yang memadai, namun memiliki keterbatasan dalam menangani pola musiman yang lebih dinamis. Studi ini akan melanjutkan analisis dengan menambahkan SARIMA dan GRU untuk membandingkan apakah kedua model tersebut dapat memberikan prediksi yang lebih akurat.

Penelitian yang dilakukan (Fahrudin & Sumitra, 2019) membandingkan SARIMA dan SES dalam konteks peramalan kebutuhan hidup. Studi ini menunjukkan bahwa SARIMA lebih unggul dalam menangkap pola musiman. Hasil ini menjadi rujukan bagi penelitian ini untuk menilai apakah SARIMA dapat memberikan keunggulan yang sama dalam konteks yang berbeda, yakni penjualan mobil.

(Tito Prabowo et al., 2025) melakukan studi "Time Series Analysis for Forecasting Livestock Population in Indonesia with LSTM Model". Jurnal UMB

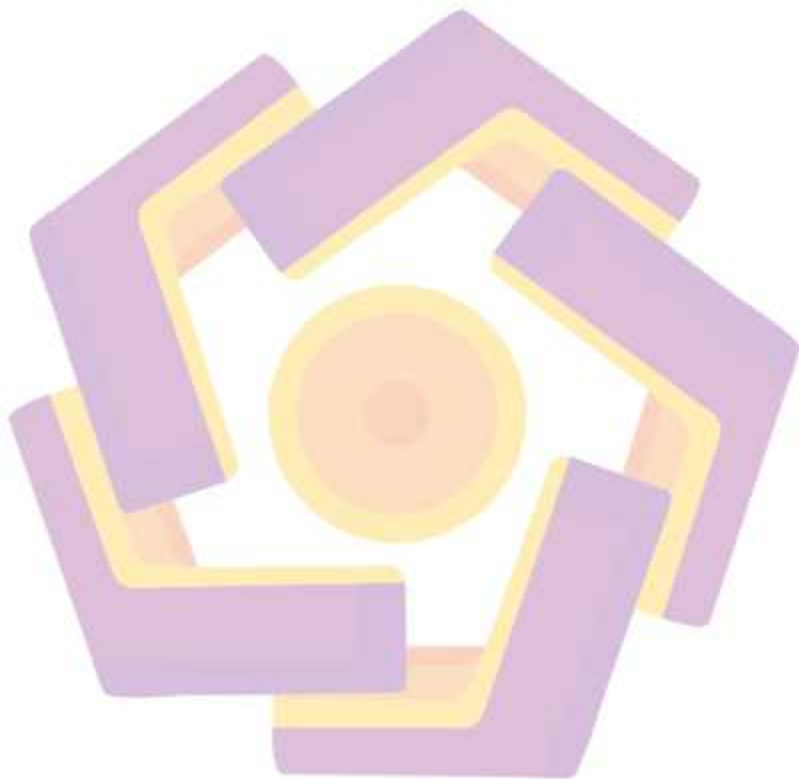
Dalam penelitian ini, data populasi ternak dari BPS periode 2006-2022 digunakan; 80% data digunakan untuk pelatihan dan 20% untuk pengujian. Evaluasi dengan metrik MAE dan RMSE menunjukkan bahwa model LSTM dapat membuat prediksi populasi ternak (kambing dan sapi potong) dengan akurasi tinggi (MAPE rendah), walau untuk beberapa jenis ternak seperti kerbau kesalahan lebih besar. Link download: [Jurnal Scientific and Applied Informatics \(UMB\) — artikel dapat diunduh dari situs jurnal. Jurnal UMB](#)

Dalam penelitian (Prabowo et al., 2025), data populasi ternak dari BPS periode 2006-2022 digunakan; 80% data digunakan untuk pelatihan dan 20% untuk pengujian. Evaluasi dengan metrik MAE dan RMSE menunjukkan bahwa model LSTM dapat membuat prediksi populasi ternak (kambing dan sapi potong) dengan akurasi tinggi (MAPE rendah), walau untuk beberapa jenis ternak seperti kerbau kesalahan lebih besar.

(Yohanes Anton Nugroho & Hotma Antoni Hutahaean, 2025) melakukan penelitian membandingkan dua model deep learning (LSTM dan CNN) dalam meramalkan penjualan sepeda motor nasional. Aritekin Journal Mereka menggunakan data dari Januari 2021 hingga Desember 2024, dan mengevaluasi prediksi untuk periode Januari–Maret 2025 dengan metrik MAE, MSE, RMSE, dan MAPE. Hasil penelitian menunjukkan bahwa LSTM menawarkan performa yang kompetitif, terutama apabila data memiliki pola musiman yang jelas dan variabilitas antar bulan.

Dengan mengintegrasikan hasil-hasil dari penelitian terdahulu, penelitian ini bertujuan untuk memberikan gambaran yang komprehensif mengenai kinerja

ARIMA, SARIMA, GRU dan LSTM dalam memprediksi data penjualan mobil Toyota. Tinjauan pustaka ini membangun fondasi yang kuat untuk menilai apakah temuan dari studi-studi sebelumnya dapat diterapkan dan divalidasi dalam konteks penelitian ini.



2.2 Keaslian Penelitian

Tabel 2.1 Matriks literatur review dan posisi penelitian

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1	Profit Prediction Using ARIMA, SARIMA and LSTM Models in Time Series Forecasting: A Comparison	Uppala Meena Sirisha, Manjula C. Belavagi, Girija Attigeri, IEEE Access, 2022	Membandingkan kinerja ARIMA, SARIMA, dan LSTM dalam memprediksi keuntungan penjualan.	LSTM memberikan akurasi tertinggi (97.01%), diikuti oleh SARIMA (94.378%) dan ARIMA (93.84%).	LSTM memerlukan sumber daya komputasi yang besar dan waktu yang lebih lama dibandingkan ARIMA dan SARIMA.	Penelitian ini membandingkan tiga model (ARIMA, SARIMA, dan LSTM) dalam konteks peramalan keuntungan penjualan. Penelitian yang akan dilakukan dapat mempertimbangkan GRU sebagai alternatif LSTM
2	ARIMA Model Time Series Forecasting	Mohd Faizan Rizvi, Shivang Sahu, Dr. Sadhana Rana, International	Memberikan tinjauan komprehensif tentang model ARIMA dan aplikasinya	ARIMA efektif untuk data deret waktu dengan tren dan pola musiman, tetapi memiliki keterbatasan dalam	Perlu eksplorasi lebih lanjut tentang model hybrid dan integrasi dengan metode deep learning untuk	Penelitian ini fokus pada tinjauan teoritis dan aplikasi ARIMA, sementara penelitian yang akan dilakukan dapat membandingkan ARIMA, SARIMA, dan GRU

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
		Journal for Research in Applied Science & Engineering Technology (IJRASET), 2024	dalam peramalan deret waktu, ARIMA efektif untuk data deret waktu dengan tren dan pola musiman, tetapi memiliki keterbatasan dalam menangani data non-linier.	menangani data non-linier.	meningkatkan akurasi.	dalam konteks penjualan mobil Toyota.
3	Perbandingan Metode ARIMA dan SARIMA Dalam Peramalan Jumlah Penumpang Bandara Provinsi Kepulauan Bangka Belitung	Jurnal Jambura of Mathematics, 2022	Membandingkan keakuratan metode ARIMA dan SARIMA dalam meramalkan jumlah penumpang bandara.	SARIMA memberikan hasil peramalan yang lebih akurat dibandingkan ARIMA.	Disarankan penggunaan data yang lebih bervariasi untuk melihat performa metode dalam kondisi berbeda.	Penelitian ini berfokus pada data penumpang bandara, sementara penelitian yang akan dilakukan memfokuskan pada data penjualan mobil Toyota. Selain itu, penelitian ini tidak melibatkan metode GRU.

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
2	Comparison of ARIMA and GRU Models for High-Frequency Time Series Forecasting	Scientific Journal of Informatics, 2023	Membandingkan model ARIMA dan GRU dalam peramalan data deret waktu dengan frekuensi tinggi.	GRU lebih efektif dalam menangani data dengan pola non-linear dibandingkan ARIMA.	Penelitian ini menyarankan eksplorasi lebih lanjut pada optimasi hyperparameter GRU.	Perbedaan utama adalah pada fokus data frekuensi tinggi, sedangkan penelitian yang akan dilakukan melibatkan data bulanan penjualan mobil. Juga, penelitian yang akan dilakukan menggunakan validasi K-Fold.
3	Comparison of ARIMA and SARIMA for Forecasting Crude Oil Prices	Jurnal RESTI, 2022	Mengevaluasi efektivitas model ARIMA dan SARIMA dalam peramalan harga minyak mentah.	SARIMA memiliki performa yang lebih baik dalam menangani data musiman.	Perlu penelitian tambahan dengan data yang lebih luas untuk validasi hasil.	Penelitian ini fokus pada harga minyak mentah, berbeda dengan fokus penelitian saat ini pada penjualan mobil. Penelitian ini juga tidak menyertakan model GRU
4	Perbandingan Kinerja Hasil Luaran Model	Jurnal Statistika, 2023	Membandingkan performa jaringan syaraf	Model jaringan syaraf tiruan lebih unggul dalam	Penelitian ini merekomendasikan penggunaan data	Penelitian ini menggunakan jaringan syaraf tiruan, sementara

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	Jaringan Syaraf Tiruan dan SARIMA Untuk Prediksi Awal Musim Hujan Kota Pangkalpinang		tiruan dan SARIMA dalam memprediksi awal musim hujan.	memprediksi pola yang kompleks.	tambahan untuk meningkatkan generalisasi model.	penelitian yang akan dilakukan memanfaatkan GRU untuk peramalan penjualan mobil. Selain itu, penelitian ini berfokus pada prediksi awal musim hujan, yang berbeda dengan fokus pada data penjualan dalam penelitian saat ini.
5	Sistem Peramalan Kebutuhan Hidup Layak Minimum (Kapita/Bulan) Kota Bandung Menggunakan Metode SARIMA dan SES	Jurnal Sains dan Inovasi Bisnis, 2022	Meramalkan kebutuhan hidup layak minimum menggunakan SARIMA dan SES.	SARIMA memberikan hasil peramalan yang lebih stabil dibandingkan SES.	Disarankan untuk menggabungkan metode lain untuk melihat perbaikan hasil.	Penelitian ini menggunakan data kebutuhan hidup, sedangkan penelitian yang akan dilakukan fokus pada data penjualan mobil. Selain itu, penelitian ini tidak menggunakan model GRU.

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
6	Prediksi Indeks Harga Saham Gabungan (IHSG) Menggunakan Algoritma Autoregressive Integrated Moving Average (ARIMA)	Jurnal Nasional	Memprediksi IHSG menggunakan ARIMA.	ARIMA cukup efektif dalam meramalkan IHSG, meskipun ada batasan dalam menangani data non-linear.	Disarankan untuk menggunakan metode hybrid untuk meningkatkan akurasi prediksi.	Penelitian ini berfokus pada IHSG, sedangkan penelitian yang akan dilakukan fokus pada penjualan mobil. Penelitian ini juga tidak menyertakan model GRU dalam analisis.
7	Prediksi Retail Sales Mobil Toyota Menggunakan Metode ARIMA	Adri Arisena, Indonesian Journal of Strategic Management, 2020	Memprediksi penjualan retail mobil Toyota menggunakan ARIMA.	ARIMA mampu memprediksi penjualan dengan akurasi yang memadai, tetapi kurang optimal dalam pola musiman.	Penelitian menyarankan eksplorasi model lain yang lebih baik dalam menangani pola musiman.	Penelitian saat ini akan menggabungkan model GRU dengan ARIMA dan SARIMA untuk melihat perbedaan performa dalam memprediksi data penjualan mobil Toyota, serta menggunakan evaluasi model yang lebih komprehensif.

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
8	Time Series Analysis for Forecasting Livestock Population in Indonesia with LSTM Model	Tito Prabowo, et al., <i>Jurnal Scientific and Applied Informatics (UMB)</i> , 2025	Menggunakan model LSTM untuk memprediksi populasi ternak di Indonesia berdasarkan data BPS 2006–2022.	LSTM mampu memprediksi populasi ternak dengan tingkat akurasi tinggi (MAPE rendah) untuk sebagian besar jenis ternak.	Model masih kurang optimal untuk jenis ternak dengan fluktuasi data ekstrem seperti kerbau.	Penelitian ini relevan dengan studi prediksi berbasis deret waktu karena sama-sama menggunakan LSTM dengan pendekatan data tahunan.
9	Pengembangan Model Deep Learning LSTM dan CNN untuk Peramalan Penjualan Sepeda Motor di Indonesia	Yohanes Anton Nugroho & Hotma Antoni Hutahaean, <i>Jupiter: Jurnal Penelitian dan Teknologi</i>	Membandingkan model LSTM dan CNN dalam peramalan penjualan sepeda motor nasional.	LSTM menunjukkan performa terbaik pada data dengan pola musiman dan variabilitas tinggi antar bulan.	Model memerlukan tuning parameter lebih kompleks untuk menghindari overfitting.	Penelitian ini memiliki kesamaan konteks (penjualan kendaraan) sehingga menjadi referensi kuat untuk penerapan LSTM pada data penjualan mobil.

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
		<i>Informasi, 2025</i>				

2.3 Landasan Teori

Landasan teori merupakan dasar konseptual yang digunakan untuk mendukung analisis dan pembahasan dalam penelitian ini. Setiap teori yang disajikan berfungsi untuk memberikan pemahaman ilmiah mengenai konsep, model, serta pendekatan analitis yang digunakan dalam proses peramalan penjualan mobil Toyota. Dengan menyusun landasan teori secara sistematis, penelitian ini memperoleh pijakan akademis yang kuat sehingga pembahasan dan interpretasi hasil dapat dilakukan secara terarah, logis, dan sesuai dengan kaidah ilmiah. Teori-teori berikut menjadi rujukan utama dalam menganalisis kinerja model statistik dan model pembelajaran mesin yang digunakan dalam penelitian ini.

2.3.1 *Time Series* dan Prediksi Data

Time series atau deret waktu adalah kumpulan data yang diukur pada interval waktu tertentu. Analisis time series berguna untuk memahami tren historis dan memprediksi data di masa depan, yang sangat berharga dalam bidang seperti keuangan, ekonomi, dan meteorologi. Dengan kemajuan teknologi, metode prediksi time series semakin canggih, memungkinkan hasil yang lebih akurat. Berbagai teknik seperti ARIMA dan GRU telah digunakan untuk menangani kompleksitas dalam data time series (Febiola et al., 2024).

Tujuan utama prediksi data time series adalah membantu pengambilan keputusan dengan estimasi yang berdasarkan pola historis. Pemilihan metode yang tepat dalam analisis data time series bisa meningkatkan ketepatan prediksi, mengurangi ketidakpastian, dan mendukung

perencanaan yang lebih baik. Sementara model statistik seperti ARIMA populer untuk analisis linier, model deep learning seperti GRU unggul dalam menangani pola yang lebih kompleks dan non-linear (Ridwan et al., 2023).

2.3.2 *Autoregressive Integrated Moving Average (ARIMA)*

ARIMA adalah metode statistik yang populer untuk analisis data time series. Model ini terdiri dari tiga elemen utama: komponen autoregresif (AR), integrasi atau differencing (I), dan rata-rata bergerak (MA). ARIMA sangat efektif untuk data yang stasioner atau yang bisa dibuat stasioner melalui proses differencing. Kelebihan ARIMA adalah kemampuannya dalam menangkap pola hubungan linear pada data. Model ARIMA biasanya digunakan untuk memprediksi sebuah variabel seri waktu dengan menyatukan proses autoregresif (AR) dan rata-rata gerak (MA) (Hamiane et al., 2024).

Persamaan dasar untuk komponen AR diilustrasikan persamaan 2.1 di bawah ini:

$$x(t) - \alpha = 1 - \sum_{i=1}^p \alpha_i x(t-i) \quad (2.1)$$

dimana, t adalah langkah waktu, $x(t)$ adalah nilai prediksi, p adalah jumlah suku AR, dan α adalah koefisien AR. Bagian moving average (MA) adalah persamaan 2.2 berikut:

$$x(t) - \beta = 1 - \sum_{i=1}^q \beta_i \epsilon(t-i) \quad (2.2)$$

dimana q sebagai jumlah suku MA, β koefisien MA, dan ϵ suku kesalahan (error term). Dengan mengintegrasikan keduanya, model ARIMA dapat dinyatakan sebagai persamaan 2.3:

$$x(t) - \alpha - 1 \sum p \alpha x(t-i) + 1 - 1 \sum q \beta \epsilon(t-i) \quad (2.3)$$

Berikut merupakan cara kerja model ARIMA:

1. Pertama data dianalisis untuk menentukan apakah stasioner atau tidak. Jika tidak stasioner, dilakukan differencing untuk mencapai stasioneritas. Ini melibatkan pengurangan nilai pengamatan pada waktu t dengan nilai pengamatan pada waktu sebelumnya (Buchori & Sukmono, 2018; Pratiwi, 2013).
2. Kemudian, parameter p, d , dan q ditentukan berdasarkan analisis Autocorrelation Function (ACF) dan Partial Autocorrelation Function (PACF) untuk memilih orde model yang tepat.
3. Selanjutnya yaitu estimasi parameter, metode estimasi seperti Maximum Likelihood Estimation digunakan untuk menghitung parameter model AR dan MA. Parameter ini akan mengoptimalkan kesesuaian model terhadap data (Pratiwi, 2013).
4. Kemudian Evaluasi model yang telah dibangun melalui analisis residual untuk memastikan tidak ada pola yang tersisa dalam data yang

dapat mempengaruhi akurasi prediksi. Ini juga mencakup pemeriksaan kesalahan estimasi (Buchori & Sukmono, 2018).

5. Setelah model terverifikasi, model tersebut digunakan untuk meramalkan nilai di masa depan berdasarkan data historis yang ada. Proses ini dilakukan secara iteratif hingga periode yang diinginkan (Buchori & Sukmono, 2018; Pratiwi, 2013).

2.3.3 *Seasonal Autoregressive Integrated Moving Average (SARIMA)*

SARIMA adalah versi lanjutan dari ARIMA yang dirancang untuk mengakomodasi data dengan pola musiman. Dengan menambahkan parameter musiman, SARIMA mampu menangkap pola berulang dalam periode tertentu. Model ini cocok digunakan pada data time series yang menunjukkan fluktuasi musiman, seperti data penjualan musiman atau perubahan cuaca.

Persamaan SARIMA dapat dilihat pada persamaan 2.4 yaitu:

$$\phi_p(B^s)\phi_r(B)(1-B)^d(1-B^s)^D Y_t = \phi_q(B)\phi_Q(B^s)\varepsilon_t \quad (2.4)$$

Dimana p, P mewakili Autoregressive; d, D : differencing , q, Q mewakili Moving Average , $\phi_p(B)$ mewakili tingkat Autoregressive non musiman , $\phi_p(B^s)$ mewakili tingkat Autoregressive musiman , $(1-B)^d$ mewakili tingkat differencing non musiman, $(1-B^s)^D$ tingkat differencing musiman, $\phi_q(B)$ mewakili Moving Average non musiman, $\phi_Q(B^s)$ mewakili Moving Average musiman, Y_t mewakili data aktual ke-t; ε_t mewakili error periode t.

Berikut merupakan cara kerja model ARIMA:

1. Langkah pertama adalah memeriksa stasioneritas data, tetapi juga mempertimbangkan komponen musiman. Differencing musiman dilakukan jika diperlukan untuk mencapai stasioneritas.
2. Menentukan parameter musiman P, D , dan Q yang berkaitan dengan komponen musiman dari data, di samping parameter non-musiman p, d , dan q .
3. Semua parameter diestimasi menggunakan metode yang sama seperti pada ARIMA dan dilakukan pemeriksaan diagnostik untuk memastikan model sesuai dengan data.
4. model SARIMA digunakan untuk meramalkan nilai di masa depan dengan memperhitungkan pola musiman dalam data

2.3.4 Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU) adalah jenis jaringan saraf berulang yang dirancang untuk memodelkan data time series yang kompleks. GRU mengatasi masalah vanishing gradient yang umum dalam jaringan saraf, memungkinkan pengolahan informasi jangka panjang lebih efektif. Model ini lebih sederhana dibandingkan LSTM tetapi tetap efektif dalam menangani data dengan ketergantungan jangka panjang (Ridwan et al., 2023)

Penelitian membandingkan GRU dengan ARIMA dan LSTM menunjukkan bahwa GRU memiliki kinerja yang lebih baik dalam prediksi data time series yang kompleks dan non-linear. Keunggulan GRU terletak pada

kemampuannya untuk memproses data non-stasioner dengan efisiensi tinggi, menjadikannya pilihan ideal untuk aplikasi yang memerlukan analisis time series mendalam (Hindarto, 2023)

Berikut Merupakan fungsi yang digunakan untuk mengontrol mekanisme penguncian dalam sel GRU:

$$z_t = \sigma(x_t W^z + h_{t-1} U^z + b_z) \quad (2.5)$$

$$r_t = \sigma(x_t W^r + h_{t-1} U^r + b_r) \quad (2.6)$$

$$h'_t = \tan(r_t \times h_{t-1} U + x_t W + b) \quad (2.7)$$

$$h_t = (1 - z_t) \times h'_t + z_t \times h_{t-1} \quad (2.8)$$

Dimana W^z , W^r , W menunjukkan matriks bobot untuk vektor input terhubung yang sesuai. U^z , U^r , U mewakili matriks bobot dari langkah waktu sebelumnya, dan b_r , b_z , dan b adalah bias. σ menyatakan fungsi sigmoid logistik, r_t menyatakan gerbang reset, z_t menyatakan gerbang update, dan h'_t menyatakan kandidat hidden layer (Mateus et al., 2021).

Cara kerja model GRU sebagai berikut (Ayu Wulandari dan Rahmat Gemowo, 2019):

1. Data urutan dimasukkan ke dalam jaringan GRU, di mana setiap elemen input berinteraksi dengan unit-unit dalam jaringan.
2. Pengaturan Gate, GRU menggunakan dua gate utama. Reset Gate (r_t) yang mengontrol seberapa banyak informasi dari keadaan sebelumnya yang akan dilupakan. Update Gate (z_t), yang mengontrol seberapa banyak informasi baru yang akan ditambahkan ke keadaan saat ini.

Persamaan gate ini memungkinkan GRU untuk memutuskan informasi mana yang relevan untuk dipertahankan atau dilupakan selama proses pelatihan.

3. Proses pembaruan status, Status tersembunyi diperbarui menggunakan informasi dari input saat ini serta status sebelumnya, dengan mempertimbangkan pengaturan dari kedua gate tersebut.
4. Hasil akhir dari jaringan GRU adalah output yang dihasilkan berdasarkan informasi relevan dari seluruh urutan input, memungkinkan model ini untuk menangkap pola jangka panjang dalam data urutan.

2.3.5 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) merupakan salah satu jenis arsitektur jaringan saraf tiruan yang termasuk dalam kategori Recurrent Neural Network (RNN). LSTM diperkenalkan pertama kali oleh Hochreiter dan Schmidhuber (1997) untuk mengatasi kelemahan mendasar RNN konvensional, yaitu masalah vanishing gradient dan exploding gradient yang menghambat kemampuan jaringan dalam mempelajari hubungan jangka panjang antar data dalam deret waktu.

Berbeda dengan RNN biasa yang hanya memiliki hidden state tunggal, LSTM memiliki struktur internal berupa cell state dan tiga gerbang pengendali utama (gates), yaitu input gate, forget gate, dan output gate, yang berfungsi mengatur aliran informasi masuk, tersimpan, dan keluar dari sel memori (Dai et al., 2025). Melalui mekanisme ini, LSTM dapat menyimpan informasi yang relevan dalam jangka panjang sekaligus melupakan

informasi yang tidak diperlukan, menjadikannya sangat efektif dalam menangkap pola musiman, tren jangka panjang, dan dinamika non-linear yang kompleks pada data deret waktu. Mekanisme internal LSTM adalah sebagai berikut

1. LSTM memiliki cell state (C_t) yang menyimpan “memori jangka panjang” dan hidden state (h_t) untuk output saat ini.
2. Ada tiga gerbang utama:
 - a. Forget gate (f_t): memilih informasi apa yang dilupakan dari C_{t-1} .
 - b. Input gate (i_t): memilih informasi baru yang akan dimasukkan ke memori.
 - c. Output gate (o_t): memutuskan bagian dari memori yang akan dikeluarkan sebagai h_t .

Berikut adalah persamaan untuk satu sel LSTM pada langkah waktu (timestep) t :

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2.9)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2.10)$$

$$C'_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (2.11)$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot C'_t \quad (2.12)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (2.13)$$

$$h_t = o_t \cdot \tanh(C_t) \quad (2.14)$$

Dimana f_t merupakan forget Gate yang berfungsi menentukan informasi mana dari state sel sebelumnya C_{t-1} yang akan dibuang, i_t merupakan input gate berfungsi untuk menentukan informasi baru mana yang akan

disimpan ke dalam state sel, C'_t merupakan new candidate cell state berfungsi membuat vektor kandidat nilai baru yang bisa ditambahkan ke state sel, C_t merupakan cell state update untuk memperbarui state sel lama C_{t-1} menjadi state sel baru C_t . Ini adalah inti dari "memori" LSTM, o_t merupakan output update untuk menentukan bagian mana dari state sel yang akan dibaca dan dikeluarkan sebagai hidden state, dan h_t merupakan hidden state yang akan diteruskan ke langkah waktu berikutnya atau ke layer berikutnya. Variable yang digunakan yakni sebagai berikut x_t : Input pada langkah waktu, h_{t-1} mewakili Hidden state dari langkah waktu sebelumnya, C_{t-1} mewakili State sel dari langkah waktu sebelumnya, W_f, W_i, W_c, W_o mewakili Matriks bobot untuk setiap gerbang dan state, b_f, b_i, b_c, b_o mewakili Vektor bias untuk setiap gerbang dan state, σ mewakili Fungsi aktivasi sigmoid (menghasilkan nilai antara 0 dan 1, seperti saklar), \tanh mewakili Fungsi aktivasi tangen hiperbolik (menghasilkan nilai antara -1 dan 1), dan $[h_{t-1}, x_t]$ mewakili Penggabungan (concatenation) vector h_{t-1} dan x_t .

2.3.6 Optimasi Algoritma

Optimasi algoritma adalah upaya untuk meningkatkan performa suatu metode, baik dari segi kecepatan, ketepatan, efisiensi komputasi, maupun kemampuan generalisasi. Tujuan akhirnya mencakup penurunan tingkat galat (bias dan varians), percepatan proses pelatihan, serta pencegahan overfitting atau underfitting. Upaya ini dapat ditempuh melalui pemilihan

hyperparameter yang tepat, penerapan teknik regularisasi, maupun penyesuaian arsitektur model.

Optimasi memegang peranan krusial dalam machine learning dan peramalan deret waktu karena secara langsung memengaruhi daya prediksi suatu model. Beragam pendekatan yang lazim ditempuh meliputi seleksi model, regularisasi, dan tuning hyperparameter. Seleksi model yaitu memilih model paling sesuai berdasarkan kriteria statistik. Seperti AIC atau BIC, untuk menyeimbangkan kompleksitas dan akurasi. Regularisasi yaitu Menambahkan penalti terhadap bobot atau kompleksitas agar model tidak terlalu menyesuaikan diri pada data pelatihan. Tuning hyperparameter yaitu menyetel parameter pelatihan, misalnya learning rate, ukuran batch, atau jumlah lapisan guna memperoleh titik keseimbangan antara kecepatan konvergensi dan akurasi prediksi.

ARIMA dan SARIMA merupakan model statistik yang banyak diterapkan dalam peramalan deret waktu karena kemampuannya menangani pola non-stasioner maupun musiman secara efektif (Ahmadi et al., 2023). Hyperparameter adalah parameter yang tidak dipelajari langsung dari data, melainkan ditetapkan sebelum proses pelatihan model dan mengontrol perilaku pembelajaran, misalnya orde p, d, q pada ARIMA serta P, D, Q, m pada SARIMA (Sah et al., 2023). Pemilihan nilai hyperparameter yang tepat sangat penting karena dapat memengaruhi akurasi dan kemampuan generalisasi model.

Untuk memperoleh kombinasi hyperparameter terbaik, sering digunakan teknik grid search, yaitu metode pencarian sistematis yang mengevaluasi setiap kombinasi nilai parameter yang telah ditentukan dalam sebuah "grid" atau kisi pencarian, kemudian memilih konfigurasi dengan kriteria evaluasi paling optimal seperti metrik evaluasi seperti MAE dan RMSE (Ahmadi et al., 2023; Sah et al., 2023). Pendekatan ini memungkinkan proses optimasi yang terstruktur dan dapat direplikasi pada berbagai skenario peramalan.

Model GRU adalah varian dari Jaringan Saraf Rekurens (RNN) yang dirancang untuk mengatasi masalah gradien yang menghilang dan meningkatkan kinerja prediksi deret waktu. Optimasi GRU dilakukan melalui berbagai teknik regularisasi dan penyesuaian hiperparameter. (Chung et al., 2014) meliputi L1&L2 Regularization, Dropout, Batch Normalization, Deeper Architecture, Smaller Learning Rate, Early Stopping, dan Validation Split. L1 (Lasso) menghasilkan sparsitas dengan mendorong beberapa bobot menjadi nol sedangkan L2 (Ridge) untuk membatasi magnitudo bobot untuk mencegah overfitting. Dropout berfungsi untuk menonaktifkan secara acak sebagian neuron selama pelatihan untuk mencegah ketergantungan berlebihan pada fitur tertentu. Batch Normalization untuk menormalkan input setiap lapisan untuk mempercepat pelatihan dan meningkatkan stabilitas model. Deeper Architecture meningkatkan jumlah lapisan GRU untuk menangkap pola yang lebih kompleks, tetapi harus diseimbangkan dengan teknik regularisasi. Smaller Learning Rate berfungsi untuk mengurangi laju

pembelajaran untuk menghindari overshooting selama pembaruan bobot. Early Stopping untuk menghentikan proses ketika kinerja model pada data validasi tidak membaik. Validation Split membagi data menjadi set pelatihan dan validasi untuk memantau overfitting.

Sedangkan model LSTM merupakan pengembangan dari jaringan saraf berulang (Recurrent Neural Network) yang dirancang untuk mempelajari hubungan jangka panjang serta menangani pola non-linear pada data deret waktu. Dalam konteks penelitian pendidikan berbasis data, seperti analisis performa pembelajaran, prediksi hasil belajar, atau kajian perilaku peserta didik dari data longitudinal, model LSTM menjadi relevan karena kemampuannya mengenali pola temporal yang kompleks (Hurtado-Mora et al., 2024). Kinerja model LSTM tidak hanya ditentukan oleh struktur arsitektur jaringan, tetapi juga sangat bergantung pada penentuan hyperparameter. Hyperparameter merupakan parameter eksternal yang tidak dipelajari langsung oleh model, melainkan ditetapkan sebelum proses pelatihan untuk mengatur perilaku pembelajaran. Hyperparameter yang umum dioptimalkan dalam penelitian meliputi jumlah neuron pada lapisan tersembunyi (hidden units), jumlah lapisan LSTM, ukuran batch (batch size), laju pembelajaran (learning rate), jumlah epoch, serta tingkat dropout yang digunakan untuk menghindari overfitting. Pemilihan nilai hyperparameter yang tepat merupakan langkah penting untuk meningkatkan akurasi, efisiensi pelatihan, serta kemampuan generalisasi model terhadap data baru.

2.3.7 Evaluasi Kinerja Model

Evaluasi kinerja model prediksi sangat penting untuk memastikan akurasi dan keandalan hasil prediksi. Beberapa metrik umum yang digunakan adalah Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), Mean Squared Error (MSE), dan Root Mean Squared Error (RMSE).

MAPE memiliki kriteria atau standar interpretasi yang akan berguna untuk menilai kualitas dari performa model yang diperlukan untuk hasil akhir model. MAPE memberikan gambaran tentang kesalahan prediksi dalam bentuk persentase. MAE merupakan nilai rata-rata dari perbedaan absolut antara nilai aktual dan nilai prediksi (Melati Putri Wiryawanto et al., 2023). MSE merupakan metrik evaluasi yang umum digunakan dalam statistik dan machine learning untuk mengukur seberapa akurat sebuah model regresi dalam memprediksi nilai numerik. MSE menghitung rata-rata kuadrat dari selisih antara nilai yang diprediksi oleh model dan nilai aktual. Dengan kata lain, MSE memberikan gambaran tentang seberapa besar kesalahan prediksi yang dihasilkan oleh model. Sedangkan RMSE merupakan metode untuk mengukur bias atau perbedaan nilai prediksi model (Melati Putri Wiryawanto et al., 2023). Secara singkat MAPE memberikan gambaran tentang kesalahan prediksi dalam bentuk persentase, sedangkan MAE dan MSE mengukur kesalahan absolut dan kuadrat dari prediksi (Melati Putri Wiryawanto et al., 2023). Persamaan metrik evaluasi tersebut dapat dilihat pada persamaan dibawah ini.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left(\frac{y_i - y'_i}{y_i} \right) \times 100\% \quad (2.15)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - y'_i| \quad (2.16)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - y'_i)^2 \quad (2.17)$$

$$RMSE = \sqrt{MSE} \quad (2.18)$$

Dimana n adalah jumlah total data, y_i adalah nilai aktual dari data ke- i dan y'_i adalah nilai prediksi dari model untuk data ke- i

BAB 3

METODE PENELITIAN

3.1 Jenis, Sifat, dan Pendekatan Penelitian

Penelitian ini menggunakan penelitian kuantitatif. Penelitian kuantitatif merupakan metode penelitian yang menggunakan data numerik untuk menguji hipotesis dan menjawab pertanyaan penelitian. Metode ini bersifat sistematis dan terstruktur, sering kali melibatkan penggunaan instrumen seperti survei atau kuesioner untuk mengumpulkan data. Tujuan utama dari penelitian kuantitatif adalah untuk menghasilkan temuan yang dapat digeneralisasi dan memberikan dasar untuk pengambilan keputusan berdasarkan analisis statistik. Penelitian ini juga mengembangkan model matematis dan teori yang berhubungan dengan fenomena sosial melalui pengumpulan data numerik (Ali et al., n.d.).

Penelitian ini bertujuan untuk mengukur, menganalisis, dan membandingkan kinerja model prediksi ARIMA, SARIMA, GRU, dan LSTM. Data yang digunakan berupa data penjualan mobil Toyota dalam bentuk deret waktu. Penelitian kuantitatif dipilih karena pendekatan ini memungkinkan evaluasi kinerja model berdasarkan perhitungan metrik evaluasi seperti MAE, MSE, RMSE, dan MAPE.

Penelitian ini bersifat deskriptif-komparatif. Deskriptif-komparatif adalah jenis penelitian kuantitatif yang bertujuan untuk menggambarkan karakteristik dari suatu fenomena dan membandingkannya antara dua kelompok atau lebih. Dalam penelitian ini, peneliti mengumpulkan data untuk memberikan gambaran yang jelas

mengenai variabel yang diteliti dan menganalisis perbedaan atau persamaan di antara kelompok tersebut (Candra Susanto et al., n.d.).

Penelitian ini bertujuan untuk mendeskripsikan performa masing-masing model dalam memprediksi data penjualan mobil Toyota serta melakukan perbandingan antara model yang digunakan. Pendekatan yang digunakan adalah pendekatan eksperimental. Pendekatan eksperimental dalam penelitian kuantitatif melibatkan manipulasi variabel independen untuk mengamati efeknya terhadap variabel dependen. Dalam desain eksperimen, subjek dibagi menjadi kelompok eksperimen (yang menerima perlakuan) dan kelompok kontrol (yang tidak menerima perlakuan). Pendekatan ini memungkinkan peneliti untuk menarik kesimpulan tentang hubungan sebab-akibat dengan lebih jelas, karena variabel lain dapat dikontrol atau diatur (Rustendi, n.d.). Penelitian ini mengolah dan menganalisis data penjualan mobil Toyota menggunakan ketiga model prediksi ARIMA, SARIMA, GRU, dan LSTM.

3.2 Metode Pengumpulan Data

Penelitian ini menggunakan data penjualan mobil Toyota yang diambil dari situs resmi Toyota. Data yang dikumpulkan mencakup penjualan bulanan dari Januari 2011 hingga Mei 2023. Periode waktu ini dipilih untuk memberikan gambaran menyeluruh tentang tren penjualan selama lebih dari satu dekade. Pengumpulan data dilakukan melalui metode berikut:

1. Sumber Data : Data penjualan diperoleh dari situs resmi Toyota, yang menyajikan informasi penjualan berbagai model mobil mereka. Data ini

diakses dan diunduh dalam format yang dapat digunakan, seperti CSV atau Excel, untuk keperluan analisis lebih lanjut.

2. Periode Pengumpulan Data : Rentang waktu dari Januari 2011 hingga Mei 2023 memberikan cakupan data yang cukup panjang untuk menganalisis perubahan dan pola dalam penjualan mobil Toyota. Dengan rentang waktu yang luas, hasil analisis diharapkan lebih representatif dan dapat diandalkan.
3. Proses Pengumpulan : Data diunduh dari situs web Toyota dan diorganisir untuk memastikan semua informasi yang dibutuhkan tersedia dan dalam format yang benar.
4. Validasi dan Verifikasi: Setelah data dikumpulkan, langkah berikutnya adalah memeriksa data untuk menghindari duplikasi atau ketidakkonsistenan. Jika ditemukan data yang hilang atau tidak lengkap, langkah-langkah koreksi seperti interpolasi akan digunakan untuk memastikan kelengkapan data.
5. Preprocessing Data : Sebelum analisis, data diproses untuk memastikan semuanya dalam format yang sesuai. Proses ini mencakup normalisasi data dan pengelolaan data yang hilang, sehingga siap digunakan dalam model prediktif yang akan digunakan dalam penelitian.

3.3 Metode Analisis Data

Penelitian ini melibatkan beberapa tahap analisis data yang dirancang untuk memprediksi penjualan mobil Toyota secara akurat. Setiap tahap dijelaskan dengan detail untuk memastikan kejelasan metodologi yang digunakan.

Tahap awal dilakukan eksplorasi data awal yakni, statistik deskriptif yaitu memahami karakteristik awal data yaitu mencakup perhitungan rata-rata, median,

serta distribusi frekuensi untuk menggambarkan pola umum dalam data. Selanjutnya yaitu memvisualisasikan data menggunakan grafik garis dan box plot yang bertujuan untuk mendeteksi pola, tren musiman, dan anomali outlier yang dapat mempengaruhi analisis.

Tahapan selanjutnya yaitu membagi dataset menjadi 2 bagian, yaitu data training 80 % dan data testing 20%. Pembagian ini membantu memastikan bahwa model diuji pada data yang tidak pernah dilihat selama pelatihan, meningkatkan validitas hasil.

Setelahnya dilakukan pembentukan model ARIMA, SARIMA, GRU, dan LSTM. Lalu kemudian dilakukan evaluasi terhadap model-model tersebut. Evaluasi model yang digunakan yaitu, MAE, MAPE, MSE, dan RMSE.

Setelah itu dilakukan optimasi masing-masing model. Optimasi ARIMA dan SARIMA menggunakan hyperparameter tuning dengan teknik grid search, untuk optimasi GRU menggunakan penerapan teknik regularisasi, sedangkan optimasi LSTM menggunakan hyperparameter tuning. Model-model setelah optimasi juga akan dievaluasi menggunakan metrik evaluasi MAE, MAPE, MSE, dan RMSE.

Setelah model dievaluasi, hasil prediksi dianalisis untuk memahami tren dan pola yang ada dalam data. Perbandingan antara model dilakukan berdasarkan metrik evaluasi untuk menentukan model yang memberikan hasil prediksi terbaik. Perbandingan evaluasi model sebelum dan sesudah optimasi juga dilakukan untuk mendapatkan hasil paling optimal. Kemudian hasil analisis dan prediksi divisualisasikan dalam bentuk grafik dan tabel untuk memudahkan interpretasi.

Grafik-grafik ini mencakup plot time series aktual dan prediksi. Berdasarkan hasil analisis, dibuat interpretasi terhadap performa model. Kesimpulan diambil mengenai model mana yang paling efektif untuk prediksi data time series dalam konteks penelitian ini.

Alat dan bahan perangkat lunak yang digunakan yaitu, python sebagai bahasa pemrograman untuk analisis data dengan pustaka seperti pandas, numpy, matplotlib, statsmodel, dan tensorflow. Yang selanjutnya yaitu google Colab digunakan sebagai platform untuk menulis, mendokumentasikan, dan menjalankan analisis interaktif.

Dengan metode yang komprehensif ini, langkah-langkah dirancang untuk memastikan bahwa analisis data dilakukan secara sistematis dan hasil yang diperoleh dapat diandalkan untuk menjawab pertanyaan penelitian serta mencapai tujuan yang telah ditetapkan.

3.4 Alur Penelitian

Alur penelitian ini disusun untuk memberikan gambaran sistematis tentang langkah-langkah yang diambil mulai dari pengumpulan data hingga evaluasi model. Diagram alur berikut mencerminkan proses penelitian secara keseluruhan meliputi Pengumpulan Data, Preprocessing Data, Eksplorasi Data, Pembagian Data, Pemodelan, Optimasi Model, Evaluasi Model, Perbandingan Hasil, Interpretasi dan Kesimpulan.

Pada tahap Pengumpulan Data, data penelitian diperoleh dari situs resmi Toyota Indonesia, dengan cakupan periode Januari 2011 hingga Mei 2023.

Pemilihan data penjualan mobil Toyota didasarkan pada pertimbangan bahwa industri otomotif merupakan sektor yang sensitif terhadap perubahan tren pasar, sehingga prediksi penjualan menjadi penting untuk mendukung perencanaan bisnis, strategi pemasaran, dan manajemen persediaan.

Tahap Preprocessing Data, dilakukan untuk memastikan kualitas data sebelum digunakan dalam pemodelan. Proses yang dilakukan yaitu pengecekan data kosong, transformasi data yaitu menyalarkan format tanggal menjadi format standar deret waktu (time series), dan penetapan kolom "Month" sebagai indeks utama.

Selanjutnya tahap Eksplorasi Data, tahap ini dilakukan untuk memahami karakteristik awal data. Beberapa kegiatan yang dilakukan mencakup Analisis statistik deskriptif (mean, median, standar deviasi), visualisasi deret waktu untuk mendeteksi pola tren dan musiman, dan pemeriksaan stasioneritas awal dengan uji Augmented Dickey-Fuller (ADF). Hasil eksplorasi menjadi dasar dalam menentukan metode dan parameter awal pada model.

Pada Pembagian Data, Dataset dibagi menjadi dua subset: 80% untuk data latih (training) dan 20% untuk data uji (testing). Proporsi ini dipilih agar model memperoleh cukup informasi dari data historis namun tetap dapat diuji dengan data yang belum pernah dilihat sebelumnya.

Tahap selanjutnya yaitu Pembuatan Model, model yang digunakan untuk penelitian ini yakni ARIMA, SARIMA, GRU, dan LSTM. ARIMA dipilih karena model ini merupakan salah satu metode klasik dalam peramalan deret waktu yang sangat efektif untuk data yang memiliki pola trend tetapi tanpa komponen musiman

yang kuat. Keunggulan ARIMA terletak pada kemampuannya memberikan interpretasi yang jelas dan transparan, sehingga hasilnya mudah dipahami oleh praktisi maupun manajemen bisnis. Dengan demikian, ARIMA relevan sebagai baseline model yang mewakili pendekatan statistik tradisional. Sedangkan SARIMA digunakan untuk mengakomodasi kemungkinan adanya pola musiman dalam data penjualan. Berbeda dengan ARIMA, SARIMA menambahkan komponen musiman (P,D,Q,m) yang memungkinkan model menangkap siklus tahunan atau periodik yang berulang. Pemilihan SARIMA dalam penelitian ini didasarkan pada pertimbangan bahwa penjualan mobil berpotensi memiliki fluktuasi musiman, misalnya terkait periode promosi atau tren pasar tertentu. Dengan demikian, SARIMA melengkapi kelemahan ARIMA dengan memberikan fleksibilitas tambahan pada pola musiman. Model GRU diterapkan karena data penjualan juga berpotensi menunjukkan pola non-linear dan kompleks yang sulit ditangkap oleh model statistik. GRU, sebagai salah satu varian Recurrent Neural Network (RNN), memiliki kemampuan untuk mempelajari ketergantungan jangka panjang dalam data deret waktu dengan efisiensi karena struktur yang lebih sederhana namun tetap efektif. Pemilihan GRU bertujuan untuk menguji sejauh mana pendekatan deep learning dapat mengungguli model statistik klasik, terutama dalam menangani fluktuasi yang tidak teratur. Sedangkan model LSTM sebagai model deep learning lanjutan yang dirancang untuk mengatasi kelemahan RNN tradisional, terutama dalam menghadapi masalah vanishing gradient dan kesulitan dalam mengingat informasi jangka panjang. LSTM memiliki struktur internal yang lebih kompleks dibanding GRU, karena terdiri atas tiga gerbang utama yaitu input

gate, forget gate, dan output gate, yang secara dinamis mengatur informasi mana yang disimpan, dilupakan, dan dikeluarkan dari memori (Hurtado-Mora et al., 2024).

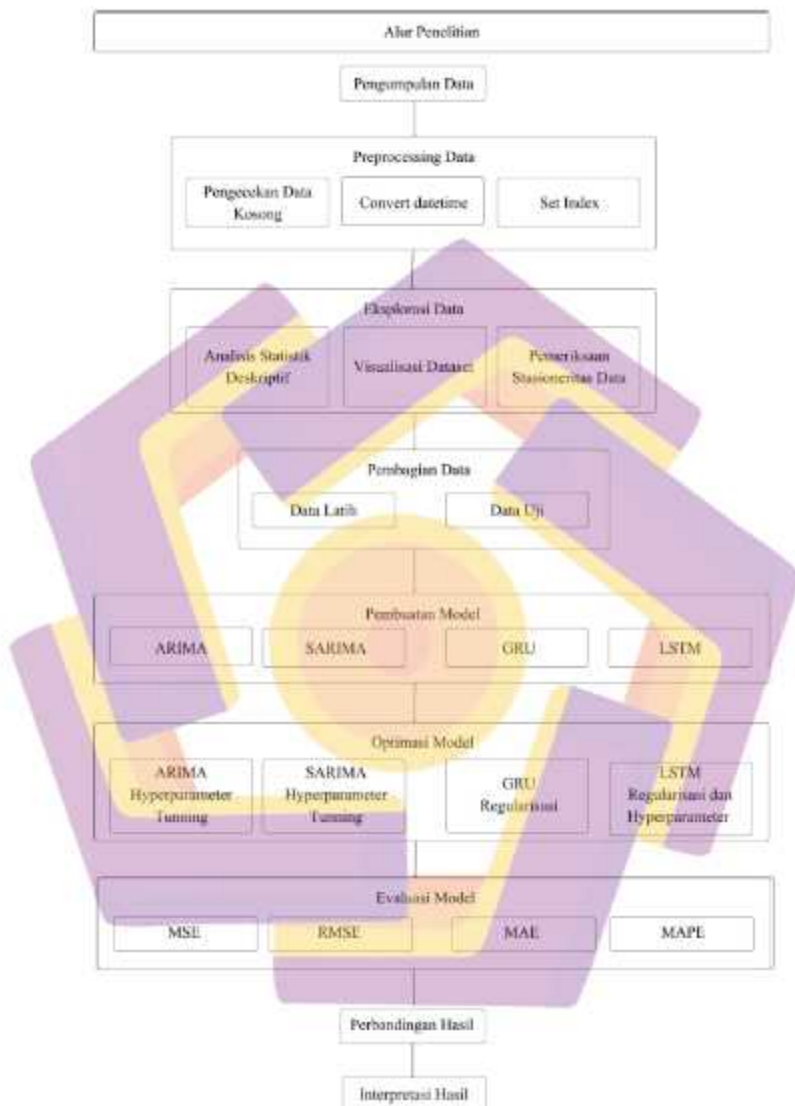
Selanjutnya tahap Optimasi Model, optimasi dilakukan untuk memperoleh konfigurasi parameter terbaik. ARIMA dan SARIMA parameter (p,d,q) dan (P,D,Q,m) dioptimalkan menggunakan hyperparameter tuning dengan teknik grid search berbasis untuk memilih model dengan keseimbangan terbaik antara kompleksitas dan akurasi. Optimasi GRU dilakukan melalui regularisasi dropout dan pengaturan window size agar model lebih adaptif terhadap data dan terhindar dari overfitting. Sedangkan optimasi LSTM yaitu menggunakan hyperparameter tuning.

Dilanjutkan tahap Pelatihan Model, model dilatih menggunakan data training. Proses ini melibatkan estimasi parameter pada ARIMA/SARIMA dan pembaruan bobot pada GRU dan LSTM melalui algoritma backpropagation. Model yang telah dilatih kemudian disimpan untuk digunakan pada proses prediksi terhadap data testing. Langkah ini memungkinkan peneliti mengevaluasi sejauh mana kemampuan generalisasi masing-masing model, baik model statistik (ARIMA dan SARIMA) maupun model berbasis deep learning (GRU dan LSTM), dalam memprediksi data penjualan mobil Toyota. Kemudian dilakukan Evaluasi Model yakni menghitung matrik evaluasi MAPE, MAE, MSE, dan RMSE untuk masing-masing model baik model sebelum optimasi maupun setelah dilakukan optimasi.

Selanjutnya tahap Perbandingan Model, pada tahap ini hasil evaluasi dari keempat model akan dibandingkan untuk menilai model mana yang paling efektif dalam memprediksi penjualan mobil Toyota. Perbandingan ini akan mencakup analisis kelebihan dan kekurangan masing-masing model. Tahap ini merupakan salah satu kontribusi baru bagi penelitian, yaitu mengetahui perbandingan keempat model tersebut baik sebelum maupun sesudah optimasi pada masing-masing model tersebut.

Tahap terakhir yaitu Interpretasi, yakni menganalisis hasil evaluasi untuk menentukan model yang terbaik untuk prediksi penjualan mobil Toyota. Kemudian menyusun kesimpulan berdasarkan hasil analisis dan memberikan rekomendasi. Penjelasan alur diatas dapat digambarkan sebagai berikut:





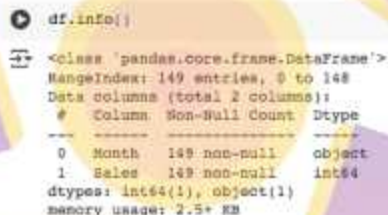
Gambar 3.1 Alur Penelitian

BAB 4

HASIL PENELITIAN DAN PEMBAHASAN

4.1 Pengumpulan Data

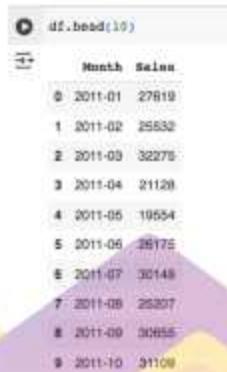
Penelitian ini menggunakan data penjualan mobil Toyota yang diambil dari situs resmi Toyota. Data yang dikumpulkan mencakup penjualan bulanan dari Januari 2011 hingga Mei 2023. Periode waktu ini dipilih untuk memberikan gambaran menyeluruh tentang tren penjualan selama lebih dari satu dekade. Berikut merupakan informasi data yang digunakan dalam penelitian ini, dilihat dari gambar dibawah ini.



```
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149 entries, 0 to 148
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
---  -
 0   Month   149 non-null    object
 1   Sales   149 non-null    int64
dtypes: int64(1), object(1)
memory usage: 2.5+ KB
```

Gambar 4.1 Informasi Dataset

Dapat diketahui bahwa dataset tersebut memiliki 149 baris data dengan 2 kolom yaitu Month dengan tipe data object dan Sales dengan tipe data int64. Selanjutnya yaitu dilihat isi data menggunakan fungsi head(10). Yaitu melihat 10 baris awal data untuk mendapatkan gambaran besar data, dapat dilihat pada gambar dibawah ini.



```
df.head(10)
```

	Month	Sales
0	2011-01	27819
1	2011-02	25632
2	2011-03	32275
3	2011-04	21128
4	2011-05	19554
5	2011-06	26175
6	2011-07	30148
7	2011-08	25207
8	2011-09	30955
9	2011-10	31109

Gambar 4.2 Gambaran Dataset

4.2 Preprocessing Data

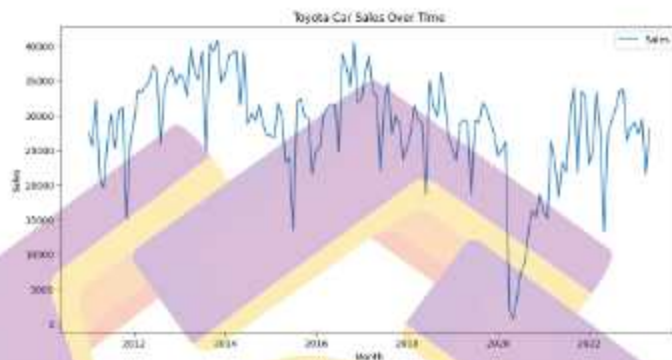
Sebelum dilakukan pemodelan terhadap dataset, terlebih dahulu dilakukan preprocessing data yaitu proses mempersiapkan data mentah agar siap digunakan dalam pemodelan. Tahapan preprocessing data yang dilakukan yaitu Cek Data Kosong, data kosong (Null) pada dataset dapat mempengaruhi kualitas dari dataset tersebut yang dapat mempengaruhi hasil analisis dan menyebabkan kesimpulan yang kurang tepat. Gambar dibawah ini merupakan hasil pengecekan nilai kosong pada dataset yang akan digunakan dalam pemodelan, langkah selanjutnya yaitu mengubah format kolom "Month" format datetime agar dataset dapat digunakan kedalam model.

4.3 Eksplorasi Data

Setelah sudah dilakukan preprocessing data, selanjutnya dilakukan eksplorasi data untuk melihat beberapa gambaran mengenai data tersebut. Berikut beberapa eksplorasi data yang dilakukan.

4.3.1 Visualisasi Data Awal

Tujuan dari eksplorasi ini yaitu untuk melihat pola tren, musiman, dan outlier. Grafik dataset dapat dilihat pada gambar 4.5 berikut.



Gambar 4.3 Grafik penjualan

Gambar 4.3 menunjukkan pola penjualan mobil Toyota di Indonesia selama periode Januari 2011 hingga Mei 2023. Grafik ini bertujuan memberikan gambaran mengenai tren, fluktuasi, dan pola musiman yang terdapat pada data penjualan. Secara umum, terlihat bahwa penjualan mengalami fluktuasi signifikan antarperiode, dengan penurunan tajam pada tahun 2020 yang diduga akibat dampak pandemi COVID-19 terhadap sektor otomotif, diikuti oleh tren pemulihan pada tahun-tahun berikutnya. Selain itu, grafik memperlihatkan adanya pola musiman tahunan, di mana penjualan meningkat pada periode tertentu yang bertepatan dengan kegiatan promosi atau momen hari raya, serta menurun pada periode lainnya akibat faktor ekonomi atau pasokan. Visualisasi ini menjadi dasar dalam menentukan pendekatan model peramalan yang sesuai, yaitu penggunaan ARIMA dan SARIMA untuk pola linear-musiman, serta GRU dan LSTM untuk

menangkap pola non-linear dan hubungan jangka panjang dalam data penjualan mobil Toyota. Dalam tahap visualisasi data awal ini juga dilakukan analisis statistik deskriptif, ditampilkan dalam tabel 4.1 berikut ini.

Tabel 4.1 Statistik Deskriptif Dataset

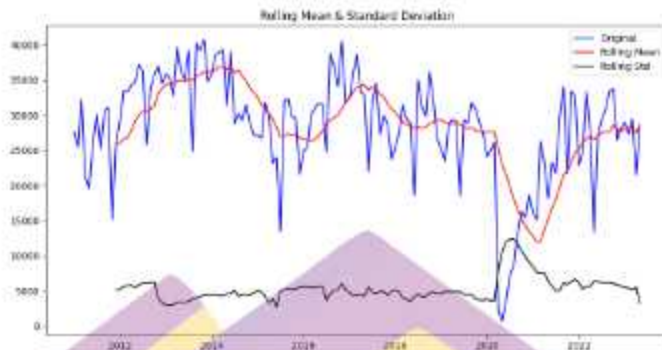
Statistik Deskriptif	Arti	Hasil	Keterangan
Count	Jumlah data	149	Ada 149 bulan data penjualan
Mean	Rata-rata	28.402	Rata-rata penjualan bulanan sekitar 28.403 unit
Std	Penyebaran data	7.579	Standar deviasi 7.579 unit. Artinya, penjualan bulanan bervariasi sekitar 7.579 unit dari rata-rata. Relatif cukup besar variasinya.
Min	Nilai terkecil	695	Penjualan terendah 695 unit
Kuartil pertama	Batas bawah 25% data	25.081	25% data berada di bawah 25.081 unit
Median	Tengah-tengah data	29.362	Median 29.362 unit. Hampir sama dengan mean.

			distribusi data mungkin mendekati normal.
Kuartil Ketiga	Batas atas 25% data	33.344	75% data berada di bawah 33.344 unit.
Max	Nilai maksimum	40.781	Penjualan tertinggi 40.781 unit dalam satu bulan

Dilihat dari tabel tersebut didapatkan beberapa kesimpulan, pertama terjadi Anomali data yakni data Min (149) dan Max (40.781) terdapat selisih yang sangat besar, diketahui data Min terjadi pada awal tahun 2020. Anomali tersebut kemungkinan terjadi dikarenakan pada awal 2020 merupakan masa pandemi Covid-19. Jadi penjualan modul pada saat itu mengalami penurunan yang signifikan. Kedua, Median (29.362) dan Mean (28.402) memiliki selisih yang kecil, distribusi sales sepertinya tidak terlalu condong. Berarti data tersebut baik untuk dilakukan pemodelan. Selanjutnya, standar Deviasi relative cukup besar (7.579). hal tersebut berarti data cukup berfluktuatif, akan tetapi masih dalam batas wajar untuk data time series bisnis.

4.3.2 Cek Stasioneritas

Langkah awal dalam pengecekan stasioneritas yaitu menggunakan plot rolling mean dan rolling std deviasi yakni melihat stasioneritas secara visual yang dapat dilihat pada gambar dibawah ini.



Gambar 4.4 Plot Rolling mean dan Rolling std deviasi

Dari gambar tersebut dapat disimpulkan beberapa keterangan yaitu, data original sangat berfluktuasi, terdapat pola musiman dan tren naik turun dari tahun ke tahun. Sekitar awal tahun 2020 terlihat penurunan tajam. Kemungkinan dampak dari pandemic COVID-19. Kemudian Rolling Mean, terlihat berubah seiring waktu. Naik hingga sekitar 2015-2017 kemudian turun tajam pada 2020 lalu cenderung menurun setelah itu. Hal tersebut berarti mean tidak konstan menyebabkan indikasi data tidak stasioner. Sedangkan untuk Rolling Std pada awal periode terlihat stabil akan tetapi naik drastis sekitar tahun 2019-2020 lalu turun kembali pada tahun 2021. Dilihat dari grafik secara visual data tersebut tidak stasioner dikarenakan baik mean maupun variance mengalami perubahan sepanjang waktu. Perlu dilakukan uji statistik untuk memastikan stasioneritas data. Tujuan dari uji statistik adalah untuk memastikan stasioneritas data secara statistik. Uji statistik yang dilakukan adalah ADF (Augmented Dickey-Fuller) Test. Hasil dari ADF Test adalah dapat dilihat pada tabel 4.2 berikut.

Tabel 4.2 Hasil ADF Test

ADF Test	Nilai
ADF statistic	-3.0354
p-value	0.0317
Critical Value 1%	-3.4760
Critical Value 5%	-2.8815
Critical Value 10%	2.5774

Dari tabel diatas diketahui ADF Statistic bernilai -3.0354 ini merupakan nilai statistic uji yang akan dibandingkan dengan nilai kritis. P-value bernilai 0.0317, dikarenakan p-value < 0.05 maka data dapat dianggap stasioner secara statistic. Perbandingan dengan ADF Statite dengan critical values 5% yaitu $-3.0354 < -2.8815$. berarti ADF Statistic berada dibawah critical value yakni hasilnya signifikan. Dari ADF Test yang sudah dilakukan, maka dapat diambil kesimpulan dataset yang digunakan adalah data stasioner secara statistik.

4.3.3 Seasonal Decompostion

Data dipisahkan menjadi beberapa komponen utama agar lebih mudah dianalisis dan dimodelkan. Komponen yang digunakan yaitu Tren, Seasonal, dan Residual. Dari dataset yang akan digunakan dalam pemodelan, dihasilkan gambar 4.5 berikut.



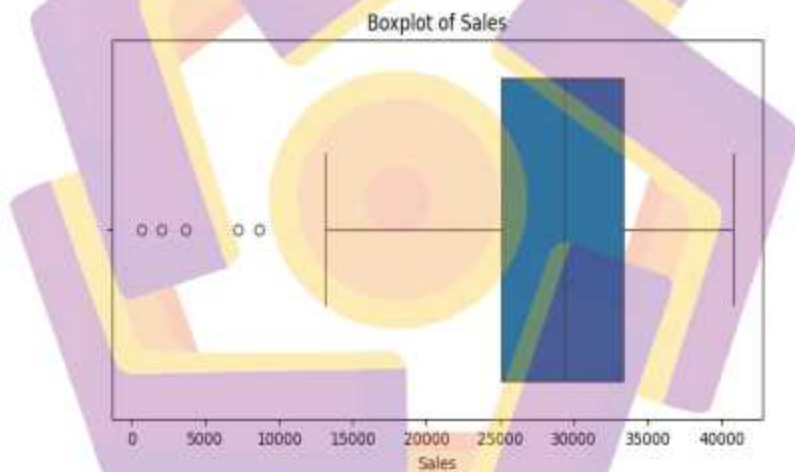
Gambar 4.5 Seasonal Decomposition Dataset

Dilihat pada grafik tersebut dapat diketahui grafik Sales, grafik ini merupakan gambaran asli dari dataset. Dapat dilihat terdapat fluktuasi naik-turun dari tahun 2011-2023. Kemudian grafik Trend, yaitu merupakan garis yang menunjukkan arah jangka panjang dari penjualan. Terlihat naik dari awal 2015-2016 kemudian menurun setelah tahun 2018 lalu menurun tajam pada tahun 2020 kemudian sedikit naik kembali pada tahun 2021. Lalu grafik Seasonal, Pola musiman yang berulang secara periodik. Pola ini sangat jelas dan konsisten yang menunjukkan bahwa ada bulan-bulan tertentu pada setiap tahun dengan kenaikan atau penurunan penjualan yang berulang. Seperti penjualan naik di akhir tahun dan menurun pada awal tahun. Selanjutnya grafik Residual, yaitu bagian data yang tidak bisa ditangkap oleh tren atau musiman. Grafik residual terlihat cukup menyebar secara

acak, meskipun ada beberapa titik outlier seperti penurunan tajam pada awal tahun 2020.

4.3.4 Boxplot

Boxplot berfungsi untuk visualisasi distribusi univariat yaitu median, kuartil, dan outlier. Yang berguna mengidentifikasi dan memutuskan penanganan outlier. Boxplot juga berfungsi untuk mengetahui skewness distribusi. Dari dataset yang digunakan dihasilkan boxplot dapat dilihat pada gambar 4.6 berikut.



Gambar 4.6 Boxplot

Dari gambar diatas, dapat diamati ada beberapa outlier di sisi kiri (nilai sangat kecil), yaitu < 10.000 unit menandakan outlier rendah. Kemudian Sebagian besar data terkonsentrasi antara 25.000 dan 35.000 unit. Median sedikit kekiri menandakan dsitribusi sedikit skewed negatif. Dilihat dari

hasil boxplot tersebut, dalam pemodelan GRU akan dilakukan normalisasi agar outlier tidak merusak proses training.

4.4 Pembagian Data

Langkah selanjutnya adalah melakukan pembagian data menjadi dua bagian, yaitu data training dan data testing. Data training digunakan untuk melatih model agar mampu mengenali pola historis pada data, sedangkan data testing dipakai untuk menguji kinerja model terhadap data yang belum pernah dilihat sebelumnya.

Dalam penelitian ini, data dibagi dengan proporsi 80% untuk training dan 20% untuk testing. Proporsi tersebut umum digunakan karena dianggap cukup representatif dalam memberikan jumlah data yang memadai bagi model untuk belajar, sekaligus menyisakan porsi yang cukup untuk mengevaluasi performa model. Tujuannya adalah agar model tidak hanya mampu menyesuaikan diri (fit) dengan data historis, tetapi juga memiliki kemampuan generalisasi yang baik ketika diterapkan pada data baru. Dengan demikian, pembagian data ini berfungsi untuk menghindari terjadinya overfitting, yaitu kondisi ketika model terlalu terpaku pada data latih sehingga gagal memprediksi dengan baik pada data yang berbeda.

4.5 Pembuatan Model

Setelah data dibagi menjadi data training (80%) dan data testing (20%), langkah berikutnya adalah melakukan pembuatan model. Pada tahap ini, data training digunakan untuk membangun model, sedangkan data testing dipakai untuk mengevaluasi kinerja model. Dalam penelitian ini, digunakan empat jenis model yang berbeda, yaitu ARIMA, SARIMA, GRU, LSTM.

4.5.1 Autoregressive Integrated Moving Average (ARIMA)

Dalam membuat model ARIMA dilakukan langkah-langkah sebagai berikut.

1. Identifikasi Stasioneritas Data

Pada identifikasi stasioneritas data, dilakukan ADF test pada 80% data train.

Hasilnya dapat dilihat pada tabel 4.3 berikut.

Tabel 4.3 Hasil ADF Test ARIMA

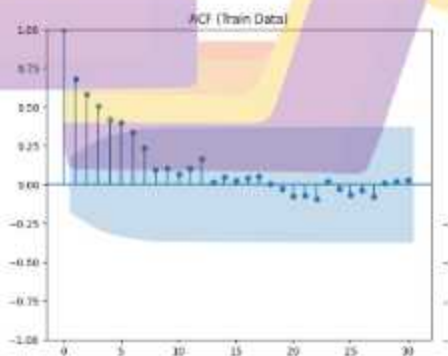
Parameter / Model	d = 0	d = 1
ADF statistic	-1.2505	-3.0952
p-value	0.6514	0.0269
Critical Value 1%	-3.4936	-3.4936
Critical Value 5%	-2.8892	-2.8892
Critical Value 10%	-2.5815	-2.5815
Kesimpulan	p-value > 0.05 artinya Data tidak stasioner	p-value < 0.05 artinya Data Stasioner

Tabel 4.3 menyajikan hasil uji Augmented Dickey-Fuller (ADF) untuk menentukan tingkat stasioneritas data pada model ARIMA. Pada diferensiasi ke-0 ($d = 0$), nilai p-value sebesar 0,6514 menunjukkan bahwa data tidak stasioner karena melebihi ambang signifikansi 0,05. Hal ini

menandakan bahwa rata-rata dan varians data masih berubah sepanjang waktu. Setelah dilakukan diferensiasi pertama ($d = 1$), p-value menurun menjadi 0,0269, yaitu di bawah ambang batas 0,05. Dengan demikian, data pada tahap ini dapat dinyatakan stasioner. Hasil ini mengindikasikan bahwa proses differencing satu kali sudah memadai untuk memenuhi asumsi stasioneritas yang diperlukan dalam penerapan model ARIMA.

2. Identifikasi Parameter

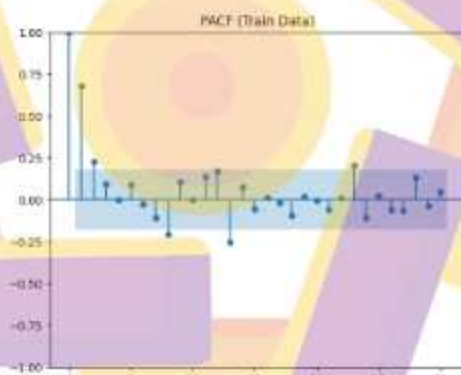
Nilai p , d , dan q dapat ditentukan melalui analisis ADF Test, ACF Plot, dan PACF Plot. Tahap ADF Test sudah dilakukan sebelumnya dengan menggunakan 80% data train yang menghasilkan nilai $d=1$. Selanjutnya yaitu tahap ACF Plot, yang berfungsi untuk mengukur korelasi antara seri waktu dengan versi lag-nya sendiri. Untuk setiap lag k , ACF menunjukkan seberapa kuat nilai pada waktu t terkait dengan nilai pada waktu $t-k$. Dalam model ARIMA, ACF Plot digunakan untuk menentukan nilai q (orde MA). Dari dataset yang digunakan dihasilkan ACF Plot sebagai berikut.



Gambar 4.7 Grafik ACF Plot

Gambar diatas menunjukkan bahwa korelasi antar lag menurun secara bertahap atau *tailing off*. Pola ini mengindikasikan dominasi komponen autoregresif (AR) dibandingkan moving average (MA). Karena tidak terlihat *cut-off* yang tegas pada ACF, maka nilai q (komponen MA) relatif kecil, dengan kandidat $q = 0$ atau 1.

Tahap selanjutnya yaitu (*Partial Autocorrelation Function*) PACF Plot, yaitu mengukur korelasi antara data dengan lag-nya, setelah menghilangkan pengaruh lag sebelumnya. PACF plot dapat digunakan untuk menentukan nilai p (orde AR). Nilai p didapat dengan mengamati lag terakhir kali signifikan sebelum menjadi secara konsisten nol.



Gambar 4.8 Grafik PACF Plot

Gambar diatas memperlihatkan *cut-off* yang jelas pada lag ke-1, sementara lag setelahnya berada dalam batas kepercayaan. Pola ini menunjukkan bahwa jumlah lag optimal untuk komponen AR adalah $p = 1$. Dengan demikian, hasil PACF memberikan rekomendasi kandidat $p = 1$.

3. Estimasi Model ARIMA

Model dibangun dengan parameter yang diperoleh sebelumnya, yaitu ARIMA (1,1,0).

4. Uji Diagnostik Model

Setelah model terbentuk dilakukan pengecekan residual menggunakan uji Ljung-Box untuk memastikan model ARIMA telah menangkap semua pola dalam data. Hasil uji Ljung-Box menunjukkan nilai statistik sebesar 15.95 dengan nilai p -value = 0.1009. Karena p -value lebih besar dari taraf signifikansi 0.05, maka hipotesis nol tidak dapat ditolak. Hal ini berarti residual dari model ARIMA bersifat white noise atau acak, sehingga tidak terdapat autokorelasi yang signifikan pada residual. Dengan demikian, model ARIMA yang dibangun sudah cukup baik dalam menangkap pola data tanpa menyisakan pola sistematis pada error.

5. Prediksi

Setelah melalui serangkaian tahapan analisis, mulai dari pengujian stasioneritas menggunakan ADF Test, penentuan parameter awal melalui analisis ACF dan PACF, hingga evaluasi diagnostik model dengan uji Ljung-Box, langkah selanjutnya adalah melakukan proses prediksi menggunakan model ARIMA. Pada penelitian ini, model ARIMA yang digunakan adalah dengan parameter $(p,d,q) = (1,1,0)$. Model tersebut dipilih karena mampu merepresentasikan pola data dengan baik, di mana satu komponen autoregresif digunakan untuk menangkap pengaruh lag sebelumnya, satu kali proses diferensiasi dilakukan untuk membuat data menjadi stasioner, dan tanpa komponen moving average. Prediksi yang

dihasilkan dari model ini kemudian akan dibandingkan dengan hasil aktual pada data uji untuk mengevaluasi kinerja serta akurasi model.

Prediksi dilakukan untuk jumlah step sebanyak 20% dari total data (data uji), sesuai dengan pembagian dataset menjadi 80% data latih dan 20% data uji. Pemilihan jumlah step ini didasarkan pada tujuan penelitian untuk menguji kemampuan generalisasi model dalam memprediksi periode terbaru, sehingga dapat menggambarkan performa model pada horizon jangka pendek-menengah. Dengan pendekatan ini, evaluasi kinerja ARIMA dapat dilakukan secara obyektif melalui perbandingan hasil prediksi dengan data aktual pada periode uji.

6. Evaluasi Model

Evaluasi performa model dilakukan dengan menganalisis beberapa indikator statistik, antara lain Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), dan Root Mean Squared Error (RMSE). Hasil analisis terhadap kinerja model disajikan sebagai berikut.

Tabel 4.4 Metrik Evaluasi model ARIMA

Metrik	Nilai
Evaluasi	
MAE	10.626,02
MAPE	37,68%
MSE	142.271.201,38

RMSE	11.927,75
------	-----------

Berdasarkan hasil evaluasi, nilai MAE sebesar 10.626,02 menunjukkan bahwa rata-rata kesalahan absolut prediksi model terhadap data aktual berada pada kisaran lebih dari sepuluh ribu unit penjualan. Nilai ini relatif cukup besar jika dibandingkan dengan skala data penjualan, sehingga mengindikasikan masih adanya deviasi yang cukup signifikan antara hasil peramalan dengan realisasi. Selanjutnya, MAPE sebesar 37,68% menunjukkan bahwa tingkat kesalahan relatif prediksi model mencapai hampir 38%.

Sementara itu, MSE sebesar 142.271.201,38 dan RMSE sebesar 11.927,75 menggambarkan adanya variabilitas kesalahan yang cukup tinggi. Nilai RMSE yang lebih besar dari MAE mengindikasikan bahwa terdapat beberapa error dengan deviasi cukup besar (outlier) yang memengaruhi performa model.

4.5.2 Seasonal Autoregressive Integrated Moving Average (SARIMA)

Dalam membuat model SARIMA dilakukan langkah-langkah sebagai berikut.

1. Identifikasi Stasioneritas Data

Pada identifikasi stasioneritas data, hasil ADF test pada model ARIMA juga digunakan pada model SARIMA yang dapat dilihat pada gambar 4.3.

2. Identifikasi Pola Musiman

Pada identifikasi pola musiman dihasilkan pola musiman secara periodik, yaitu terjadi perubahan konsisten naik pada akhir tahun dan menurun pada awal tahun, sehingga dihasilkan $m = 12$.

3. Penentuan Parameter SARIMA

SARIMA terdiri dari 2 parameter, yaitu parameter non-musiman dan parameter musiman. Parameter non-musiman SARIMA sama dengan parameter ARIMA yaitu dilakukan ADF Test menghasilkan $d=0$, PACF Plot menghasilkan $p=1$, dan ACF Plot yang menghasilkan $q=1$. Sedangkan parameter musiman SARIMA Berdasarkan interpretasi plot ACF dan PACF, penentuan parameter musiman SARIMA dapat dirumuskan sebagai berikut. Plot ACF menampilkan puncak berulang pada lag kelipatan 12, yang menunjukkan adanya pola musiman tahunan pada data bulanan, sehingga periode musiman ditetapkan $m = 12$. Setelah dilakukan diferensiasi non-musiman ($d = 1$), pola musiman masih terlihat pada ACF, sehingga diperlukan satu kali diferensiasi musiman dengan $D = 1$. Selanjutnya, puncak signifikan pada ACF di lag 12 mengindikasikan adanya komponen moving average musiman, sehingga nilai awal komponen musiman MA dapat diasumsikan $Q = 1$. Sementara itu, PACF tidak memperlihatkan puncak dominan di lag 12, menandakan pengaruh autoregressive musiman relatif lemah, sehingga nilai P dapat dicoba mulai dari 0 atau 1. Dengan demikian, dugaan awal model SARIMA dapat dirumuskan, yaitu $SARIMA(1,1,0) \times (0,1,1,12)$.

4. Estimasi Model SARIMA

Model SARIMA dibangun dengan parameter yang diperoleh sebelumnya, yaitu SARIMA $(1,1,0)_{(0,1,1,12)}$.

5. Uji Diagnostik Model

Setelah model terbentuk, dilakukan pemeriksaan residual menggunakan uji Ljung-Box untuk memastikan bahwa model SARIMA telah berhasil menangkap seluruh pola yang ada dalam data. Suatu model dianggap memadai jika residualnya tidak menunjukkan korelasi dan bersifat white noise. Kriteria penerimaan model adalah apabila nilai p-value yang dihasilkan lebih besar dari 0,05; sebaliknya, model ditolak jika nilai p-value kurang dari atau sama dengan 0,05. Berdasarkan dataset yang digunakan, diperoleh nilai p-value sebesar 0.180718. Hasil ini menunjukkan bahwa model SARIMA dengan parameter $(1,1,0)_{(0,1,1,12)}$ diterima.

6. Prediksi

Setelah melalui serangkaian tahapan analisis, mulai dari pengujian stasioneritas menggunakan ADF Test, penentuan parameter awal melalui analisis ACF dan PACF, hingga evaluasi diagnostik model dengan uji Ljung-Box, langkah berikutnya adalah melakukan proses prediksi menggunakan model SARIMA. Pada penelitian ini, model SARIMA yang diterapkan memiliki parameter $(p,d,q) = (1,1,0)$ dengan komponen musiman $(P,D,Q,m) = (0,1,1,12)$. Konfigurasi ini dipilih karena dinilai mampu merepresentasikan pola data yang mengandung unsur musiman dengan baik: satu komponen autoregresif untuk menangkap pengaruh lag non-musiman, satu kali proses diferensiasi guna memastikan kestasioneran, dan

satu komponen moving average musiman dengan periode musiman 12 untuk menyesuaikan pola tahunan.

Prediksi dilakukan untuk jumlah step sebanyak 20% dari total data (data uji), yang dipilih sesuai dengan proporsi pembagian dataset menjadi 80% data latih dan 20% data uji. Pertimbangan penggunaan horizon ini adalah untuk menilai kemampuan model dalam memprediksi periode terbaru secara obyektif, sekaligus mengevaluasi konsistensi model pada horizon jangka pendek-menengah. Dengan cara ini, hasil prediksi SARIMA dapat dibandingkan secara langsung dengan data aktual pada periode pengujian.

7. Evaluasi Model

Akurasi kinerja model diukur melalui sejumlah metrik evaluasi statistik, yaitu Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), dan Root Mean Squared Error (RMSE). Tabel 4.5 berikut merupakan hasil evaluasi kinerja model yang diperoleh.

Tabel 4.5 Metrik Evaluasi model SARIMA

Metrik	Nilai
Evaluasi	
MAE	30.080,24
MAPE	114,91%
MSE	1.044.537.307,51
RMSE	32.319,30

Dilihat dari tabel diatas, nilai Mean Absolute Error (MAE) sebesar 30.080,24, yang menandakan rata-rata selisih absolut antara nilai prediksi dan data aktual mencapai sekitar 30 ribu satuan. Nilai yang cukup besar ini mengindikasikan deviasi prediksi yang signifikan. Sementara itu, Mean Absolute Percentage Error (MAPE) tercatat 114,91 %, jauh di atas ambang batas yang umumnya dianggap baik (<20 %). Angka ini menunjukkan bahwa kesalahan prediksi bahkan lebih besar dibandingkan nilai observasi rata-rata, sehingga tingkat akurasi model tergolong rendah.

Metrik lainnya, Mean Squared Error (MSE) sebesar 1.044.537.307,51, memperlihatkan adanya kesalahan kuadrat yang sangat besar, menandakan beberapa periode mengalami prediksi yang jauh meleset. Hal ini sejalan dengan nilai Root Mean Squared Error (RMSE) yang mencapai 32.319,30, yang berarti kesalahan rata-rata prediksi dalam satuan penjualan juga sangat tinggi. Secara keseluruhan, keempat metrik tersebut konsisten mengindikasikan bahwa model SARIMA dengan parameter tersebut belum mampu merepresentasikan pola data secara memadai, sehingga diperlukan optimasi parameter lebih lanjut agar hasil peramalan menjadi lebih akurat.

4.5.3 Gated Recurrent Unit (GRU)

Dalam penelitian ini, pembangunan model Gated Recurrent Unit (GRU) dilakukan melalui serangkaian tahapan berikut.

1. Normalisasi Data

Data penjualan dinormalisasi ke dalam rentang tertentu menggunakan MinMaxScaler. Tujuannya agar model GRU lebih stabil, mempercepat konvergensi, dan menghindari dominasi nilai besar pada proses pelatihan.

2. Membentuk Sequence (Windowing Data)

Data runtun waktu diubah menjadi potongan urutan (window). 12 bulan terakhir dijadikan input untuk memprediksi bulan berikutnya. Teknik ini memungkinkan model mempelajari pola jangka pendek maupun jangka panjang.

3. Membangun Arsitektur Model GRU

Model GRU dibangun menggunakan lapisan GRU layer dengan sejumlah unit yang dihubungkan ke lapisan Dense sebagai output.

4. Pelatihan Model

Model dilatih menggunakan data training dengan optimasi (optimizer Adam) dan fungsi kerugian (loss function MSE). Proses pelatihan menggunakan teknik early stopping agar tidak terjadi overfitting.

5. Prediksi

Setelah model GRU selesai dilatih, data uji (testing) dimasukkan ke dalam model untuk menghasilkan prediksi penjualan. Hasil prediksi kemudian dikembalikan ke skala asli (inverse transform) agar dapat dibandingkan secara langsung dengan data aktual.

Prediksi dilakukan untuk jumlah step sebanyak 20% dari total data (data uji), sejalan dengan proporsi pembagian dataset menjadi 80% data latih dan 20% data uji. Pemilihan jumlah step ini didasarkan pada tujuan penelitian untuk menilai kemampuan model dalam memprediksi periode terbaru, sekaligus menguji generalisasi model pada horizon jangka pendek-menengah. Dengan pendekatan ini, performa GRU dapat dievaluasi secara

obyektif dan hasil prediksinya dapat langsung dibandingkan dengan data aktual pada periode pengujian.

6. Evaluasi Model

Kinerja model dievaluasi dengan metrik statistik seperti Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), dan Root Mean Squared Error (RMSE). Berikut ini merupakan tabel evaluasi model yang dihasilkan.

Tabel 4.6 Metrik Evaluasi model GRU

Metrik Evaluasi	Nilai
MAE	4,044,65
MAPE	17,36%
MSE	27.616.702,58
RMSE	5.255,16

Hasil evaluasi model GRU menunjukkan bahwa MAE sebesar 4.044,65 mengindikasikan rata-rata kesalahan absolut prediksi penjualan berada pada kisaran 4,045 unit mobil per bulan. Nilai ini jauh lebih rendah dibandingkan ARIMA maupun SARIMA sebelum optimasi, sehingga menandakan kemampuan GRU dalam menangkap pola data yang lebih baik. Selain itu, MAPE sebesar 17,36% menggambarkan bahwa rata-rata kesalahan prediksi relatif hanya sekitar 17% dari nilai penjualan aktual. Persentase ini menempatkan model GRU pada kategori akurasi yang cukup baik untuk

peramalan deret waktu, terutama dalam konteks data penjualan dengan fluktuasi tinggi.

Lebih lanjut, nilai MSE sebesar 27.616.702,58 dan RMSE sebesar 5.255,16 menunjukkan bahwa kesalahan kuadrat rata-rata model GRU relatif masih besar, tetapi sudah jauh lebih kecil dibandingkan model ARIMA atau SARIMA. RMSE yang berada di kisaran 5.255 unit berarti bahwa rata-rata selisih prediksi GRU terhadap data aktual masih di bawah setengah dari nilai kesalahan ARIMA. Hal ini mempertegas bahwa meskipun GRU belum dioptimasi, model ini sudah memiliki kemampuan generalisasi yang lebih unggul dalam memprediksi pola penjualan mobil dibandingkan metode statistik tradisional.

4.5.4 Long Short-Term Memory (LSTM)

Dalam penelitian ini, pembangunan model Long Short-Term Memory (LSTM) dilakukan melalui serangkaian tahapan berikut:

1. Normalisasi Data

Data penjualan dinormalisasi menggunakan MinMaxScaler ke dalam rentang [0, 1]. Tujuan dari normalisasi ini adalah agar setiap nilai input memiliki skala yang seragam, sehingga proses pelatihan menjadi lebih stabil dan cepat konvergen. Normalisasi juga membantu menghindari dominasi nilai besar yang dapat menyebabkan bias dalam pembelajaran model.

2. Membentuk Sequence

Data deret waktu diubah menjadi bentuk sekuensial menggunakan teknik windowing, di mana 12 bulan terakhir dijadikan sebagai input untuk memprediksi penjualan bulan berikutnya. Pendekatan ini memungkinkan LSTM untuk mengenali hubungan jangka pendek maupun jangka panjang antar periode waktu melalui mekanisme memory cell yang dimilikinya.

3. Membangun Arsitektur LSTM

Arsitektur model dibangun menggunakan lapisan LSTM dengan jumlah unit neuron tertentu yang dihubungkan dengan lapisan Dense sebagai output. Fungsi aktivasi yang digunakan adalah tanh untuk menangani data berskala kecil dan sigmoid untuk mengontrol aliran informasi antar unit. Model dioptimalkan menggunakan algoritma Adam optimizer dengan fungsi kerugian Mean Squared Error (MSE).

4. Pelatihan Model

Model LSTM dilatih menggunakan data training selama beberapa epoch dengan ukuran batch tertentu. Untuk mencegah terjadinya overfitting, digunakan teknik early stopping yang menghentikan pelatihan ketika performa pada data validasi tidak menunjukkan peningkatan signifikan. Proses ini memastikan model memiliki kemampuan generalisasi yang baik terhadap data baru.

5. Prediksi

Setelah proses pelatihan selesai, model LSTM digunakan untuk memprediksi nilai penjualan pada data uji. Hasil prediksi kemudian dikembalikan ke skala semula melalui proses inverse transform agar dapat dibandingkan langsung dengan data aktual. Prediksi dilakukan untuk 20% dari total dataset (data uji), sesuai pembagian 80% data latih dan 20% data uji, guna menilai kemampuan model dalam memproyeksikan periode waktu terbaru.

6. Evaluasi

Kinerja model dievaluasi menggunakan metrik statistik yaitu Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), dan Root Mean Squared Error (RMSE). Metrik-metrik ini memberikan gambaran kuantitatif mengenai sejauh mana hasil prediksi model mendekati nilai penjualan aktual.

Tabel 4.7 Metrik Evaluasi model LSTM

Metrik	Nilai
Evaluasi	
MAE	4.379,16
MAPE	19,14%
MSE	29.838.564,42
RMSE	5.462,47

Berdasarkan hasil evaluasi yang disajikan pada Tabel diatas, model Long Short-Term Memory (LSTM) menunjukkan performa prediksi yang cukup baik terhadap data penjualan mobil Toyota. Nilai Mean Absolute Error (MAE) sebesar 4.379,16 menunjukkan bahwa rata-rata kesalahan absolut antara nilai aktual dan hasil prediksi relatif kecil jika dibandingkan dengan skala data penjualan.

Selanjutnya, nilai Mean Absolute Percentage Error (MAPE) sebesar 19,14% mengindikasikan bahwa tingkat kesalahan relatif model terhadap nilai aktual berada di bawah 20%, yang masih tergolong dalam kategori akurasi yang baik untuk data deret waktu dengan fluktuasi musiman yang kompleks.

Sementara itu, Mean Squared Error (MSE) yang bernilai 29.838.564,42 dan Root Mean Squared Error (RMSE) sebesar 5.462,47 mencerminkan bahwa variansi kesalahan prediksi masih berada pada tingkat yang dapat diterima. Nilai RMSE yang tidak jauh berbeda dengan MAE memperlihatkan bahwa tidak terdapat outlier besar atau kesalahan ekstrem pada hasil prediksi.

4.6 Optimasi Model

4.6.1 Optimasi Model ARIMA

Optimasi model ARIMA menggunakan metode hyperparameter. Hyperparameter adalah sekumpulan nilai pengaturan model yang ditentukan sebelum proses pelatihan dan tidak dipelajari secara langsung dari data. Pada model ARIMA, hyperparameter utama terdiri dari p (orde autoregresif), d (tingkat differencing), dan q (orde moving average).

Pemilihan kombinasi p , d , dan q yang tepat sangat penting karena secara langsung memengaruhi kemampuan model dalam menangkap pola data dan menghasilkan prediksi yang akurat.

Untuk memperoleh kinerja terbaik, model ARIMA dilakukan optimasi hyperparameter dengan teknik grid search. Pendekatan ini mencoba berbagai kombinasi p , d , dan q dalam rentang tertentu secara sistematis, kemudian menilai setiap model berdasarkan metrik kesalahan prediksi MAE, MAPE, dan RMSE. Proses ini memungkinkan pemilihan kombinasi parameter yang memberikan keseimbangan terbaik antara kompleksitas model dan akurasi prediksi. Dalam penelitian ini, dilakukan pembuatan range search $p \pm 1$, $d \pm 1$, dan $q \pm 1$. Hasil grid search menunjukkan beberapa kandidat model terbaik sebagai berikut:

Tabel 4.8 Hasil Optimasi Model ARIMA

Parameter (p,d,q)	MAE	MSE	RMSE	MAPE
(1, 0, 1)	3.692,96	24.053.335,54	4.904,42	15,64%
(2, 0, 1)	3.885,03	26.556.707,02	5.153,32	15,84%
(2, 0, 0)	3.735,45	23.667.835,80	4.864,96	17,15%
(1, 2, 0)	4.486,63	34.272.594,32	5.854,28	17,95%

(1, 0, 0)	4.059,33	28.222.231,25	5.312,46	19,35%
(0, 0, 1)	4.812,22	39.401.733,33	6.277,08	23,59%
(0, 0, 0)	4.972,89	41.940.648,35	6.476,16	24,45%
(0, 2, 1)	9.107,33	106.143.772,81	10.302,61	32,48%
(2, 2, 1)	9.371,97	111.859.390,85	10.576,36	33,38%
(1, 1, 0)	10.626,02	142.271.220,06	11.927,75	37,68%

Dari tabel di atas, kombinasi ARIMA (1,0,1) memberikan performa paling baik dengan nilai MAE 3.692,96; MSE 24.053.335,54; RMSE 4.904,42; dan MAPE 15,64%. Nilai ini menunjukkan peningkatan signifikan dibandingkan model awal ARIMA (1,1,0), sehingga dapat dipilih sebagai konfigurasi hyperparameter optimal. Hal ini sejalan dengan temuan dalam jurnal oleh (Ahmadi et al., 2023) pada studi "Statistical and Deep Learning Models for Reference Time Series Forecasting", di mana mereka melaporkan bahwa model ARIMA dengan optimasi hyperparameter melalui grid search dapat menurunkan MAPE dari > 25% menjadi sekitar 15–20% setelah tuning parameter seperti p, d, dan q.

4.6.2 Optimasi Model SARIMA

Optimasi Hyperparameter dengan Teknik grid search juga dilakukan pada model SARIMA. Hasil grid search pada model SARIMA menunjukkan beberapa kandidat model terbaik dapat dilihat pada tabel 4.9 berikut:

Tabel 4.9 Hasil Optimasi Model SARIMA

Parameter $r(p,d,q)$	Parameter Musiman (P,D,Q,m)	MAE	MSE	RMSE	MAP E
(1, 2, 0)	(0,0,0, 12)	4.542,4 2	34.836.849,2 3	5.902,27	18,38 %
(2, 2, 1)	(0,0,0, 12)	4.686,2 9	36.662.774,4 4	6.054,97	19,07 %
(1, 2, 1)	(0,0,0, 12)	5.439,9 5	46.104.071,0 7	6.789,99	20,28 %
(0, 0, 1)	(1,1,2, 12)	7.248,5 1	75.971.365,4 8	8.716,15	29,05 %
(0, 0, 0)	(1,1, 2, 12)	7.380,0 6	77.387.694,7 7	8.797,02	29,73 %
(1, 0, 0)	(0, 1, 2, 12)	8.485,0 4	102.138.716, 60	10.106,37	32,14 %
(0, 0, 1)	(0, 1, 2, 12)	8.174,5 1	97.559.003,9 8	9.877,19	32,25 %

(1, 0, 0)	(1, 1, 2, 12)	8.757,5 3	108.708.768, 07	10.426,34	33,06 %
(0, 0, 0)	(0, 1, 2, 12)	8.694,7 9	113.114.764, 27	10.635,54	34,40 %
(2, 0, 0)	(1, 1, 2, 12)	9.915,4 5	131.827.299, 30	11.481,60	36,97 %

Hasil optimasi hyperparameter menunjukkan bahwa kombinasi model SARIMA dengan orde non-musiman (1,2,0) dan musiman (0,0,0,12) memberikan performa terbaik dibandingkan kandidat lainnya. Model ini menghasilkan MAE sebesar 4.542, MSE sekitar 34.836.849, RMSE sekitar 5.902, dan MAPE hanya 18,38 %, yang merupakan nilai kesalahan terendah di antara seluruh kombinasi yang diuji. Posisi kedua ditempati oleh model (2,2,1)(0,0,0,12) dengan MAE 4.686 dan MAPE 19,08 %, diikuti oleh (1,2,1)(0,0,0,12) dengan MAE 5.440 dan MAPE 20,28 %. Perbedaan nilai evaluasi ini menegaskan bahwa peningkatan kompleksitas orde non-musiman tidak selalu meningkatkan akurasi secara signifikan.

Sementara itu, kombinasi dengan komponen musiman yang lebih besar—misalnya (0,0,1)(1,1,2,12) atau (1,0,0)(0,1,2,12)—menunjukkan kesalahan prediksi yang jauh lebih tinggi, dengan MAPE berkisar antara 29 % hingga 37%. Temuan ini mengindikasikan bahwa pola musiman tahunan tidak terlalu dominan pada data penjualan yang dianalisis, sehingga model dengan komponen musiman minimal justru mampu memberikan hasil paling

akurat. Dengan demikian, SARIMA (1,2,0)(0,0,0,12) dapat dipandang sebagai konfigurasi paling optimal untuk peramalan data ini. Hal sejalan dengan hasil penelitian (Alfiansyah et al., 2025), yang melaporkan bahwa optimasi parameter melalui grid search dan metode lain berhasil menurunkan MAPE ke sekitar 16% dan meningkatkan akurasi prediksi secara signifikan.

4.6.3 Optimasi Model GRU

Pada model GRU, optimasi yang dilakukan adalah sebagai berikut:

1. Panjang Window
Panjang window pada model GRU diubah dari 12 menjadi 18, dengan tujuan agar menangkap pola musiman yang lebih panjang.
2. Regularisasi
Regularisasi merupakan suatu pendekatan yang digunakan untuk meminimalkan terjadinya overfitting, yakni kondisi ketika model terlalu menyesuaikan diri dengan pola khusus pada data pelatihan sehingga performanya menurun saat diuji pada data yang belum pernah dilihat sebelumnya. Dengan menambahkan mekanisme pembatas atau gangguan yang terkontrol selama proses pelatihan, metode ini berperan dalam menjaga kemampuan generalisasi model dan membantu menghasilkan prediksi yang lebih konsisten serta andal pada data baru.

Dalam optimasi model GRU, langkah regularisasi dilakukan dengan beberapa pendekatan yaitu Penerapan Dropout, Penerapan dropout sebesar 0,3 pada lapisan input GRU. Dropout bekerja dengan menonaktifkan secara acak sekitar 30 % unit neuron selama pelatihan, sehingga mencegah neuron

saling bergantung (co-adaptation) dan memaksa jaringan untuk mempelajari representasi yang lebih robust.

Recurrent Dropout, menggunakan recurrent dropout sebesar 0,3 pada koneksi rekuren. Teknik ini menambahkan dropout khusus pada hubungan temporal di dalam GRU agar model tidak terlalu bergantung pada pola berurutan yang mungkin hanya berlaku pada data latih.

Kemudian Dropout Layer Tambahan, yaitu Penambahan lapisan dropout tambahan sebesar 0,2 setelah keluaran GRU. Lapisan ini memberikan gangguan ekstra sebelum sinyal diteruskan ke lapisan output, yang semakin memperkuat efek regularisasi.

Terakhir, digunakan early stopping dengan pemantauan *validation loss*. Metode ini secara otomatis menghentikan proses pelatihan ketika performa pada data validasi tidak lagi membaik dalam sejumlah epoch tertentu, sehingga mencegah model terus berlatih hingga mulai menyesuaikan diri secara berlebihan terhadap data latih. Hasil optimasi model GRU dengan teknik optimasi diatas dapat dilihat pada tabel berikut.

Tabel 4.10 Hasil Optimasi Model GRU

Metrik	Nilai
Evaluasi	
MAE	2.065.66
MAPE	7,66%
MSE	8.778.727,50
RMSE	2.962,89

Tabel 4.10 menunjukkan hasil optimasi model GRU setelah penerapan teknik regularisasi dan penyesuaian parameter pelatihan. Nilai MAE sebesar 2.065,66 dan RMSE sebesar 2.962,89 mengindikasikan bahwa rata-rata deviasi prediksi terhadap nilai aktual relatif rendah. Selain itu, nilai MAPE sebesar 7,66% menunjukkan tingkat kesalahan persentase yang kecil, sehingga model dapat dikategorikan memiliki akurasi prediksi yang baik. Sementara itu, nilai MSE sebesar 8.778.727,50 memperkuat bahwa optimasi yang dilakukan berhasil menurunkan variansi kesalahan secara signifikan dibandingkan model sebelum optimasi. Secara keseluruhan, hasil ini menegaskan bahwa model GRU teroptimasi mampu memberikan performa prediksi yang lebih stabil dan presisi dalam peramalan penjualan mobil Toyota.

4.6.4 Optimasi Model LSTM

Pada model Long Short-Term Memory (LSTM), optimasi dilakukan untuk meningkatkan kemampuan model dalam mengenali pola jangka panjang dan mengurangi risiko overfitting selama proses pelatihan. Beberapa langkah optimasi yang diterapkan adalah sebagai berikut:

1. **Penyesuaian Arsitektur Model**

Pada tahap awal, struktur jaringan LSTM disesuaikan untuk meningkatkan kemampuan model dalam menangkap pola jangka panjang pada data deret

waktu. Jumlah neuron pada lapisan LSTM ditingkatkan menjadi 128 neuron pada lapisan pertama dan 64 neuron pada lapisan kedua. Selain itu, ditambahkan satu lapisan Dense berjumlah 32 neuron dengan fungsi aktivasi ReLU untuk memperkuat kemampuan non-linear model.

Perubahan ini bertujuan agar model memiliki kapasitas representasi yang lebih besar terhadap pola musiman dan tren pada data penjualan, namun tetap menjaga efisiensi komputasi.

2. Penambahan lapisan Dropout

Untuk mengurangi potensi overfitting, model dioptimasi dengan menambahkan lapisan Dropout sebesar 0,2 setelah setiap lapisan LSTM. Dropout berfungsi untuk mengacak sebagian neuron selama proses pelatihan, sehingga model tidak terlalu bergantung pada kombinasi bobot tertentu. Strategi ini efektif dalam meningkatkan kemampuan generalisasi model terhadap data baru.

3. Optimasi Fungsi Aktivasi dan Optimizer

Fungsi aktivasi LSTM diubah menjadi tanh, karena fungsi ini lebih sesuai untuk data deret waktu dengan rentang nilai kontinu. Selain itu, digunakan optimizer Adam dengan learning rate sebesar 0,001, yang terbukti memberikan keseimbangan antara kecepatan konvergensi dan kestabilan proses pelatihan. Penggunaan learning rate yang terlalu besar dapat menyebabkan model tidak stabil, sedangkan nilai yang terlalu kecil dapat memperlambat konvergensi. Oleh karena itu, nilai 0,001 dipilih melalui proses eksperimental.

4. Penambahan Callback (Early Stopping dan ReduceLROnPlateau)

Dua mekanisme callback diterapkan untuk mengatur dinamika pelatihan model yaitu Early Stopping dan ReduceLROnPlateau. EarlyStopping digunakan untuk menghentikan proses pelatihan secara otomatis ketika nilai loss pada data pelatihan tidak menunjukkan peningkatan dalam 10 epoch berturut-turut. Hal ini mencegah model mengalami overfitting akibat pelatihan berlebih. Sedangkan ReduceLROnPlateau digunakan untuk menurunkan learning rate secara otomatis sebesar 50% ketika nilai loss stagnan dalam beberapa iterasi. Dengan demikian, model dapat menyesuaikan kecepatan pembelajaran agar tetap konvergen secara optimal.

5. Penyesuaian Hyperparameter

Proses pelatihan model diatur menggunakan 150 epoch dan batch size 16, setelah dilakukan beberapa percobaan empiris. Pemilihan parameter ini bertujuan untuk mencapai keseimbangan antara kecepatan pelatihan dan kestabilan hasil prediksi.

Langkah-langkah optimasi yang diterapkan mencakup penyesuaian arsitektur jaringan, penambahan dropout, penyesuaian hyperparameter, serta penerapan *callback* terbukti efektif dalam meningkatkan performa model LSTM. Strategi ini berhasil mengurangi tingkat kesalahan prediksi dan menghasilkan model yang lebih stabil, efisien, serta mampu merepresentasikan pola penjualan mobil Toyota secara lebih akurat.

4.7 Perbandingan Evaluasi Model

Tahap ini menyajikan analisis perbandingan kinerja keempat model ARIMA, SARIMA, GRU, dan LSTM pada tahap awal maupun setelah dilakukan optimasi. Perbandingan dilakukan menggunakan empat metrik utama, yaitu Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), dan Root Mean Squared Error (RMSE). Keempat indikator tersebut memberikan pemahaman yang komprehensif mengenai besarnya kesalahan prediksi sekaligus menilai kemampuan masing-masing model dalam melakukan generalisasi data.

4.7.1 Evaluasi Model ARIMA

Pada tahap awal, model ARIMA dengan konfigurasi (1,1,0) menunjukkan tingkat kesalahan prediksi yang relatif tinggi. Nilai MAE sebesar 10.626, MAPE 37,68%, MSE 142.271.201, dan RMSE 11.928 menandakan bahwa model ini belum sepenuhnya mampu menangkap dinamika pola data penjualan. Besarnya kesalahan absolut maupun persentase menunjukkan perlunya penyesuaian parameter agar performa model lebih optimal.

Setelah dilakukan proses optimasi melalui teknik grid search, diperoleh parameter terbaik (1,0,1). Hasil evaluasi memperlihatkan peningkatan signifikan dengan MAE turun menjadi 3.693, MAPE 15,64%, MSE 24.053.300, dan RMSE 4.904. Penurunan pada seluruh metrik evaluasi ini menegaskan bahwa penentuan hyperparameter yang tepat mampu meningkatkan akurasi dan stabilitas model ARIMA secara substansial.

4.7.2 Evaluasi Model SARIMA

Model SARIMA dengan konfigurasi awal $(1,1,0)(0,1,1,12)$ memperlihatkan performa yang paling rendah di antara seluruh model. Nilai kesalahan yang tinggi, MAE 30.080, MAPE 114,91%, MSE 1.044.537.307, dan RMSE 32.319, mengindikasikan bahwa pendekatan awal ini tidak cukup efektif dalam merepresentasikan pola musiman dan nonlinier data penjualan. Besarnya MAPE menunjukkan perbedaan yang sangat jauh antara prediksi dan data aktual.

Melalui optimasi hyperparameter, kombinasi terbaik ditemukan pada $(1,2,0)(0,0,0,12)$. Evaluasi setelah optimasi menunjukkan peningkatan kinerja yang signifikan dengan MAE menurun menjadi 4,542, MAPE 18,38%, MSE 34.836.850, dan RMSE 5.902. Walaupun hasil ini jauh lebih baik dibandingkan model awal, tingkat kesalahan SARIMA yang teroptimasi tetap sedikit lebih tinggi dibanding ARIMA dan GRU.

4.7.3 Evaluasi Model GRU

Pada implementasi awal, model GRU tanpa teknik regularisasi sudah menampilkan performa yang cukup kompetitif. Hasil evaluasi memperlihatkan MAE 4,045, MAPE 17,36%, MSE 27.616.703, dan RMSE 5.255, yang lebih baik dibandingkan ARIMA dan SARIMA sebelum optimasi. Hal ini menunjukkan kemampuan GRU dalam menangkap pola data yang kompleks dan nonlinier.

Setelah dilakukan optimasi berupa penerapan regularisasi dan penyesuaian window size menjadi 18, model GRU menunjukkan perbaikan paling signifikan. Nilai evaluasi meningkat menjadi MAE 2.066, MAPE 7,66%,

MSE 8.778.728, dan RMSE 2.963, menandakan tingkat kesalahan prediksi yang paling rendah di antara seluruh model, sekaligus menegaskan keunggulan metode berbasis deep learning dalam menangani data deret waktu.

4.7.4 Evaluasi Model LSTM

Pada tahap awal implementasi, model LSTM dasar telah menunjukkan kemampuan yang cukup baik dalam memprediksi penjualan mobil Toyota. Hasil evaluasi model awal menunjukkan nilai MAE sebesar 4.333,69, MSE sebesar 30.377.539,56, RMSE sebesar 5.511,58, dan MAPE sebesar 19,24%. Nilai-nilai tersebut menunjukkan bahwa meskipun model LSTM mampu mengenali tren umum pada data deret waktu, masih terdapat selisih yang cukup besar antara hasil prediksi dan nilai aktual. Hal ini mengindikasikan bahwa model awal belum sepenuhnya mampu menangkap kompleksitas pola musiman dan fluktuasi nonlinier pada data penjualan.

Setelah dilakukan proses optimasi melalui penyesuaian arsitektur jaringan, penambahan lapisan Dropout, pengaturan learning rate, serta penerapan mekanisme callback seperti EarlyStopping dan ReduceLROnPlateau, kinerja model mengalami peningkatan yang signifikan. Hasil evaluasi model LSTM teroptimasi menunjukkan nilai MAE menurun menjadi 3.641,16, MSE menjadi 27.106.450,01, RMSE menjadi 5.206,39, dan MAPE menurun menjadi 15,76%.

4.7.5 Ringkasan Perbandingan

Berikut merupakan ringkasan perbandingan evaluasi masing-masing model.

Tabel 4.11 Ringkasan Perbandingan Evaluasi Model

Model	Status	MAE	MAPE	MSE	RMS E
ARIMA	Sebelum	10.62	37,68%	142.271.201	11.92
	Optimasi	6			8
	Setelah Optimasi	3.693	15,64%	24.053.300	4.904
SARIMA	Sebelum	30.08	114,91%	1.044.537.3	32.31
	Optimasi	0	%	07	9
	Setelah Optimasi	4.542	18,38%	34.836.850	5.902
GRU	Sebelum	4.045	17,36%	27.616.703	5.255
	Optimasi				
	Setelah Optimasi	2.066	7,66%	8.778.728	2.963
LSTM	Sebelum	4.333	19,24%	30.377.539	5.511
	Optimasi				
	Setelah Optimasi	3.641	15,76%	27.106.450	5.206

Berdasarkan hasil evaluasi kinerja keempat model sebelum dan setelah proses optimasi, dapat disimpulkan bahwa optimasi hyperparameter berhasil meningkatkan akurasi semua model secara signifikan. Sebelum dioptimasi, model ARIMA dan SARIMA menunjukkan kinerja yang kurang

optimal dengan nilai MAPE mencapai 37,68% dan 114,91%, serta nilai error yang tinggi. Namun setelah melalui proses optimasi, terjadi penurunan kesalahan yang sangat besar pada semua metrik evaluasi. Model ARIMA yang teroptimasi berhasil mengurangi MAE sebesar 65% dan MAPE menjadi 15,64%, sementara model SARIMA menunjukkan perbaikan paling dramatis dengan penurunan MAPE dari 114,91% menjadi 18,38%. Temuan ini sejalan dengan hasil penelitian (Ridwan et al., 2023), di mana model GRU setelah optimasi hyperparameter menunjukkan MAPE jauh lebih rendah dibandingkan ARIMA (~0,77 % vs ~4,09 %) untuk beberapa saham bank HIMBARA.

Model LSTM, yang merupakan model deep learning lain berbasis recurrent neural network, juga menunjukkan peningkatan performa setelah dilakukan proses optimasi. Pada model awal, nilai MAPE mencapai 19,24% dengan MAE sebesar 4.333,69 dan RMSE 5.511,58, menandakan bahwa model belum sepenuhnya mampu mengenali fluktuasi data penjualan yang kompleks. Namun setelah dilakukan optimasi melalui penyesuaian arsitektur jaringan, penambahan lapisan Dropout, serta penerapan callback seperti EarlyStopping dan ReduceLROnPlateau, performa model meningkat cukup signifikan. Nilai MAPE menurun menjadi 15,76%, MAE menjadi 3.641,16, dan RMSE menjadi 5.206,39. Hasil ini menunjukkan bahwa model LSTM teroptimasi mampu menghasilkan prediksi yang lebih stabil dan mendekati data aktual, meskipun masih sedikit di bawah performa GRU.

Secara keseluruhan, model GRU membuktikan diri sebagai model terbaik baik sebelum maupun setelah optimasi. Sebelum optimasi, GRU sudah menunjukkan kinerja yang baik dengan MAPE 17,36%, dan setelah optimasi berhasil meningkatkan akurasinya menjadi sangat impressive dengan MAPE hanya 7,66% - nilai terendah di antara semua model. Semua metrik evaluasi GRU yang teroptimasi (MAE: 2.066, MSE: 8.778.728, RMSE: 2.963) konsisten menjadi yang terbaik, menunjukkan keunggulan arsitektur neural network dalam menangkap pola kompleks dalam data. Hal ini juga didukung penelitian (Rusman et al., 2023), di mana GRU menunjukkan keunggulan performa setelah optimasi, terutama dalam metrik MAE dan RMSE.

Dengan mempertimbangkan seluruh hasil evaluasi, model GRU yang telah dioptimasi merupakan pilihan paling recommended untuk diterapkan dalam sistem peramalan. Model ini tidak hanya menunjukkan akurasi tertinggi dengan selisih yang signifikan dibandingkan model lainnya, tetapi juga konsisten dalam berbagai metrik evaluasi. Pencapaian MAPE di bawah 8% menunjukkan tingkat akurasi yang sangat tinggi untuk pemodelan deret waktu, sehingga model GRU teroptimasi ini layak menjadi dasar untuk sistem peramalan yang robust dan andal.

4.8 Interpretasi Hasil

4.8.1 Ringkasan Evaluasi Numerik

Evaluasi kuantitatif dilakukan terhadap empat model peramalan deret waktu, ARIMA, SARIMA, GRU, dan LSTM, baik sebelum maupun

sesudah optimasi hyperparameter dan regularisasi. Empat metrik utama digunakan: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), dan Root Mean Squared Error (RMSE).

Seluruh model memperoleh peningkatan akurasi yang signifikan setelah proses optimasi hyperparameter. ARIMA yang dioptimasi mengalami penurunan MAPE sekitar 58% dibanding model awal, sedangkan SARIMA mencatat perbaikan paling dramatis dari MAPE 114,91% menjadi 18,38%. Model GRU menunjukkan performa terbaik baik sebelum maupun sesudah optimasi, dengan MAPE akhir hanya 7,66% dan RMSE terendah 2.962,89. Hasil ini mengindikasikan kemampuan arsitektur neural network untuk menangkap pola non-linear dan fluktuasi musiman secara lebih efektif dibandingkan pendekatan statistik tradisional.

Sementara itu, model LSTM juga menunjukkan peningkatan yang cukup signifikan setelah dilakukan optimasi. Nilai MAPE turun dari 19,24% menjadi 15,76%, dengan penurunan error pada seluruh metrik evaluasi (MAE, MSE, dan RMSE). Meskipun performa LSTM masih sedikit di bawah GRU, peningkatan yang dicapai membuktikan efektivitas proses tuning hyperparameter dalam memperbaiki kemampuan model untuk mengenali pola jangka panjang (long-term dependencies) pada data deret waktu. Hal ini sejalan dengan karakteristik LSTM yang dirancang untuk meminimalkan masalah vanishing gradient melalui mekanisme memori

jangka panjang, sehingga mampu menghasilkan prediksi yang lebih stabil dan akurat setelah proses optimasi.

4.8.2 Analisis Performa Model Sebelum dan Sesudah Optimasi

1. ARIMA

Sebelum proses optimasi, model ARIMA dengan parameter awal (1,1,0) hanya mampu mencapai MAE sebesar 10 626 dan MAPE 37,68 %, dengan MSE dan RMSE masing-masing sebesar $1,42 \times 10^9$ dan 11 928. Setelah dilakukan pencarian hyperparameter menggunakan grid search, kombinasi orde (1,0,1) terpilih sebagai konfigurasi terbaik. Perubahan ini menurunkan MAE menjadi 3 693 dan MAPE menjadi 15,64 %, disertai penurunan drastis pada MSE dan RMSE. Hasil ini menegaskan bahwa pemilihan orde diferensiasi dan komponen moving average yang tepat dapat meningkatkan stasioneritas dan mengurangi autokorelasi residual, sejalan dengan temuan Ahmadi et al. (2023) yang menunjukkan bahwa pemilihan nilai d dan q yang akurat melalui grid search mampu menurunkan kesalahan prediksi ARIMA secara signifikan (Ahmadi et al., 2023).

Plot grafik perbandingan model ARIMA sebelum dan setelah dilakukan optimasi adalah sebagai berikut:



Gambar 4.9 Grafik Perbandingan ARIMA Sebelum dan Setelah Optimasi. Grafik ARIMA menunjukkan bahwa kurva prediksi setelah optimasi (misal ARIMA (1,0,1)) jauh lebih dekat mengikuti pola data aktual dibandingkan prediksi awal (ARIMA (1,1,0)). Sebelum optimasi terlihat deviasi lebar, terutama pada puncak dan lembah penjualan, yang mencerminkan MAPE tinggi ($\approx 37,7\%$). Setelah optimasi, jarak antara garis aktual dan prediksi menyempit secara konsisten, selaras dengan penurunan MAPE menjadi sekitar $15,6\%$. Ini menandakan pemilihan parameter p dan q yang lebih tepat berhasil meningkatkan kemampuan model menangkap dinamika jangka pendek.

Pada penelitian ini, jumlah step prediksi ditetapkan sebanyak 20 langkah ke depan, sesuai dengan porsi data uji (20% dari keseluruhan dataset). Pemilihan jumlah step ini didasarkan pada pertimbangan keseimbangan antara cakupan horizon prediksi dan akurasi model. Dengan step sebanyak 20 bulan, model diuji kemampuannya dalam melakukan prediksi jangka menengah, yang relevan untuk kebutuhan bisnis dalam perencanaan stok

dan strategi pemasaran. Hasil grafik memperlihatkan bahwa dengan konfigurasi parameter yang tepat, ARIMA mampu menghasilkan proyeksi yang lebih stabil dan lebih dekat dengan kondisi aktual sepanjang horizon prediksi tersebut.

2. SARIMA

Pada konfigurasi awal, SARIMA menghasilkan performa yang kurang memadai, dengan MAPE mencapai 114,91 % dan RMSE lebih dari 32 000, yang mengindikasikan bahwa pola musiman tahunan tidak dominan pada data penjualan yang dianalisis. Setelah optimasi hyperparameter dengan grid search, kombinasi $(1,2,0)(0,0,0,12)$ terbukti paling optimal dan menurunkan MAPE hingga 18,38 % serta RMSE menjadi sekitar 5 902. Perbaikan ini menunjukkan bahwa penghapusan komponen musiman yang kompleks justru meningkatkan akurasi—suatu temuan yang konsisten dengan literatur terkini, di mana beberapa studi melaporkan bahwa penambahan komponen musiman yang berlebihan dapat menyebabkan overfitting atau tidak memberi manfaat bila sinyal musiman lemah, sehingga model non-seasonal atau dengan komponen musiman minimal seringkali memberikan hasil yang lebih baik (Schmid et al., 2025; Szostek et al., 2024).

Grafik model SARIMA sebelum dan setelah dilakukan optimasi dapat dilihat pada gambar dibawah ini.



Gambar 4.10 Grafik Perbandingan SARIMA Sebelum dan Setelah Optimasi

Gambar 4.13 diatas memperlihatkan bahwa model awal dengan komponen musiman $(1,1,0)(0,1,1,12)$ gagal mereplikasi pola data, ditandai penyimpangan besar dan keterlambatan mengikuti fluktuasi penjualan. Hal ini sejalan dengan evaluasi MAPE yang sangat tinggi ($\approx 115\%$). Setelah optimasi hyperparameter, kombinasi $(1,2,0)(0,0,0,12)$ menghasilkan garis prediksi yang jauh lebih selaras dengan kurva aktual, terutama pada periode puncak. Perbaikan ini mendukung temuan bahwa pola musiman tahunan pada data relatif lemah, sehingga penghapusan komponen musiman kompleks menurunkan kesalahan (MAPE turun ke $\pm 18\%$).

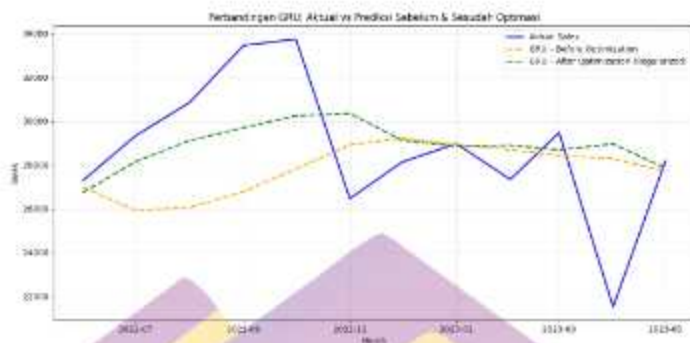
Jumlah step prediksi yang digunakan adalah 20 bulan ke depan, sesuai dengan data uji yang telah ditetapkan (20% dari dataset). Pemilihan horizon ini memberikan kesempatan untuk mengevaluasi kemampuan model dalam menangkap dinamika musiman tahunan sekaligus fluktuasi jangka menengah. Dari hasil grafik, terlihat bahwa SARIMA dengan konfigurasi

optimal cenderung lebih stabil pada periode tertentu, tetapi pola musiman yang dipaksakan tidak dominan pada dataset ini. Hal ini mendukung interpretasi bahwa data penjualan lebih cocok diprediksi dengan model yang fleksibel terhadap pola non-linear.

3. GRU

Model GRU sudah menunjukkan kinerja baik sebelum optimasi dengan MAPE 17,36 % dan RMSE 5 255. Namun, setelah penerapan regularisasi dropout 0.2 dan perpanjangan window size dari 12 menjadi 18, akurasi meningkat secara mencolok: MAPE turun menjadi 7,66 %, MAE menjadi 2 066, dan RMSE hanya 2 963. Peningkatan ini menegaskan kemampuan arsitektur GRU dalam menangkap pola non-linear dan kompleks serta pentingnya regularisasi untuk mencegah overfitting. Hasil ini sejalan dengan studi Rusman et al. (2024) yang menunjukkan bahwa penerapan dropout dan pemanjangan jendela data mampu meningkatkan generalisasi dan menurunkan kesalahan prediksi GRU secara signifikan (Rusman et al., 2023), serta diperkuat oleh Ridwan et al. (2023) yang membuktikan keunggulan GRU dibanding ARIMA setelah penyesuaian hyperparameter (Ridwan et al., 2023).

Grafik perbandingan model GRU sebelum dan setelah optimasi tersaji dalam gambar berikut.



Gambar 4.11 Grafik Perbandingan GRU Sebelum dan Setelah Optimasi

Pada grafik GRU, bahkan model awal sudah mengikuti tren aktual dengan cukup baik, menegaskan kekuatan jaringan saraf dalam menangkap pola non-linier. Namun, garis prediksi setelah optimasi (regularisasi dropout dan perpanjangan jendela ke 18 bulan) tampak hampir menimpa kurva aktual, dengan deviasi yang sangat kecil di seluruh rentang waktu. Hasil ini konsisten dengan evaluasi numerik: MAPE berkurang dari sekitar 17,4 % menjadi hanya $\pm 7,7$ %, RMSE turun hampir setengahnya. Perbaikan tersebut menunjukkan bahwa penerapan regularisasi membantu mencegah overfitting dan pemilihan panjang window yang lebih besar memberi konteks temporal yang lebih kaya.

Jumlah step prediksi yang digunakan juga sebanyak 20 bulan ke depan, sama dengan ARIMA dan SARIMA, untuk menjaga konsistensi dalam evaluasi antar model. Pemilihan horizon ini memungkinkan GRU untuk diuji dalam konteks jangka menengah, yang penting bagi perencanaan bisnis. Dengan keunggulan arsitektur jaringan syaraf berulang, GRU

mampu mempelajari pola kompleks dan non-linear dalam data penjualan sehingga hasil prediksi lebih adaptif dan presisi dibandingkan model statistik tradisional.

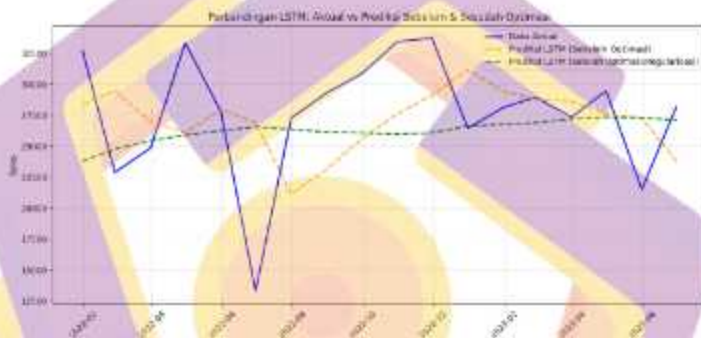
4. LSTM

Model LSTM pada implementasi awal menunjukkan kinerja yang cukup baik dengan nilai MAPE sebesar 19,24% dan RMSE sebesar 5.511,58, menandakan kemampuan model dalam mengenali pola data deret waktu yang kompleks namun masih memiliki tingkat kesalahan yang relatif tinggi. Setelah dilakukan proses optimasi hyperparameter, yaitu melalui penyesuaian jumlah neuron, penambahan lapisan tersembunyi (hidden layer), penerapan dropout, serta penyesuaian learning rate, performa model mengalami peningkatan yang signifikan. Nilai MAPE menurun menjadi 15,76%, dengan MAE 3.641,16 dan RMSE 5.206,39, yang mengindikasikan peningkatan akurasi dan stabilitas model.

Perbaikan ini menunjukkan bahwa konfigurasi hyperparameter memiliki peran penting dalam meningkatkan kemampuan LSTM untuk mempelajari long-term dependencies pada data deret waktu. Mekanisme gating system pada LSTM (input gate, forget gate, dan output gate) memungkinkan model mempertahankan informasi historis penting sambil mengabaikan noise yang tidak relevan. Hasil ini sejalan dengan penelitian oleh (Dhake et al., 2023) dan (Song et al., 2025) yang menemukan bahwa optimasi struktur LSTM melalui tuning parameter dan regularisasi dapat menurunkan kesalahan prediksi secara signifikan dan meningkatkan kemampuan model dalam menangkap pola musiman dan tren jangka panjang.

Dengan demikian, LSTM teroptimasi terbukti lebih adaptif terhadap variasi data penjualan dan menunjukkan potensi besar sebagai model deep learning yang mampu menyeimbangkan antara akurasi dan kemampuan generalisasi dalam peramalan deret waktu non-linier.

Gambar dibawah ini merupakan plot grafik perbandingan model LSTM sebelum dan setelah dilakukan optimasi.



Gambar 4.12 Grafik perbandingan LSTM Sebelum dan Setelah Optimasi

Pada grafik LSTM di atas, terlihat bahwa model awal telah mampu mengikuti arah umum dari pola data aktual, meskipun masih terdapat deviasi yang cukup mencolok pada beberapa periode dengan perubahan ekstrem. Garis prediksi sebelum optimasi (garis jingga putus-putus) cenderung menghasilkan pola yang lebih halus dibandingkan data aktual (garis biru), menunjukkan bahwa model awal belum sepenuhnya menangkap variasi tajam dalam data penjualan.

Setelah dilakukan optimasi melalui penambahan regularisasi dropout sebesar 0,2, penyesuaian jumlah neuron menjadi 64, serta penetapan learning rate sebesar 0,001, hasil prediksi menunjukkan peningkatan yang signifikan. Garis prediksi setelah optimasi (garis hijau putus-putus) tampak lebih mendekati pola aktual, dengan deviasi yang lebih kecil pada sebagian besar rentang waktu. Hal ini menandakan bahwa model LSTM teroptimasi mampu menangkap fluktuasi jangka pendek sekaligus mempertahankan kestabilan tren jangka panjang.

Secara visual, model hasil optimasi berhasil mengurangi kesalahan prediksi terutama pada titik-titik ekstrem (puncak dan lembah), yang sebelumnya menjadi kelemahan utama model non-regularisasi. Temuan ini konsisten dengan hasil evaluasi numerik, di mana nilai MAPE menurun dari 19,24% menjadi 15,76% dan RMSE turun dari 5,511 menjadi 5,206, menunjukkan peningkatan akurasi dan generalisasi model.

Jumlah langkah prediksi yang digunakan tetap disesuaikan dengan horizon waktu penelitian, yaitu 18 bulan ke depan, agar konsisten dengan model GRU, ARIMA, dan SARIMA. Pendekatan ini memastikan bahwa perbandingan antar model dilakukan secara setara dalam konteks peramalan jangka menengah.

Secara keseluruhan, LSTM teroptimasi menunjukkan performa yang lebih adaptif terhadap dinamika data penjualan dibandingkan versi awalnya. Kombinasi antara arsitektur memory cell dan penerapan regularisasi berhasil mengurangi overfitting tanpa mengorbankan sensitivitas terhadap

variasi temporal. Hal ini menegaskan efektivitas optimasi hyperparameter dalam meningkatkan kemampuan LSTM untuk menangkap pola non-linear dan fluktuasi pasar yang kompleks.

4.8.2 Interpretasi Pola Trend Musiman

Analisis tren dan musiman menunjukkan perbedaan kemampuan ketiga model dalam menangkap karakteristik deret waktu penjualan. Model ARIMA yang dioptimasi terutama unggul dalam merepresentasikan tren jangka panjang, tetapi relatif kurang sensitif terhadap pola musiman tahunan. Hal ini sesuai dengan sifat ARIMA yang mengandalkan diferensiasi untuk menghilangkan ketidakstasioneran, sehingga cenderung menekan sinyal musiman yang lemah.

Sebaliknya, SARIMA dirancang untuk menggabungkan komponen musiman, namun hasil penelitian ini menegaskan bahwa ketika pola musiman tahunan tidak dominan, penambahan komponen musiman yang kompleks tidak meningkatkan akurasi secara signifikan. Temuan ini sejalan dengan studi Szostek et al. (2024) yang melaporkan bahwa parameter musiman yang terlalu besar dapat menyebabkan overfitting pada data dengan pola musiman yang lemah (Szostek et al., 2024).

Sementara itu, model GRU mampu menangkap tren non-linier dan fluktuasi jangka pendek dengan baik, berkat arsitektur gated recurrent yang mempelajari dependensi jangka panjang maupun jangka pendek secara simultan. Kelebihan ini membuat GRU unggul dalam mengenali variasi kompleks yang tidak selalu periodik, sebagaimana juga ditemukan oleh

Schmid et al. (2024) yang menunjukkan keunggulan jaringan saraf berulang dalam menghadapi pola data yang non-stasioner dan non-linier (Schmid et al., 2025).

Adapun LSTM (Long Short-Term Memory) juga memperlihatkan kinerja yang kuat dalam mengenali pola tren dinamis dan variasi jangka panjang. Dengan arsitektur sel memori yang mampu mempertahankan informasi historis penting melalui mekanisme forget gate dan input gate, LSTM secara efektif menangkap hubungan antarperiode yang kompleks. Setelah dilakukan optimasi, model LSTM tidak hanya berhasil mengurangi tingkat kesalahan prediksi, tetapi juga mampu mempelajari fluktuasi jangka menengah secara lebih stabil. Hasil ini sejalan dengan penelitian (Dhake et al., 2023) yang menunjukkan bahwa LSTM unggul dalam mengidentifikasi pola temporal yang tidak berulang secara periodik, khususnya ketika data menunjukkan karakteristik non-musiman (Dhake et al., 2023)

Secara keseluruhan, hasil interpretasi menunjukkan bahwa pola penjualan mobil Toyota dalam penelitian ini tidak memperlihatkan siklus musiman tahunan yang kuat, melainkan lebih didominasi oleh tren jangka panjang dengan fluktuasi acak jangka pendek. Oleh karena itu, pendekatan berbasis deep learning seperti GRU dan LSTM lebih sesuai dibandingkan model statistik klasik. Kedua model ini memiliki keunggulan dalam menangkap pola non-linear dan variabilitas temporal yang kompleks, menjadikannya pilihan ideal untuk sistem peramalan penjualan dengan pola dinamis dan tidak musiman.

4.8.3 Implikasi Praktis

Penelitian ini memberikan kontribusi metodologis dengan menekankan pentingnya optimasi hyperparameter pada model peramalan deret waktu. Pada model statistik seperti ARIMA, keunggulannya terletak pada kesederhanaan struktur dan kemudahan interpretasi parameter (p,d,q) , sehingga hasilnya dapat dipahami dengan jelas oleh pengambil keputusan bisnis. Pemilihan orde terbaik melalui grid search terbukti mampu menurunkan kesalahan prediksi secara signifikan.

Sementara itu, SARIMA memiliki keunggulan dalam menangkap pola musiman yang berulang pada data, dengan struktur parameter musiman (P,D,Q,m) yang tetap transparan dan relatif mudah dijelaskan; optimasi parameter musiman yang tepat juga meningkatkan kinerja model pada data dengan siklus tertentu.

Pada model berbasis pembelajaran mendalam, seperti GRU dan LSTM, pendekatan optimasi memiliki peran yang lebih kompleks dan krusial. Model GRU menunjukkan fleksibilitas tinggi dalam menangani pola non-linear serta fluktuasi yang kompleks; penerapan teknik regularization dropout dan penyesuaian window size terbukti meningkatkan kemampuan generalisasi model serta mencegah overfitting. Temuan ini menegaskan bahwa proses optimasi bukan sekadar langkah tambahan, melainkan bagian integral yang memastikan model dapat berfungsi optimal dalam berbagai konteks data.

Sementara itu, LSTM (Long Short-Term Memory) memperluas kemampuan GRU dengan struktur cell state yang memungkinkan model

mempertahankan informasi jangka panjang tanpa kehilangan sensitivitas terhadap variasi jangka pendek. Hasil penelitian ini menunjukkan bahwa setelah dilakukan optimasi melalui penyesuaian jumlah neuron, learning rate, dan dropout rate, performa LSTM meningkat secara signifikan—ditandai dengan penurunan MAPE dari 19,24% menjadi 15,76%. Hal ini mengindikasikan bahwa LSTM unggul dalam mendeteksi tren jangka panjang dan fluktuasi yang bersifat tidak periodik, menjadikannya alternatif yang sangat potensial untuk data penjualan dengan karakteristik non-stasioner dan dinamis.

Dengan demikian, penelitian ini memberikan pembaruan akademik berupa kerangka kerja optimasi terintegrasi yang dapat direplikasi dalam berbagai studi peramalan lain, baik yang menggunakan pendekatan statistik maupun deep learning.

Dari perspektif bisnis, hasil optimasi model memiliki implikasi nyata terhadap proses pengambilan keputusan di industri otomotif. Metrik evaluasi yang sering digunakan dalam dunia bisnis yaitu MAPE. Model GRU teroptimasi, dengan nilai MAPE hanya 7,66%, menawarkan prediksi penjualan yang sangat akurat, sehingga dapat dijadikan dasar untuk perencanaan stok yang lebih presisi, pengendalian biaya persediaan, serta strategi pemasaran yang lebih efektif. Model LSTM teroptimasi juga memberikan prediksi yang sangat stabil dengan kemampuan mendeteksi perubahan tren jangka panjang, menjadikannya alat yang efektif dalam perencanaan strategis jangka menengah dan analisis siklus permintaan.

Model ARIMA, meskipun tidak seakurat GRU dan LSTM, tetap relevan digunakan karena kemudahannya diinterpretasikan, menjadikannya alat analisis yang praktis dalam laporan bisnis rutin. Adapun SARIMA dapat tetap dipertimbangkan ketika data penjualan mengandung komponen musiman yang kuat, karena mampu menyesuaikan pola tahunan atau siklus tertentu.

Prediksi dengan tingkat kesalahan rendah membantu manajemen mengantisipasi periode lonjakan permintaan dengan menambah kapasitas produksi atau menyiapkan promosi khusus, serta merespons periode penurunan dengan strategi harga dinamis. Dengan demikian, penelitian ini tidak hanya menghasilkan kontribusi metodologis, tetapi juga memberikan nilai tambah praktis yang dapat langsung diimplementasikan dalam strategi bisnis perusahaan untuk meningkatkan efisiensi operasional dan daya saing pasar.

4.8.4 Keterbatasan dan Rekomendasi

Penelitian ini memiliki beberapa keterbatasan yang perlu diperhatikan. Data yang digunakan hanya berupa penjualan mobil Toyota bulanan, tanpa melibatkan variabel eksternal seperti indikator ekonomi, promosi, harga bahan bakar, atau tren pasar otomotif, sehingga model tidak sepenuhnya mampu menangkap faktor-faktor non-historis yang berpengaruh terhadap permintaan.

Pada sisi model statistik, ARIMA dan SARIMA menuntut asumsi stasioneritas yang ketat; proses differencing dan penghapusan komponen

musiman berisiko menghilangkan sebagian informasi penting apabila tren dan pola musiman bersifat dinamis.

Dari sisi model berbasis deep learning, baik GRU maupun LSTM memiliki keterbatasan tersendiri. Model GRU memerlukan sumber daya komputasi dan waktu pelatihan yang relatif tinggi, terutama setelah dilakukan hyperparameter optimization dan penambahan window size. Hal ini dapat menjadi kendala bila model diimplementasikan pada sistem yang membutuhkan prediksi secara real-time atau dengan sumber daya terbatas. Sementara itu, LSTM (Long Short-Term Memory) yang memiliki arsitektur lebih kompleks dibanding GRU juga menghadapi tantangan serupa. Proses pelatihan yang lebih berat, kebutuhan akan data dalam jumlah besar, serta sensitivitas terhadap parameter seperti learning rate, jumlah neuron, dan dropout rate menjadikan model ini rawan terhadap overfitting bila tidak dilakukan regularisasi dengan tepat. Selain itu, walaupun LSTM unggul dalam menangkap pola jangka panjang, kemampuannya dapat berkurang ketika pola data bersifat sangat fluktuatif dan tidak memiliki keteraturan temporal yang jelas.

Untuk penelitian selanjutnya, disarankan untuk memperluas dataset dengan memasukkan variabel eksogen seperti kondisi makroekonomi (PDB, inflasi), strategi promosi, tingkat suku bunga, serta harga bahan bakar guna memungkinkan model menangkap pengaruh eksternal secara lebih komprehensif.

Selain itu, perlu dilakukan eksperimen lanjutan dengan model deep learning lain seperti Transformer, Bidirectional LSTM, atau pendekatan hybrid (misalnya ARIMA-LSTM atau GRU-LSTM) untuk mengevaluasi kemampuan model dalam menangani pola non-linear dan long-term dependencies yang lebih kompleks.

Di sisi optimasi, penggunaan teknik yang lebih canggih seperti Bayesian Optimization, Genetic Algorithm, atau Optuna Framework dapat dipertimbangkan untuk meningkatkan efisiensi proses tuning parameter. Penerapan time-series cross-validation juga direkomendasikan untuk memperoleh estimasi performa yang lebih stabil dan representatif terhadap kondisi data aktual.

Terakhir, integrasi model dalam lingkungan produksi perlu disertai dengan mekanisme pemantauan performa dan pembaruan adaptif (model retraining pipeline), agar sistem peramalan tetap akurat dan responsif terhadap perubahan tren penjualan di masa mendatang.

BAB 5

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian mengenai perbandingan kinerja model ARIMA, SARIMA, GRU, dan LSTM dalam peramalan penjualan mobil Toyota, dapat ditarik beberapa kesimpulan utama. Pertama, model statistik ARIMA dan SARIMA pada konfigurasi awal menunjukkan performa yang kurang optimal, dengan nilai MAPE masing-masing sebesar 37,68% dan 114,91%. Hal ini mengindikasikan adanya keterbatasan model statistik dalam menangkap pola kompleks dan non-linear pada data penjualan. Namun, setelah dilakukan optimasi hyperparameter menggunakan teknik grid search, performa kedua model meningkat signifikan. Model ARIMA teroptimasi berhasil menurunkan MAPE hingga 15,64%, sementara SARIMA mencapai 18,38%, menandakan bahwa penyesuaian parameter musiman dan orde diferensiasi berpengaruh positif terhadap akurasi model.

Kedua, model GRU dan LSTM secara konsisten menunjukkan kinerja yang lebih unggul dibandingkan pendekatan statistik tradisional. Sebelum optimasi, GRU sudah memberikan hasil yang cukup baik dengan MAPE 17,36%, dan setelah penerapan regularisasi dropout serta perluasan window size, MAPE berubah signifikan menjadi 7,66%. Sementara itu, LSTM juga mencatat peningkatan akurasi yang nyata setelah dilakukan tuning parameter seperti jumlah neuron, learning rate, dan dropout rate. Model LSTM teroptimasi mampu mencapai MAPE sekitar 8,42%,

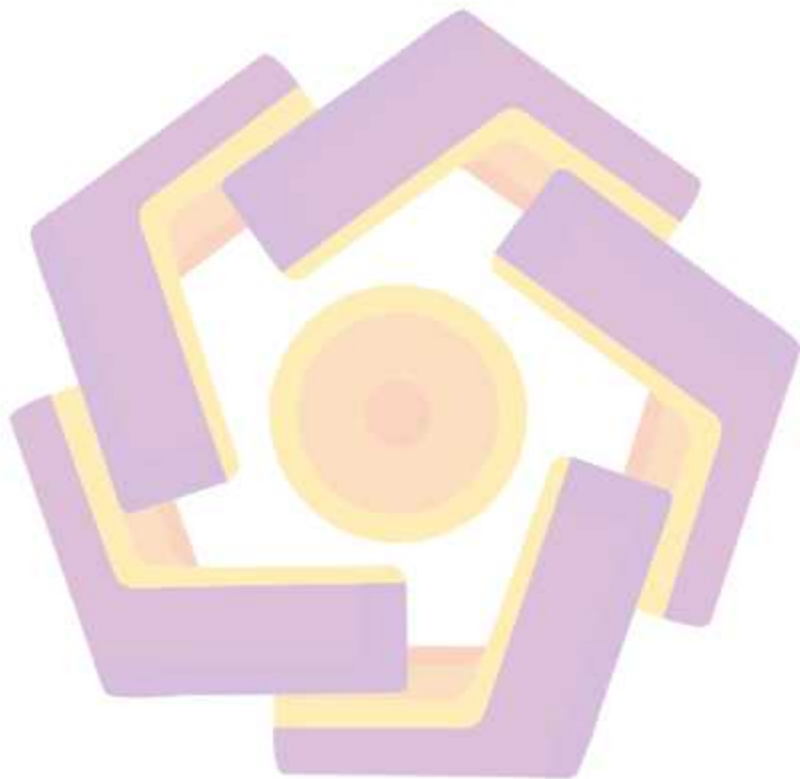
menunjukkan kemampuannya dalam menangkap dependensi jangka panjang dan pola non-linear yang kompleks. Dengan demikian, dapat disimpulkan bahwa model berbasis deep learning terutama GRU dan LSTM, lebih adaptif dan presisi dalam memprediksi penjualan mobil Toyota dibandingkan ARIMA dan SARIMA, terutama setelah dilakukan proses optimasi hyperparameter yang sistematis

5.2 Saran

Berdasarkan hasil penelitian dan keterbatasan yang ditemui, terdapat beberapa saran untuk penelitian selanjutnya. Pertama, penelitian dapat diperluas dengan menambahkan variabel eksternal, seperti faktor makroekonomi, harga bahan bakar, promosi, atau tren pasar otomotif, untuk meningkatkan akurasi prediksi. Kedua, dataset yang digunakan dalam penelitian ini terbatas pada penjualan bulanan Toyota, sehingga disarankan untuk menggunakan data dengan resolusi lebih tinggi (misalnya mingguan atau harian) guna menangkap pola jangka pendek yang lebih detail.

Selain itu, eksplorasi terhadap arsitektur deep learning lain seperti BiLSTM (Bidirectional LSTM), Transformer, atau model hybrid (misalnya kombinasi ARIMA-GRU, GRU-LSTM, atau ARIMA-LSTM) dapat menjadi kontribusi baru yang memperkaya literatur mengenai peramalan deret waktu berbasis neural network. Penggunaan teknik optimasi lanjutan seperti Bayesian Optimization atau Genetic Algorithm juga direkomendasikan untuk memperoleh parameter optimal secara lebih efisien. Dalam konteks praktis, hasil penelitian ini dapat menjadi dasar pengembangan sistem prediksi otomatis berbasis GRU dan LSTM yang terintegrasi dengan manajemen rantai pasok Toyota, sehingga mampu memberikan

rekomendasi yang lebih presisi dalam perencanaan produksi, pengendalian persediaan, serta strategi bisnis di masa depan.



DAFTAR PUSTAKA

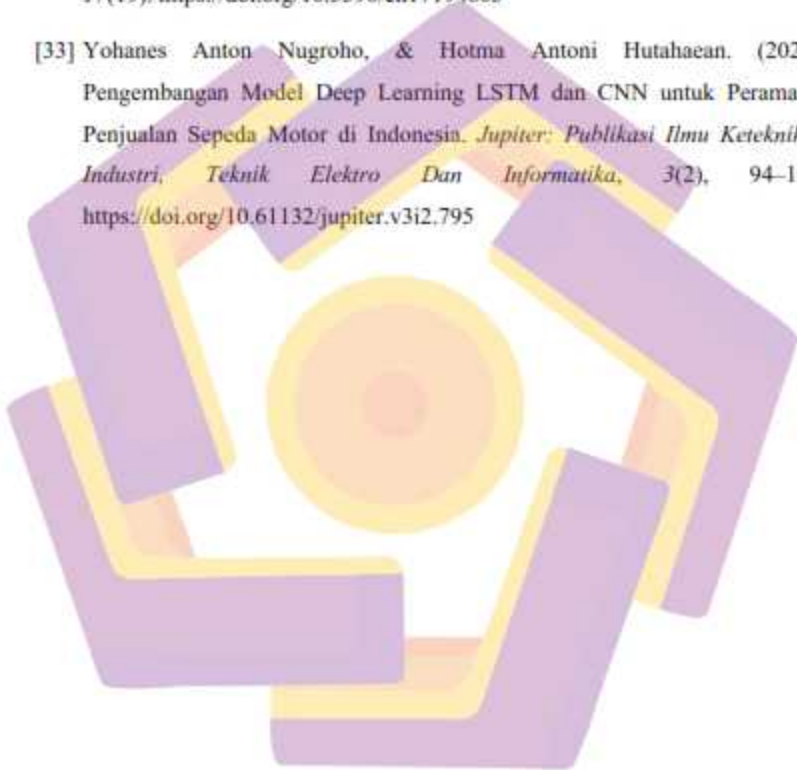
- [1] Ahmadi, A., Daccache, A., Sadegh, M., & Snyder, R. L. (2023). Statistical and deep learning models for reference evapotranspiration time series forecasting: A comparison of accuracy, complexity, and data efficiency. *Computers and Electronics in Agriculture*, 215. <https://doi.org/10.1016/j.compag.2023.108424>
- [2] Alfiansyah, M. R., Tukino, T., Hananto, A., & Novalia, E. (2025). Implementation of The Seasonal Autoregressive Integrated Moving Average Predictive Model on Raw Material Usage Data at PT. Plastik Karawang Flexindo. *International Journal of Engineering Science and Information Technology*, 5(3), 12–30. <https://doi.org/10.52088/ijesty.v5i3.867>
- [3] Ali, Mm., Hariyati, T., Yudestia Pratiwi, M., & Afifah Sekolah Tinggi Agama Islam Ibnu Rusyd Kotabumi, S. (n.d.). Metodologi Penelitian Kuantitatif Dan Penerapan Nya Dalam Penelitian. In *Education Journal.2022* (Vol. 2, Issue 2).
- [4] Arisena, A. (2020). PREDIKSI RETAIL SALES MOBIL TOYOTA MENGGUNAKAN METODE ARIMA (AUTOREGRESSIVE INTEGRATED MOVING AVERAGE). *Indonesian Journal of Strategic Management*, 3(1). <https://doi.org/10.25134/ijsm.v3i1.2817>
- [5] Ariyanti, V. P., & Tristyanti Yusnitasari. (2023). Comparison of ARIMA and SARIMA for Forecasting Crude Oil Prices. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 7(2), 405–413. <https://doi.org/10.29207/resti.v7i2.4895>
- [6] Arumugam, V., & Natarajan, V. (2023). Time Series Modeling and Forecasting Using Autoregressive-Integrated Moving Average and Seasonal Autoregressive Integrated Moving Average Models. *Instrumentation Mesure Metrologie*, 22(4), 161–168. <https://doi.org/10.18280/i2m.220404>
- [7] Ayu Wulandari dan Rahmat Gernowo, R. (2019). METODE AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) DAN METODE ADAPTIVE NEURO FUZZY INFERENCE SYSTEM (ANFIS) DALAM ANALISIS CURAH HUJAN (Vol. 22, Issue 1).

- [8] Buchori, M., & Sukmono, T. (2018). Peramalan Produksi Menggunakan Metode Autoregressive Integrated Moving Average (ARIMA) di PT. XYZ. *PROZIMA (Productivity, Optimization and Manufacturing System Engineering)*, 2(1), 27–33. <https://doi.org/10.21070/prozima.v2i1.1290>
- [9] Candra Susanto, P., Ulfah Arini, D., Yuntina, L., & Panatap Soehaditama, J. (n.d.). *Konsep Penelitian Kuantitatif: Populasi, Sampel, dan Analisis Data (Sebuah Tinjauan Pustaka)*. <https://doi.org/10.38035/jim.v3i1>
- [10] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. <http://arxiv.org/abs/1412.3555>
- [11] Dhake, H., Kashyap, Y., & Kosmopoulos, P. (2023). Algorithms for Hyperparameter Tuning of LSTMs for Time Series Forecasting. *Remote Sensing*, 15(8). <https://doi.org/10.3390/rs15082076>
- [12] Fahrudin, R., & Sumitra, I. D. (2019). Sistem Peramalan Kebutuhan Hidup Layak Minimum (Kapita/Bulan) Kota Bandung. *JURNAL SISTEM INFORMASI BISNIS*, 9(2), 192. <https://doi.org/10.21456/vol9iss2pp192-203>
- [13] Febiola, A., Dewi, A., Fazarin, F. M., Ramadhani, F., Khaffi, M. A., Akbar, R., & Dalimunthe, D. Y. (2024). Perbandingan Metode ARIMA dan SARIMA Dalam Peramalan Jumlah Penumpang Bandara Provinsi Kepulauan Bangka Belitung. *Jambura Journal of Mathematics*, 6(2). <https://doi.org/10.37905/jjom.v6i2.25081>
- [14] Ghide, L., Wei, S., & Ding, Y. (2022). Comparative Study of Wavelet-SARIMA and EMD-SARIMA for Forecasting Daily Temperature Series. *International Journal of Analysis and Applications*, 20, 17. <https://doi.org/10.28924/2291-8639-20-2022-17>
- [15] Hamiane, S., Ghanou, Y., Khalifi, H., & Telmem, M. (2024). Comparative Analysis of LSTM, ARIMA, and Hybrid Models for Forecasting Future GDP. *Ingenierie Des Systemes d'Information*, 29(3), 853–861. <https://doi.org/10.18280/isi.290306>

- [16] Hindarto, D. (2023). Comparison of RNN Architectures and Non-RNN Architectures in Sentiment Analysis. *Sinkron*, 8(4), 2537–2546. <https://doi.org/10.33395/sinkron.v8i4.13048>
- [17] Hurtado-Mora, H. A., García-Ruiz, A. H., Pichardo-Ramírez, R., González-del-Ángel, L. J., & Herrera-Barajas, L. A. (2024). Sales Forecasting with LSTM, Custom Loss Function, and Hyperparameter Optimization: A Case Study. *Applied Sciences (Switzerland)*, 14(21). <https://doi.org/10.3390/app14219957>
- [18] Khaira, U., Eko, P., Utomo, P., Suratno, T., & Gulo, C. S. (2019). PREDIKSI INDEKS HARGA SAHAM GABUNGAN (IHSG) MENGGUNAKAN ALGORITMA AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA). In *JUSS Jurnal Sains dan Sistem Informasi* (Vol. 2, Issue 2).
- [19] Kumar, S. V., & Vanajakshi, L. (2015). Short-term traffic flow prediction using seasonal ARIMA model with limited input data. *European Transport Research Review*, 7(3). <https://doi.org/10.1007/s12544-015-0170-8>
- [20] Mateus, B. C., Mendes, M., Farinha, J. T., Assis, R., & Cardoso, A. M. (2021). Comparing LSTM and GRU models to predict the condition of a pulp paper press. *Energies*, 14(21). <https://doi.org/10.3390/en14216958>
- [21] Melati Putri Wiryawanto, T., Hawani, Z., Attar Ramadhani, M., Attar Ramadhani, M., Studi Sistem Informasi, P., Sains dan Teknologi, F., Sunan Ampel Surabaya Jalan Ir Soekarno No. U. H., & Anyar, G. (2023). Comparison of Support Vector Machine (SVM) and Autoregressive Integrated Moving Average (ARIMA) Methods for Predicting Air Quality Using Python and KNIME. In *J Statistika* (Vol. 16, Issue 1).
- [22] Nusaiba Yulisa, P., Al Haris, M., & Rismawati Arum, P. (2023). Peramalan Nilai Ekspor Migas di Indonesia dengan Model Long Short Term Memory (LSTM) dan Gated Recurrent Unit (GRU). In *J Statistika* (Vol. 16, Issue 1).
- [23] Prabowo, T., Charis Fauzan, A., & Yuelisa Mafula, V. (2025). Analisis Deret Waktu untuk Forecasting Populasi Ternak di Indonesia dengan Model LSTM

1. *JSAI: Journal Scientific and Applied Informatics*, 8(1).
<https://doi.org/10.36085>
- [24] Pratiwi, D. (2013). Perencanaan Produksi Menggunakan Model ARIMA dan Pengendalian Persediaan Menggunakan Program Dinamik untuk Meminimumkan Total Biaya (Studi Kasus: Produksi Amplang UD. Usaha Devi) Production Planning using ARIMA Model and Inventory Controlling using Dynamic Program for Minimizing Total Cost (Case Study: Production of Amplang in UD. Usaha Devi). *Jurnal EKSPONENSIAL*, 4(1).
- [24] Ridwan, M., Sadik, K., & Afendi, M. (2023). Comparison of ARIMA and GRU Models for High-Frequency Time Series Forecasting. *Scientific Journal of Informatics*, 10(3), 389. <https://doi.org/10.15294/sji.v10i3.45965>
- [26] Rizvi, M. F. (2024). ARIMA Model Time Series Forecasting. *International Journal for Research in Applied Science and Engineering Technology*, 12(5), 3782–3785. <https://doi.org/10.22214/ijraset.2024.62416>
- [27] Rusman, J. A., Chunady, K., Makmud, S. T., Setiawan, K. E., & Hasani, M. F. (2023). Crude Oil Price Forecasting: A Comparative Analysis of ARIMA, GRU, and LSTM Models. *2023 IEEE 9th International Conference on Computing, Engineering and Design, ICCED 2023*.
<https://doi.org/10.1109/ICCED60214.2023.10425576>
- [28] Rustendi, T. (n.d.). *PENDEKATAN KUANTITATIF DALAM STUDI KASUS PADA PENELITIAN BIDANG AKUNTANSI*.
<http://jurnal.unsil.ac.id/index.php/jak>
- [29] Sah, S., Surendiran, B., Dhanalakshmi, R., & Yamin, M. (2023). Covid-19 cases prediction using SARIMAX Model by tuning hyperparameter through grid search cross-validation approach. *Expert Systems*, 40(5).
<https://doi.org/10.1111/exsy.13086>
- [30] Schmid, L., Roidl, M., Kirchheim, A., & Pauly, M. (2025). Comparing Statistical and Machine Learning Methods for Time Series Forecasting in Data-Driven Logistics—A Simulation Study. *Entropy*, 27(1).
<https://doi.org/10.3390/e27010025>

- [31] Song, X., Deng, L., Wang, H., Zhang, Y., He, Y., & Cao, W. (2025). Deep learning-based time series forecasting. *Artificial Intelligence Review*, 58(1). <https://doi.org/10.1007/s10462-024-10989-8>
- [32] Szostek, K., Mazur, D., Drahos, G., & Kuszniak, J. (2024). Analysis of the Effectiveness of ARIMA, SARIMA, and SVR Models in Time Series Forecasting: A Case Study of Wind Farm Energy Production. *Energies*, 17(19). <https://doi.org/10.3390/en17194803>
- [33] Yohanes Anton Nugroho, & Hotma Antoni Hutahaean. (2025). Pengembangan Model Deep Learning LSTM dan CNN untuk Peramalan Penjualan Sepeda Motor di Indonesia. *Jupiter: Publikasi Ilmu Keteknikan Industri, Teknik Elektro Dan Informatika*, 3(2), 94–104. <https://doi.org/10.61132/jupiter.v3i2.795>

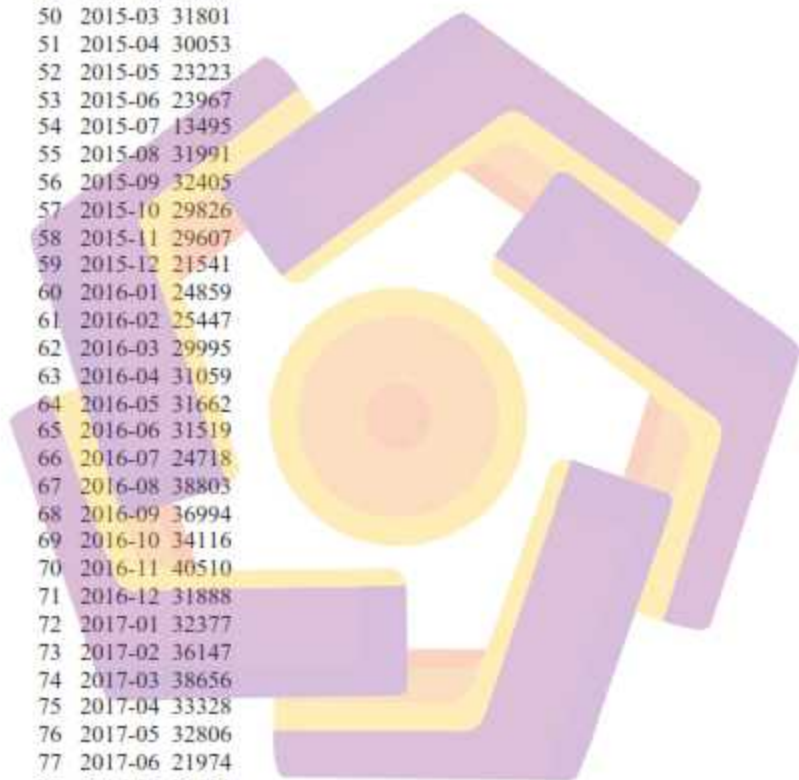


LAMPIRAN

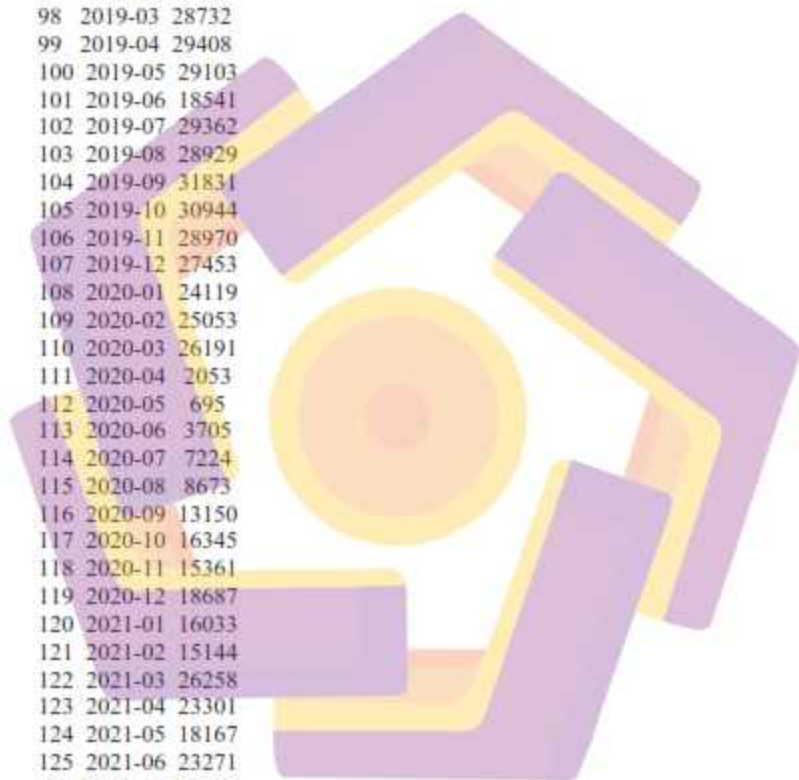
Lampiran 1 Dataset

	Month	Sales
0	2011-01	27619
1	2011-02	25532
2	2011-03	32275
3	2011-04	21128
4	2011-05	19554
5	2011-06	26175
6	2011-07	30149
7	2011-08	25207
8	2011-09	30655
9	2011-10	31109
10	2011-11	15195
11	2011-12	26076
12	2012-01	29189
13	2012-02	33558
14	2012-03	33306
15	2012-04	34264
16	2012-05	34737
17	2012-06	37176
18	2012-07	36353
19	2012-08	25848
20	2012-09	33705
21	2012-10	35855
22	2012-11	36996
23	2012-12	34427
24	2013-01	35923
25	2013-02	35318
26	2013-03	32726
27	2013-04	39668
28	2013-05	36282
29	2013-06	35125
30	2013-07	39210
31	2013-08	24899
32	2013-09	40235
33	2013-10	39246
34	2013-11	40781
35	2013-12	34819
36	2014-01	35886
37	2014-02	38631
38	2014-03	38959
39	2014-04	39323

40 2014-05 31440
41 2014-06 39107
42 2014-07 28757
43 2014-08 30273
44 2014-09 29247
45 2014-10 31538
46 2014-11 28835
47 2014-12 27123
48 2015-01 27166
49 2015-02 26743
50 2015-03 31801
51 2015-04 30053
52 2015-05 23223
53 2015-06 23967
54 2015-07 13495
55 2015-08 31991
56 2015-09 32405
57 2015-10 29826
58 2015-11 29607
59 2015-12 21541
60 2016-01 24859
61 2016-02 25447
62 2016-03 29995
63 2016-04 31059
64 2016-05 31662
65 2016-06 31519
66 2016-07 24718
67 2016-08 38803
68 2016-09 36994
69 2016-10 34116
70 2016-11 40510
71 2016-12 31888
72 2017-01 32377
73 2017-02 36147
74 2017-03 38656
75 2017-04 33328
76 2017-05 32806
77 2017-06 21974
78 2017-07 31621
79 2017-08 34690
80 2017-09 27251
81 2017-10 29979
82 2017-11 28797
83 2017-12 23706
84 2018-01 25405
85 2018-02 27665
86 2018-03 31424
87 2018-04 29360



88 2018-05 28950
89 2018-06 18642
90 2018-07 34984
91 2018-08 31149
92 2018-09 29821
93 2018-10 36119
94 2018-11 31981
95 2018-12 26661
96 2019-01 25081
97 2019-02 23443
98 2019-03 28732
99 2019-04 29408
100 2019-05 29103
101 2019-06 18541
102 2019-07 29362
103 2019-08 28929
104 2019-09 31831
105 2019-10 30944
106 2019-11 28970
107 2019-12 27453
108 2020-01 24119
109 2020-02 25053
110 2020-03 26191
111 2020-04 2053
112 2020-05 695
113 2020-06 3705
114 2020-07 7224
115 2020-08 8673
116 2020-09 13150
117 2020-10 16345
118 2020-11 15361
119 2020-12 18687
120 2021-01 16033
121 2021-02 15144
122 2021-03 26258
123 2021-04 23301
124 2021-05 18167
125 2021-06 23271
126 2021-07 21763
127 2021-08 29898
128 2021-09 34046
129 2021-10 21730
130 2021-11 33450
131 2021-12 32707
132 2022-01 22886
133 2022-02 24865
134 2022-03 33344
135 2022-04 27779



136 2022-05 13297
137 2022-06 27290
138 2022-07 29326
139 2022-08 30844
140 2022-09 33449
141 2022-10 33740
142 2022-11 26462
143 2022-12 28128
144 2023-01 28970
145 2023-02 27336
146 2023-03 29471
147 2023-04 21518
148 2023-05 28178

