



ELECTRIC VEHICLE DEMAND FORECASTING USING MACHINE LEARNING
AND CHARGING INFRASTRUCTURE SUITABILITY ANALYSIS IN THAILAND

MR. CHARTRIN KRONGHINLAD

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE IN
MATHEMATICS WITH COMPUTER SCIENCE
DEPARTMENT OF MATHEMATICS
GRADUATE COLLEGE
KING MONGKUT'S UNIVERSITY OF TECHNOLOGY NORTH BANGKOK
ACADEMIC YEAR 2025
COPYRIGHT OF KING MONGKUT'S UNIVERSITY OF TECHNOLOGY NORTH
BANGKOK

ELECTRIC VEHICLE DEMAND FORECASTING USING
MACHINE LEARNING AND CHARGING INFRASTRUCTURE
SUITABILITY ANALYSIS IN THAILAND

The seal of King Mongkut's University of Technology North Bangkok is a large, circular emblem in a light red color. It features a central sunburst with rays emanating from a central point. Below the sunburst are three tiered, pagoda-like structures. The entire emblem is surrounded by a circular border containing Thai text. The text at the top of the border reads 'พระจอมเกล้าธนบุรี' (Phra Chomklao Thonburi) and the text at the bottom reads 'มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี' (Mahavithayalai Tecthonoi Phra Chomklao Thonburi).

MR. CHARTRIN KRONGHINLAD

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE IN
MATHEMATICS WITH COMPUTER SCIENCE
DEPARTMENT OF MATHEMATICS
GRADUATE COLLEGE
KING MONGKUT'S UNIVERSITY OF TECHNOLOGY NORTH BANGKOK
ACADEMIC YEAR 2025
COPYRIGHT OF KING MONGKUT'S UNIVERSITY OF TECHNOLOGY
NORTH BANGKOK



Thesis Certificate

The Graduate College, King Mongkut's University of Technology North Bangkok

Title Electric Vehicle Demand Forecasting Using Machine Learning and Charging
Infrastructure Suitability Analysis in Thailand

By Mr. Chartrin Kronghinlad

Accepted by the Graduate College, King Mongkut's University of Technology
North Bangkok in Partial Fulfillment of the Requirements for the Master of
Science in Mathematics with Computer Science

..... Dean / Head of Department

(Professor Dr. Saowanit Sukparungsee)

Thesis Examination Committee

..... Chairman

(Assistant Professor Dr. Mathuros Panmuang)

..... Advisor

(Assistant Professor Dr. Sakchai Tangprasert)

..... Committee

(Assistant Professor Chanakarn Kiataramkul)

..... Committee

(Dr. Ekkachai Kunnawuttipreechachan)

Name : Mr. Chartrin Kronghinlad
Thesis Title : Electric Vehicle Demand Forecasting Using Machine Learning and Charging Infrastructure Suitability Analysis in Thailand
Major Field : Mathematics with Computer Science
King Mongkut's University of Technology North Bangkok
Thesis Advisor : Assistant Professor Dr. Sakchai Tangprasert
Academic Year : 2025

ABSTRACT

This research employs a range of data analysis techniques, including Genetic Algorithm (GA), Artificial Neural Networks (ANN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), the Grey Model (GM), a hybrid Grey Model with Genetic Algorithm (GM–GA), and the Residual Revised Grey Model with Genetic Algorithm (R2GM–GA), to forecast future demand for electric vehicles (EVs) in Thailand. The findings are intended to support strategic planning and development of EV charging infrastructure. The dataset was obtained from the Automotive Information Center of Thailand and comprises 22,288 records of new vehicle registrations from 2019 to 2023. After filtering for EV-specific entries, 5,320 records were retained for analysis. Additional data on EV charging station locations were collected from official websites and mobile applications of charging service providers in Thailand, yielding 3,174 records. Data on EV charging usage, comprising 3,740 records, were also incorporated; although some information was partially restricted due to confidentiality, the dataset remains sufficiently comprehensive and current for analysis. The selected models were evaluated based on forecasting performance using error-based metrics. The results indicate that the R2GM–GA model achieved the highest predictive accuracy, producing the lowest error and estimates most consistent with observed values. The forecasting error was 2.84%, which is considered highly reliable according to standard accuracy evaluation criteria. Furthermore, the forecasting results were used to examine the relationship between EV adoption and charging infrastructure requirements. The analysis suggests that Thailand would require approximately 53,543 charging connectors and at least 16,818 charging stations nationwide to adequately support daily EV usage. These findings provide practical insights for policymakers and relevant stakeholders in guiding the efficient planning and development of EV charging infrastructure in the future.

(Total 156 Pages)

Keywords: Forecasting, Grey Forecasting Model, Genetic Algorithm, Electric car.

Advisor

ชื่อ : นาย ถัดกรีน ครอบหินลาด
ชื่อวิทยานิพนธ์ : การพยากรณ์ปริมาณรถยนต์ไฟฟ้าด้วยการเรียนรู้ของเครื่องและการวิเคราะห์
ความเหมาะสมของสถานีอัดประจุไฟฟ้าในประเทศไทย
สาขาวิชา : คณิตศาสตร์เชิงวิทยาศาสตร์คอมพิวเตอร์
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
อาจารย์ที่ปรึกษาวิทยานิพนธ์ : ผู้ช่วยศาสตราจารย์ ดร.ศักดิ์ชาย ตั้งประเสริฐ
ปีการศึกษา : 2568

บทคัดย่อ

งานวิจัยนี้ประยุกต์ใช้เทคนิคการวิเคราะห์ข้อมูลหลายรูปแบบ ได้แก่ อัลกอริทึมเชิงพันธุกรรม (Genetic Algorithm: GA) โครงข่ายประสาทเทียม (Artificial Neural Networks: ANN) โครงข่ายประสาทเทียมแบบวนซ้ำ (Recurrent Neural Networks: RNN) โครงข่ายประสาทเทียมแบบหน่วยความจำระยะยาวและระยะสั้น (Long Short-Term Memory: LSTM) แบบจำลองเกรย์ (Grey Model: GM) แบบจำลองเกรย์ร่วมกับอัลกอริทึมเชิงพันธุกรรม (GM-GA) และแบบจำลองเกรย์ปรับปรุงเศษเหลือร่วมกับอัลกอริทึมเชิงพันธุกรรม (Residual Revised Grey Model with Genetic Algorithm: R2GM-GA) เพื่อพยากรณ์ความต้องการใช้รถยนต์ไฟฟ้าในประเทศไทย และสนับสนุนการวางแผนพัฒนาโครงสร้างพื้นฐานสถานีชาร์จรถยนต์ไฟฟ้า ข้อมูลที่ใช้ในการวิจัยประกอบด้วยข้อมูลการจดทะเบียนรถยนต์ใหม่ระหว่างปี พ.ศ. 2562–2566 จากศูนย์ข้อมูลอุตสาหกรรมยานยนต์แห่งประเทศไทย จำนวน 22,288 รายการ โดยคัดกรองเฉพาะข้อมูลรถยนต์ไฟฟ้าจำนวน 5,320 รายการ นอกจากนี้ ยังมีการรวบรวมข้อมูลสถานีชาร์จรถยนต์ไฟฟ้าจำนวน 3,174 รายการ และข้อมูลการใช้งานสถานีชาร์จจำนวน 3,740 รายการ จากเว็บไซต์และแอปพลิเคชันของผู้ให้บริการสถานีชาร์จในประเทศไทย แบบจำลองทั้งหมดถูกประเมินประสิทธิภาพด้วยค่าความคลาดเคลื่อน ผลการวิจัยพบว่า แบบจำลอง R2GM-GA ให้ประสิทธิภาพการพยากรณ์ดีที่สุด โดยมีค่าความคลาดเคลื่อนเพียงร้อยละ 2.84 ซึ่งอยู่ในระดับที่มีความน่าเชื่อถือสูง นอกจากนี้ ผลการวิเคราะห์ยังชี้ให้เห็นว่า ประเทศไทยควรมีหัวชาร์จประมาณ 53,543 หัวชาร์จ และสถานีชาร์จอย่างน้อย 16,818 แห่งทั่วประเทศ เพื่อรองรับการใช้งานรถยนต์ไฟฟ้าในอนาคต ผลการวิจัยนี้สามารถใช้เป็นแนวทางสำหรับผู้กำหนดนโยบายและหน่วยงานที่เกี่ยวข้องในการวางแผนและพัฒนาโครงสร้างพื้นฐานสถานีชาร์จรถยนต์ไฟฟ้าได้อย่างมีประสิทธิภาพในอนาคต

(วิทยานิพนธ์มีจำนวนทั้งหมด 156 หน้า)

คำสำคัญ : การพยากรณ์, แบบจำลองการพยากรณ์เกรย์, อัลกอริทึมเชิงพันธุกรรม, รถยนต์ไฟฟ้า

อาจารย์ที่ปรึกษาวิทยานิพนธ์

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to the university administrators and the Dean of the Faculty of Applied Science, King Mongkut's University of Technology North Bangkok, for providing me with the opportunity and financial support to pursue my master's degree. I would like to express my deepest appreciation to Assistant Professor Dr. Sakchai Tangprasert, Faculty of Applied Science, King Mongkut's University of Technology North Bangkok, for his invaluable guidance, encouragement, and continuous support throughout this research. His insightful advice on model development and thesis preparation has been greatly appreciated. I would also like to extend my gratitude to all lecturers of the Department of Mathematics, King Mongkut's University of Technology North Bangkok, for their valuable knowledge and guidance that contributed to the completion of this research. Furthermore, I would like to sincerely thank government agencies, private organizations, and individuals who provided valuable data and support for this research, including the Thailand Automotive Information Center; the Office of Vehicle Registration and Tax Standards (OBEC); Provincial Electricity Authority (PEA Volta); Evolt Technology; EA Anywhere; Electricity Generating Authority of Thailand (EGAT, Elexa); Metropolitan Electricity Authority (MEA); and PTT EV Station. Finally, I would like to express my deepest gratitude to my family for their constant encouragement, understanding, and support throughout my academic journey

Chartrin Kronghinlad

TABLE OF CONTENTS

	Page
Abstract(in English)	iv
Abstract(in Thai)	V
Acknowledgements	vii
Table of contents	Vii
List of figures	Xi
List of abbreviations	Xiv
Chapter 1 Introduction	1
1.1 Background and importance of the problem	1
1.2 Research objectives	2
1.3 Research hypothesis	2
1.4 Scope of research	2
1.5 Research methodology	3
1.6 Benefits of research	4
Chapter 2 Literature review and research theory	5
2.1 Electric car	5
2.2 Electric car charger	7
2.3 Forecasting	12
2.4 Grey model (GM)	13
2.5 Genetic algorithm (GA)	17
2.6 Artificial neural networks (ANN)	25
2.7 Recurrent neural network (RNN)	28
2.8 Long short-term memory networks (LSTM)	32
2.9 Errors analysis	34

TABLE OF CONTENTS(CONTINUE)

	Page
2.10 Literature Review	38
2.11 The process of optimizing a residual revised grey model using a genetic algorithm (R2GM-GA).	53
Chapter 3 Analysis Process	56
3.1 electric car charging stations analysis process	56
3.2 Data Collection	57
3.3 Data cleaning	58
3.4 Model generating	60
3.5 Gm optimization	64
3.6 Grey model (GM) integrated with genetic algorithm(GA)	65
3.7 Compare the performance of the algorithms	67
3.8 The demand for electric cars dashboard	74
3.9 Check errors	74
Chapter 4 Operating results	76
4.1 Application to two-year forecasts	76
4.2 Relationship between electric vehicles and electric vehicle charging stations	84
Chapter 5 Discussion and Conclusions	90
5.1 Summary of Study Results	90
5.2 Discussion of Results	93
5.3 Recommendations and Development Guidelines	93
REFERENCES	95
APPENDIX A	105

TABLE OF CONTENTS(CONTINUE)

	Page
APPENDIX B	116
APPENDIX C	131
APPENDIX D	147
APPENDIX E	151
BIOGRAPHY	156



LIST OF TABLES

		Page
2-1	Table of sum example of the MAE	35
2-2	Table of sum example of the Root mean squared error (RMSE)	36
2-3	Table of the Mean Absolute Percentage Error (MAPE)	37
2-4	Shows related literature review research	38
2-5	The table of techniques for checking the answers	46
3-1	Raw data table on new vehicle registrations	57
3-2	Updated raw data table on new vehicle registrations	59
3-3	Cleaning data	60
3.4	Table comparing the accuracy values obtained from the results of Each algorithm.	74
4-1	Forecasting 2 years	76
4-2	Check error table	83
4-3	EV Charging Station Performance Table	85
4-4	population EV Charging stations	86
4-5	Model Forecasting with Users charging stations	86
4-6	EV Charging Cars Per Day Table	87
A-1	Raw Data	106

LIST OF FIGURES

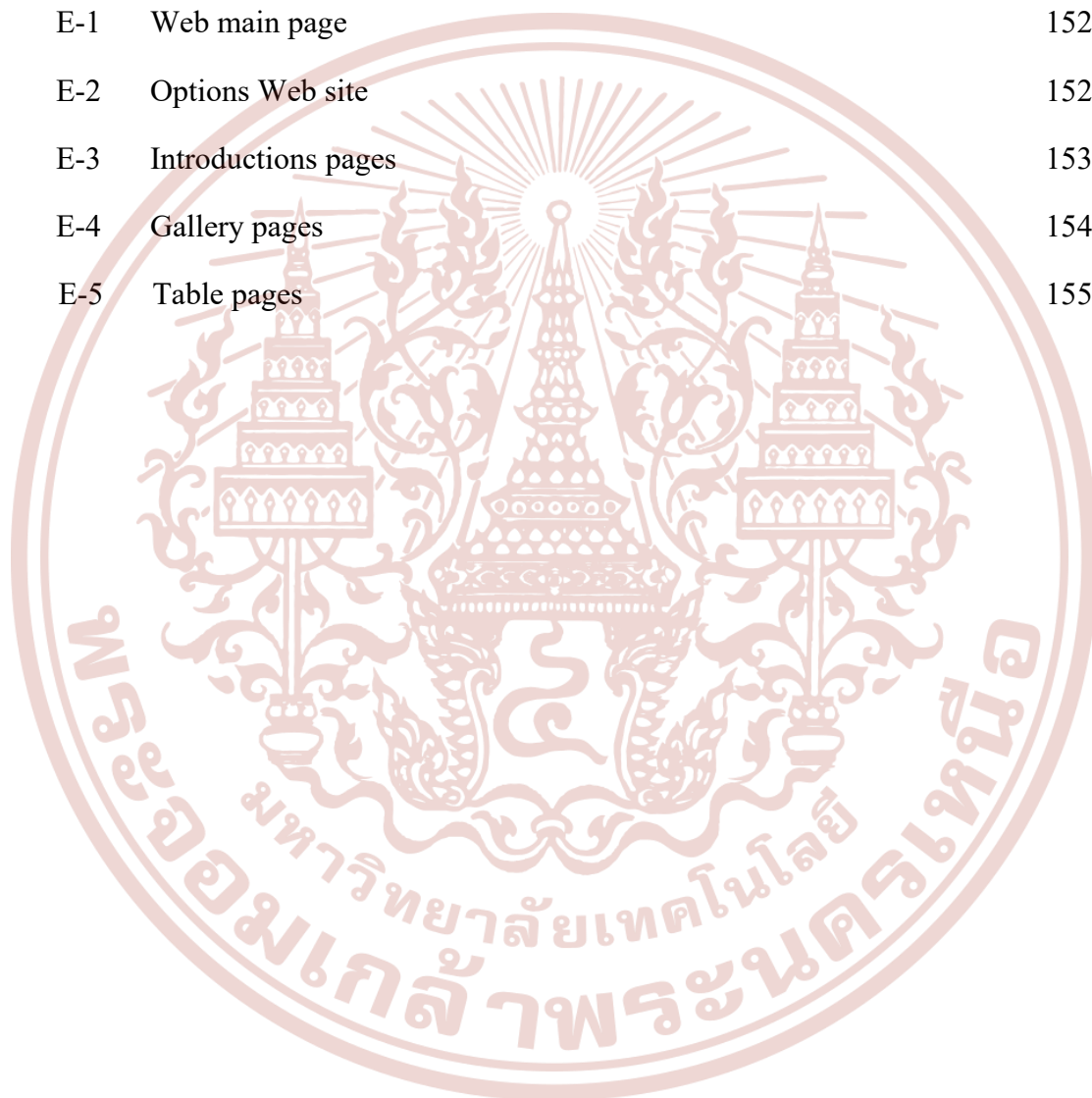
		Page
2-1	The world's first electric car.	5
2-2	Components of an electric car.	6
2-3	Normal charge	7
2-4	Normal Fast charge head	8
2-5	Fast charge head	8
2-6	Quick charge	9
2-7	Quick charge 7-hole	10
2-8	Home electric car charging equipment	11
2-9	Electric car charging station	12
2-10	The Grey model (GM(1,1))process	16
2-11	The Genetic Algorithm process	18
2-12	Pseudo-code	24
2-13	A Basic description of the parts of an ANN diagram	28
2-14	RNN equation process	29
2-15	RNN process	30
2-16	RNN example	31
2-17	RNN equation example	32
3-1	Flow chart research process	56
3-2	Flow chart grey model process	62
3-3	Flow chart Genetic Algorithm model process	63
3-4	Flow chart grey model (GM) integrated with genetic algorithm (GA)	66
3-5	Graph of the Forecast result using the ANN algorithm	67

LIST OF FIGURES (CONTINUE)

		Page
3-6	Graph of the Forecast results using the RNN algorithm	68
3-7	Graph of the Forecast results using the LSTM algorithm	69
3-8	Graph of the Forecast results using the LSTM algorithm	70
3-9	Graph of the Forecast results using the GM model algorithm	71
3-10	Graph of the forecast results using the grey model genetic algorithm model (GM-GA)	72
3-11	Graph of the forecast results using the Residual revised gray model integrated with the genetic algorithm (R2GM-GA)	73
4-1	Home web Forecasting	78
4-2	Graph of the forecast 2-year results using the ANN algorithm	79
4-3	Graph of the forecast 2-year results using the LSTM, RNN algorithm	79
4-4	Graph of the forecast 2-year results using the GM algorithm	80
4-5	Graph of the forecast 2-year results using the GA algorithm	81
4-6	Graph of the forecast 2-year results using the GM-GA algorithm	81
4-7	Graph of the forecast 2-year results using the R2GMGA algorithm	82
A-1	Raw Data from the Evolt app	107
A-2	Raw Data from the PTT app	108
A-3	Raw Data from the MEA app	109
A-4	Raw Data from the MEA All head app	110
A-5	Raw Data from the EA Anywhere app	111
A-6	Raw Data from the MG website	112
A-7	Raw Data on Electric Vehicle Charging Station Users from (MEA)	113

LIST OF FIGURES (CONTINUE)

	Page
A-8 Raw Data on Electric Vehicle Charging Station Users from (PEA)	114
E-1 Web main page	152
E-2 Options Web site	152
E-3 Introductions pages	153
E-4 Gallery pages	154
E-5 Table pages	155



LIST OF ABBREVIATIONS

MAE	Mean absolute error
RMSE	Root mean squared error
MAPE	Mean absolute percentage error
EV	Electric vehicle
GM	Grey model
GA	Genetic algorithm
ANN	Artificial neural network
RNN	Recurrent neural network
LSTM	Long short-term memory
GMGA	Grey model to genetic algorithm
R2GMGA	Residual revised grey model using a genetic algorithm



CHAPTER 1

INTRODUCTION

Forecasting the demand for electric cars affects the construction of electric car charging stations. Using an algorithm optimizing a residual revised gray model (GM) using a genetic algorithm (GA), the authors studied and discussed the Background and importance of the problem research, objectives research hypothesis, scope, and procedures, benefits of research, including research structure to know the purpose and methods of carrying out the researcher has set the following schedule:

- 1.1 Background and importance of the problem
- 1.2 Research objectives
- 1.3 Research hypothesis
- 1.4 Scope of research
- 1.5 Conduct research
- 1.6 Benefits of research

1.1 Background and importance of the problem

In the past five years, technological advancements have significantly influenced various industries, including the automotive sector. One of the most notable developments is the emergence of electric vehicles (EVs). The concept of electric vehicles dates back to 1881, when Gustave Trouvé presented an electric vehicle at the International Exposition of Electricity in Paris, France. However, this invention was primarily a demonstration model rather than a commercially viable vehicle (Chureungsakun, 2014). Subsequently, in 1884, Thomas Parker, an electrical engineer, developed an electric vehicle powered by rechargeable batteries in Wolverhampton, England. This innovation marked a significant step forward, as it introduced the concept of rechargeable energy storage. Nevertheless, it was not widely adopted as the dominant form of transportation at that time (Phromwong, 2018). During the same period, gasoline-powered vehicles experienced rapid development. These vehicles could be produced more efficiently, were less expensive, and allowed for faster refueling compared to electric vehicles. In addition, the invention of the electric starter motor made internal combustion engine vehicles easier to operate. As a result, gasoline-powered vehicles gained widespread popularity, leading to a decline in the use of electric vehicles during that era. The resurgence of electric vehicles began in the 1990s, particularly in the United States, when environmental concerns became more prominent. Regulatory initiatives, such as those introduced by the California Air Resources Board, encouraged the development of fuel-efficient and low-emission vehicles. The long-term objective of these policies was to reduce environmental pollution and promote cleaner transportation alternatives (Phromwong, 2018). At present, electric vehicles are widely accepted in many regions, particularly in China, Europe, and the United States (kritanu, 2022). Meanwhile, conventional vehicles powered by gasoline and diesel engines continue to contribute significantly to air pollution. As a result, many countries have introduced policies to phase out internal combustion engine vehicles. For example, France has announced plans to ban the sale and production of gasoline and diesel vehicles by 2040. Similarly, the United Kingdom has implemented policies to restrict internal combustion vehicles within the same timeframe. Other countries, such as Germany and India, have proposed earlier targets,

aiming to implement similar bans by 2030. In addition, countries such as the Netherlands and Norway have set more ambitious goals, targeting the phase-out of combustion vehicles as early as 2025. (Lye, 2017) These global trends are expected to accelerate the growth of the electric vehicle market and increase the demand for EV charging infrastructure. Consequently, many countries have begun expanding EV charging networks to support this transition. Therefore, this research focuses on analyzing and forecasting the demand for both internal combustion vehicles and electric vehicles in Thailand. These two categories represent the primary vehicle types preferred by consumers. The study aims to examine long-term market trends and identify the future direction of the automotive industry in Thailand. The scope of this research includes forecasting the demand for vehicles using monthly registration data categorized by fuel type from January 2019 to December 2023. The results are expected to support policymakers, executives, and stakeholders in the automotive industry in planning and developing EV charging infrastructure effectively in the future (Lye, 2017).

1.2 Research objectives

1.2.1 Plan and analyze data on electric vehicle demand to understand the connection with the need to build electric vehicle charging stations.

1.2.2 To examine the relationship between the number of electric vehicle purchases and the number of electric vehicle charging stations, with a focus on analyzing the growth trend of electric vehicle adoption alongside the development of electric vehicle charging infrastructure.

1.3 Research hypothesis

Analysis and forecasting of demand for electric cars affect the construction of electric car charging stations using a genetic algorithm (R2GM-GA) that has high accuracy in forecasting, more accurate than forecasts using other algorithms.

1.4 Scope of research

1.4.1 This research examines the demand for electric vehicles (EVs) in Thailand using vehicle registration data reported by the Thailand Automotive Information Center (AIC) for the period 2019–2023. The dataset consists of 22,288 records derived from official registration statistics published by the AIC and compiled for the purposes of this research. The analysis is limited to Battery Electric Vehicles (BEVs), in accordance with the AIC classification, which defines BEVs as vehicles powered exclusively by electricity. Plug-in Hybrid Electric Vehicles (PHEVs) are excluded because they rely partly on conventional fuels and are not classified as fully electric vehicles under the AIC definition (Chou et al., 2022) Accordingly, the scope of this research is confined to the analysis and forecasting of BEV demand in Thailand. The findings are intended to provide insights into BEV adoption trends and to support policymakers, energy planners, and stakeholders in the automotive and charging infrastructure sectors for future planning and decision-making.

1.4.2 Scope of the Study on Electric Vehicle Charging Station Construction

This research focuses on analyzing the volume of electric vehicle (EV) charging station construction in Thailand in 2023. The data were obtained from the official websites and mobile applications of EV charging station manufacturers and service

providers operating in Thailand. The analysis covers the number, capacity, and locations of charging stations constructed across the country during the specified year. Only data from verified and publicly accessible sources were included to ensure the reliability and accuracy of the research. Accordingly, the scope of this research is limited to examining the current state of EV charging infrastructure in 2023, which serves as a basis for assessing the relationship between EV adoption trends and the development of charging facilities in Thailand.

1.4.3 Scope of the Study on Forecasting Accuracy Evaluation. This section focuses on evaluating the forecasting accuracy of various algorithms used to predict the demand for electric vehicles (EVs). The objective is to identify the model that provides the highest accuracy and efficiency in predicting future EV demand. Several forecasting algorithms are analyzed and compared based on standard performance metrics, including Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Mean Absolute Error (MAE). Among these models, the Residual Revised Grey Model optimized by a Genetic Algorithm (R2GM-GA) demonstrates the highest level of accuracy and the lowest error, making it the most suitable model for predicting electric vehicle demand in this research.

1.4.4 Scope of the Study on the Relationship Between EV Registrations and Charging Stations. This research investigates the relationship between the number of electric vehicle (EV) registrations and the development of EV charging infrastructure in Thailand. The analysis is based on data including the number of registered EVs, the number of EV charging stations constructed, the number of charging heads per station, the operating hours of charging stations, and the average charging time per vehicle. Due to the lack of detailed data in 2023 on the actual number of vehicles using each charging station and the exact time spent per charging session, the analysis focuses on establishing a correlative relationship between EV adoption and the construction of charging stations using available aggregated data. Previous research indicates that such analysis can provide meaningful insights into the relationship between EV adoption and the expansion of charging infrastructure. Accordingly, the scope of this analysis is limited to evaluating trends and correlations between EV registration growth and charging station deployment, serving as a foundation for strategic planning in EV infrastructure development (Xydas et al., 2013).

1.5 Research methodology

1.5.1 Search for and collect information on total vehicle registrations and electric vehicle charging stations nationwide for use in analyzing and forecasting the demand for electric vehicles.

1.5.2 Study car registration data and filter out unnecessary information, keeping only what is necessary to forecast the demand for electric vehicles.

1.5.3 Design and forecast the demand for electric cars using each algorithm.

1.5.4 Analyze the data and forecast the demand for electric vehicles in Thailand using various algorithms, with a focus on (R2GM-GA).

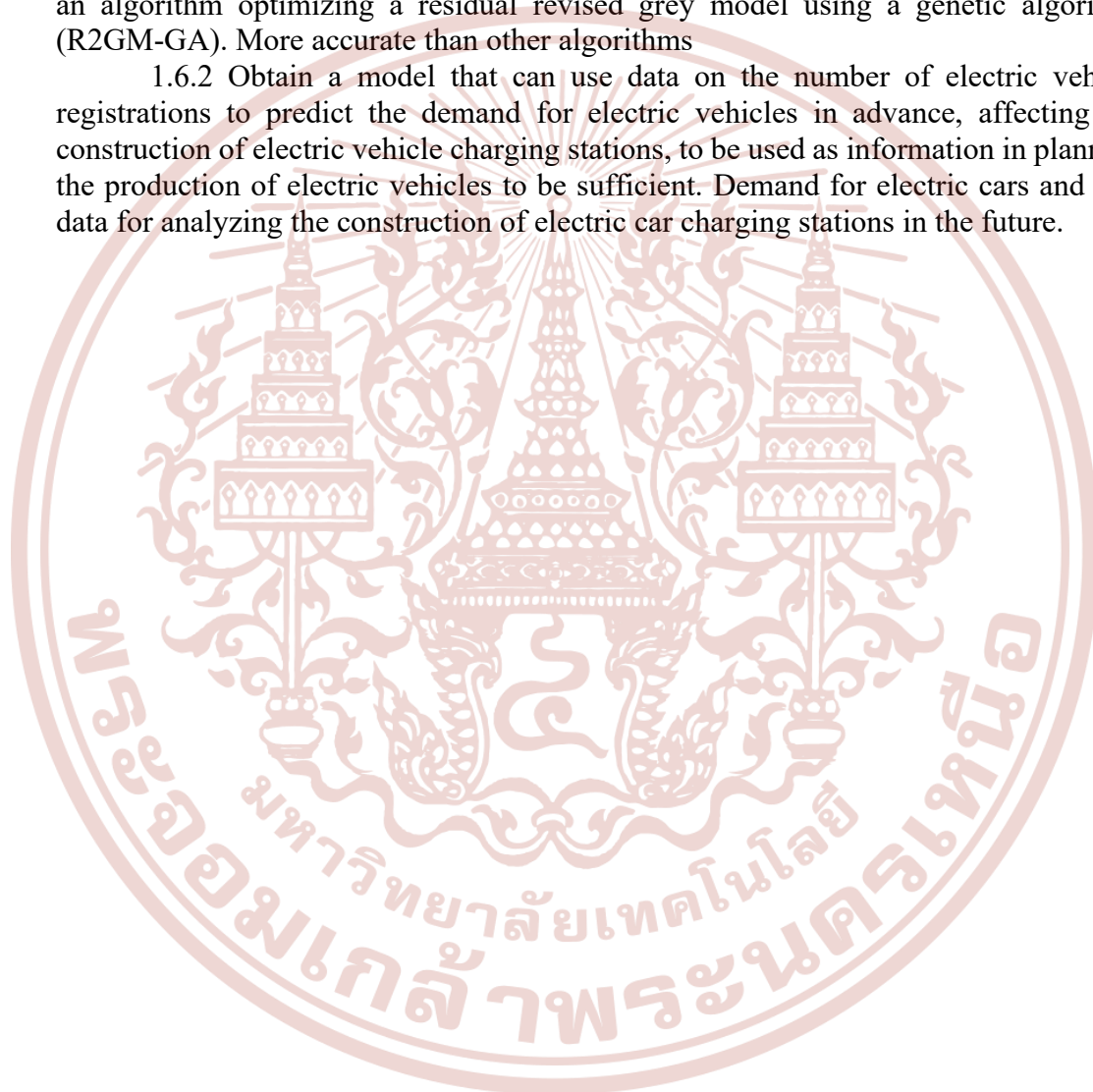
1.5.5 Evaluate the accuracy of each algorithm to determine which one is most accurate in forecasting.

1.5.6 Create a data visualization to present the results of the forecast.

1.6 Benefits of research

1.6.1 Obtain a forecasting model. Analysis and forecasting of demand for electric cars (forecasting) affect the construction of electric car charging stations using an algorithm optimizing a residual revised grey model using a genetic algorithm (R2GM-GA). More accurate than other algorithms

1.6.2 Obtain a model that can use data on the number of electric vehicle registrations to predict the demand for electric vehicles in advance, affecting the construction of electric vehicle charging stations, to be used as information in planning the production of electric vehicles to be sufficient. Demand for electric cars and it is data for analyzing the construction of electric car charging stations in the future.



CHAPTER 2

LITERATURE REVIEW AND RESEARCH THEORY

As stated in Chapter 1, the objective of this research is to develop a forecasting model for electric vehicle demand and to examine its impact on the construction of EV charging infrastructure. To achieve this objective, relevant documents and previous studies were reviewed to provide fundamental knowledge related to various types of vehicles, with a particular focus on electric vehicles. In addition, information regarding EV charging stations in Thailand was examined. This chapter also covers forecasting theories, relevant algorithms, and mathematical models applied in this research. These elements serve as the foundation for developing an appropriate forecasting model and for analyzing the relationship between EV demand and the development of charging station infrastructure

- 2.1 Electric car
- 2.2 Electric car charger
- 2.3 Forecasting
- 2.4 Grey model (GM)
- 2.5 Genetic algorithm (GA)
- 2.6 Artificial neural networks (ANN)
- 2.7 Recurrent neural network (RNN)
- 2.8 Long Short-Term Memory Networks (LSTM)
- 2.9 Errors analysis

2.1 Electric car

In the past five years, technological advancements have significantly influenced various industries, including the automotive sector. One of the most notable developments is the advancement of electric vehicles. Electric vehicles were first introduced in 1881, when Mr. Gustave Trouvé presented an electric vehicle at the International Exposition of Electricity (Exposition Internationale d'Électricité) in Paris, France. However, this invention was primarily a demonstration and is not considered the first fully developed electric vehicle, as illustrated in Figure 2-1 (Chureungsakun, 2024).



Figure 2-1 The world's first electric car. (Chureungsakun, 2024)

Three years later, in 1884, Mr. Thomas Parker, an electrical engineer, developed an electric vehicle powered by rechargeable batteries in Wolverhampton, England. Photographic evidence of this vehicle has been documented, confirming its historical existence. The key feature of this electric vehicle was its rechargeable battery system, which represented a significant technological advancement at that time. However, it is not considered the first fully developed electric vehicle. In 1888, Andreas Flocken, a German inventor, designed an electric vehicle known as the Flocken Elektrowagen. This vehicle is widely recognized as the first practical electric car. It is currently preserved and displayed at the PS. Speicher Museum in Einbeck, Germany. The following section describes the main components of an electric vehicle, as illustrated in Figure 2-2.

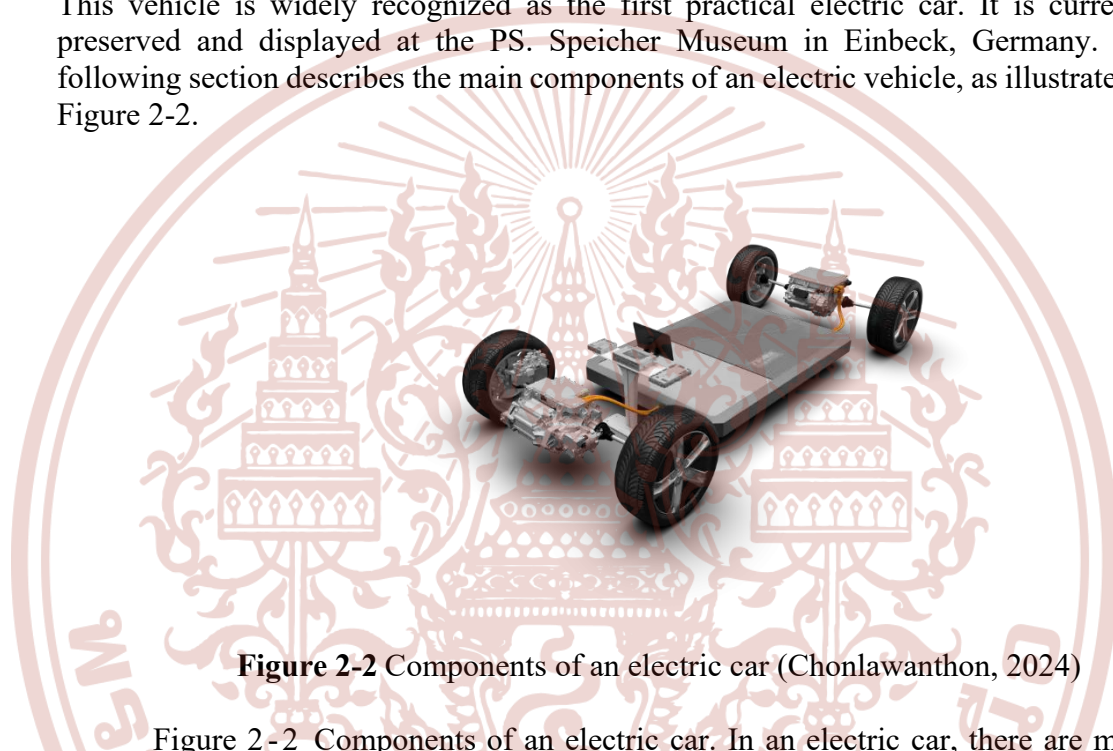


Figure 2-2 Components of an electric car (Chonlawanthon, 2024)

Figure 2-2 Components of an electric car. In an electric car, there are many components, which will consist of the first one is the motor from Figure 2-2, which is in the middle between the wheels, both front and back, depending on whether the car is a 4-wheel drive or 2-wheel drive. In Figure 2-2, it is a 4-wheel drive type. The electric motor is compared to a combustion car, an engine, or a car's power source. It functions to produce electricity to control the different parts of the car. If you notice, the size will be small for driving efficiency, which is easy to maintain, and the car is more agile. Next is the battery from Figure 2-2. It is the middle one, arranged in 2 blocks like a combustion car, which is the fuel tank that acts as the main power source in electric vehicles. It supplies energy to various parts of the car, which is the heaviest part of the car, because it stores energy to provide electric cars with sufficient energy for each drive. The unit of capacity is kilowatt-hours (KWH). The battery capacity depends on the type of car and the size of the car itself. (Chonlawanthon, 2024) Next is an important part of importing energy to the car called the on-board charger or the car charger. It is like the fuel tank in a combustion car. It functions to receive electricity into the electric car by converting the current into direct current (DC) to prevent short circuits. The electricity entering the machine will have another electrical converter that acts as an inverter, which is to control the electric current so that the current is not too strong. The electric current controller controls depending on the size of each battery. How much strength can it support? (Anurakjaturong, 2020) The driving system is similar to the

driving system of a combustion car. It starts by starting the motor to work and then sending an electric current from the battery obtained from the motor to control the car through the steering wheel and gear, and our accelerator pedal. (Lutsey and Nicholas, 2019)

2.2 Electric car charger

An electric car charging station is a source that charges electricity for electric car batteries. There are three types of charging available at present. (kritanu, 2022)

2.2.1. Normal charging

Ordinary charging means charging from a household outlet by connecting to a household master with a minimum supported current of 15-45 amperes, which must be alternating current (AC) only, as the ordinary charger supports. There is little electrical current. Because it receives little current, normal charging takes the longest, at 12–16 hours for each full charge, depending on the size of the battery in each vehicle. And normal charging can support only. (kritanu, 2022) the charging head. Normal type only, the general appearance is as shown in Figure 2-3.

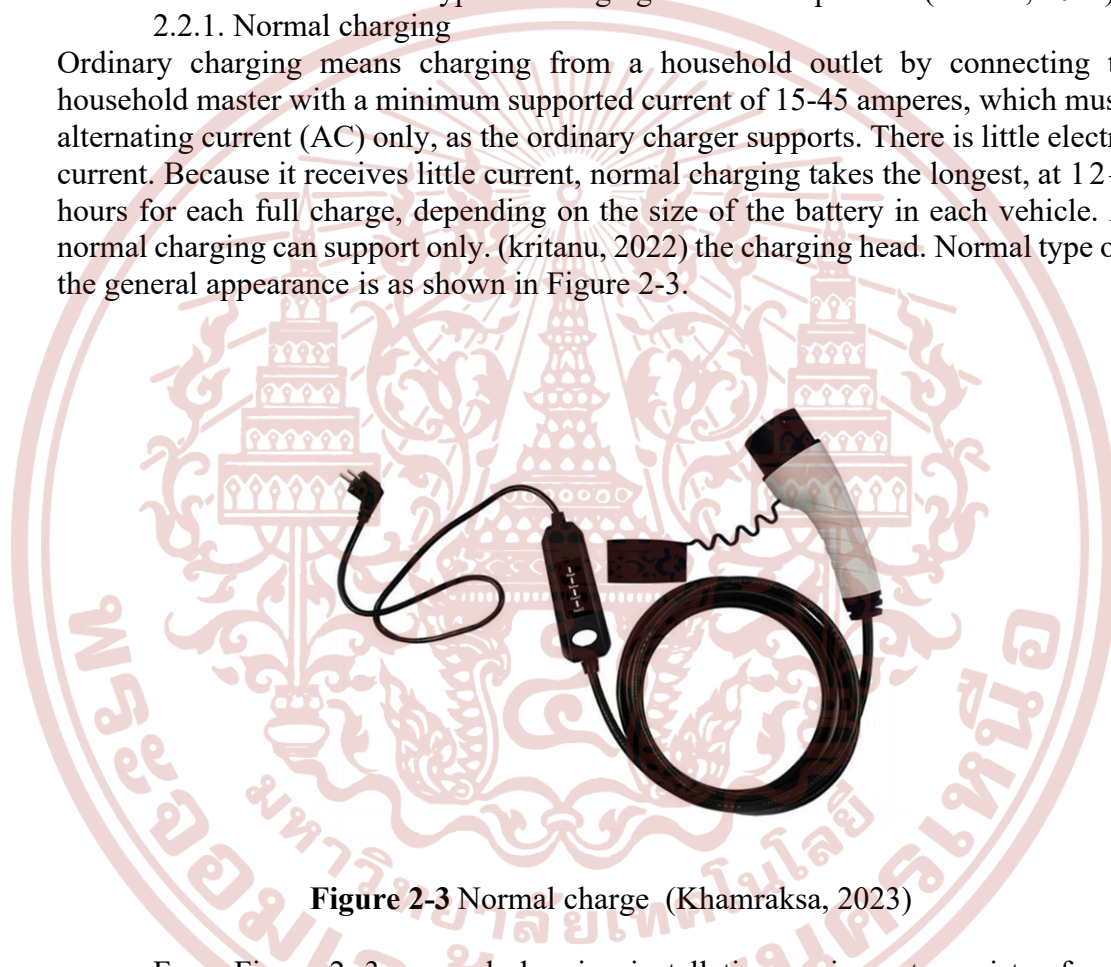


Figure 2-3 Normal charge (Khamraksa, 2023)

From Figure 2-3, normal charging installation equipment consists of a plug connected to home motor equipment, an electric current converter into the electric car battery, and a normal charging head that can accept electric current from 15 - 45 Amperes. This device can provide electricity to charge cars at 120–240 watts. This device is popularly used in North America and Japan due to its cheap installation price, but in Europe, it is not commonly used because European electric car brands do not support it. It is considered that it takes a long time to charge and wastes a lot of time. It is not popular among European cars, but it is popular in Japan because it is cheap and suitable for charging at home.

2.2.2. Fast charging

Fast charging is normal charging using alternating current charging. But it's quicker and safer because it's charged through a wall charger (Wall Box), causing more electricity to come out than normal, making charging a 24 kilowatt (KW) electric car take only 4–7 hours, and electric vehicles with a size of more than 30 kilowatts (KW)

take only 9 hours, but because they are still a switching current type of charging, they can only support a normal charger. The general (kritanu, 2022) appearance is as shown in Figure 2-4.



Figure 2-4 Normal Fast charge head (Khamraksa, 2023)

From Figure 2-4, the equipment for installing fast electric vehicle charging consists of 1. A normal charger can handle 32 amperes and can emit 250 watts of electricity. This type of fast charger is popular in Asia and China because China is the origin of this type of charger. Many of them are produced and exported to car brands in China. They are popular but not popular in Europe and America. Because it emits a small amount of electricity and takes a long time as normal, it is not popular in Thailand, as shown in Figure 2-5.



Figure 2-5 Fast charge head (Khamraksa, 2023)

From Figure 2-5, the equipment for installing fast electric vehicle charging consists of a normal charger head, but the head size is a 7-hole type, which is larger than the head in Figure 2-2. There are 2 types of connector ends: 1 pipe and 3 pipes. 1 pipe can receive current at 70 amperes but can release electricity at 250 watts, while 3

pipes can receive current. It is at 63 amperes but can release electricity at 450 watts because it can release a lot of electricity in Asia, especially Thailand. It is very popular because it is a 7-hole charger head that can be used to charge European brand cars. Can and other brands sell electric cars in Thailand, and the production cost is considered cheap if compared to quick charging.

2.2.3. Quick charging

Urgent charging is a direct current (DC) charging method. It is a quick charging method suitable for people who want to charge in a hurry and quickly. The average time it takes to charge from 0 percent to 80 percent is 40–60 minutes each time, which is a very fast time suitable for city life that requires the hustle and bustle of fast charging. Because urgent charges are charged with a direct electrical connection, it can support. There are two types of chargers: the normal type and the fast-charging type. The general (kritanu, 2022)The appearance is as shown in Figures 2–6.

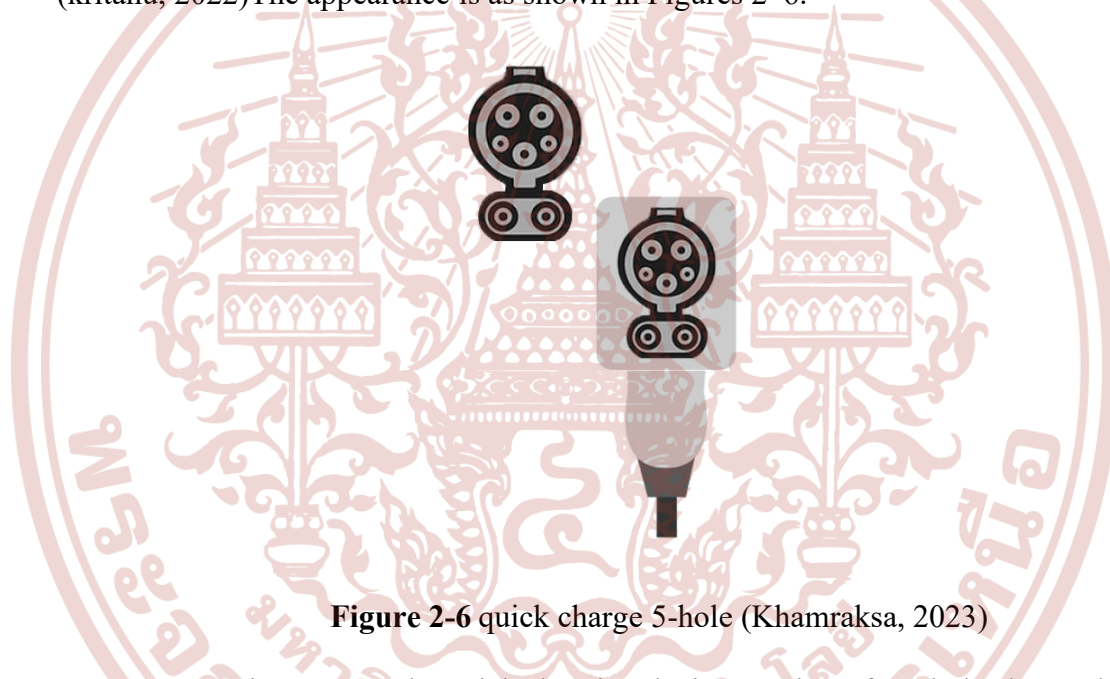


Figure 2-6 quick charge 5-hole (Khamraksa, 2023)

From Figures 2–6, the quick charging device consists of a 5-hole charger head, like a normal charger head, but with 2 additional quick charge holes added at the bottom, as shown in Figures 2–6. This makes this type of charger able to receive more electricity at 200 amperes, and because it receives a large amount of electricity, it can release up to 600 watts of electricity, making it possible to charge electric cars more quickly. But it is still not popular in Thailand because the 5-hole quick charger head is not very popular or expensive. Even though European electric car brands support it, it's not as popular as the 7-hole model. We've said that the 7-hole model charges faster, and the building price isn't very different, as shown in Figures 2-7.

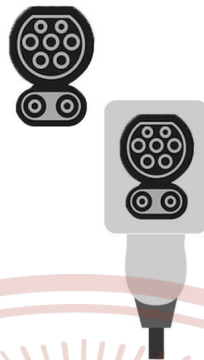


Figure 2-7 quick charge 7-hole (Limited, 2023)

From Figure 2-7, the quick charging device consists of a 7-hole charger head, like the quick charger head, but with 2 additional quick charge holes added at the bottom, as shown in the picture. 2-4 This makes this type of charger able to receive more electricity at 250 amperes, and because it receives a large amount of electricity, it can release up to 1000 watts of electricity, making it possible to charge electric cars more quickly. It is popular in Thailand because of the 7-hole quick charger head. European brand electric cars are very popular because the charger head is the fastest, causing Asian countries to produce more. It is not popular for retail sales because of the cost price. Expensive production makes the installation price expensive as well, causing it to usually be manufactured only by large companies. At present in Thailand, there are many products used by famous brands that are created by themselves.

2.2.4. The difference between a normal charger and a fast charger

An ordinary charger head (AC) is a normal charger head that supports only a small amount of current, making it able to support only alternating current connections because the charger can support an electric current not exceeding 200 amperes, but in A fast charger (DC) is a charger that can support a large current, with a minimum of more than 200 (master, 2023)amperes, making it able to support direct current. Therefore, it is often popular to use the quick charging method. The general characteristics and differences of the two heads are shown in Figures 2-4 (kritanu, 2022)

In summary, the construction of electric car charging stations today can be divided into two types: those in the owner's own home and the type of service point where the owner has the equipment included with the purchase of the vehicle will be obtained in the form shown in Figure 2-1. Not commonly used because delayed charging is not popular. Another type that is normally sold by the owner in Thailand in the format shown in pictures 2-4. Because it charges quickly and can support many brands. And the installation price is not expensive if compared to the service point type. The installation price for the owner is 25,000-60,000 baht, depending on the brand the customer purchases and the amount of electricity the customer chooses. As shown in Figures 2-8.



Figure 2-8 Home electric car charging equipment (Chaidee, 2023)

Finally, the service point type, where the service point form consists of 3 chargers, 1 quick charge (AC), and 2 quick charges (DC), can be chosen by customers. Most of the service point types in Thailand are. By a gas station mall and a famous tourist spot, because the price of installing a service point is expensive at 1 million-3 million baht, depending on the amount of electrical amperage. As for the price of the service, it is divided according to the amount of electrical amperage in the cabinet. Amperage cabinets are less than 300 kilowatts, quick charge is 4.5 baht per 1 kilowatt, quick charge is 6.9 baht per 1 kilowatt, and for amperage cabinets over 300 kilowatts, quick charge is 5.5 baht per 1 kilowatt. kilowatts and urgent is 8.8 baht per 1 kilowatt.(Authority, 2023) As shown in Figures 2–9.



Figure 2-9 electric car charging station (Authority, 2023)

2.3 Forecasting

Forecasting is the act of predicting or predicting the future by using behavior or data from the past over a long period to create events that can be seen and predicted in the future to be able to plan and prepare for action. Optimized future responses to business needs with the highest accuracy. In general, the use of forecasting is as follows: As a tool in planning effectively and efficiently. (Barus et al., 2019) Forecasts can be divided into Quantitative forecasting and Qualitative forecasting

2.3.1 Qualitative forecasting

Qualitative forecasting is a type of forecast that does not focus on statistical data methods but relies on descriptive information from opinions or those with experience knowledge and the ability to make forecasts, not mathematical principles. This method is suitable for gathering information in the past and is then used to summarize the results as a prophecy forecast or prediction of the company's sales in advance based on past experiences (Leenatham and Khemavuk, 2020) In this technique, there are many methods, such as the Delphi method, which is a method of bringing together various disadvantages to brainstorm and then meeting together to find solutions or find better ways, such as sales falling in 2023. What should we do to increase sales? In the year 2024, the researcher got the most profit and no loss by analyzing the disadvantages in 2023 for development in 2024. Let's get better later, how? (Chou et al., 2022) forecasting by surveying the market is a forecast from random users to see what they are using in their current trade to make it better and to meet the needs of customers in the future, such as a random group of 100 KMUTNB students was asked what brand of phones students currently use and why they chose that brand so that they could use it to develop their brand in the future as a forecast. Ask an expert to collect past data and ask executives or experts to analyze future trends, etc., and then forecast ideas. It is a forecast that requires all officials to come together and brainstorm opinions, which are not very popular because they have a very high degree of bias. Finally, forecast the

score prediction or estimate. rely on the most experienced people in the organization to make predictions using only historical data sets (Stekler and Symington, 2016)

2.3.2 Quantitative forecasting

Quantitative forecasting is the use of quantitative data sets or past numbers to create events and predictions. In the future, by relying on the principles of creating mathematical models by techniques that use quantitative principles (Y. L. Li et al., 2020) Quantitative forecasting techniques can be divided into two main types: the first is the forecast value obtained by relying on data that has occurred in the past as an important basis for forecasting, without using other data as an aid. Forecasting is called simple forecasting. This technique uses various types of forecasting methods, such as native smoothing techniques, time series, Box, adaptive filtering, etc. Another type of forecast value is obtained from the forecast, which is determined by the relationship between the necessary forecast data. Relying on data relationships in forecasting is called correlation forecasting. Including techniques for analyzing regression equations and econometric forecasting. (Baratsas et al., 2022) Time series forecasting is a simple forecasting technique that uses machine learning to analyze past data and predict the future of time series data, such as predicting the temperature each day and predicting how much rain will fall. What time of day and what time does it fall, and can also predict the amount of water stored for use in the dry season. Sometimes it is applied to predict the sales of new products in the market, what the results will be, or can be used to analyze sales in advance. Therefore, in this research, the researcher has used time series to help with the analysis. The demand for electric cars (Chou et al., 2022)

2.4 Grey model (GM)

The Grey Model (GM) is a theory of the grey system developed by Professor Julong Deng in the 1980s. The word "Grey Theory" means bad, incomplete, and uncertain, as the name suggests. (C.-N. Wang et al., 2020) Computer scientists have used the theory. in modeling to bridge the gap that exists between social science and science. The theoretical steps of the Grey Theory system are applied to a variety of data and are currently being used to solve forecasting problems by being called the "GM(n,m) Forecasting Model." Analyze incomplete or incomplete data into regression models to make predictions that cover a variety of data. (Wu-yong et al., 2013) Before starting the GM(n,m) process, starting from (n,m), n is the sequence of different equations and m is the number of variables. The steps involved in GM (n, m) are: Grey forecasting or (GM (1,1)) procedure can be done with the following steps: (Yao et al., 2022)

$$x^{(0)} = (x^{(0)}(1), x^{(0)}(2), \dots, x^{(0)}(n)) \quad (2-1)$$

After the original data accumulated operation, gray GM (1,1) first-order differential equations is set up as follows:

$$\frac{dx^{(1)}}{dt} + ax^{(1)} = u \quad (2-2)$$

The variables are defined as follows:

$x^{(1)}$ is the original data sequence,

$\frac{dx^{(1)}}{dt}$ is the first order accumulated generating operation (AGO)

sequence,

A is the development coefficient,

U is the step function.

The corresponding solution is:

$$x^{(1)}(k+1) = (x^{(0)}(1) - \frac{u}{a})e^{-ak} + \frac{u}{a} \quad (2-3)$$

The variables are defined as follows:

$x^{(1)}(k+1)$ Is the predicted value for the next time period

$x^{(0)}(1)$ is the initial value of the data

K is the time index.

A is the development coefficient, which determines the grey incident generation.

U is the grey action coefficient, representing the grey dynamic response-ability.

This model is often used for forecasting when only a few data points are available. Here's an example to illustrate how it works:

2.4.1 Gm Example

Let's say we have the following data points for a time series:

$$x = [10, 20, 30, 40, 50] \quad (2-4)$$

We want to forecast the next value in the series using the GM (1,1) model. First, we need to calculate the accumulated generating sequence (AGS) of the original data:

$$x^{(1)} = [10, 20, 30, 40, 50] \quad (2-5)$$

Then, we can calculate the development coefficient a and the grey input u using the first two terms of the AGS:

$$\begin{aligned}
 a &= \frac{1 - \frac{x^{(1)}n}{x^{(1)}(1)}}{n-1} \\
 a &= \frac{1 - \frac{150}{10}}{5-1} \\
 a &= \frac{1-15}{4} \\
 a &= -3.5 \\
 u &= x^{(1)}(1) * (1 - e^{-3.5}) \\
 u &\approx 25.292
 \end{aligned} \tag{2-6}$$

Finally, we can use the GM (1,1) equation to forecast the next value:

$$\begin{aligned}
 x(6) &= \left(10 - \frac{25.292}{-3.5}\right) \cdot e^{-3.5 \cdot 6} + \frac{25.292}{-3.5} \\
 &\approx 56.57
 \end{aligned} \tag{2-7}$$

So, according to the GM(1,1) model, the forecasted value for the next time step is approximately 56.57. (W. Zhou and He, 2013)

GM prediction models (1, 1) have a small number of required samples, are easy to calculate, and can be tested. Suitable for short-term forecasts for long-term forecasts, Only the most recent data is meaningfully useful in the absence of historical data. However, the model takes some time to integrate past data to make predictions. (C.-N. Wang et al., 2020) The Grey model (GM (1,1)) process as shown in Figures 2-10.

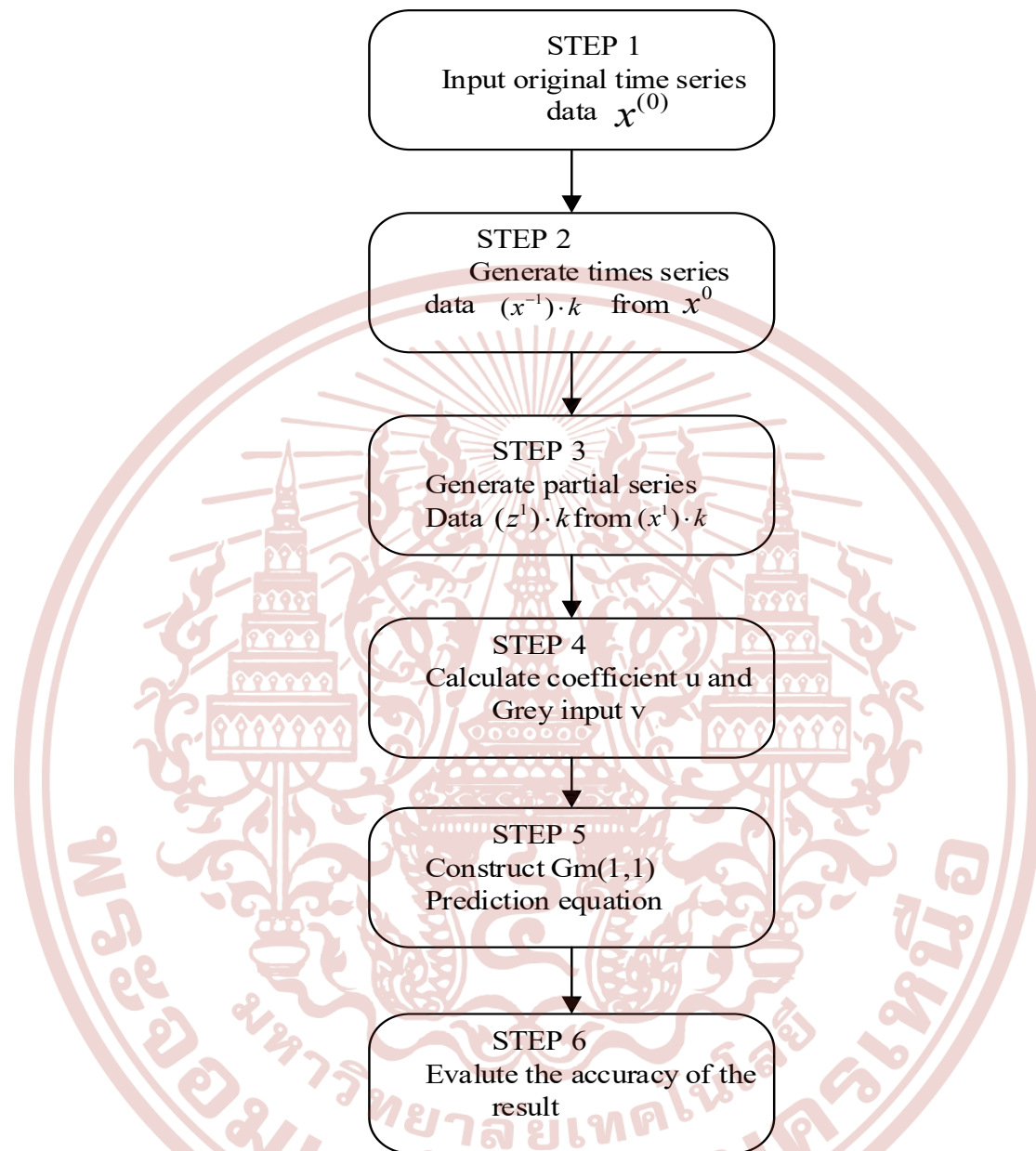


Figure 2-10 The Grey model (GM(1,1)) process (C.-N. Wang et al., 2020)

From Figure 2-10 Show the grey model process

Step 1: input original time series data $x^{(0)}$ Collect the historical data that represents the behavior of the system you want to model. Ensure the data is sequential and represents a trend or pattern.

Step 2: Generate time series data $(x^1) \cdot k$ from $x^{(0)}$ Generate the first-order accumulated generating operation (AGO) series $(x^1) \cdot k$ from the original data $x^{(0)}$

Step 3: Generate partial series data $(z^1) \cdot k$ from $(x^1) \cdot k$

Use the AGS to formulate the Grey Differential Equation, which is a first-order ordinary differential equation.

Step 4: Calculate coefficients u and grey input v Calculate the mean values of the adjacent elements in the accumulated data sequence.

Step 5: Construct GM (1,1) prediction equation
Calculate the GM (1,1) model based on the differential equation solution.

Step 6: Evaluate the accuracy of the results Use the GM (1,1) model to forecast future values of the time series. Plot the original data and the forecasted values to compare the performance of the GM(1,1) model(C.-N. Wang et al., 2020)

The GM (1, 1) forecast model can still be improved in many ways and can be used with other algorithms.(Xie and Xie, 2009b) In this research, the researcher conducted a GM experiment using the principle of optimizing a residual revised gray model (GM) to find answers in forecasting. By finding the answer by using a genetic algorithm (GA). To help find the answer itself to compare and find the best forecast for the demand for electric cars.

2.5 Genetic algorithm (GA)

GA is a type of evolutionary algorithm that is well-known and widely used. Its inventor was a Dutchman who invented this algorithm by getting the main idea from the selection of the population of chromosomes by each chromosome. Mozo finds the most suitable ditch. (Kushwaha et al., 2024) To find the most appropriate chromosomes for the problem and display them in the results of a genetic algorithm. (Mahmoudinazlou and Kwon, 2024) The main features of the genetic algorithm are: calculations of the genetic algorithm are mainly in the parameters of the bit string after encoding, rather than the parameters themselves, so the search analysis is not subject to restrictions on the continuity of the object parameters. The genetic algorithm for solving the problem uses multi-point random search methods (copy-mating - mutation) rather than the traditional single-point search methods, therefore avoiding the algorithm converging to the optimal solution in the region and getting the overall optimal solution. When operating a genetic algorithm, the only need is to give the required objective function, and there is no need for other auxiliary information (such as the continuity of the function, differential sexual). So it is suitable for all kinds of objective functions (Xie and Xie, 2009b) Due to the genetic algorithm with the above characteristics, in the past few years, genetic algorithms have been widely used to find optimal solutions to various problems. and get better results. Here, we use a genetic algorithm to (GM (1, 1)) prediction model parameters to optimize the solution-finding process of the model, accessing the (GM (1), 1) prediction model based on genetic algorithm optimization. A flow chart of a general genetic algorithm is shown in Fig. 2-5, which shows how each step of GM works.(Xie and Xie, 2009a, 2009b) As shown in Figures 2-11.

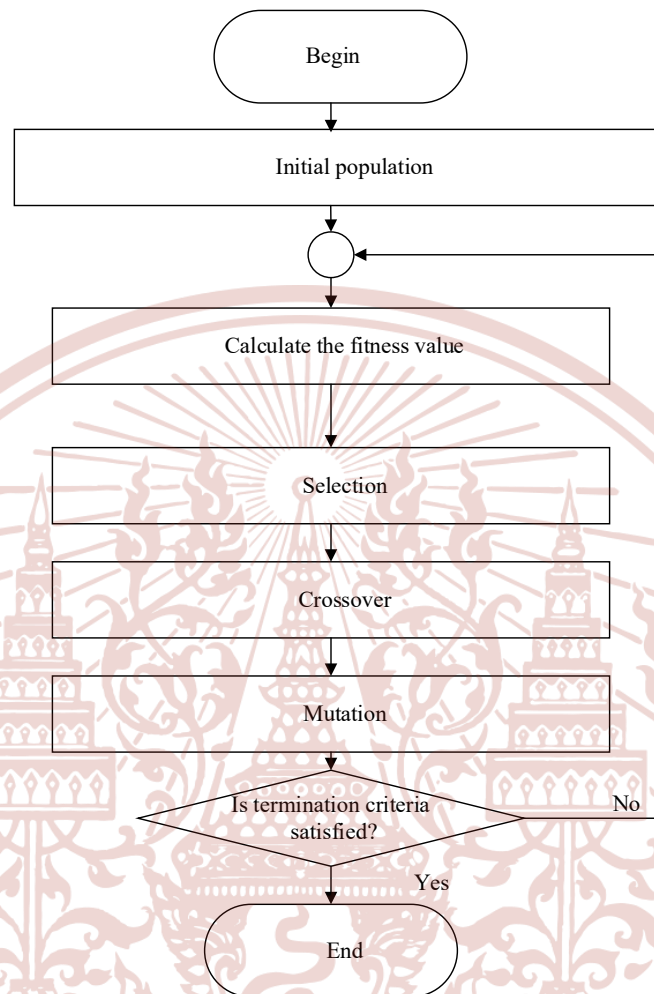


Figure 2-11 The genetic algorithm process (Albadr et al., 2020)

GM (1,1) Forecast model In this paper, a more accurate GM (1,1) prediction model optimized with a genetic algorithm for the total amount of import-export trade will be presented, which can reduce the modeling error of the traditional GM forecast model (1,1) and can be used to analyze the sales in advance itself.(Xie and Xie, 2009b) There are 5 phases in the genetic algorithm which are as follows:

2.5.1 Initial population This process starts with a set of individuals which is called a population. an individual has its characteristics and these individuals are known as genes. genes are combined into a string to form a chromosome.

2.5.2 Fitness Function The fitness function determines how fit an individual is to survive in the next generation or not. Fitness function plays a vital role in genetic algorithms. The fitness function gives a score to everyone. The probability that an individual will be selected for the next generation is based on his fitness score. the fittest ones will survive in the next generation (Cavallaro et al., 2024)

2.5.3 Selection The idea of the selection phase is to select the fittest individuals from the population and give approval to them for the next generation. Individuals are selected based on their fitness scores. Individuals with high fitness scores have more chances to be chosen for reproduction.

2.5.4 Crossover The Crossover is the most important phase in a genetic algorithm. A crossover point is randomly selected from the generation, and Offspring are created by exchanging the genes of parents among themselves until the crossover point is reached. New offspring are added to the population, and a new population will be generated.

2.5.5 Mutation The Mutation process is used to maintain diversity in the generation, and it prevents premature convergence. In mutation, genes are randomly replaced in a position with a new value. termination the algorithm terminates if the population has converged. On termination, the algorithm provides the optimal answer(Samavati et al., 2011) "Here are examples of applications that use genetic algorithms to solve the problem of combination. Suppose there is equality $a + 2 \cdot b + 3 \cdot c + 4 \cdot d = 30$ genetic algorithm will be used to find the value of a,b,c, and d that satisfy the above equation. "(Hermawanto, 2013) The variables are defined as follows:

$$f(x) = a + 2 \cdot b + 3 \cdot c + 4 \cdot d = 30 \quad (2-8)$$

Since there are four variables in the equation, namely a, b, c, and d, we can compose the chromosome as follows: To speed up the computation, we can restrict the values of variables a, b, c, and d are integers between 0 and 30. (MacNeil and Schultz, 2010)

2.5.6. GA example

For example, we define the number of chromosomes in the population are 6, then we generate the random values of genes a,b,c, and d for 6 chromosomes (Joshi and Krishna, 2014)

Step 1: initialization

Create an initial population of people. Try using a population size of 6 people according to equations (2-9) divided into 6 chromosomes.

$$\begin{aligned} c[1] &= [a : b : c : d] = [12 : 5 : 23 : 8] \\ c[2] &= [a : b : c : d] = [2 : 21 : 18 : 3] \\ c[3] &= [a : b : c : d] = [10 : 4 : 13 : 14] \\ c[4] &= [a : b : c : d] = [20 : 1 : 10 : 6] \\ c[5] &= [a : b : c : d] = [1 : 4 : 13 : 19] \\ c[6] &= [a : b : c : d] = [20 : 5 : 17 : 1] \end{aligned} \quad (2-9)$$

Step 2: Computation of fitness function

Evaluate the fitness of each individual using equation (2-8) and find f(x) to equation (2-10)

$$f(1) = \text{abs}(12 + 2 \cdot 5 + 3 \cdot 23 + 4 \cdot 8) - 30$$

$$f(1) = \text{abs}(123 - 30)$$

$$f(1) = 93$$

$$f(2) = \text{abs}(2 + 2 \cdot 21 + 3 \cdot 18 + 4 \cdot 3) - 30$$

$$f(2) = \text{abs}(110 - 30)$$

$$f(2) = 80$$

$$f(3) = \text{abs}(10 + 2 \cdot 4 + 3 \cdot 13 + 4 \cdot 14) - 30$$

$$f(3) = \text{abs}(113 - 30)$$

$$f(3) = 83$$

$$f(4) = \text{abs}(20 + 2 \cdot 1 + 3 \cdot 10 + 4 \cdot 6) - 30$$

$$f(4) = \text{abs}(76 - 30)$$

$$f(4) = 46$$

$$f(5) = \text{abs}(1 + 2 \cdot 4 + 3 \cdot 13 + 4 \cdot 19) - 30$$

$$f(5) = \text{abs}(124 - 30)$$

$$f(5) = 94$$

$$f(6) = \text{abs}(20 + 2 \cdot 5 + 3 \cdot 17 + 4 \cdot 1) - 30$$

$$f(6) = \text{abs}(85 - 30)$$

$$f(6) = 55$$

(2-10)

Step 3: Selection

The fittest chromosome has a higher prob. Will be selected for the next generation. To avoid divided by zero problem, the value of $f(x)$ is added by 1

$$fitness[1] = \frac{1}{(1 + f(1))}$$

$$fitness[1] = \frac{1}{94} = 0.0106$$

$$fitness[2] = \frac{1}{(1 + f(2))}$$

$$fitness[2] = \frac{1}{81} = 0.0123$$

$$fitness[3] = \frac{1}{(1 + f(3))}$$

$$fitness[3] = \frac{1}{84} = 0.0119$$

$$fitness[4] = \frac{1}{(1 + f(4))}$$

$$fitness[4] = \frac{1}{47} = 0.0213$$

$$fitness[5] = \frac{1}{(1 + f(5))}$$

$$fitness[5] = \frac{1}{95} = 0.0105$$

$$fitness[6] = \frac{1}{(1 + f(6))}$$

$$fitness[6] = \frac{1}{56} = 0.0179$$

$$total = 0.0106 + 0.0123 + 0.0119$$

$$total = 0.0348 + 0.0213 + 0.0105 + 0.0179$$

$$total = 0.0845$$

(2-11)

Step 4: prob. For each chromosome by

$$p[i] = \frac{fitness[i]}{total}$$

(2-12)

$$\begin{aligned}
p[1] &= \frac{0.0106}{0.0845} = 0.1254 \\
p[2] &= \frac{0.0123}{0.0845} = 0.1456 \\
p[3] &= \frac{0.0119}{0.0845} = 0.1408 \\
p[4] &= \frac{0.0213}{0.0845} = 0.2521 \\
p[5] &= \frac{0.0105}{0.0845} = 0.1243 \\
p[6] &= \frac{0.0179}{0.0845} = 0.2118
\end{aligned} \tag{2-13}$$

After the calculation in Eq. (2-12), We can see that the 4th chromosome has the highest performance. This chromosome has the highest probability of being selected for the next generation of chromosomes. for the selection process, we use a roulette wheel. For that, we should calculate the probability value to find the appropriate value for the next step(Hermawanto, 2013)

Step 5: selection is performed using the roulette wheel

$$\begin{aligned}
c[1] &= 0.1254 \\
c[2] &= p[1] + p[2] \\
c[2] &= 0.1254 + 0.1456 \\
c[2] &= 0.2710 \\
c[3] &= p[1] + p[2] + p[3] \\
c[3] &= 0.1254 + 0.1456 + 0.1408 \\
c[3] &= 0.4118 \\
c[4] &= p[1] + p[2] + p[3] + p[4] \\
c[4] &= 0.1254 + 0.1456 + 0.1408 + 0.2521 \\
c[4] &= 0.6639 \\
c[5] &= p[1] + p[2] + p[3] + p[4] + p[5] \\
c[5] &= 0.1254 + 0.1456 + 0.1408 + 0.2521 + 0.1243 \\
c[5] &= 0.7882 \\
c[6] &= p[1] + p[2] + p[3] + p[4] + p[5] + p[6] \\
c[6] &= 0.1254 + 0.1456 + 0.1408 + 0.2521 + 0.1243 + 0.2118 \\
c[6] &= 1.0
\end{aligned} \tag{2-14}$$

After calculating the cumulative probability of the selection process using the roulette wheel in Equation (2-14), the chromosome value is obtained and the process of generating random numbers R in the range 0-1 can be continued. Goes into this equation. (2-15) to find the R-value (Hermawanto, 2013)

$$\begin{aligned}
 R[1] &= 0.201 \\
 R[2] &= 0.284 \\
 R[3] &= 0.099 \\
 R[4] &= 0.822 \\
 R[5] &= 0.398 \\
 R[6] &= 0.501
 \end{aligned}
 \tag{2-15}$$

if $R[1]$ lies bet. $C[1]$ and $c[2]$ then select chromosome[2] as a chromosome in the new pop for the next gen. as it is a higher value.

$$\begin{aligned}
 NewChromosome[1] &= Chromosome[2] \\
 NewChromosome[2] &= Chromosome[3] \\
 NewChromosome[3] &= Chromosome[1] \\
 NewChromosome[4] &= Chromosome[6] \\
 NewChromosome[5] &= Chromosome[3] \\
 NewChromosome[6] &= Chromosome[4]
 \end{aligned}
 \tag{2-16}$$

Chromosomes in the population thus became:

$$\begin{aligned}
 Chromosomes[1] &= [2 : 21 : 18 : 3] \\
 Chromosomes[2] &= [10 : 4 : 13 : 4] \\
 Chromosomes[3] &= [12 : 5 : 23 : 8] \\
 Chromosomes[4] &= [20 : 5 : 17 : 1] \\
 Chromosomes[5] &= [10 : 4 : 13 : 14] \\
 Chromosomes[6] &= [20 : 1 : 10 : 6]
 \end{aligned}
 \tag{2-17}$$

Step 7: crossover

Chromosomes are controlled using a crossover rate. Pseudo-code for the crossover process is as follows: Figure 2-12

```

begin
k ← 0;
while(k < population) do
R[k] = random(0-1);
if(R[k] < pc) then
select Chromosome[k] as parent;
end;
k = k + 1;
end;
end;

```

Figure2-12 Pseudo-code

accordingly for $R[1], R[4], R[5]$ parents are chromosome[1], chromosome[4], chromosome[5] selected .

$$\begin{aligned}
Chromosome[1] &= Chromosome[1] \times Chromosome[4] \\
Chromosome[4] &= Chromosome[4] \times Chromosome[5] \\
Chromosome[5] &= Chromosome[5] \times Chromosome[1]
\end{aligned} \tag{2-18}$$

Step 8: deciding the crossover point

$$\begin{aligned}
c[1] &= 1 \\
c[2] &= 2 \\
c[3] &= 3
\end{aligned} \tag{2-19}$$

The researcher selected the random values between 1 to 3

$$\begin{aligned}
Chromosome[1] &= Chromosome[1] \times Chromosome[4] \\
Chromosome[4] &= Chromosome[4] \times Chromosome[5] \\
Chromosome[5] &= Chromosome[5] \times Chromosome[1]
\end{aligned} \tag{2-20}$$

$$\begin{aligned}
[2:5:17:1] &= [2:21:18:3] \times [20:5:17:1] \\
[20:4:13:14] &= [20:5:17:1] \times [10:4:13:14] \\
[10:4:18:3] &= [10:4:13:14] \times [2:21:18:3]
\end{aligned} \tag{2-21}$$

Thus, Chromosome population after experiencing a crossover process:

$$\begin{aligned}
Chromosomes[1] &= [2:5:17:1] \\
Chromosomes[2] &= [10:4:13:4] \\
Chromosomes[3] &= [12:5:23:8] \\
Chromosomes[4] &= [20:4:13:14] \\
Chromosomes[5] &= [10:4:18:3] \\
Chromosomes[6] &= [20:1:10:6]
\end{aligned} \tag{2-21}$$

Step 9: mutation is done by replacing the gene at a random position with

The number of chromosomes that have mutations in a population is determined by the mutation rate parameter. a new value. we must compute the total length of the gen

$$\begin{aligned}
total_{gen} &= numberofgeninChromosome \cdot numberofpopulation \\
total_{gen} &= 4 \cdot 6 \\
total_{gen} &= 24
\end{aligned} \tag{2-22}$$

The mutation process generates a random no. bet. 1 to 24 if the generated random no is smaller than the mutation rate(pm) variable, suppose we define the pm mutation rate as 10%. It is expected that 10% (0.1) of total genes will be in the pop. That will be mutated: no. of mutations = $0.1 \cdot 24 = 2.4 = 2$ approximating functions suppose generation of random number yield 12 and 18 then the chromosome which has the mutation is chromosome number 3, gen number 4, and chromosome 5 gen. number 2. The value of the mutated point is replaced by a random number between 0-30. Suppose the generated random numbers are 2 and 5, then the chromosome composition after mutation is:(Doğan et al., 2011)

$$\begin{aligned}
\text{Chromosomes}[1] &= [2:5:17:1] \\
\text{Chromosomes}[2] &= [10:4:13:4] \\
\text{Chromosomes}[3] &= [12:5:23:8] \\
\text{Chromosomes}[4] &= [20:4:13:14] \\
\text{Chromosomes}[5] &= [10:4:18:3] \\
\text{Chromosomes}[6] &= [20:1:10:6]
\end{aligned} \tag{2-23}$$

These new chromosomes will undergo chromosomes such as evaluation selection, crossover, and mutation, and at the end, it produces a new generation of chromosomes for the next iteration. This process will be repeated until a predetermined no. of gen. after running 50 generations the best chromosome is obt. $A = 7, b = 5, c = 3$ and $d = 1$. (MacNeil and Schultz, 2010)

2.6 Artificial neural networks (ANN)

An ANN is one of the data mining techniques and one of the algorithms of Neural Networks. It is a mathematical model or computational model that attempts to simulate the structure and behavior of biological artificial neural networks (G.-q. Li et al., 2010) for data processing by calculation to simulate the operation of artificial neural networks in the human brain. To create tools that can learn to recognize Pattern Recognition and create new Knowledge Extraction, like the ability of the human brain. Artificial neural networks consist of artificial neurons, which are called units called neutrons, which are connected as a network. Like the way neurons in the brain are connected. The working principle of an ANN consists of an Input Layer, Hidden Layers, and an Output Layer. It starts with the Input Layer, which acts to input data by feeding it into this layer, the first layer of the network. Examples of data input include images, numbers, text, etc. After inputting data, it goes to the Hidden Layers, which is the data processing layer that can be calculated, consisting of neurons that will process the data. By calculating the weights and biases, the activation function is then used to pass the results to the next layer. The activation function helps to convert the sum of the calculated weights into a non-linear form so that the ANN can learn and work with complex data (Gao and Yang, 2024).

2.6.1 ANN equation

For example, the Rectified Linear Unit (ReLU) will be released as 0 if the calculated value is less than 0, otherwise, the value will be released as the calculated value. After the calculation is complete, it will go to the Output Layer. All processed data will be summarized in this layer, which will be a prediction of the results learned from the input data, such as image classification or prediction, etc. (Hao et al., 2023) to the equation for calculating the ANN algorithm. The calculation equations in the ANN algorithm, according to equations 2-24, are as follows.

$$z = \sum_{i=1}^n w_i \bullet x_i + b \tag{2-24}$$

The variables are defined as follows: z is the weighted sum of the delays, x_i is the input, i

w_i is the weight value associated with the input, i

b is a constant,

n is the number of inputs.

Since the ANN calculates the activation function in the Hidden Layers, resulting in Equation 2-25, the last equation in the calculation of the ANN. is calculated as follows:

$$a_j^l = a_j^l \left(\sum_{i=1}^{n_{l-1}} (a_i^{l-1} w_{ij}) + b_j^l \right) \quad (2-25)$$

Where a_j^l represents the output of the j th node in layer l a_i^{l-1} represents the output of the i th node in the layer $l-1$ σ is the activation function w_{ij} and b_j^l are weight and bias. As shown in Fig. 2, we utilize the ANN model with 1 input layer, 3 hidden layers, and 1 output layer. The activation function is selected as the ReLU function (X. Huang et al., 2020)

2.6.2 ANN example

Starting with the input data, assuming that the input data x_1 and x_2 have the following values:

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.8 \end{bmatrix} \quad (2-26)$$

Then, the weight values in the calculation of the hidden layer are as follows:

$$w^{(1)} = \begin{bmatrix} 0.2 & 0.4 \\ 0.7 & 0.3 \\ 0.6 & 0.5 \end{bmatrix} \quad (2-27)$$

The biased values are:

$$b^{(1)} = \begin{bmatrix} 0.1 \\ 0.2 \\ 0.3 \end{bmatrix} \quad (2-28)$$

Calculate the weighted joint value for the delay values in the hidden layer.

$$z^{(1)} = w^{(1)} \cdot x + b^{(1)} \quad (2-29)$$

$$z^{(1)} = \begin{bmatrix} 0.2 & 0.4 \\ 0.7 & 0.3 \\ 0.6 & 0.5 \end{bmatrix} \cdot \begin{bmatrix} 0.5 \\ 0.8 \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0.2 \\ 0.3 \end{bmatrix}$$

$$z^{(1)} = \begin{bmatrix} 0.5 \\ 0.66 \\ 0.94 \end{bmatrix}$$
(2-30)

Then, continue the calculation using the ReLU function as the activation function.

$$a^{(1)} = \sigma(z^{(1)}) = \text{ReLU}(z^{(1)})$$
(2-31)

$$a^{(1)} = \begin{bmatrix} \max(0, 0.5) \\ \max(0, 0.66) \\ \max(0, 0.94) \end{bmatrix}$$

$$a^{(1)} = \begin{bmatrix} 0.5 \\ 0.66 \\ 0.94 \end{bmatrix}$$
(2-32)

Then, before reaching the output layer, the weight values are set to be used in the final calculation, which are:

$$w^{(2)} = [0.3 \quad 0.4 \quad 0.5]$$
(2-33)

And the biased values are:

$$b^{(2)} = 0.1$$
(2-34)

Calculate the weighted joint value for the delay values in the output layer.

$$z^{(2)} = w^{(2)} \cdot a^{(1)} + b^{(2)}$$
(2-35)

$$z^{(2)} = [0.3 \quad 0.4 \quad 0.5] \cdot \begin{bmatrix} 0.5 \\ 0.66 \\ 0.94 \end{bmatrix} + 0.1$$
(2-36)

$$z^{(2)} = 1.107$$

Then, continue the calculation using the Sigmoid function as the activation function

$$\hat{y} = \sigma(z^{(2)}) = \frac{1}{1 + e^{-z^{(2)}}} \quad (2-37)$$

$$\hat{y} = \frac{1}{1 + e^{-1.107}} \quad (2-38)$$

$$\hat{y} \approx 0.7515$$

The final answer of the ANN calculation result is approximately 0.7515. (Zhang, 2000) Next, the working principle explanation according to the ANN flowchart in Figure 2-8 shows the working principle of the ANN algorithm process As shown in Figures 2–12.

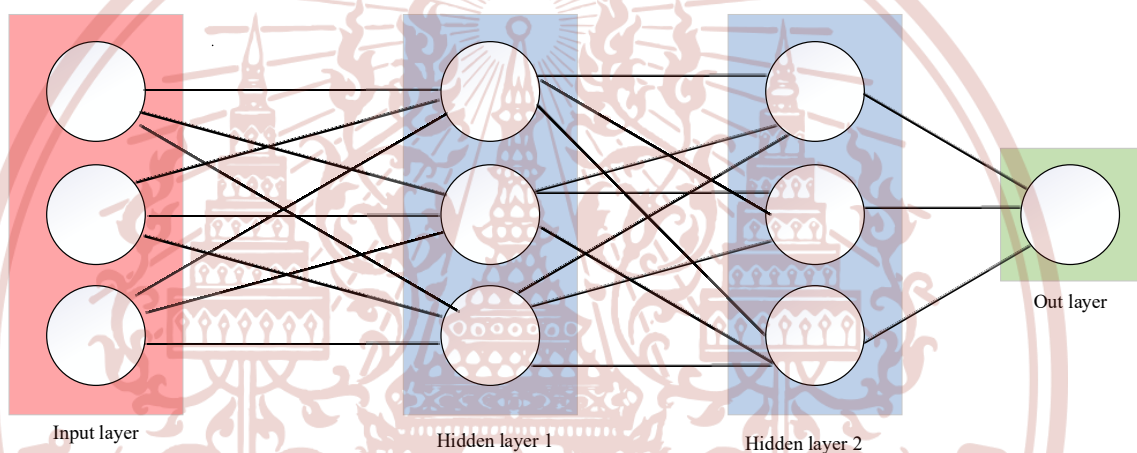


Figure 2-13 A basic description of the parts of an ANN diagram (Johnsson et al., 2020)

Input Layer: The input layer receives the initial data or features. Each node in this layer represents one feature of the data input.

Hidden Layers: Hidden layers are intermediate layers between the input and output layers. They perform computations on the input data using weighted connections.

Output Layer: The output layer produces the final output of the network. The number of nodes in this layer depends on the type of problem the network is solving (e.g., regression, classification)(Medium, 2024)

2.7 Recurrent neural network (RNN)

Recurrent neural networks are feed-forward neural networks. It organizes the forecast information systematically and helps to manage the information accordingly when we send the input values. Both the past and the current work of the network are entered into the system. Recurrent neural networks analyze data systematically and process the results into values that are outputs of the network system. For example, our system arranges our eating in order of rice, pork, and water. If we eat rice with us, the recurrent neural networks predict that next, we will drink water. Or if we eat pork and water, recurrent neural networks predict that we will eat that rice ourselves.

(ArunKumar et al., 2021) .” RNNs work under the assumption that they can predict a layer’s output by saving and feeding that layer’s output back into the input. The most important property of the RNN is its hidden state, which considers sequencerelated input.”(Roy and Tanveer, 2023) As shown in Figures 2–13.

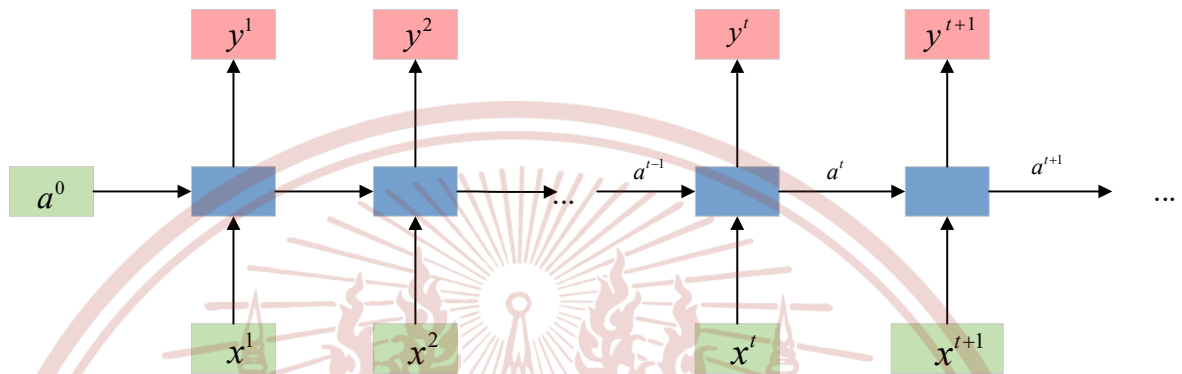


Figure 2-14 RNN equation process (Aishwarya, 2024)

Figure 2-13 shows the basic structure of a Recurrent Neural Network (RNN), which consists of several components that represent the internal working process of RNN as follows: Starting with Node: The blue node in the middle of the image represents the hidden state (a) At each time step, the green node at the bottom represents the input (x) and the pink node at the top represents the output (y)

2.7.1 Linking Principles:

The horizontal arrow line connecting the blue nodes represents the transmission of the hidden state (a) from time t to t+1.

The vertical arrow line connecting the green node to the blue node represents the transition of the input (x) into the hidden state at each time period.

The vertical arrow line from the blue node to the pink node represents the output (y) that is calculated from the hidden state at each time step. (Aishwarya, 2024)

Meaning of Each Variable:

a^0 is the initial hidden state.

$x^1, x^2, x^t, \dots, x^{t+1}$ are the inputs at each time step.

$y^1, y^2, y^t, \dots, y^{t+1}$ are the outputs generated from the hidden state at each time step.

a^{t-1}, a^t, a^{t+1} are the hidden states at each time step. The hidden state value is updated at each time step by the input at the same time and the previous hidden state then creates the output from the current hidden state.(Aishwarya, 2024)

For each timestep t, the activation a^t and the output y^t are expressed as follows:

$$\begin{aligned} a^t &= g_1(w_{aa}a^{t-1} + w_{ax}x^t + b_a) \\ y^t &= g_2(w_{ya}a^t + b_y) \end{aligned} \quad (2-39)$$

Where w_{ax}, w_{aa}, b_a, b_y are coefficients that are shared temporally and g1, and g2 activation functions, as shown in Figures 2–14.

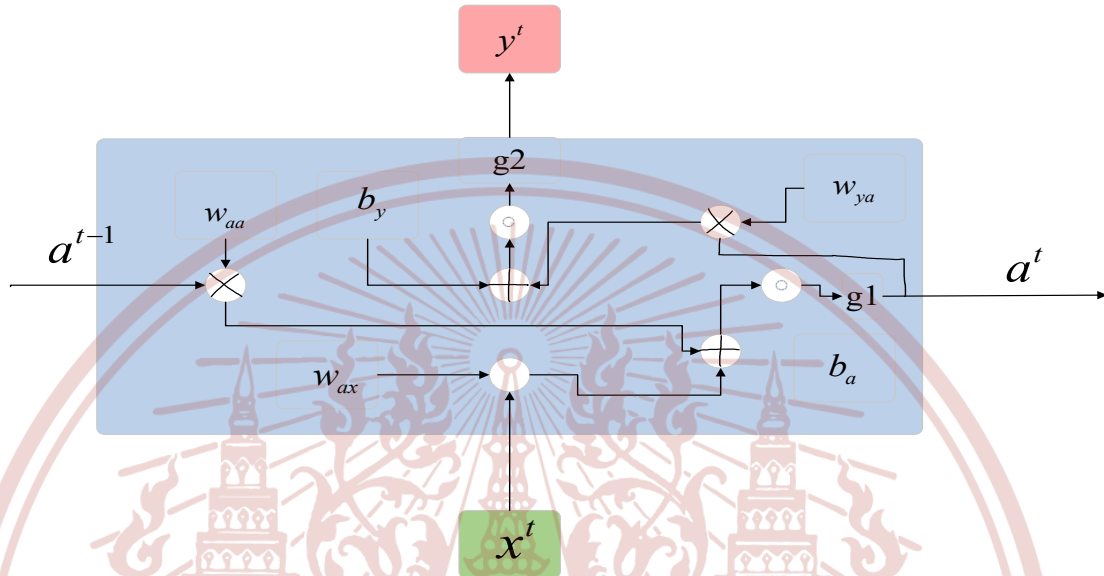


Figure 2-15 RNN process(Aishwarya, 2024)

Figure 2-14 shows the basic structure of an RNN, similar to the characteristics of Figure 2-8, but with additional details regarding weight calculations.

Weights and Biases:(Aishwarya, 2024)

x^t Input data at time period t

a^{t-1} The hidden state from the previous time period t-1

Weight and Bias Variables:

w_{aa} Weight matrix for the hidden state a^{t-1}

w_{ax} Weight matrix for the input data x^t

w_{ay} Weight matrix for the output data y^t

b_a Bias for the hidden states.

b_y Bias for the output.

2.7.2 Operation:

The previous hidden state a^{t-1} is multiplied by the weight matrix w_{aa} . The input x^t is multiplied by the weight matrix w_{ax} . The results of these multiplications are added to the bias b_a and passed through the activation function (usually a non-linear function such as tanh or ReLU) to create a new hidden state a^t . The new hidden state a^t is multiplied by the weight matrix w_{ay} and added to the bias b_y then passed through the activation function (or sometimes directly) to produce the output y^t (Aishwarya, 2024)

2.7.3 A trivial example

Take the word “dogs,” where we want to train an RNN to predict the letter “s” given the letters “d”-“o”-“g”. The architecture above would look like the following:

$$d = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} o = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} g = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} s = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (2-40)$$

From equation (2-26), the input values of the equations d , o , g , and s show the working principle diagram of a Recurrent Neural Network (RNN) as illustrated As shown in Figures 2–15.

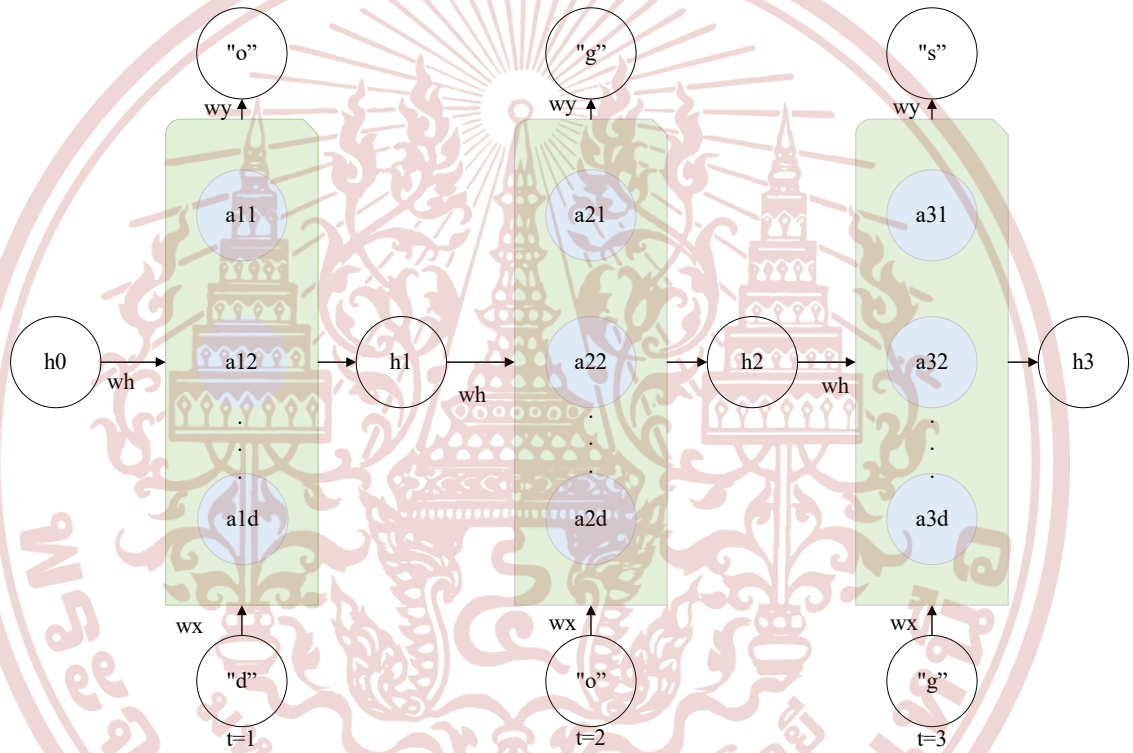


Figure 2-16 RNN example(Aishwarya, 2024)

Figure 2-15 shows the details of the steps of a recurrent neural network (RNN), starting from input: d , o , and g are the inputs in each time sequence ($t = 1, 2, 3$) and there are also hidden states: States h_0, h_1, h_2 , and h_3 represent the hidden states in each step of the processing, and the connection weights: w_x, w_h , and w_y represent the weights of the connections between the different states. Output: o, g, s is the predicted result from each hidden state. The input in each time sequence is passed to the hidden state of the model. The hidden state will be calculated according to the formula 2-25. When the calculation is complete, the obtained results and the previous state will be calculated according to the hidden state, and the connection weights will be the result in the output.(Aishwarya, 2024) this example, we'll use 3 hidden nodes in our RNN ($d=3$). The dimensions for each of our variables are As shown in Figures 2–16.

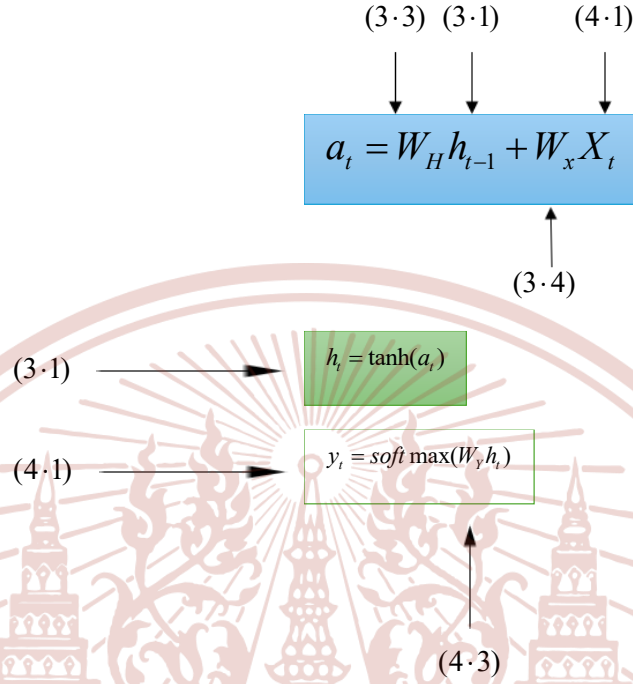


Figure 2-17 RNN equation example(Aishwarya, 2024)

Figure 2-17 shows where $k = 4$, because our input x is a 4-dimensional one-hot vector for the letters in “dogs.”(G. Chen, 2016)

2.8 Long Short-Term Memory Networks (LSTM)

Long Short-Term Memory (LSTM) is a neural network with memory capabilities that is suitable for processing and predicting important events. It is designed to be more effective than recurrent neural networks (RNN).(He, 2023) conclusion: long short-term memory (LSTM) belongs to recurrent neural networks (RNN) and is better because long short-term memory (LSTM) can remember important and similar previous information (Roy and Tanveer, 2023) ignore the information. They have better long-term memory and may be able to predict data better than recurrent neural networks (RNN).(C. Jin, 2022) The Long Short-Term Memory (LSTM) cell is a type of recurrent neural network (RNN) unit that is capable of learning long-term dependencies. The LSTM unit includes several gates (input, forget, and output gates) that control the flow of information into and out of the cell, allowing it to selectively remember or forget information over time. The equations for an LSTM unit are as follows:

Forget gate:

$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] + b_f) \quad (2-41)$$

Input gate:

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_i) \quad (2-42)$$

Candidate value:

$$\tilde{c}_t = \tanh(w_c \cdot [h_{t-1}, x_t] + b_c) \quad (2-43)$$

Cell state:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (2-44)$$

Output gate:

$$o_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o) \quad (2-45)$$

Hidden state:

$$h_t = o_t \odot \tanh(c_t) \quad (2-46)$$

In these equations:

f_t is the forget gate output at time step t , which decides how much of the previous cell state c_t to forget.

i_t is the input gate output at time step t , which decides how much of the new candidate value $\tilde{c}_t$ to incorporate into the cell state.

$\tilde{c}_t$ is the candidate value at time step t , which represents the new information that could be added to the cell state.

c_t is the cell state at time step t , which represents the memory of the lstm unit.

o_t is the output gate output at time step t , which decides how much of the cell state c_t to output.

h_t is the hidden state/output of the LSTM unit at time step t , which is based on the cell state c_t and is also the output of the LSTM unit at that time step.

2.8.1 Long Short-Term Memory Networks (LSTM) Example

The W matrices and b vectors are the weights and biases of the LSTM unit, respectively, which are learned during the training process. The σ the function represents the sigmoid activation function, and $\odot$ represents element-wise multiplication. (Hochreiter and Schmidhuber, 1997) input Data: Suppose we have the following sentence: "The cat sat on the mat." Tokenization: First, we tokenize this sentence into individual words: ["The", "cat", "sat", "on", "the", "mat"]. Word Embedding: We represent each word as a vector using techniques like Word2Vec or GloVe. For simplicity, let's say our embedding for each word is a 3-dimensional vector.

"The" => [0.1, 0.2, 0.3]

"cat" => [0.4, 0.5, 0.6]

"sat" => [0.7, 0.8, 0.9]

"on" => [0.2, 0.3, 0.4]

"the" => [0.1, 0.2, 0.3]
 "mat" => [0.5, 0.6, 0.7]

Sequence Preparation: We create input sequences and corresponding output labels. For example, if we set the sequence length to 3, our input-output pairs would be:

Input: [0.1, 0.2, 0.3], Output: "cat"
 Input: [0.4, 0.5, 0.6], Output: "sat"
 Input: [0.7, 0.8, 0.9], Output: "on"
 Input: [0.2, 0.3, 0.4], Output: "the"
 Input: [0.1, 0.2, 0.3], Output: "mat"

LSTM Architecture: We define an LSTM network with input size 3 (for the 3-dimensional word vectors) and output size equal to the vocabulary size (number of unique words). We can add additional layers as needed. Training: We train the LSTM network using the input-output pairs generated in step 4. During training, the LSTM learns to predict the next word in a sequence based on the previous words. Prediction: Once the LSTM is trained, we can use it to predict the next word in a given sequence. For example, if we provide the input [0.2, 0.3, 0.4], the LSTM might predict the word "the". This example demonstrates how an LSTM network can learn to capture dependencies within sequential data, making it well-suited for tasks like language modeling, time series prediction, and more. (ArunKumar et al., 2021)

2.9 Errors analysis

Errors analysis is the finding of indicators for evaluating models or various techniques in which we use the errors of the predictors, which use the values 1Mean absolute error (MAE), Root mean squared error (RMSE), and Mean Absolute Percentage Error (MAPE). In the evaluation, because it is a technique commonly used to evaluate forecast errors the accuracy of the evaluation can be calculated according to the technique.

2.9.1 Mean absolute error (MAE)

Before knowing the MAE value, know the actual demand value first. Actual Demand first is represented by y_i of that product, and The predicted value of demand (Forecast Demand) is represented by $\hat{y}_i$ of that product obtained from the model. In this case, we will determine the error value of the model by specifying the error value. (Buzna et al., 2021)

$$e_i = y_i - \hat{y}_i \quad (2-47)$$

If the predicted value is higher than the actual value, the error is positive. If the actual value is higher than the predicted value, the error is negative. It is another indicator that is highly popular. It can be calculated. By averaging the absolute value of the error (Absolute Error) as shown in the equation.

$$\text{The formula is } MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad \text{Mean absolute error} \quad (2-48)$$

The variables are defined as follows:

y_i is the actual value.

$\hat{y}_i$ is the predicted value.

n is the number of observations.

(MAE) is less sensitive to outliers compared to MSE because it doesn't involve squaring. (Shrinvisan et al., 2023)

MAE example

An example of how to calculate the Mean Absolute Error (MAE) Suppose we have the following actual and predicted values: As shown in Table 2-1.

Table 2-1 Table of sum example of the MAE

Actual (y)	Predicted ($\hat{y}$)
3	2
7	5
2	4
5	6

Calculate the absolute errors:

$$\begin{aligned}
 |3 - 2| &= 1 \\
 |7 - 5| &= 2 \\
 |2 - 4| &= 2 \\
 |5 - 6| &= 1
 \end{aligned}
 \tag{2-49}$$

Sum the absolute errors:

$$1 + 2 + 2 + 1 = 6 \tag{2-50}$$

Divided by the number of observations ($n=4$)

$$MAE = \frac{6}{4} = 1.5 \tag{2-51}$$

So, the MAE is 1.5

2.9.2 Root mean squared error (RMSE)

RMSE is simply the square root of MSE. The formula is $RMSE = \sqrt{MSE}$ It is in the same unit as the dependent variable, making it easier to interpret. (Shrinvisan et al., 2023) Key Points All three metrics provide a measure of the average magnitude of errors between predicted and actual values. MSE penalizes larger errors more heavily due to squaring. MAE is more robust to outliers as it uses absolute differences. RMSE is the square root of MSE and provides a measure in the same unit as the dependent variable. When choosing between these metrics, it often depends on the specific

characteristics of the problem and the importance of different types of errors in the context of the application.(Shrinvisan et al., 2023) The RMSE value is very popular for training purposes. And select a model Since the formula is quite easy to find the derivative, it makes adjusting the model. It can be done more naturally. And is a measure used in the competition for many Regression-type problems on the Kaggle platform. MSE equation is

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2-52)$$

RMSE example how to calculate the Root mean squared error (RMSE) Suppose we have the following actual and predicted values: As shown in Table 2-2.

Table 2-2 Table of sum example of the Root mean squared error (RMSE)

Actual (y)	Predicted ($\hat{y}$)
3	2
7	5
2	4
5	6

Calculate the squared errors:

$$\begin{aligned} (3-2)^2 &= 1 \\ (7-5)^2 &= 4 \\ (2-4)^2 &= 4 \\ (5-6)^2 &= 1 \end{aligned} \quad (2-53)$$

Sum the squared errors

$$1+4+4+1=10 \quad (2-54)$$

Divide by the number of observations (n=4)

$$RMSE = \frac{10}{4} = 2.5 \quad (2-55)$$

Take the square root of the result

$$RMSE = \sqrt{2.5} \approx 1.5811 \quad (2-56)$$

So, the RMSE is 1.5811

2.9.3 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error can be found by averaging the average percentage error values as shown in the equation.(Rashid et al., 2023)

$$MAPE = \frac{100}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (2-57)$$

The variables are defined as follows:

y_i is the actual value.

$\hat{y}_i$ is the predicted value.

n is the number of observations.

2.9.4 MAPE example

An example of how to calculate the Mean Absolute Percentage Error (MAPE) Suppose we have the following actual and predicted values: As shown in Table 2-3.

Table 2-3 Table of example the Mean Absolute Percentage Error (MAPE)

Actual (y)	Predicted ($\hat{y}$)
3	2
7	5
2	4
5	6

Calculate the absolute percentage of errors:

$$\begin{aligned} \left| \frac{3-2}{3} \right| \times 100 &= 33.33\% \\ \left| \frac{7-5}{7} \right| \times 100 &= 28.57\% \\ \left| \frac{2-4}{2} \right| \times 100 &= 100\% \\ \left| \frac{5-6}{5} \right| \times 100 &= 20\% \end{aligned} \quad (2-58)$$

Sum the absolute percentage errors:

$$33.33 + 28.57 + 100 + 20 = 181.9 \quad (2-59)$$

Divide by the number of observations ($n=4$)

$$MAPE = \frac{181.9}{4} = 45.475\% \quad (2-60)$$

So, the MAPE is 45.475% Mean Absolute Percentage Error uses percentage reporting to understand the level of accuracy. Regardless of the size of what is being predicted.

(scale-independent) makes reporting easy for stakeholders to understand and can make decisions quickly (Kumar et al., 2023) Summary Mean Absolute Percentage Error has Divisor closest to 0: In case the actual value (Actual Demand) is 0, the value cannot be found. In the case where the value is close to 0, it will cause the expected value to fluctuate. Has an abnormally high value although Asymmetric Formula: In the case where the predicted value is Less than the actual value, the Absolute Percentage Error will be 0-100%, but in cases where the predicted value is higher than the actual value, the Absolute Percentage Error has a chance of being higher than 100%. Therefore, using this value To choose a model There will be a tendency to get models that predict lower-than-normal values. (Alkhafaij et al., 2023)

2.10 Literature Review

There is a wide variety of research related to future predictions. This is similar to the research that uses an algorithm for optimizing a residual revised gray model using a genetic algorithm (GM-GA) in many studies, as follows: As shown in Table 2-4.

Table 2-4 shows related literature review research.

Research	Forecasting techniques					
	LSTM	RNN	ANN	GM	GA	GM-GA
Alkhafaij et al. (2023)	✓	✓				
B. Liu et al. (2023)	✓					
Rashid et al. (2023)				✓		
Shrinvisan et al. (2023)				✓		
T. Zhang et al. (2023)				✓		
Ding and Li (2021)					✓	
Kayacan et al. (2010)	✓			✓		
Xie and Xie (2009b)				✓		✓
Jia et al. (2008)				✓		✓
Kluabwang (2018)				✓		
Chenchen et al. (2023)				✓		
G. Jin and Yang (2023)				✓		
Y. Chen (2023)				✓		
L. Huang et al. (2023)				✓		
Pu et al. (2022)				✓		
Y. Zhou et al. (2022)				✓		
Pan (2022)				✓		
S. Li (2022)				✓		

Table 2-4 (Continued) shows related literature review research

Research	Forecasting techniques					
	LSTM	RNN	ANN	GM	GA	GM-GA
R. Zhang et al. (2023)	✓			✓		
Yang et al. (2023)	✓			✓		
J. J. Zhou (2013)				✓		
J. Li et al. (2021)				✓		
Guo et al. (2012)					✓	
Zhao et al. (2015)					✓	
Lee and Tong (2011)				✓		
Sapnken et al. (2022)				✓		✓
Sapnken et al. (2023)				✓		
Yao et al. (2022)				✓	✓	✓
X. Wang (2011)				✓		
X. Liu et al. (2014)				✓		
Yuhong Wang et al. (2010)				✓		
Ou (2012)				✓		✓
Niu et al. (2008)				✓		
S. H. Wang et al. (2007)				✓		
Kumar et al. (2023)	✓					
B. Liu et al. (2023)	✓					
Rashid et al. (2023)	✓	✓	✓			
Shrinvisan et al. (2023)	✓					
T. Zhang et al. (2023)	✓					
Ding and Li (2021)				✓		
Kayacan et al. (2010)				✓		
Z. Wu and Li (2019)				✓		
L. Wu et al. (2021)				✓		
He (2023)	✓					
Kumar et al. (2023)				✓		
He (2023)	✓					
Y. Wang (2014)					✓	
Buzna et al. (2021)		✓				
Summary	12	3	1	34	5	5

Researchers have studied various algorithms used for forecasting and/or related to forecasting through journals of various databases. They found that 7 algorithms are commonly used for forecasting: recurrent neural network (RNN), long short-term memory (LSTM), artificial neural network (ANN), convolutional neural network

(CNN), grey model (GM), and genetic algorithm (GA). In addition, researchers have reviewed a large number of academic journals, magazines, publications, and videos. Including data sources from ACM Digital Library, Business Source Complete, IEEE / IET Electronic Library (IEL), ScienceDirect, SpringerLink – Journal, and many others. It was found that many of these algorithms are often used in forecasting because they are popular algorithms for forecasting, consisting of LSTM 11 times, RNN 3 times, ANN 1 times, CNN 3 times, GM 35 times, GA 5 times, and finally GM-GA 6 times. From studying various data sources and journals, it appears that the GM model is used in forecasting the most at 35 times, followed by LSTM 11 times and GM-GA 6 times, followed by the remaining numbers mentioned above. However, the details and limitations of each model have been summarized by the researchers, and the process of proceeding is detailed below.

2.10.1 Long Short-Term Memory (LSTM)

Long short-term memory (LSTM) is a type of recurrent neural network (RNN) designed to overcome the vanishing gradient problem and record long-term dependencies. LSTM cells contain a memory unit that can store and retrieve information over extended periods. (Van Houdt et al., 2020) It belongs to the category of deep learning and is designed to recognize patterns over long periods. It is effective for sequential prediction problems because it can collect previous data and use it in processing. It can solve the long-term dependency problem that traditional RNNs face. Challenges in Long-range Dependency and Vanishing Gradient Problems LSTM is designed to recognize long-term information using uniquely designed gating mechanisms with great flexibility. LSTM can model both long-term and short-term temporal sequences. (Van Houdt et al., 2020) The working principle of (LSTM) consists of 3 main important steps: Forget Gate, Input Gate, and Output Gate. Starting with Forget Gate, it is responsible for determining information in cell whether to store data or discard data The collected data is imported into that node and further calculated. Next, the Input Gate receives new input data and then saves or writes the data to each node. The work is divided into 2 parts. (He, 2023) The first part is responsible for receiving data to select data as a data controller or data definer. The second part is responsible for receiving data to update the data create functions and import data. The last one, the Output Gate, is responsible for controlling data when it is issued. Let's get it neat and get the results that we want. (Alkhafaj et al., 2023). However, LSTM has revolutionized time series forecasting by providing a robust method for modeling uncertain data, demonstrating its versatility and effectiveness. (Sherstinsky, 2020) Various applications such as Financial Time Series LSTM networks are widely used to forecast financial time series or LSTM is also used to predict energy consumption. The ability to handle multiple seasonal patterns and nonlinearities makes it suitable for this domain. This has shown promise in its agility and effectiveness, however, continued research is needed to address existing challenges and further improve efficiency. (Feng et al., 2021)

Alkhafaj et al. (2023) mentions that LSTM is one of the deep learning (DL) approaches. A research study was conducted on Hybrid Metaheuristics with Sparse-Trained Deep Learning for Sustainable Electric Vehicle Charging Demand Forecasting using the HMSDL-EVCDF model. The forecast using LSTM was compared to evaluate accuracy and find the least error.

2.10.2 Recurrent Neural Network (RNN)

Recurrent Neural Networks (RNN) is an Artificial Neural Network (ANN) architecture designed to solve problems for processing sequential data such as text, audio, or time series data() in the forecasting of (RNN) as this method is particularly effective for tasks with continuous data and relationships between past and future data points, such as cost forecasting. (Aishwarya, 2024)Sales forecast, weather forecast, etc.(Buzna et al., 2021)An artificial neural network with a unique structure that can remember past data and use that data to predict future values. RNNs have internal processing links called "hidden states" used to recognize knowledge or previous information. Let's combine it with the next data to match the new data to use in the next forecast.(Rashid et al., 2023) RNN is used in forecasting. There are the following working principles. Sequential data processing: RNN processes data step by step. Each step receives input and after that, it goes into a "hidden state" after which it stores information about all previous sequences. If the data has a problem, it will be improved by looping, depending on how long the problem is in each data. When the desired data is obtained, the final result will be displayed, making RNN able to capture sequential and dependent data. with the context that occurred before well and is suitable for processing continuous data such as text and time data(Aishwarya, 2024). However, there is a large amount of long data that is not suitable for this algorithm because it takes a long time and is difficult to sort. Most of the time, if there is a lot of sorted data, the LSTM algorithm will be used to help.(Alkhafaj et al., 2023)

Tsai et al. (2018) states that RNN is a step-by-step process of processing data results as the main basis for making predictions. A research study was conducted on air pollution forecasting using RNN with LSTM, creating data sets in each set to connect them to each other in stages to create relationships and forecast values. PM2.5 concentration can be obtained by using a time series sequence data set to forecast. The method was used to compare accuracy and to find the least amount of error, as this research found other better techniques.

2.10.3 Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) are one of the techniques of data mining. It is a mathematical model that simulates an artificial neural network, which is a computer system designed to imitate the work of the human brain. (ANN) is used for solving complex problems that are not easily solved with conventional mathematical methods, such as image recognition, speech recognition, and predicting the behavior of time-series data (Hao et al., 2023). The main components of an ANN include nodes, or neurons. It acts like neurons in the human brain. Each neuron receives input data, processes the data, and then sends out the data to the next neuron. Layers: (ANN) is divided into layers, including the input layer, hidden layers, and output layer (weights): Each link between neurons has a weight that indicates the importance of that link and the final component. Activation Functions: Used to process input data into an appropriate format. For the next layer, such as the sigmoid, tanh, and ReLU functions (Gao and Yang, 2024), the working principle of ANN is that it takes data into the input layer and then sends it to the hidden layers for processing with neurons and activation functions. Gradually check the data. If it doesn't pass, it will be returned to the input layer. If it passes, the output layer will summarize the results and show the final results. (Rayudu and Roseline, 2023) ANN has been used in many fields, such as pattern recognition with uncertainty such as handwriting, signatures, letters, facial images,

natural language processing, and prediction tasks such as weather forecasts, stock forecasts, and others (Pornpatcharapong, 2008).

Roy and Tanveer (2023) mentions that ANN is one of the neural network approaches and a technique in data mining for forecasting. A research study was conducted on stock price forecasting using DeepNet, which was used to compare accuracy values and to find the least amount of error to analyze and compare stock prices within the stock market. Additionally, the results can be presented as stocks to be used in financial analysis.

2.10.4 Grey Model (GM)

Professor Julong Deng developed the Grey Model (GM), a quantitative forecasting method, at the beginning of the decade. 1980 (Ding and Li, 2021) Originally, GM was part of the Grey System Theory. GM's goal It is a mathematical forecasting method that is used to predict the future. Especially in cases where there is little data or high uncertainty, GM is used in many different forecasts, such as regional business planning for many provinces in the country and forecasts of crop seed yields for certain provinces in the country, which can be used to analyze the agricultural economy in the country and irrigation planning (Kayacan et al., 2010). GM is the most commonly used. is GM (1,1), which has a relatively simple structure and steps before starting the GM (n,m) process. Starting from (n,m), n is the sequence of difference equations, and m is the number of variables. The steps involved in GM (n, m) are: Grey forecasting, or steps (GM(1,1)), can be done with the following steps (Z. Wu and Li, 2019): Generation of accumulated data. (AGO): Accumulate the data that needs to be forecast in order to make the data more smooth. Later, creating a gray differential equation (GDE) is creating a differential equation. gray pattern to use it to find the steps. Parameter estimation Create parameters using the least squares method to estimate parameters and perform further calculations. Then create a model (GM) to find the final forecast results. (Geng et al., 2021) The original data forecasting step (IAGO) takes the data from the model and returns the accumulated data to the original data to use in forecasting to get the final results. That's it (L. Wu et al., 2021)

Xie and Xie (2009b) states that the gray model (GM) is one of the deep learning (DL) approaches. A research study was conducted on forecasting the total volume of import-export trade based on gray modeling optimized by a genetic algorithm. The GM model was used to forecast the amount of import-export trade using data from 1989 to 2004 to find the values from 2005 to 2007. The GM was combined with a genetic algorithm (GA) to help in analyzing and forecasting the import and export value data.

S. H. Wang et al. (2007) mentions that the grey model (GM) is one type of forecasting method. The research on medium-long term load forecasting based on the improved gray model was studied by using the GM model for forecasting. The process and steps of creating the GM model were examined and compared with the optimized GM model to find the accuracy and the best value in this forecast.

Jia et al. (2008) mentions that the grey model (GM) is one type of forecasting method. A research study was conducted on "An improved GM(1,1)-genetic algorithm for short-term forecasting in power systems" by studying the process and steps of creating the GM model and using it for forecasting. The study presents the GM(1,1) model and the genetic algorithm (GA) in the forecast, using the improved genetic algorithm (IGA) as the forecasting tool. The goal was to find the accuracy and determine the best value in this forecast.

Xie and Xie (2009b) mentions that the grey model (GM) is one type of forecasting method. A research study was conducted on "Forecast of regional gross national product based on grey modeling optimized by genetic algorithm" by studying the forecast using the GM model to predict GDP values. The study examined the process and steps of creating the GM and genetic algorithm (GA) models. The GM(1,1) model, optimized by GA (GM-OGA), was used for forecasting. The study aimed to find the accuracy and value of the forecast, using GDP data from 1989 to 2004 to forecast values for 2005-2007 and compare their accuracy. The results clearly show that GM-OGA provides the highest accuracy values.

Niu et al. (2008) mentions that the grey model (GM) is one type of forecasting method. A research study was conducted on "Forecast of regional gross national product based on grey modeling optimized by genetic algorithm," by studying forecasting using the GM model and examining the process and steps of modeling with the GM and genetic algorithm (GA). GA is presented in the forecasting process using IGA as a method for forecasting with algorithms. The study used GM to assist in finding the results of the forecast and determining the accuracy of the forecast.

Yu and Zhao (2017) mentions that the grey model (GM) is one type of forecasting method. We have studied the research on "A combination forecasting model of extreme learning machine based on genetic algorithm optimization." We studied forecasting using the GM model and finding the accuracy value, as this research uses the genetic algorithm (GA) as the principle for predicting values. The GM model was used to compare forecast values and accuracy values to find the best value.

Kluabwang (2018) mentions that the grey model (GM) is one type of forecasting method. A research study was conducted on "Electricity consumption forecasting in Thailand using grey prediction models." They studied forecasting using the GM model and evaluated its accuracy in forecasting the amount of electricity consumption within Thailand. The GM model was used to analyze and forecast electricity consumption in Thailand. Additionally, another algorithm, Tune-GM, was used for forecasting and comparing the forecast values and measuring accuracy. It was concluded that forecasting with Tune-GM is better than the traditional GM.

Lee and Tong (2011) mentions that the grey model (GM) is one type of forecasting method. We have studied the research on forecasting energy consumption using a gray model improved by incorporating genetic programming. We have studied the GM model in forecasting and found it to be accurate in forecasting internal electricity consumption. China can use GM to analyze and forecast electricity consumption within Thailand. Another algorithm used for forecasting is genetic programming (GP), and GP-GM was also used to forecast the amount of electricity within China from 1990 to 2003. By comparing the forecast values and measuring the accuracy from 2004 to 2007, it was concluded that forecasting with GP-GM was better than traditional GM because it has higher accuracy and less error.

Yao et al. (2022) mentions that the grey model (GM) is one type of forecasting method. We have studied the research on forecasting energy consumption using a grey model improved by incorporating genetic programming. We have studied forecasting using the GM model and evaluated its accuracy. However, in this research, not only conventional GM was used; GM was also combined with neural networks (NN) to improve predictions. Additionally, a genetic algorithm (GA) was applied to the GM and NN models to further enhance prediction accuracy. The study concluded that the

GA-GM-NN model provided the best predictive value, demonstrating the highest accuracy and the least error.

Ou (2012) mentions that the grey model (GM) is one type of forecasting method. A study has been conducted on forecasting agricultural output with an improved grey forecasting model based on the genetic algorithm. The research involved using the GM model for forecasting and evaluating its accuracy. However, this research not only employed the traditional GM model but also utilized an improved GM(1,1) model and a genetic algorithm (GA) to enhance the predictions. Specifically, the GA-Improved GM (GAIGM) was used as the basis for forecasting agricultural yields from 1999 until 2010. The obtained prediction and accuracy values demonstrated that GAIGM provided the best accuracy and the least amount of error.

2.10.5 Genetic Algorithms (GA)

Genetic algorithms (GA) were invented by John Holland in 1975 as a technique for finding optimization solutions. (optimization) inspired by the principles of natural evolution using the principles of gene and chromosome selection. (Y. Wang, 2014) In order to find the genes or chromosomes of the best survivors and be able to have the most efficient offspring, the use of genetics (GA) in prediction is a process that takes concepts from natural selection, and genetics is used to find the best answer to a forecasting problem. GA is often used to optimize the parameters of a forecasting model. The working principle of the genetic algorithm (GA) consists of the following main steps: (Yao et al., 2022) Coding: Substitute values to find the parameters of the prediction model in the form of chromosomes (chromosomes), where each chromosome consists of genes (genes) that Substituting various parameters into the initial population model creates an initial random set of chromosomes, which are represented by different parameter values. Fitness Evaluation evaluates the ability of each chromosome using benchmark functions (e.g., MAPE, RMSE, MAE) (Zhao et al., 2015) to measure the accuracy of the Selectio prediction model. Choose the chromosome with the highest fitness value to be the parent for the next generation of chromosomes. Crossover combines the parent chromosomes to create offspring (offspring) by mixing genes from the parents. Mutation changes the gene value of some chromosomes. Diversity in the next-generation population Replace the old chromosome with a newer, higher-performing chromosome. Iteration Repeat steps 3–7 several times until the best parameters are reached or until the specified number of cycles is reached to produce the final result. Come (Guo et al., 2012)

Xie and Xie (2009b) mentions that the gry model (GM) is one type of forecasting method. We have studied the research on the forecast of the total volume of import-export trade based on gray modeling optimized by a genetic algorithm. We have analyzed the forecasting using the GM model and evaluated its accuracy. However, this research did not just use the conventional GM model; it also employed a genetic algorithm (GA) to enhance the predictions. Moreover, the research used an optimized genetic algorithm (GAO) to find the accurate values for imported and exported goods. It can be concluded that GM-GAO provided the least amount of error and the highest accuracy.

Y. Wang (2014) mentions that the Genetic Algorithm (GA) is an algorithm that searches randomly, similar to genetic characteristics in natural selection. It is one of the approaches that can be used for prediction. This was explored in a research study on short-term wind power forecasting using a genetic algorithm of wavelet neural

networks. We studied the forecast by examining the principles of the GA model in forecasting and evaluating its accuracy. However, this research did not solely focus on the GA algorithm. Instead, it also utilized a Wavelet Neural Network (WNN) to aid in data prediction. The study combined GA and WNN to create the GA-WNN model for forecasting wind energy within China. The results concluded that the GA-WNN model has the highest accuracy and the least error in forecasting wind energy.

Yao et al. (2022) mentions that the Genetic Algorithm (GA) is one of the algorithms that search randomly, similar to genetic characteristics in natural selection. It is one of the approaches that can be used in research. This was explored in a research study on a combination forecasting method of gray neural network based on a genetic algorithm. We studied forecasting by examining the principles of the GA and gray model (GM) models and also incorporating a neural network (NN) for forecasting and finding accuracy values. However, in this research, we did not study just the GA algorithm alone. Instead, we used the neural network to help predict data, producing the GM-NN model. Additionally, the GA was used with GM and NN to create the GA-GM-NN model for forecasting elevator purchases. The results concluded that the GA-GM-NN model has the highest accuracy and the least error in forecasting values.

Zhao et al. (2015) mentions that the Genetic Algorithm (GA) is one of the algorithms that search randomly, similar to genetic characteristics in natural selection. It is one of the approaches that can be used in research. This was explored in a research study on system reliability prediction by support vector regression (SVR) with analytic selection and genetic algorithm parameter selection. We studied the prediction by examining the principles of the GA model and SVR, incorporating analytic selection to forecast and determine accuracy. However, this research did not solely focus on the GA algorithm alone. Instead, we used Grid-search to aid in data forecasting. The study did not use only GA; rather, it combined GA and SVR, resulting in the GA-SVR model for forecasting. GA was used to compare the accuracy and error values of the forecasts. It can be concluded that the GA-SVR model provides the highest accuracy and the least error in estimating wind energy values.

Guo et al. (2012) mentions that the Genetic Algorithm (GA) is one of the algorithms that search randomly, similar to genetic characteristics in natural selection. It is one of the approaches that can be used in research. This was explored in a research study on the topic of an improved support vector machine oil price forecast model based on genetic algorithm optimization parameters. The forecast was studied by examining the principles of the GA model and the support vector machine (SVM). In this research, we did not study just the GA algorithm alone but also used SVM to aid in data forecasting. The study combined GA and SVM, resulting in the GA-SVM model to forecast engine oil prices in China. GA was used to compare the accuracy and error of the predictions. It can be concluded that the GA-SVM model has the highest accuracy and the least error in predicting oil prices. Additionally, the cost of forecasting wind energy was the lowest.

In conclusion, after studying various research studies related to the gray model (GM) and genetic algorithm (GA) in various predictions, it is evident that this type of algorithm is widely used and applied in many ways. For instance, GM-IGA has been used to forecast the total volume of import-export trade, (Xie and Xie, 2009b) and GAIGM has been utilized for forecasting agricultural output with an improved gray forecasting model based on a genetic algorithm. Additionally, GA-SVM has been

applied to forecast fuel oil prices in China, (Ou, 2012) and an optimized GM model has been used for medium-long-term load forecasting(S. H. Wang et al., 2007)

Despite these advances, researchers have studied a variety of information sources and various sources of journals both domestically and internationally. No one has studied and invented a method to combine the optimized GM model with the genetic algorithm (GA) to find the forecast value, which is called RGM-GA, which is gap and the main issue in the research on the forecasting of the demand for electric vehicles this time. Therefore, the researchers named the invention for use in the forecast this time as R²GM-GA. This research uses R²GM-GA to forecast the demand for electric vehicles. The results aim to identify the links related to the approach to creating electric vehicle charging stations.

2.10.6 Check Answer

In this step, the technique is selected to check the answers from the forecast to be the most accurate or have the least error. The researcher has studied various techniques for checking the error or accuracy from various e-journals. There are examples of using techniques to check the answers from the forecast, summarized in Table 3-4, the table of techniques for checking the answers, consisting of the column Research, the name of the research that uses the checking technique, and the column Check Errors, which is the name of the technique used to check the answers from the forecast, according to Table 2-5, the table of techniques for checking the answers. As shown in Table 2-5 the table of techniques for checking the answers.

Table 2-5 The table of techniques for checking the answers.

Research	Check Errors		
	MAPE	MAE	RMSE
Draper et al. (2013)			✓
Ou (2012)	✓		✓
Lee and Tong (2011)	✓		
de Myttenaere et al. (2016)	✓		
Ding and Li (2021)	✓	✓	✓
Michańków et al. (2024)		✓	
Akanbi et al. (2024)			
Rashid et al. (2023)	✓	✓	
Olawumi et al. (2024)		✓	✓
Alkhafajj et al. (2023)		✓	
Jonathan et al. (2019)		✓	
Frías-Paredes et al. (2018)		✓	✓
Karunasingha (2022)		✓	✓
Kluabwang (2018)	✓		
Mendo (2009)		✓	
Gani et al. (2024)	✓		✓
Gao and Yang (2024)		✓	

Table 2-5 (Continued) The table of techniques for checking the answers.

Research	Check Errors		
	MAPE	MAE	RMSE
Tsai et al. (2018)		✓	✓
Chaaban and Alfadl (2024)		✓	✓
Aldosari (2024)	✓	✓	✓
TOTAL	8	13	10

Table 2-5 shows that the research that has been studied from many sources and journals, both domestically and internationally, found that many techniques were used to check the error value and the accuracy value of the forecast, including MAPE, MAE, RMSE, R2, and finally actual data value. After studying, it was found that all of the above models are widely used to check the forecast. Therefore, we collected data on the number of models used to check the forecast value and used neural network (NN) models because the researcher used the NN model for forecasting. It was found that MAE was used to check the forecast the most at 13 times, followed by RMSE and actual data value at 10 times, followed by MAPE 8 times, and finally R2 at 4 times. The reason for choosing these techniques to check the research in each journal is that the researcher has summarized and explained the working principles of the forecast-checking techniques as follows:

This research shows the impact on the assessment of underground moisture data by using the moisture values obtained to predict the moisture values in the next basement to expect the previous layer (Draper et al., 2013). The values obtained in this report will use Root mean squared error (RMSE) and Fractional root mean squared error (FRMSE) techniques to check the error value and the forecast results. The results show that the error value is quite low and accurate. Although RMSE is popularly used to check forecasts, FRMSE has a lower error value and is more accurate in forecasting the classification of underground moisture. However, RMSE is more suitable for forecasting with uncertain data (Draper et al., 2013).

This research shows the modeling of agricultural production forecasting by using the traditional GM (1,1) model prediction method, and in this research, a new model called GAIGM (1,1) is created for this forecasting, which uses agricultural production data from the year (1998-2008) to forecast agricultural production in the year (2009-2010) and checks the error value (Ou, 2012). This research chooses to use two techniques for checking: MAPE and RMSE. The researcher claims that the two techniques are used because these two techniques are the most appropriate for checking the error value of the forecast, and the results obtained in this research are that GAIGM (1,1) gives the best forecast value. The check from MAPE is 4.53% and RMSE is 2.67%, which are values of both models less than 10%. This research claims that This model is less than 10%, and it is like checking the real data value that GAIGM (1,1) gives the best forecast value. It can be concluded that GAIGM (1,1) gives the best forecast value, and MAPE and RMSE are the two models that are most suitable for checking the forecast error value (Ou, 2012).

In this research, the results of energy forecasting are shown by using the traditional GM (1,1) model and the GPGM model. The researcher has invented the

GPGM technique to forecast energy usage in this research, specifically (Lee and Tong, 2011). The data used in this forecast is the annual energy consumption data of China from 1990 to 2003 to forecast data from 2004 to 2007. In addition to the 2 techniques mentioned earlier, this research has prepared 2 techniques for forecasting, totaling 4 techniques: the traditional GM (1,1), HSU, Chen, Linear Regression, and GPGM. The results from these 4 models will be checked for error or accuracy. The researcher used 2 models to check: PE and MAPE. The results show that the HSU, Chen, and GPGM models have the least error at 20.23%, which is the same as the mape model. However, at the same time, The principle of measuring the value of mape, if the measured value is less than 20%, it is considered the most accurate measurement. But if it is more than 20%, it is considered inappropriate. In checking the accuracy value, the PE part measures the value obtained from the lowest value, which in this forecast, it appears that gpgm gives the best forecast value, at 2.59%, the lowest. When comparing the measured values from the actual data, it appears that GPGM gives the best forecast value. In conclusion, GPGM forecasts the best and PE is the most suitable for checking the forecast error value more than MAPE (Lee and Tong, 2011).

This research equation techniques and other techniques were ready to compare with MAPE to find out which one is better, namely the MSE technique because the researcher claimed that this technique is more suitable for checking the forecast with various regression equations such as Regression model, Complexity control and Kernel regression, Quate regression and finally Primal problem (de Myttenaere et al., 2016). After checking the truth value as a percentage, it was concluded that the results of the various model values in the forecast were checked. MAPE was more accurate than MSE while MSE can still be used to check the forecast because the values in each model were checked and not much different from the values checked by MAPE, almost 1%. Each model can be checked (de Myttenaere et al., 2016).

In this research, the solar energy forecast uses a model to forecast the sunlight. In this research, 5 models are used in forecasting: LR, RFR, KNN, ANN, and LSTM. When the forecast is complete, 3 techniques are used to check the forecast results, the error value, and the accuracy value: RMSE, MAE, and R2 score (Chaaban and Alfadl, 2024). The reason for using 3 techniques is that the researcher claims that these 3 techniques are the most suitable for checking the error value and checking the forecast using the Neural Network (NN) technique. The principle of checking if the result of MAE is less than 20% is the best. The same as RMSE, but while the R2 score, the result must be found from the value of 1 as much as possible, it is considered the best model. The forecast result this time is that KNN is the best because the MAE value is the least at 0.03% and the RMSE is the least at 0.09%. The R2 score is the result value that must be found from the value of 1, the highest is at 99.05%, and when compared with the actual value, KNN gives the best forecast value, so it can be concluded that KNN forecasts the best. The RMSE, MAE, and R2 scores check the error value and the best accuracy value in forecasting using Neural network (NN) techniques and other techniques (Chaaban and Alfadl, 2024).

In this research, the sales forecast and preparation of the electric car snooker were done using the traditional GM model for forecasting (Ding and Li, 2021). There are many other techniques such as OICGM, TWSOGM, and ESOGM. The last model is the model the researcher invented for use in this forecasting specifically to forecast the number of sales sufficient to stock products in the research. When the forecast is

complete, the error value and accuracy value are checked. In this research, there are 4 techniques for checking: APE, MAPE, RMSE, and MAE. The researcher claims that the 4 techniques are the most appropriate for checking the accuracy value of electric car forecasting and the most suitable for checking the forecast using the Neural Network (NN) technique. The forecast results show that ESOGM has the best forecast value because it checks the error value and accuracy value the best. When checked with the actual data values, the ESOGM model has the closest value. In conclusion, ESOGM forecasts the best. APE, MAPE, RMSE, and MAE check the error value and accuracy value, the best in forecasting using the Neural Network (NN) technique and other techniques (Ding and Li, 2021).

In this research, we do financial time series forecasting using the LSTM technique and MADL in currency forecasting (Michańków et al., 2024). Both techniques are ML techniques in the Neural Network (NN) type. Therefore, the researcher of this work uses MAE and MADL to check the error value and accuracy value of this forecast. Find out which one is the best in checking the accuracy value this time. The conclusion is that MADL/MADL has the best error value or accuracy value, while MAE/MAE and MAE/MADL are close to each other at 0.0025, which is slightly different. In conclusion, MADL checks the error value and the best accuracy value in forecasting. While MAE can be used to check the error value and the accuracy value of forecasting (Michańków et al., 2024).

In this research, the forecasting of electricity load in Singapore is done to plan the schedule for controlling electricity release in Singapore (Aldosari, 2024). In this research, neural network types are used in forecasting, namely ANN, SVM, NB, DT, Rf, and deep learning types are applied, namely NB-GOA, DT-GOA, and ANN-GOA, and the last one is the one that the researcher invented for use in forecasting and believes that it is the best in forecasting, namely SVM-GOA. All models are used to forecast the electricity load rate in Singapore to control electricity release in Singapore. The obtained values will be used to check the error value and accuracy value. In this research, 4 techniques are used for checking, including MAPE, RMSE, MAE, and R2. The reason for using these 4 techniques is because these 4 techniques check the error value and the accuracy value from the forecast using neural network and deep learning techniques the best. The conclusion is that SVM-GOA has the least error value because the RMSE value is less than 20%. The researcher claims that it is the best value. And MAE and MAPE are the least like R2. When compared to the actual value, it can be concluded that SVM-GOA gives the best forecast value. Therefore, it can be concluded that KNN forecasts the best. It can also be concluded that MAPE, RMSE, MAE, and R2 check the error value and the best accuracy value in forecasting using neural network (NN) deep learning techniques and other techniques (Aldosari, 2024).

This research is about air pollution forecasting, especially PM2.5, to find out how much it affects human health (Tsai et al., 2018). The dataset is air pollution data and PM2.5 data from 2010 to 2016, and the forecast is for 2017. This research uses 2 models to forecast air pollution: RNN and LSTM. The results are checked for error and accuracy. This research uses 2 techniques: RMSE and MAE. The reason for choosing these 2 techniques is because the researcher claims that they are the most accurate and most suitable models for checking errors from the forecast by the Neural Network technique. The forecast results show that LSTM has the best forecasting results and the best error and accuracy. When checked with the actual values, it is concluded that the

LSTM model is the most similar. It can be concluded that LSTM forecasts the best. It can also be concluded that RMSE and MAE check the error and the best accuracy in forecasting using Neural Networks (NN) and other techniques (Tsai et al., 2018).

In this research, the prediction of the corrosion time of steel concrete is done (Gao and Yang, 2024). In this research, the corrosion time of steel concrete is predicted by using the neural network model to help in the prediction, namely ANN, to predict the duration of each type of steel. The results are checked for error and accuracy. In this research, 4 techniques are used for checking: R2, MSE, MAE, and MRE. The reason for choosing these 4 techniques is because the researcher claims that they are the most accurate and most suitable models for checking the error from the neural network technique. The results show that ANN can predict the corrosion time of steel concrete using the ANN model because the error is small and like the real value. The accuracy is the most accurate. In conclusion, ANN can predict the corrosion time of steel concrete effectively. The results can be used to study the quality of steel concrete. The technique for checking the accuracy and error values, such as R2, MSE, MAE, and MRE, can check the error and accuracy the best (Gao and Yang, 2024).

In this research, short-term electrical load forecasting was conducted (Gani et al., 2024). In this research, the researcher used Neural Network models to help in forecasting, including ANN, RNN, LSTM, k-means, PCA, and another model invented by the researcher, LSTM with PCA and k-means. After forecasting, the results were checked for error values and accurate values. This research used 3 techniques to check: RMSE, MAPE, and MSE. The reason for choosing these three techniques is because the researcher claims that they are the most accurate and most suitable models for checking the error values from the forecast by the Neural Network technique. These three techniques have been used by researchers in many papers, so they claim that they have the best accurate values. The results show that LSTM with PCA and k-means have the best forecast values because the error values and accuracy values are the best in all models. In conclusion, LSTM with PCA and k-means forecast the best. It can also be concluded that RMSE, MAPE, and MSE provide the best error and accuracy in forecasting using Neural Networks (NN) and other techniques (Gani et al., 2024).

In this research, happiness predictions were conducted using machine learning models to analyze happiness behavior data and forecast it (Akanbi et al., 2024). The data set was referenced from the World Happiness Report from 2018 and 2019 and forecasted in 2020 using 3 techniques from the machine learning model to analyze the data: LASSO, Random Forest, and XGBoost. The forecast results were used to check the error value and accuracy value, which the user used 2 techniques to check: R-squared and MSE. The researcher claimed that the two techniques were the best in checking the error value from the forecast by the machine learning technique. The results concluded that XGBoost had the best error value and accuracy value measured by the error value of 92.13%, which is very close to the truth, and the accuracy value measured by the 2 techniques is the best. Therefore, it can be concluded that XGBoost forecasts the best. It can also be concluded that R-squared and MSE checked the error value and the best accuracy value in forecasting using the machine learning technique and other techniques (Akanbi et al., 2024).

This research aims to forecast the demand for electric vehicle charging by applying the forecast results to the construction of electric vehicle charging stations in the United States (Rashid et al., 2023). This is achieved by importing real-time electric

vehicle charging data from these stations. In this research, the research uses the neural network model to help in forecasting, including RNN, CNN, LSTM, and GRU, and the last one is the one invented by the researcher for use in this forecasting, which is LSTM-HT. Then, the forecasting of electric vehicle charging demand is used to analyze and plan the construction of electric vehicle charging stations in the United States. The results are used to check the error and accuracy. In the technique, 4 techniques are used to check: R2 score, MSE, MAE, and MAPE. The reason for choosing these 4 techniques is that the researcher claims that these 4 techniques are the most accurate and precise in forecasting using the neural network model. And these 4 techniques check the errors most accurately. The results obtained show that the LSTM-HT model has the best and most appropriate forecast value because the best error value is the most accurate. In conclusion, LSTM-HT forecasts the best. It can also be concluded that the R2 score, MSE, MAE, and MAPE are the best predictors of error and accuracy using neural network techniques and other techniques (Rashid et al., 2023).

In this research, the method of recycling and improving waste management and separation is predicted (Olawumi et al., 2024). In this forecast, the neural network model is used to help in the forecast, including LSTM, linear regression, Arima, SVM, and decision tree. The results are used to plan the management and recycling of waste systematically. The results must be checked for error and accuracy. In this research, 2 techniques are used to check: MAE and RMSE. The researcher claims that these 2 techniques are the most suitable for checking the accuracy and error values of the forecast using the neural network model. The results show that LSTM predicts the best value and the best error and accuracy. It can be concluded that LSTM predicts the best. It can also be concluded that MAE and RMSE check the error and accuracy values the best in forecasting using neural network techniques and other techniques (Olawumi et al., 2024).

In this research, the deep electric vehicle charging loading time is forecasted to reduce the charging time of each electric vehicle for ease of use and installation of electric vehicle charging stations (Alkhafaj et al., 2023). This research uses deep learning models to help in forecasting, namely HMSDL-EVCDF. In addition, there are other models such as RNN, GRU, SAES, BI-LSTM, and LSTM. Then, the forecast results of the different models are checked for error and accuracy. In this research, 3 techniques are used: MAE, R2, and RMSE. The researchers claim that these two techniques are the most appropriate for checking the error and accuracy because the researchers claim that 3. Check the forecast value using deep learning and Neural Network models. The results show that HMSDL-EVCDF predicts the deep electric vehicle charging loading time to reduce the charging time of electric vehicles the best and it is the same as checking the error and accuracy. In conclusion, HMSDL-EVCDF predicts the best. It can also be concluded that MAE, R2, and RMSE provide the best error and accuracy in forecasting using deep learning and other techniques (Alkhafaj et al., 2023).

In this research, we want to test whether the MAE technique can best check the error value and the accuracy value by using the data for prediction, divided into 80% and 20%, (Jonathan et al., 2019) where 80% is the data that the model learns by itself and 20% is the model that is used to check the error value. Others are compared, such as the actual value of the data. We call this step Predicting Flow. The result is that the MAE value is very far from the actual value, at 0.582, which is very close on average.

In conclusion, MAE is the best at checking the error value and the accuracy value (Jonathan et al., 2019).

In this research, we want to test which technique can check the error value and accuracy value better between MAE and the new technique for checking, DMAE, to find the best one for checking the error value of the forecast (Frías-Paredes et al., 2018). In this research, the wind forecast data set for planning the power system and trading wind energy is the wind energy production data for 15 months from Spain. It is used in the forecasting of the operators of this research using the Predicting Flow method. The result is that DMAE can check the error value better than MAE because it is close to the actual value by about 80%, which is higher than MAE but in the same direction. In conclusion, DMAE can check the error value and accuracy value the best. And MAE can be used to check the forecast error value even though the value is lower, but it is more popular (Frías-Paredes et al., 2018).

In this research, we want to test the error-checking technique between RMSE and MAE (Karunasingha, 2022). Which one checks the error and accuracy better? Using the forecasting data set using ANN and ARIMA models, this research uses the operator forecasting method. This research uses the Predicting Flow method. The results obtained in this research are used to check the error of both techniques and compare them with the actual values to see which one checks better. The results show that RMSE has a slightly better value, but it cannot be said which one is better. In conclusion, RMSE and MAE check the error and accuracy the best in forecasting (Karunasingha, 2022).

In this research, electricity consumption in Thailand is forecasted to accurately plan for the future electricity demand in the country and control the production volume in and out of the country (Kluabwang, 2018). In this forecast, the researcher used 2 models to verify: the original GM and Tune-GM, which are models invented by the researcher for this research. The results are used to verify the accuracy of the error. The technique used for verification is MAPE. The reason for choosing MAPE is because the researcher claims that MAPE is suitable for verifying the error and accuracy of forecasts using the GM model. The data used is the electricity consumption in Thailand from 2008-2013 and the forecast is from 2014-2019. The results show that Tune-GM forecasts the best value and the best error. When compared to the actual value, it is the closest. It can be concluded that Tune-GM forecasts the best. It can also be concluded that MAPE checks the error and the best accuracy in forecasting using the GM technique and other techniques (Kluabwang, 2018).

In this research, the probability of an event from a Bernoulli experiment is estimated using a sequential stopping rule, also known as inverse binomial sampling (Mendo, 2009). The author proposes a formula to calculate the mean absolute error (MAE) used in the estimation because the researcher claims that MAE checks the best value. The experiment stops when the number of successful events reaches a certain number. In addition, the work also shows that the MAE can decrease in a hierarchical order, which means that when the number of successful events increases, the MAE will decrease. This research is useful in simulating communication systems, such as bit error rate (BER) and block error rate (BLER). In conclusion, MAE is the best at checking errors and accuracy (Mendo, 2009).

In conclusion, after studying various e-journals, it can be concluded that the research has used the Predicting Flow Method, which is a technique used in forecasting

or predicting the flow of various resources, such as traffic, product planning, or even forecasting happiness. This method uses mathematical and statistical models to predict the direction or amount that will occur in the future. The purpose is to manage and plan the use of resources efficiently. It is divided into 30 to 70, 80 to 20, or 60 to 40, depending on how far in advance we forecast. Most of it is data that we use relationships and is real data that comes continuously. On the other side is data that is used to forecast through various models, such as time series models such as ARIMA and Holt-Winters, which are suitable for predicting data with continuous data flow, and machine learning models such as random forest and gradient boosting that can learn complex flow patterns. And neural networks such as RNN and LSTM, which are suitable for predicting time series data. And the flow with an uncertain or fluctuating pattern, then take the obtained data to check with various error values and accuracy values, such as mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE), which the researcher used all 3 of these techniques to check the error value and accuracy value because MAE is suitable for evaluating models that work with small data sets or incomplete data. Because it is not necessary to have a lot of data for the results to be reliable and suitable for measuring the accuracy of forecasts with the Gray Model (GM), because GM is often used to predict data with limited samples or high uncertainty. MAE error measurement helps to show the difference between the predicted value and the actual value at an easy-to-understand level, regardless of the size of the data or the distribution of outliers. It is suitable for GA when GA adjusts the data to create a GA forecasting model. The inspection is not complicated and can be inspected (Rob J. Hyndman and Koehler, 2006). RMSE is suitable for the type of work that requires a quantitative measure of the difference between the predicted value and the actual value. RMSE provides detailed information about the error in the model, such as time series forecasting, such as sales forecasting, price forecasting, or demand forecasting. RMSE helps to measure the severity of the forecast error or to check the sales. Next, MAPE is suitable for the type of forecasting data that has no zero values or very low values: MAPE may not be appropriate in cases where the actual value is close to zero because it will show an unusually high percentage. Therefore, MAPE should be avoided when there is data with zero values or very low values. Quantitative data: MAPE is good when the forecasted data is quantitative, such as sales, production volume, or finance, which can be measured imprecisely. Data analysis in a business context: MAPE is popular in business and marketing because displaying the results in percentage terms makes it easy to understand and can be communicated well at the executive level. Therefore, in this research, the data will be divided into 80:20 parts. 80 is the original data and the actual value. 20 is the data that forecasts the demand for electric vehicles in 2023 and 2024 using the data from 2022-2023 itself using the OGM-GA model and using three techniques to check the error value and the accuracy value.

2.11 The process of optimizing a residual revised grey model using a genetic algorithm (R2GM-GA).

In this step, after the researcher has explored and studied from various sources and various sources of journals and research both domestically and internationally, it was found that there are still grasp that no one has done or has not yet invented, which is to take the GM model to the process called Optimizing by doing it called Residual

Revised Gray Model and then use it for forecasting by integrating using the Genetic Algorithm model to help in forecasting to help find the final answer as efficiently as possible. The reason for choosing to make this model is that, after studying various periods and research, it was found that the GM model is the most popular and widespread forecasting tool. However, at the same time, most of the research that was studied found that it used the GM model, but the original model did not change much, or was applied to some extent, but did not use the original GM basis much. Therefore, the researcher decided to create a model called (R2GM-GA) to use in forecasting with the highest accuracy and precision because after checking the error value and the accuracy value, it had the best value, and the researcher believed that the (R2GM-GA) model was the best forecasting model. The steps of the residual revised grey model using a genetic algorithm (R2GM-GA) are as follows:

The steps of the residual revised grey model using a genetic algorithm (R2GM-GA) are as follows: Starting from the normal steps of the Grey model, starting from the following equations:

$$X^{(0)} = [x^{(0)}(1), x^{(0)}(2), \dots, x^{(0)}(n)] \quad (2-61)$$

Where:

$X^{(0)}$ is the actual data value that the researcher chooses to input.

The researcher then calculates the grey model procedure using equations (2-2) and (2-3). The residual revised grey model is then calculated using the following equation.

$$\epsilon(K) = x^{(0)}k - \hat{x}^{(0)}(K) \quad (2-62)$$

Where:

$\epsilon(K)$ is represents the model error.

$x^{(0)}k$ is represents the actual value.

$\hat{x}^{(0)}(K)$ is represents the predicted value.

The residual revised gray model procedure involves calculating the model error. After obtaining the error values, we loop back to the gray model procedure to correct the errors in the residual revised gray model until the error is minimized. After obtaining the values, we adjust the data to be suitable for use in the genetic algorithm model. The researcher then performs the genetic algorithm process as described in Chapter 2. 5. The following section presents a series of example problems and their corresponding solution procedures for the RGMGA model, aiming to demonstrate its underlying principles and computational methodology.

(R2GM-GA). Example

Assume that the following dataset is available for forecasting

$$X^{(0)} = [120, 135, 150, 165, 181, 200, 218, 240] \quad (2-63)$$

In order to forecast the subsequent value in the series using the GM(1,1) model, the initial procedure involves deriving the accumulated generating sequence (AGS) from the original dataset.

$$x^{(1)} = [120, 255, 405, 570, 751, 951, 1169, 1409] \quad (2-64)$$

Then, given the values of a and u from the problem, $a = -0.1142$, $u = 30.210$, calculate $gm(1,1)$ from the formula.

$$\hat{x}^{(0)} = [136.6, 148.4, 162.7, 177.5, 192.5, 208.3, 225.4] \quad (2-65)$$

Next, calculate the Residual Series Gray model as follows.

$$\begin{aligned} \epsilon(K) &= x^{(0)}k - \hat{x}^{(0)}(K) \\ \epsilon(K2) &= 135 - 136.6 = -1.6 \\ \epsilon(K3) &= 150 - 148.4 = +1.6 \\ \epsilon(K4) &= 165 - 162.7 = +2.3 \\ \epsilon(K5) &= 181 - 177.5 = +3.5 \\ \epsilon(K6) &= 200 - 192.5 = +7.5 \\ \epsilon(K7) &= 218 - 208.3 = +9.7 \\ \epsilon(K8) &= 240 - 225.4 = +14.6 \end{aligned} \quad (2-66)$$

The researcher obtain $r^{(0)} = [-1.6, 1.6, 2.3, 3.5, 7.5, 9.7, 14.6]$. Next, the $r^{(0)}$ values are used to calculate $\hat{r}^{(0)}$ according to the GM(1,1) model. Applying the formula yields

$$\hat{r}^{(0)} = [-1.6, 2.0, 3.1, 4.9, 7.2, 10.5, 15.2] \quad (2-67)$$

The $\hat{r}^{(0)}$ values are then used to derive $\hat{x}^{(0)}$ according to the following equation.

$$\begin{aligned} k=2 &= \hat{x}^{(0)} = 136.6 + 2.0 = 138.6 \\ k=3 &= \hat{x}^{(0)} = 148.4 + 3.1 = 151.5 \\ k=4 &= \hat{x}^{(0)} = 162.7 + 4.9 = 167.6 \\ k=5 &= \hat{x}^{(0)} = 177.5 + 7.2 = 184.7 \\ k=6 &= \hat{x}^{(0)} = 192.5 + 10.5 = 203.0 \\ k=7 &= \hat{x}^{(0)} = 208.3 + 15.2 = 223.5 \\ k=8 &= \hat{x}^{(0)} = 225.4 + 19.8 = 245.2 \end{aligned} \quad (2-68)$$

Rearranging the equation yields:

$$\hat{x}^{(0)} = [138.6, 151.5, 167.6, 184.7, 203.0, 223.5, 245.2] \quad (2-69)$$

Using the values derived earlier, the GA model is calculated, which produces a forecasted value of 262.3 for year 9.

CHAPTER 3

ANALYSIS PROCESS

This chapter presents the research methodology and outlines the procedures undertaken in this study. It describes the overall research process, including the key steps involved in data preparation, model development, and performance evaluation. In addition, this chapter provides an overview of the forecasting algorithms employed to estimate electric vehicle (EV) demand. Furthermore, the results obtained from each algorithm are illustrated through graphical representations, and their forecasting performance is compared using accuracy evaluation metrics. The chapter also includes a summary table presenting the performance of each model. The structure of this chapter is organized into the following sections:

- 3.1 Data collection
- 3.2 Data cleaning
- 3.3 Model generating
- 3.4 GM optimization
- 3.5 Gray model (GM) integrated with genetic algorithm (GA)
- 3.6 Compare the performance of the algorithms
- 3.7 The demand for electric cars dashboard
- 3.8 Check Errors

3.1 electric car charging stations analysis process

In this research, there are 6 steps of analysis process, in each step, there is data collection. Filter data Take only what is necessary for this job. There are steps to optimize the Residual Revised Gray Model (RRGM). There are steps to optimize the Residual Revised Gray Model (RRGM) using Genetic Algorithms (GA). Finally, let's compare. Analysis and forecasting of demand for electric cars (forecasting) affects the construction of electric car charging stations analysis process as shown in figure 3.1

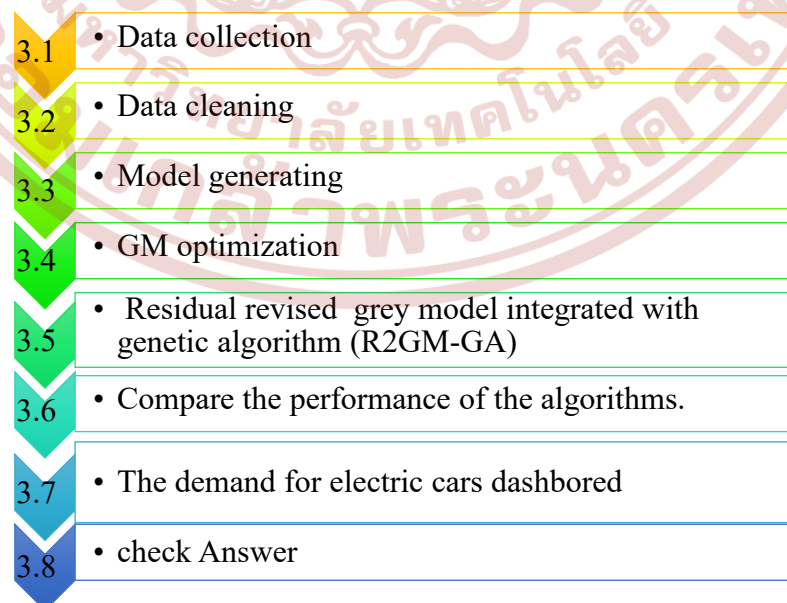


Figure 3-1 Flow chart research process

3.2 Data Collection

The data collection process began with gathering information on new vehicle registrations in Thailand from January 2019 to December 2023, along with the most recent data on electric vehicle (EV) charging stations in 2024. Additionally, data on electric vehicle registrations were obtained from the Automotive Information Center of the Thailand Automotive Institute, which provides official online statistical records of vehicle registrations (Automotive Information Center, n.d.). These sources served as the primary datasets for developing the forecasting models used in this research. The initial dataset consisted of nine variables: sequence, year-month, brand, ≤ 2000 cc, > 2000 cc, BEV, non-specific, and total. However, several of these variables were not suitable for forecasting due to redundancy or limited predictive relevance. Therefore, the dataset was reorganized to improve its suitability for forecasting analysis. The revised structure consists of six variables: sequence, year, month, brand, size, type, and number. This restructuring improved clarity and analytical consistency, ensuring that the data aligned more effectively with the methodological requirements of the forecasting models. In the revised dataset, the sequence variable represents the alphabetical ordering of automobile brands. The year variable indicates the calendar year of vehicle registration, while the month variable represents the months from January to December. The brand variable identifies each automobile brand registered within the corresponding period. The size variable classifies vehicles based on engine capacity into two categories: ≤ 2000 cc for vehicles with an engine displacement less than or equal to 2000 cc, and > 2000 cc for those exceeding 2000 cc. The type variable categorizes vehicles into combustion vehicles and battery electric vehicles (BEVs), representing conventional fuel-powered and fully electric cars, respectively. The non-specific category refers to vehicles that are no longer in production but remain available in the market. Finally, the number variable denotes the quantity of new vehicle registrations for each brand, categorized by type and size. All collected and reorganized data are presented in Table 3-1, which provides an example of the dataset used for analysis in this research.

Table 3-1 Raw data table on new vehicle registrations

No.	Year	Month	Brand	> 2000 cc	≤ 2000 cc.	Bev	Not specific	Total
1	2023	Jan	Aston Martin	0	1	0	0	1
2	2023	Jan	Audi	108	12	3	0	123
3	2023	Jan	Bentley	0	19	0	0	19
4	2023	Jan	Biz Nex	0	0	31	0	31
5	2023	Jan	BMW	1,004	163	37	0	1,204
6	2023	Jan	BYD	0	0	1,064	0	1,064
7	2023	Jan	Cadillac	0	1	0	0	1
8	2023	Jan	Chevrolet	2	3	0	0	5

Table 3-1 Raw data table on new vehicle registrations(continued)

NO	Year	Month	Brand	>2000c c.	<=2000cc.	Bev	Not spec ific	Total
9	2023	Jan	DAIHATSU	1	0	0	0	1
10	2023	Jan	DFSK	3	0	1	0	4
11	2023	Jan	E-TUCK	0	0	1	0	1
12	2023	Jan	FERRARI	0	4	0	0	4
13	2023	Jan	FORD	3,786	725	0	0	4,511
14	2023	Jan	FOTON	0	7	0	0	7
15	2023	Jan	HAVAL	810	0	0	0	810
17	2023	Jan	HINO	0	83	0	0	83
18	2023	Jan	HONDA	8,260	200	0	0	8,460
19	2023	Jan	HONDA	8,260	200	0	0	8,460
20	2023	Jan	HYUNDAI	142	358	0	0	500
21	2023	Jan	ISUZU	14,581	4,063	0	0	18,644
22	2023	Jan	JEEP	4	3	0	0	7
23	2023	Jan	KIA	0	134	0	0	134
24	2023	Jan	LAMBORG HINI	0	2	0	0	2
25	2023	Jan	LAND ROVER	8	0	0	0	8

Table 3-1, it shows the data of electric vehicle registrations that have not been adjusted and compiled into 9 columns, consisting of the columns Order, Year month brand <=200cc., >2000cc., Bev, not specific, total. The data will be adjusted, and the columns will be reduced to only 6 columns in the next step, called data cleaning.

3.3 Data cleaning

The data selection and preparation process involved systematically removing information that was not relevant to the research objectives and retaining only the data required for forecasting the demand for electric vehicles (EVs) in Thailand. Monthly vehicle registration data, particularly for battery electric vehicles (BEVs), were collected and compiled from the Automotive Information Center (AIC) of the Thailand Automotive Institute, which serves as an official source of national automotive statistics (Thailand Automotive Institute, n.d.). The dataset was obtained from the AIC online database. The dataset includes monthly vehicle registration records from January 2019 to December 2023, as presented in Table 3-1. It consists of six variables: sequence, year, month, brand, size, type, and number of registered vehicles. In the original dataset, vehicle size is categorized into three groups: vehicles with engine capacities less than or equal to 2000 cc, vehicles with engine capacities greater than 2000 cc, and battery electric vehicles (BEVs). The type variable represents the energy source used by each vehicle category during the study period, including combustion engine vehicles, which rely on fuel as the primary energy source, and BEVs, which operate using electric

power. Because the objective of this research is to forecast the demand for electric vehicles in Thailand, only BEV-related records were retained for analysis. All data associated with non-electric vehicles, including those with engine sizes ≤ 2000 cc and >2000 cc, as well as all combustion engine vehicle types, were removed. This data filtering process ensured that the dataset aligned with the research objectives and supported an accurate assessment of BEV demand trends in Thailand. The refined dataset after applying the selection criteria is presented in Table 3-2.

Table 3-2 Updated raw data table on new vehicle registrations

NO.	Year	Month	Brand	size	type	Total
1	2023	Jan	ASTON MARTIN	Bev	Bev	0
2	2023	Jan	ASTON MARTIN	>2000 cc.	Oil cars	1
3	2023	Jan	ASTON MARTIN	≤ 2000 cc.	Oil cars	0
4	2023	Jan	ASTON MARTIN	Not specific	Oil cars	0
5	2023	Jan	Audi	Bev	Bev	3
6	2023	Jan	Audi	>2000 cc.	Oil cars	12
7	2023	Jan	Audi	≤ 2000 cc.	Oil cars	108
8	2023	Jan	Audi	Not specific	Oil cars	0
9	2023	Jan	BENTLEY	Bev	Bev	0
10	2023	Jan	BENTLEY	>2000 cc.	Oil cars	19
11	2023	Jan	BENTLEY	≤ 2000 cc.	Oil cars	0
12	2023	Jan	BENTLEY	Not specific	Oil cars	0
13	2023	Jan	BIZ NEX	Bev	Bev	31
14	2023	Jan	BIZ NEX	>2000 cc.	Oil cars	0
15	2023	Jan	BIZ NEX	≤ 2000 cc.	Oil cars	0
16	2023	Jan	BIZ NEX	Not specific	Oil cars	0
17	2023	Jan	BMW	Bev	Bev	37
18	2023	Jan	BMW	>2000 cc.	Oil cars	163
19	2023	Jan	BMW	≤ 2000 cc.	Oil cars	1,004
20	2023	Jan	BMW	Not specific	Oil cars	0
21	2023	Jan	BYD	Bev	Bev	1,064
22	2023	Jan	BYD	>2000 cc.	Oil cars	0
23	2023	Jan	BYD	≤ 2000 cc.	Oil cars	0
24	2023	Jan	BYD	Not specific	Oil cars	0

From Table 3-2, the dataset was restructured for forecasting purposes by reducing and organizing it into six variables: sequence, year, month, brand, size, type, and number. The sequence variable represents the alphabetical ordering of automobile brands. The year variable indicates the calendar year of vehicle registration, while the month variable represents the months from January to December. The brand variable identifies the automobile brands registered in Thailand during each specified period. The variable size classifies vehicles based on engine capacity into two categories: ≤ 2000 cc, representing vehicles with an engine displacement less than or equal to 2000 cc, and >2000 cc, representing vehicles with an engine displacement greater than 2000 cc. The type variable categorizes vehicles into two groups: combustion engine vehicles,

which use fuel as the primary energy source, and battery electric vehicles (BEVs), which operate using electricity as the primary source of power. In addition, the non-specific category refers to vehicles that are no longer in production but are still available in the market. Finally, the number variable represents the total number of newly registered vehicles for each brand, categorized by vehicle type and size. The cleaned and structured dataset is presented in Table 3-3.

Table 3-3 cleaning data

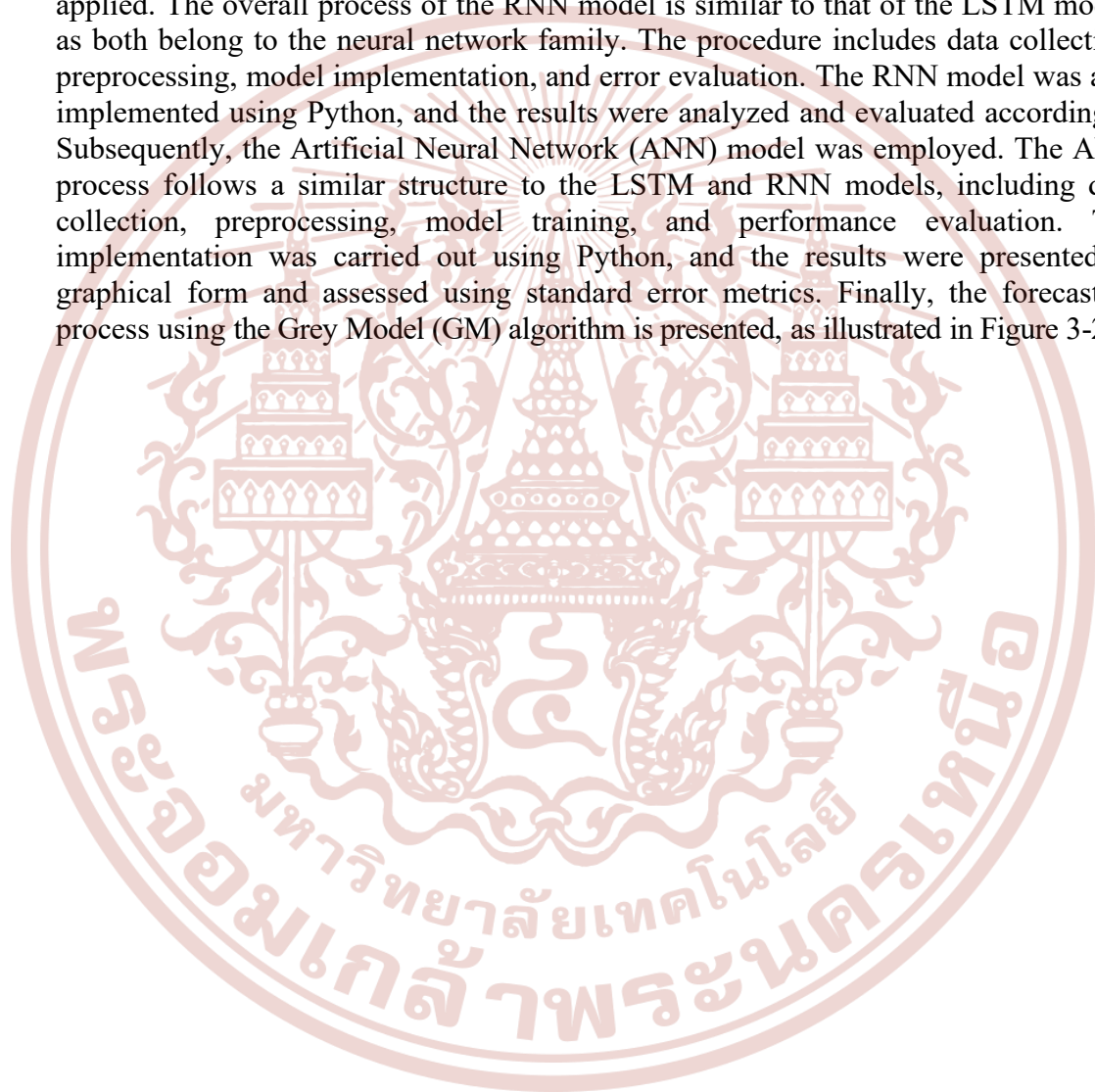
No.	Year	Month	Brand	size	type	Total
1	2023	Jan	ASTON MARTIN	Bev	Bev	0
2	2023	Jan	Audi	Bev	Bev	3
3	2023	Jan	BENTLY	Bev	Bev	0
4	2023	Jan	BIZ NEX	Bev	Bev	31
5	2023	Jan	BMW	Bev	Bev	37
6	2023	Jan	BYD	Bev	Bev	1,064
7	2023	Jan	CADILLAC	Bev	Bev	0
8	2023	Jan	CHEVROLET	Bev	Bev	0
9	2023	Jan	DAIHATSU	Bev	Bev	0
10	2023	Jan	DFSK	Bev	Bev	1
11	2023	Jan	E-TUCK	Bev	Bev	1
12	2023	Jan	FERRARI	Bev	Bev	0
13	2023	Jan	FORD	Bev	Bev	0
14	2023	Jan	FOTON	Bev	Bev	0
15	2023	Jan	HAVAL	Bev	Bev	0
16	2023	Jan	HINO	Bev	Bev	1
17	2023	Jan	HONDA	Bev	Bev	0
18	2023	Jan	HYUDAI	Bev	Bev	0
19	2023	Jan	ISUZU	Bev	Bev	0
20	2023	Jan	JEEP	Bev	Bev	0
21	2023	Jan	KIA	Bev	Bev	0
22	2023	Jan	LAMBORGHINI	Bev	Bev	0
23	2023	Jan	LAND ROVER	Bev	Bev	0
24	2023	Jan	LEXUS	Bev	Bev	0

From Table 3-3, the dataset was cleaned by retaining only the data relevant to forecasting the demand for electric vehicles. All irrelevant data were removed during the preprocessing stage. The original dataset consisted of 21,285 rows and 6 columns. After data cleaning and filtering, only 92 rows remained, while the number of columns remained unchanged at six. The refined dataset includes only the variables necessary for forecasting electric vehicle demand in Thailand and is used in the subsequent analysis.

3.4 Model generating

After data cleaning, the dataset presented in Table 3-3 was used as input for various forecasting models to predict the demand for electric vehicles. The forecasting

process began with the Long Short-Term Memory (LSTM) model. The operational framework of the LSTM model consists of several steps, including data collection, data preprocessing, model training, and error evaluation. The data collection process has been described in Section 3.1, while data preprocessing was conducted during the data cleaning stage. The LSTM model was implemented using Python, and the results were presented in graphical form. The model performance was then evaluated using error metrics. Following the LSTM model, the Recurrent Neural Network (RNN) model was applied. The overall process of the RNN model is similar to that of the LSTM model, as both belong to the neural network family. The procedure includes data collection, preprocessing, model implementation, and error evaluation. The RNN model was also implemented using Python, and the results were analyzed and evaluated accordingly. Subsequently, the Artificial Neural Network (ANN) model was employed. The ANN process follows a similar structure to the LSTM and RNN models, including data collection, preprocessing, model training, and performance evaluation. The implementation was carried out using Python, and the results were presented in graphical form and assessed using standard error metrics. Finally, the forecasting process using the Grey Model (GM) algorithm is presented, as illustrated in Figure 3-2.



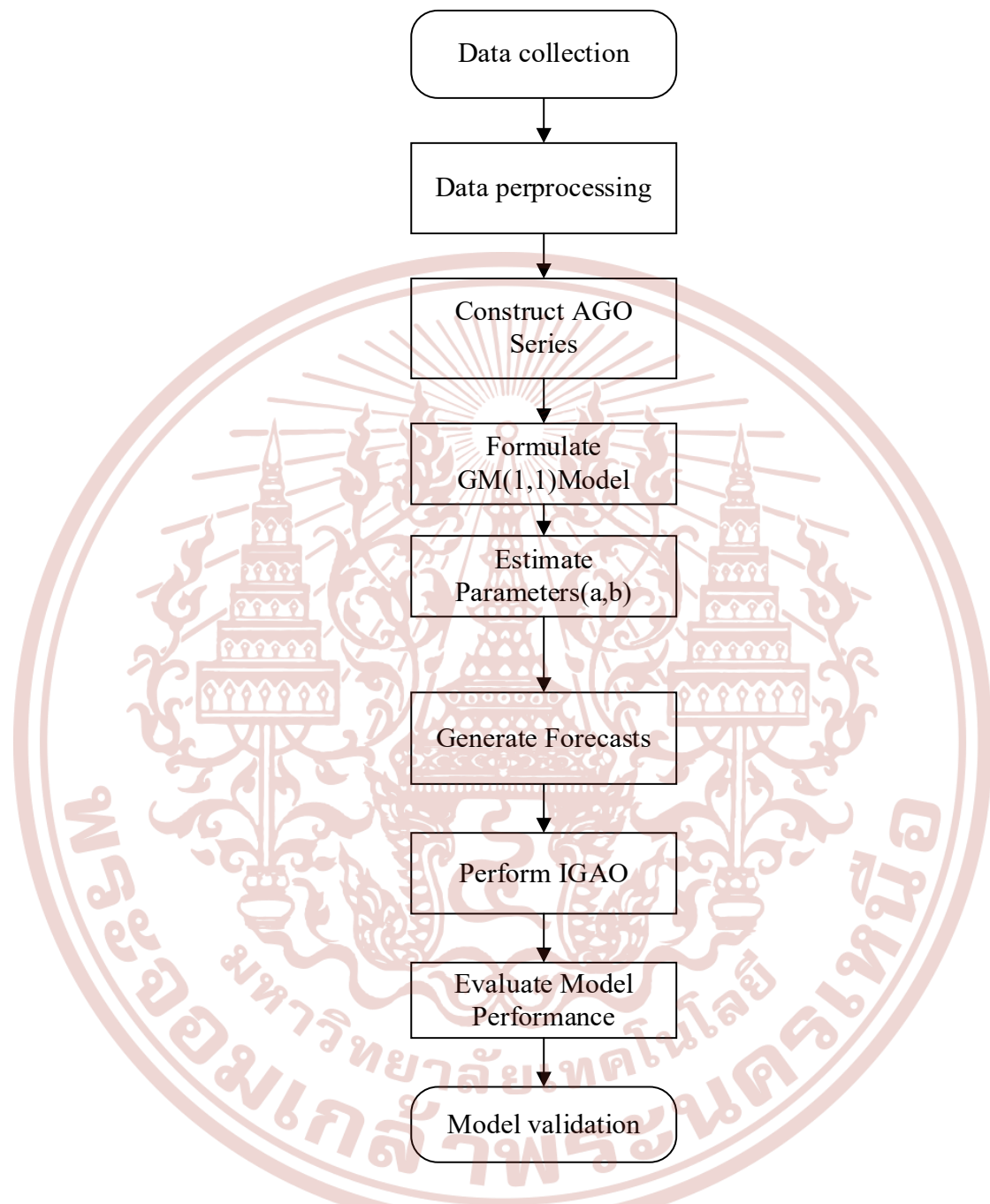


Figure 3-2 Flow chart grey model Process (Xu et al., 2024)

From Figure 3-2, the flowchart illustrates the forecasting process using the Grey Model (GM) algorithm applied in this research. The process begins with data collection, which involves gathering new vehicle registration data, as described in Section 3.1. This is followed by the data preprocessing stage, where irrelevant data are removed, and only the necessary data for forecasting are retained. This step corresponds to the data cleaning process presented in Section 3.2. Next, the Accumulated Generating Operation (AGO) is applied to the original time series data. The AGO is a fundamental concept in Grey System Theory and is used to transform the data into a smoother sequence, thereby revealing the underlying trend and improving forecasting accuracy.

After that, the GM(1,1) model is formulated by constructing a differential equation based on the AGO-transformed data. The model parameters (a and b) are then estimated using the least squares method. Once the parameters are obtained, the forecasting values are generated using the developed model. Subsequently, the Inverse Accumulated Generating Operation (IAGO) is applied to convert the accumulated data back to the original scale. The model performance is then evaluated by calculating standard error metrics, including Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Mean Absolute Error (MAE). Finally, model validation is conducted to assess the overall accuracy and reliability of the forecasting results, which are presented in graphical or dashboard form. The implementation of the GM model was carried out using Python, and the results were visualized for further analysis. The following section presents the process flow of forecasting using the Genetic Algorithm model, as illustrated in Figure 3-3.

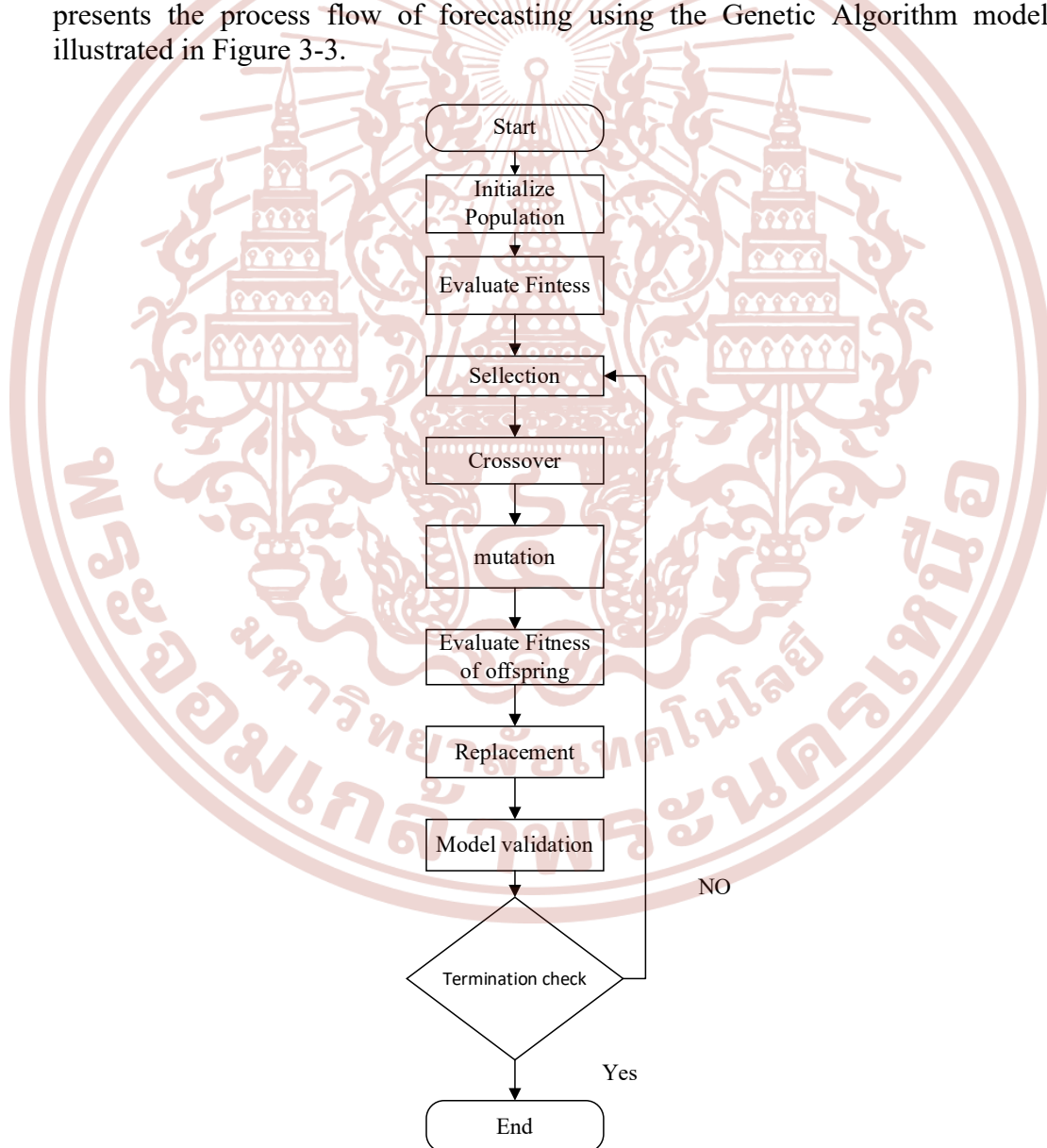


Figure 3-3 Flow chart Genetic Algorithm model process. (Semenova et al., 2020)

From Figure 3-3, the flowchart illustrates the forecasting process using the Genetic Algorithm (GA) applied in this research. The process begins with the initialization of the population, where candidate solutions are generated based on the cleaned dataset obtained in Section 3.2. These initial solutions are typically created randomly to ensure diversity within the population. Next, the fitness of each solution is evaluated using a predefined fitness function. Based on this evaluation, the selection process is performed to choose the most suitable individuals to serve as parents for the next generation. Following this, the crossover operation is applied, where selected parent solutions are combined to produce new offspring. Common crossover techniques include single-point and double-point crossover. The offspring are then subjected to mutation, which introduces random changes to maintain diversity within the population. After mutation, the fitness of the offspring is evaluated again. The results are then used in the replacement step, where less suitable individuals in the current population are replaced by the newly generated offspring. Finally, the model performance is evaluated based on accuracy and error metrics. If the predefined criteria are satisfied, the process terminates and the final results are obtained. Otherwise, the algorithm iterates through the process again. This iterative procedure is referred to as the termination check.

3.5 GM optimization

Since this research requires the use of the Grey Model (GM) together with the Genetic Algorithm (GA) because (R²GM-GA) is quite widely used in research, the researcher has implemented the use of the Residual revised grey model (R²GM) together with GA to forecast the demand data for electric vehicles. This research begins with the Residual revised Grey Model (R²GM) by first determining the original Grey Model (GM (1,1)) and then calculating the residuals according to the following steps.

$$e^{(0)}(K) = x^{(0)}(k) - \hat{x}^{(0)}(k), \dots, k = 1, 2, \dots, n \quad (3-1)$$

Where $\hat{x}^{(0)}(k)$ is the predicted value from GM (1,1).

Create a Gray Model for residuals:

$$e^{(0)} = \{e^{(0)}(1), e^{(0)}(2), \dots, e^{(0)}(n)\} \quad (3-2)$$

Create an accumulative sequence (AGO) for residuals:

$$e^{(1)}(k) = \sum_{i=1}^k e^{(0)}(i), \dots, k = 1, 2, \dots, n \quad (3-3)$$

Create a differential equation for GM (1,1) of residuals:

$$\frac{de^{(1)}t}{dt} + ce^{(1)}(t) = d \quad (3-4)$$

where c and d are the parameters to be found

Calculate the estimated value for residuals:

$$\hat{e}^{(0)}(k+1) = (e^{(0)}(1) - \frac{d}{c})e^{-ck} + \frac{d}{c}, \dots, k = 1, 2, \dots, n-1 \quad (3-5)$$

The equation for forecasting Residual revised grey model integrated with genetic algorithm (R²GM-GA)

$$\hat{x}_{RRGM}^{(0)}(k) = \hat{x}^{(0)}(k) + \hat{e}^{(0)}(k), \dots, k = 1, 2, \dots, n \quad (3-6)$$

Where:

$e^{(0)}$ is the difference (residuals) between the predicted and actual values

$e^{(0)}(k)$ is the difference value at time point k

c, d are the parameters to be evaluated in the differential equation of residuals

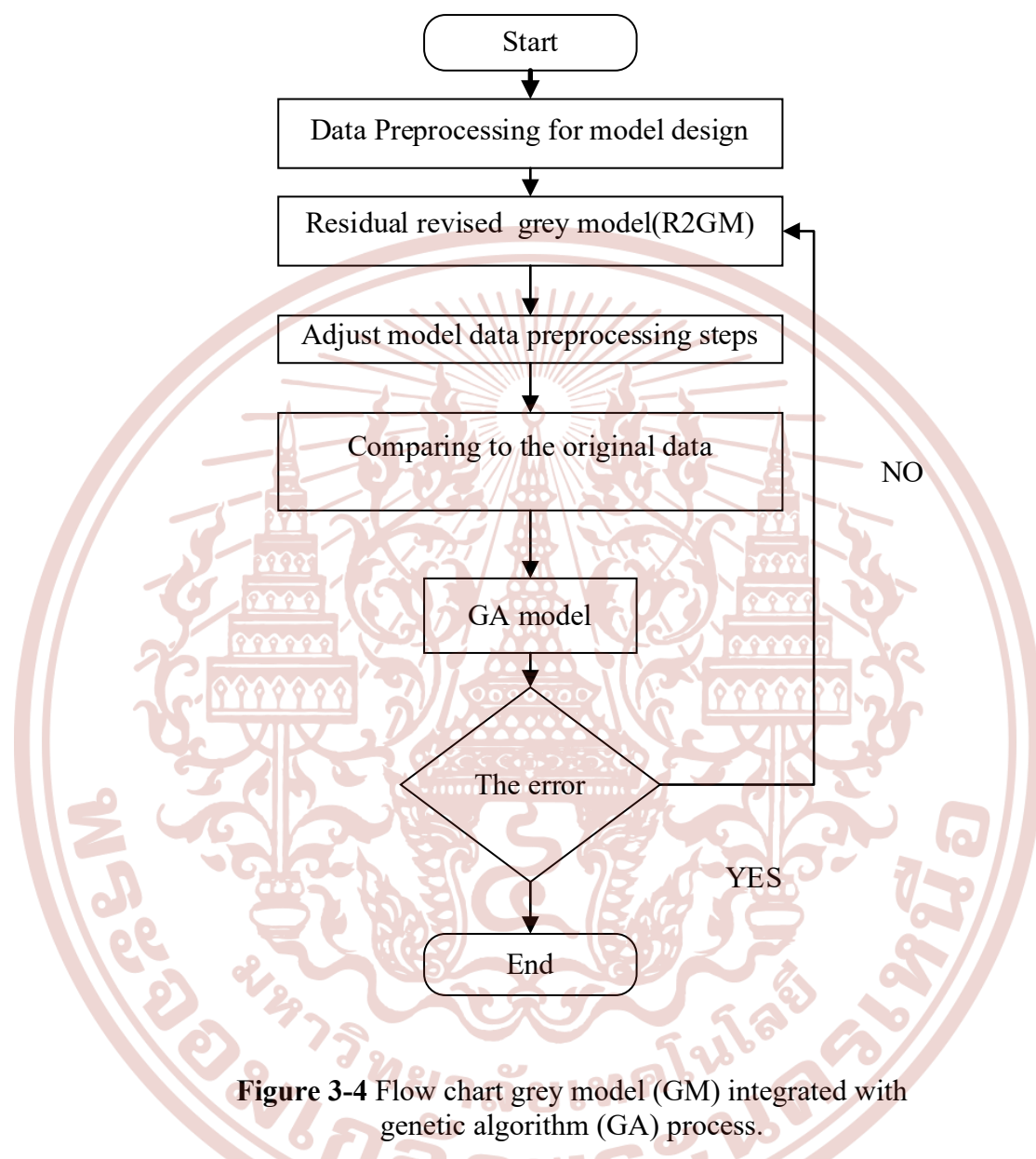
$\frac{de^{(1)}t}{dt}$ is the cumulative rate of change of residuals $e^{(1)}$ per t

$\hat{e}^{(0)}(k)$ is the predicted value of residuals at time point

After obtaining the model equation for forecasting with the Residual revised grey model integrated with genetic algorithm (R²GM-GA) algorithm, it can be used to forecast the demand for electric vehicles in the next step.

3.6 Residual revised grey model integrated with genetic algorithm (R²GM-GA)

After the prediction through various algorithms, the algorithm will run the normal GM process and then run the grey model (GM) integrated process according to step 3.4. Then, the output is adjusted to fit the original data to find the value in GA according to step 3.3 according to the flowchart in the following steps. as shown in Figure 3-4.



From Figure 3-4, the flowchart illustrates the process of the optimized Grey Model combined with the Genetic Algorithm (GA) in this research. The process begins with data preprocessing for model development, which includes data collection and data cleaning to ensure that the dataset is suitable for the optimized GM model. The optimization process of the GM model follows the procedures described in Section 3.4. After preprocessing, the data are further adjusted to ensure compatibility with the GA process. The transformed dataset is then compared with the original data to verify consistency and suitability for optimization. Next, the Genetic Algorithm (GA) is applied. The procedure of the GA model follows the steps described in Section 3.3, including population initialization, fitness evaluation, selection, crossover, mutation, and replacement. Once the optimization process is completed, the model performance is evaluated using error metrics and forecasting accuracy. If the results are not

satisfactory or are inferior to those of other models, the process returns to the GM optimization stage for further refinement. Otherwise, the final results are presented and the forecasting outcomes are summarized.

3.7 Compare the performance of the algorithms

In this step, the results of the prediction of each algorithm are shown and the table of error values or the forecast accuracy of each algorithm is shown. Starting from Figure 3-5, the graph image shows the results of the prediction of electric vehicle demand using the ANN algorithm as shown in the picture. As shown in Figure 3-5

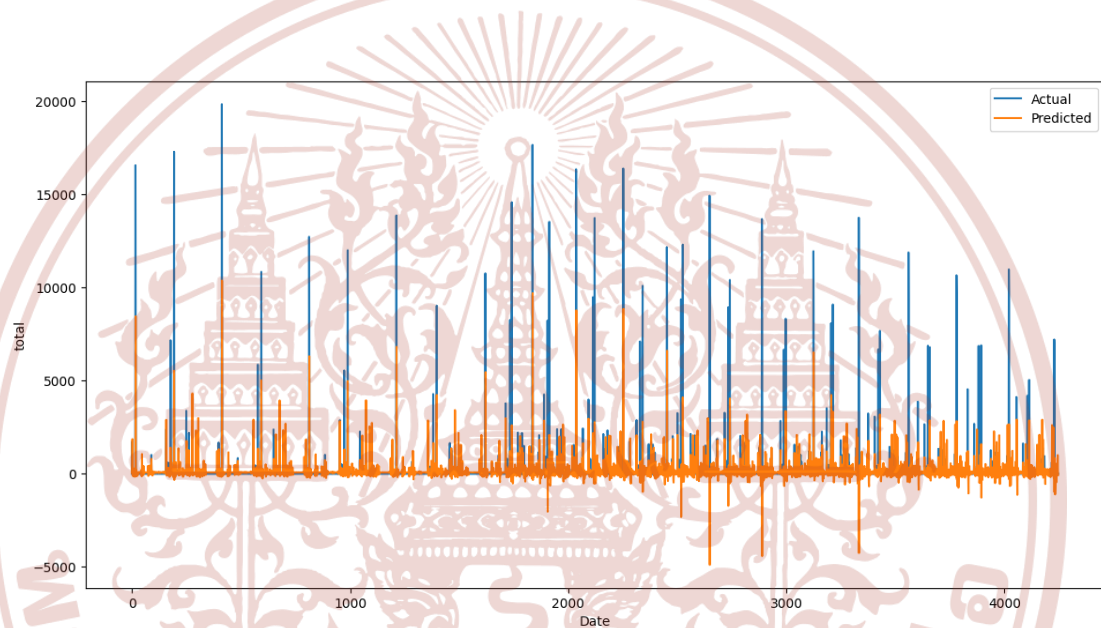


Figure 3-5 Graph of the forecast results using the ANN algorithm

Figure 3-5 shows a graph depicting the predicted electric car demand using the ANN algorithm. The blue line represents the cleaned data, specifically the number of newly registered electric cars, while the yellow line shows the predicted demand using the ANN algorithm. The forecast indicates that the demand for electric cars is continuously decreasing. The x-axis represents time, and the y-axis represents the number of electric cars in demand. Later, in Figure 3-6, the graph illustrates the predicted electric car demand using the RNN algorithm. As shown in Figure 3-6

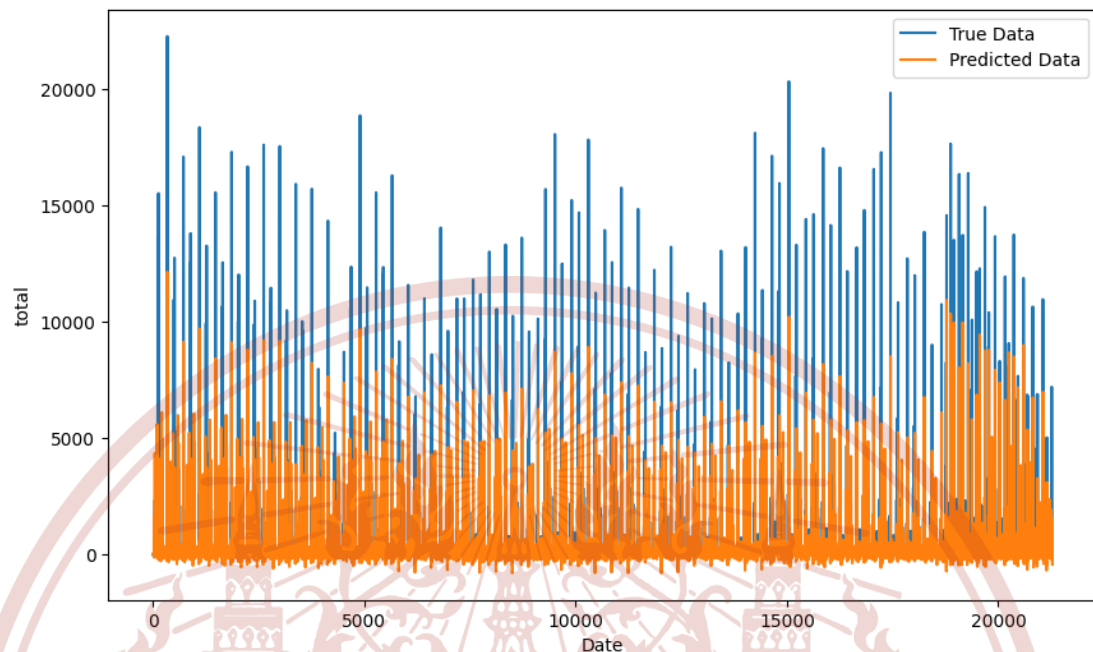


Figure 3-6 Graph of the forecast results using the RNN algorithm

From Figure 3-6, the graph shows the results of predicting the demand for electric vehicles using the RNN algorithm. The blue line is the data value that we cleared the data, the number of newly registered electric vehicles, and the yellow line is the value of the demand for electric vehicles using the RNN algorithm. Note that the forecast value says that the demand for electric vehicles is continuously decreasing, where the x-axis and y-axis are the values of the number of electric vehicle demand is shown in Figure 3-7. The graph shows the results of the electric vehicle demand prediction using the LSTM algorithm as shown in the figure. As shown in Figure 3-7

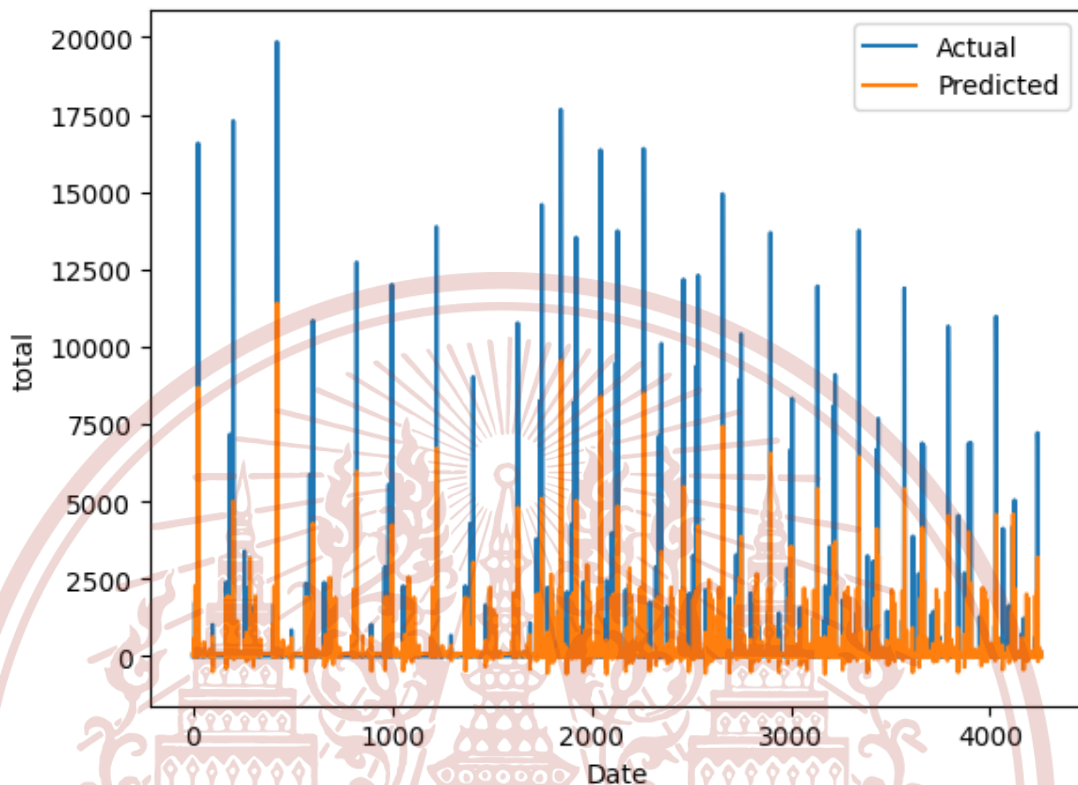


Figure 3-7 Graph of the forecast results using the LSTM algorithm

From Figure 3-7, the graph shows the results of the electric car demand prediction using the LSTM algorithm. The graph is similar to ANN and RNN because they are the same type of forecasting using neural networks, making the graph similar. The blue line is the data value that we cleared the data, the data of the number of newly registered electric cars, and the yellow line is the value of the electric car demand using the LSTM algorithm. It can be observed that the forecast value indicates that the demand for electric vehicles is decreasing continuously, where the x-axis and y-axis are the values of the number of electric vehicles demanded. Later, in Figure 3-8, the graph image shows the results of the prediction of electric vehicle demand using the GM algorithm as shown in the figure as shown in Figure 3-8

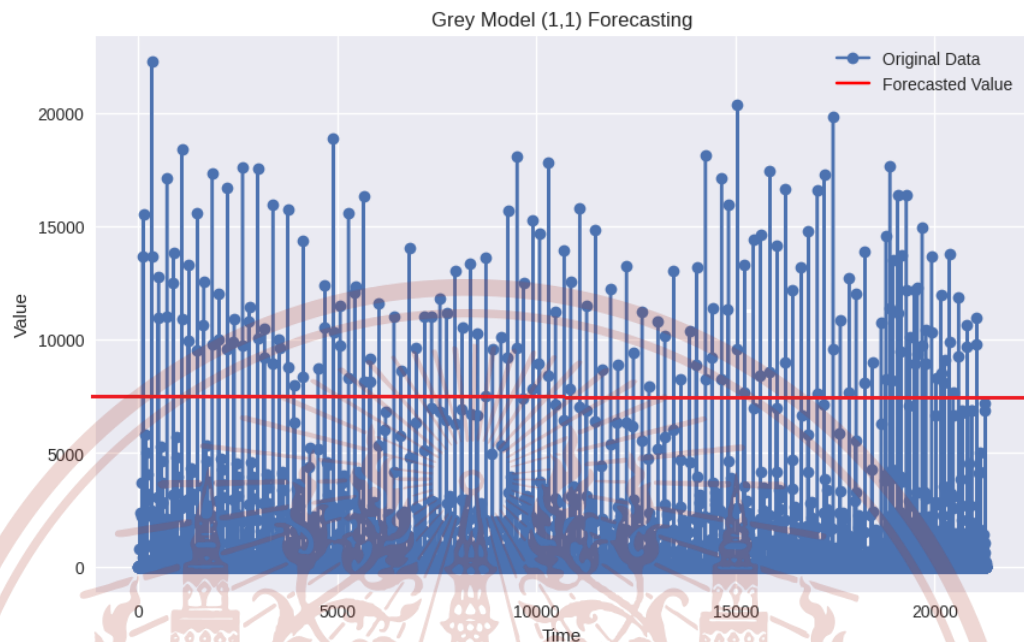


Figure 3-8 Graph of the forecast results using the GM model algorithm

From Figure 3-8, the graph shows the results of the prediction of electric car demand with the GM algorithm. The blue line is the data value that we cleared the data, the data of the number of newly registered electric cars, and the red line is the line dividing the data value of the forecast of electric car demand. Note that the characteristics of the graph are similar and the value of the number of electric car demand tends to decrease continuously. Look at the value of the x-axis and the y-axis, which is the value of the number of electric cars decreasing. Later, in Figure 3-9, the graph shows the results of the prediction of electric car demand with the genetic algorithm as shown in the picture. As shown in Figure 3-9

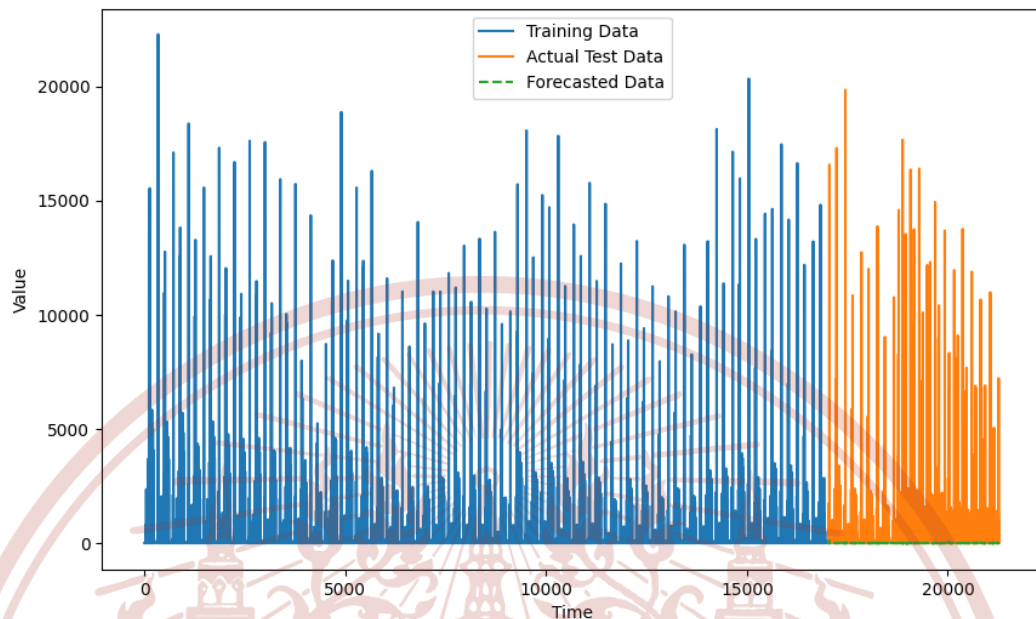


Figure 3-9 Graph of the forecast results using the genetic algorithm model

From Figure 3-9, the graph shows the results of predicting electric car demand using the Genetic Algorithm (GA). The blue line represents the actual data, which is the number of newly registered electric cars after data cleaning, and the orange line represents the predicted values of electric car demand using the GA algorithm. From the graph, it can be observed that this is the only algorithm where the forecast values do not differ much from the original data, and the data values appear to be relatively constant compared to the original data. The x-axis and y-axis values show that the predicted number of electric car demands is very close to the actual values. Finally, in Figure 3-10, the graph displays the results of predicting electric car demand using the Gray Model-Genetic Algorithm (GM-GA) as shown in the picture. As shown in Figure 3-10

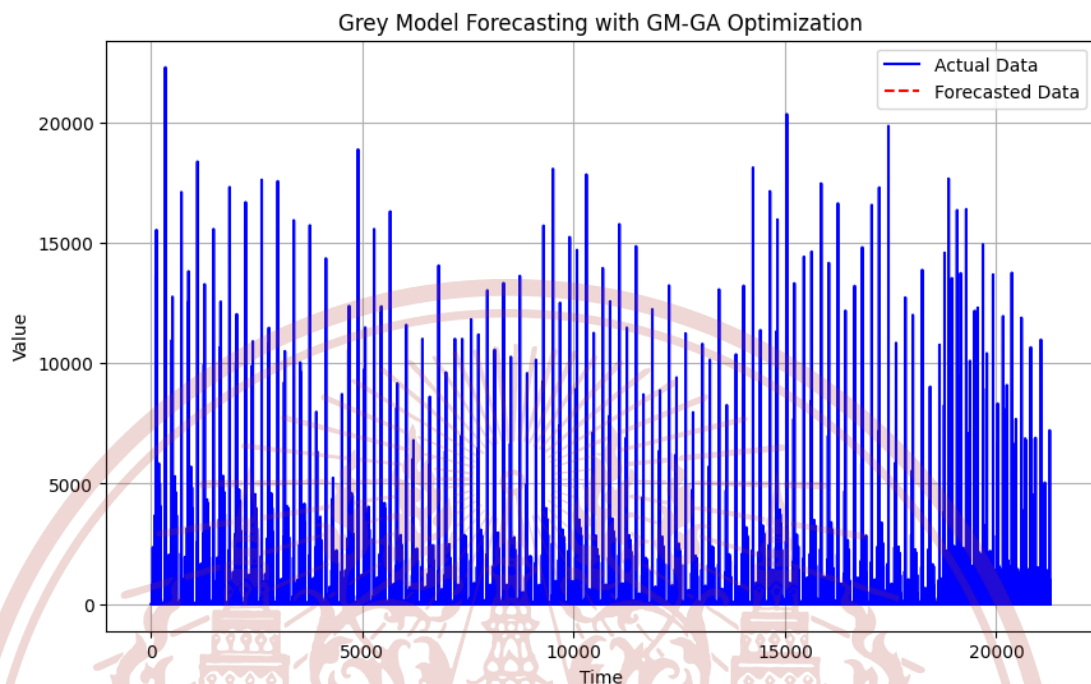


Figure 3-10 Graph of the forecast results using the grey model-genetic algorithm model (GM-GA)

Figure 3-10, the graph shows the results of the prediction of electric car demand using the grey model-genetic algorithm (GM-GA) algorithm as shown in the figure. Characteristics The blue line graph shows that the graph will gradually decrease and remain constant, indicating that the demand for electric cars will be high in the beginning and will gradually decrease and remain constant after that. This can be seen from the x-axis and y-axis, which represent the number of new electric car registrations. The graph shows the results of the forecast. After the forecast is complete, the accuracy and error values of the final algorithm will be checked in Figure 3-11. The graph shows the results of the prediction of electric car demand using the Residual revised gray model integrated with the genetic algorithm (R2GM-GA) algorithm as shown in Figure 3-11.

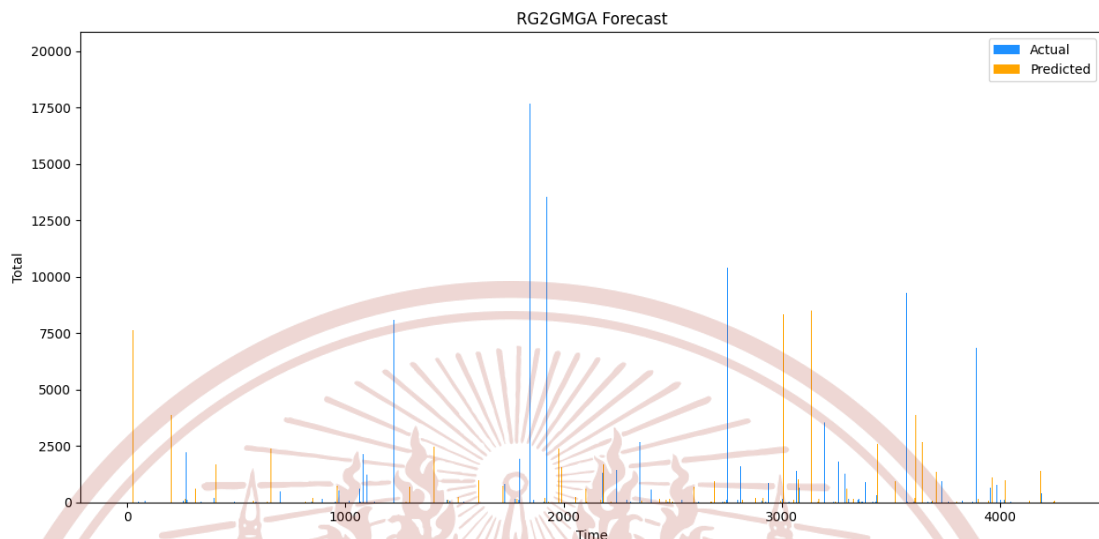


Figure 3-11 Graph of the forecast results using the Residual revised gray model integrated with the genetic algorithm (R2GM-GA)

Figure 3 - 11, the graph shows the results of predicting electric vehicle demand using the Residual revised gray model integrated with the genetic algorithm (R²GM-GA) as depicted in the figure. The characteristics of the graph indicate that the demand for electric vehicles will initially be high, then gradually decrease and stabilize. This is reflected in the blue line, which shows a trend of decreasing and leveling off. The x-axis and y-axis represent the number of new electric vehicle registrations. After completing the forecast, the accuracy and error values of the algorithm will be assessed. After getting the error value of the future demand forecast for electric cars. Every technique is then used to find the Errors analysis by taking out the error values of forecasting the demand for electric cars in the future. Let's find the accuracy of the error using the error checking technique Mean absolute error (MAE), Root mean squared error (RMSE), and Mean Absolute Percentage Error (MAPE), then compare the obtained values to see which one has the least Errors analysis value. The results showed that the grey model (GM) integrated with the genetic algorithm (GA) had the least analysis errors, showing that it had the most accurate error in forecasting future demand for electric cars. As shown in Table 3-4 Table comparing the accuracy values obtained from the results of each algorithm.

Table 3-4 Table comparing the accuracy values obtained from the results of each algorithm.

Algorithm	Mean Absolute Percentage Error	Mean Absolute Error	Root Mean Squared Error
LSTM	inf	286.97	1,183.24
RNN	inf	277.832	1,208.928
ANN	inf	221.34965	1,222.528
GM	Inf	176.69596	186.53869
GA	Inf	0.0016025	0.003126
GM-GA	Inf	2213.75	4,027.16
R2GM-GA	inf	0.0000001362122	0.0000004890965

The results indicate that the R2GM-GA model achieved the lowest MAE and RMSE values, with the minimum values being 0.0000001362122 and 0.0000004890965, respectively. This suggests that, among the models evaluated—namely ANN, RNN, LSTM, GM, GA, and GM-GA—the R2GM-GA model demonstrates the highest forecasting accuracy. Error values were assessed using MAPE, MAE, and RMSE metrics, and the R2GM-GA model consistently showed the lowest values across all measures. These findings indicate that the R2GM-GA model provides the most precise predictions for forecasting the number of electric vehicles, supporting the conclusion that it possesses the highest overall accuracy among the tested models.

3.8 The demand for electric cars on website

After developing the forecasting model, it was used to predict the number of electric vehicles for a 10-month period in 2024. The forecasted values were then compared with the actual registration data obtained from the Thailand Automotive Information Center (<https://data.thaiauto.or.th/auto/auto-stat/auto-registration/auto-register.html>) to evaluate the accuracy of the predictions. The results were subsequently visualized in graphical form and presented on a website to facilitate more efficient and accessible interpretation of the forecasting outcomes.

3.9 Check Answer

In conclusion, after studying various e-journals, it can be concluded that the research has used the Predicting Flow Method, which is a technique used in forecasting or predicting the flow of various resources, such as traffic, product planning, or even forecasting happiness. This method uses mathematical and statistical models to predict the direction or amount that will occur in the future. The purpose is to manage and plan the use of resources efficiently. It is divided into 30 to 70, 80 to 20, or 60 to 40,

depending on how far in advance we forecast. Most of it is data that we use relationships and is real data that comes continuously. On the other side is data that is used to forecast through various models, such as time series models such as ARIMA and Holt-Winters, which are suitable for predicting data with continuous data flow, and machine learning models such as random forest and gradient boosting that can learn complex flow patterns. And neural networks such as RNN and LSTM, which are suitable for predicting time series data. And the flow with an uncertain or fluctuating pattern, then take the obtained data to check with various error values and accuracy values, such as mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE), which the researcher used all 3 of these techniques to check the error value and accuracy value because MAE is suitable for evaluating models that work with small data sets or incomplete data. Because it is not necessary to have a lot of data for the results to be reliable and suitable for measuring the accuracy of forecasts with the Gray Model (GM), because GM is often used to predict data with limited samples or high uncertainty. MAE error measurement helps to show the difference between the predicted value and the actual value at an easy-to-understand level, regardless of the size of the data or the distribution of outliers. It is suitable for GA when GA adjusts the data to create a GA forecasting model. The inspection is not complicated and can be inspected (Rob J. Hyndman and Koehler, 2006). RMSE is suitable for the type of work that requires a quantitative measure of the difference between the predicted value and the actual value. RMSE provides detailed information about the error in the model, such as time series forecasting, such as sales forecasting, price forecasting, or demand forecasting. RMSE helps to measure the severity of the forecast error or to check the sales. Next, MAPE is suitable for the type of forecasting data that has no zero values or very low values: MAPE may not be appropriate in cases where the actual value is close to zero because it will show an unusually high percentage. Therefore, MAPE should be avoided when there is data with zero values or very low values. Quantitative data: MAPE is good when the forecasted data is quantitative, such as sales, production volume, or finance, which can be measured imprecisely. Data analysis in a business context: MAPE is popular in business and marketing because displaying the results in percentage terms makes it easy to understand and can be communicated well at the executive level. Therefore, in this research, the data will be divided into 80:20 parts. 80 is the original data and the actual value. 20 is the data that forecasts the demand for electric vehicles in 2023 and 2024 using the data from 2022-2023 itself using the R2GM-GA model and using three techniques to check the error value and the accuracy value.

CHAPTER 4 RESEARCH RESULTS

The results of the electric vehicle registration forecasting project were divided into two parts:

- 4.1 Application to two-year forecasting
- 4.2 Relationship between electric vehicles and electric vehicle charging stations

4.1 Application to two-year forecasting

After we have used all 7 models to forecast and check the error values in Chapter 3, this chapter will forecast 2 years in advance from the data using LSTM, RNN, ANN, GM, GA, Gm-GA, and R2GM-GA models. The results are shown in Table4-1

Table4-1 Forecasting 2 years

Date	Actual	ANN	RNN	LSTM	GM	GA	GMGA	R2-GMGA
1/1/2567	13,974	10,696	7,486	10,200	34,199	11,695	27,303	6,500
1/2/2567	4,128	13,665	12,173	11,087	392,215	13,314	30,690	6,625
1/3/2567	5,224	14,100	14,259	12,024	449,811	14,078	34,498	6,759
1/4/2567	5,677	12,826	10,130	13,013	515,861	15,397	38,778	6,903
1/5/2567	7,692	14,616	17,867	14,078	591,606	16,860	43,589	7,056
1/6/2567	5,888	16,147	19,476	15,397	678,469	18,480	48,997	7,225
1/7/2567	6,020	15,799	21,250	16,860	778,082	20,550	55,075	7,399
1/8/2567	8,354	17,580	25,771	18,480	892,316	22,983	61,908	7,590
1/9/2567	6,288	19,628	28,858	20,550	1,023,318	25,929	69,589	7,794
1/10/2567	6,631	21,918	36,064	22,983	1,173,549	29,560	78,222	8,013
TOTAL	69,876	156,974	193,156	154,673	6,837,219	224,434	488,649	71,864

Table 4-1 presents the division of the dataset into training and testing subsets using a 90:10 ratio. In this study, 90% of the dataset—comprising electric vehicle (EV) registration data from 2019 to 2023 obtained from the Thailand Automotive Industry Data Center—was utilized as the training set. This portion of the data was applied for model construction, parameter estimation, and preliminary performance evaluation, as well as for generating 10-month-ahead forecasts. The remaining 10% of the dataset, consisting of actual EV registration data from January to October 2024, was reserved as the testing set. This subset was used to assess the predictive performance and generalization ability of each forecasting model by comparing the forecasted values with the actual observed data. The selection of the 90:10 ratio is appropriate for this

study, as the primary objective is to forecast EV adoption in 2024—a period expected to exhibit substantial growth. Moreover, prior studies have demonstrated that this proportion provides a suitable balance between training efficiency and evaluation reliability (Nayak et al., 2024). As shown in Table 4-1, the first column represents the actual EV registration values for 2024, while the subsequent columns display the corresponding 10-month forecast results generated by each model. The models evaluated in this study include Artificial Neural Network (ANN), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Grey Model (GM), Genetic Algorithm (GA), hybrid Grey Model with Genetic Algorithm (GM-GA), and the proposed Residual Revised Grey Model with Genetic Algorithm (R2GM-GA). The results indicate that the R2GM-GA model achieved the highest level of forecasting accuracy among all models considered. Specifically, the predicted total number of EV registrations using the R2GM-GA model was 69,876 vehicles, which is closest to the actual observed value of 71,864 vehicles reported by the Thailand Automotive Industry Data Center. This finding is further supported by the comparative results presented in Table 4-5. These results suggest that the R2GM-GA model provides superior predictive performance in capturing the trend of EV adoption. Therefore, it can be considered the most suitable model for forecasting EV demand in this study. Subsequently, the forecasting errors of each model were evaluated using standard error metrics, as described in the following section (Rob J Hyndman, 2014).

$$Error = \left| \frac{Forecast - Actual}{Actual} \right| \cdot 100 \quad (4-1)$$

The variables are defined as follows: Error represents the percentage error between the forecasted and actual values. Forecast denotes the predicted value generated by the model. Actual refers to the observed value recorded in the dataset. Based on the data presented in Table 4-1, the following equation is used:

$$Error = \left| \frac{71,864 - 69,876}{69,876} \right| \cdot 100$$

$$Error = \left| \frac{1,988}{69,876} \right| \cdot 100 \quad (4-2)$$

$$Error = 0.02845 \cdot 100$$

$$Error = 2.845$$

Based on Equation 4 - 2 , the calculated forecasting error is 2 . 8 4 5 % . In forecasting practice (Rob J Hyndman, 2014), An error value of less than or equal to 5% is generally considered acceptable for reliable prediction performance. Therefore, the error obtained in this study indicates a high level of accuracy, as the deviation between

the forecasted and actual values is relatively small. This result suggests that the dataset and the applied forecasting models are capable of effectively capturing the underlying trend of electric vehicle (EV) registrations. Consequently, the forecasting outcomes can be regarded as reliable for further analysis and decision-making. Following the error evaluation, the forecasting results were visualized in graphical form to facilitate interpretation and comparison across models. The graph illustrates the predicted number of EV registrations generated by each model alongside the actual observed values. To further enhance accessibility and usability, a web-based application was developed to present the forecasting results interactively. This platform allows users to conveniently explore the data and compare model performance through graphical visualization. An example of the graphical output is presented in Figure 4-1.

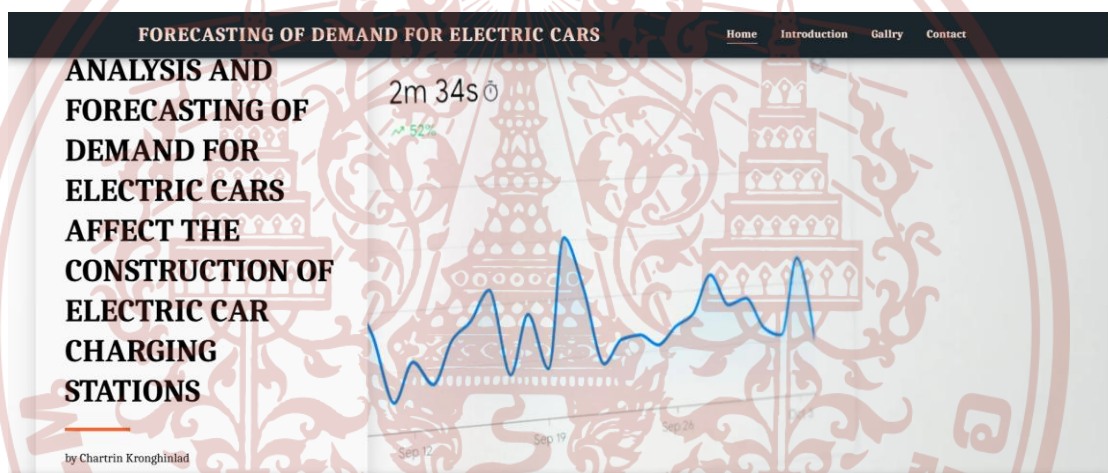


Figure 4-1 Home web Forecasting

Figure 4-1 illustrates the website interface used to present the forecasting results, which is organized into several functional sections. The Home section presents the title of the research. The Introduction section provides an overview of the research topic and a summary of the key findings. The Gallery section displays the forecasting results for each model in graphical form. These results are presented across seven forecasting models: ANN, RNN, LSTM, GM, GA, GM-GA, and R2GM-GA. The Table section presents the forecast results in tabular form for each model and also includes tables demonstrating the relationship between electric vehicle purchases and the number of electric vehicle charging stations constructed. Additionally, links are provided to show the number of electric vehicles charged per day and the annual number of charging stations. The information is organized in a table format for ease of interpretation. Finally, the Contact section provides the researcher's contact information and institutional affiliation for further communication. Figure 4-2 presents the graphical forecast results of the seven models: ANN, RNN, LSTM, GM, GA, GM-GA, and R2GM-GA.

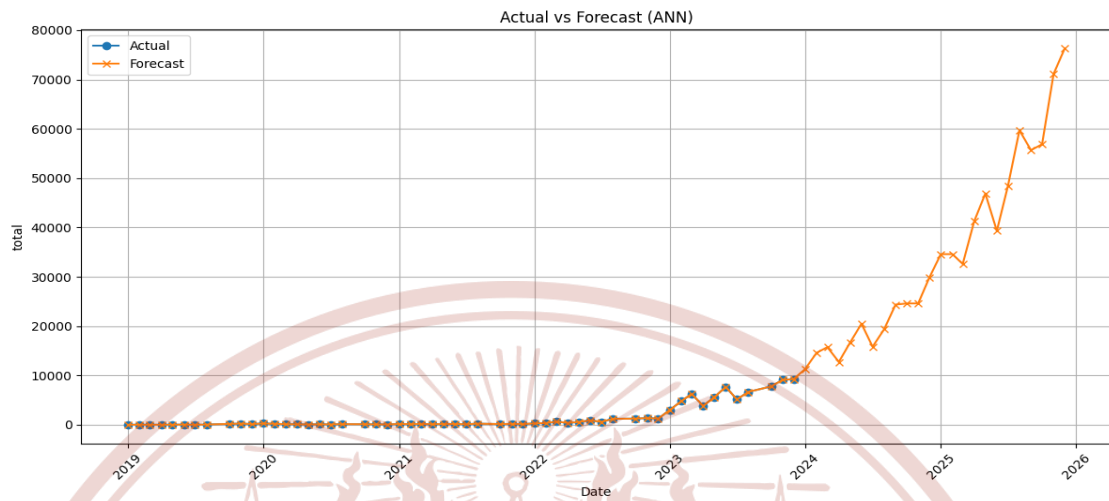


Figure 4-2 Graph of the forecast 2-year results using the ANN algorithm

Figure 4-2 presents the forecasted number of electric vehicles (EVs) generated using the artificial neural network (ANN) model. The horizontal axis represents the year used for forecasting, while the vertical axis indicates the total number of EVs. The figure shows that the number of EVs remains relatively constant during the initial years, reflecting the early stage of EV adoption in Thailand. In subsequent years, the forecast indicates a steadily increasing growth trend through 2024. This upward trajectory is consistent with the expanding interest in and adoption of EVs in Thailand. Next are shown Figure 4-3 presents the advanced forecast of electric vehicle registrations generated using the long short-term memory (LSTM) and recurrent neural network (RNN) models.

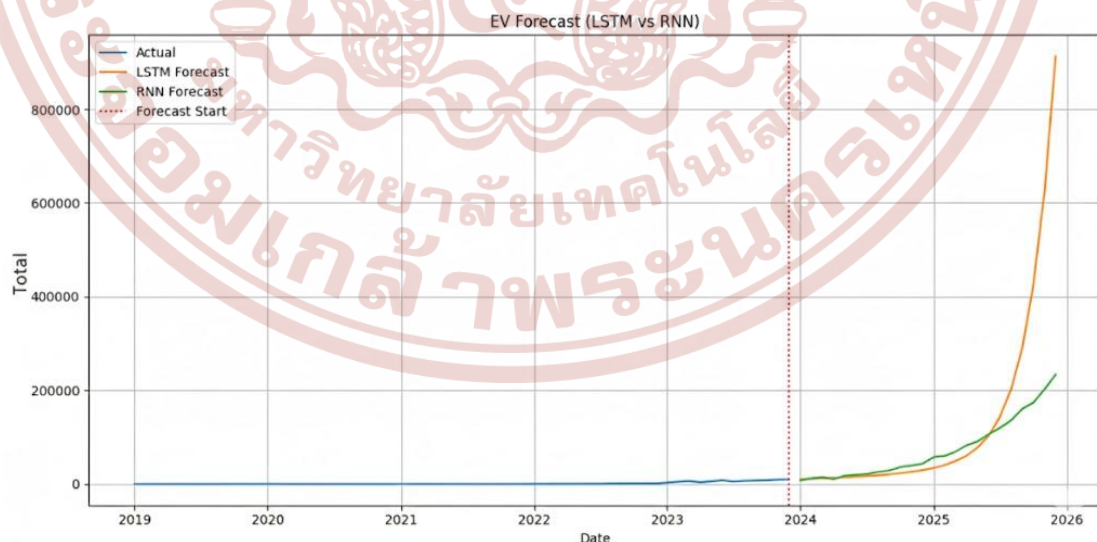


Figure 4-3 Graph of the forecast 2-year results using the LSTM, RNN algorithm

Figure 4-3 illustrates the forecasted number of electric vehicles (EVs) obtained using the recurrent neural network (RNN) and long short-term memory (LSTM) models. The horizontal axis represents the year used for forecasting, and the vertical

axis indicates the total number of EVs. The green line corresponds to the RNN-based forecast, while the yellow line represents the LSTM-based forecast. The red dotted line denotes the boundary between the test data and the forecasted values for 2024. These two models are compared due to their similar underlying structures, with the primary distinction being that the LSTM model incorporates additional gating mechanisms to enhance long-term dependency learning. The figure shows that the number of EVs remains relatively constant during the initial years, reflecting the early stage of EV adoption in Thailand. In the later years, and through the forecast horizon up to 2024, both models indicate a clear upward trend, which is consistent with the increasing interest in EVs in Thailand. A notable difference between the models is observed in the first nine years, during which the RNN model produces higher forecasted values than the LSTM model. Although the LSTM forecasts are slightly lower during this period, its predicted value toward the end of 2024 exceeds the long-term estimates of the RNN model, which extends into 2026. However, despite these differences in magnitude, a higher forecasted value does not necessarily indicate superior model performance. Instead, model suitability should be evaluated based on error metrics, which provide a more reliable measure of forecasting accuracy in accordance with APA reporting standards. Next are shown Figure 4-4 presents the forecasted number of electric vehicles generated using the Grey Model (GM).

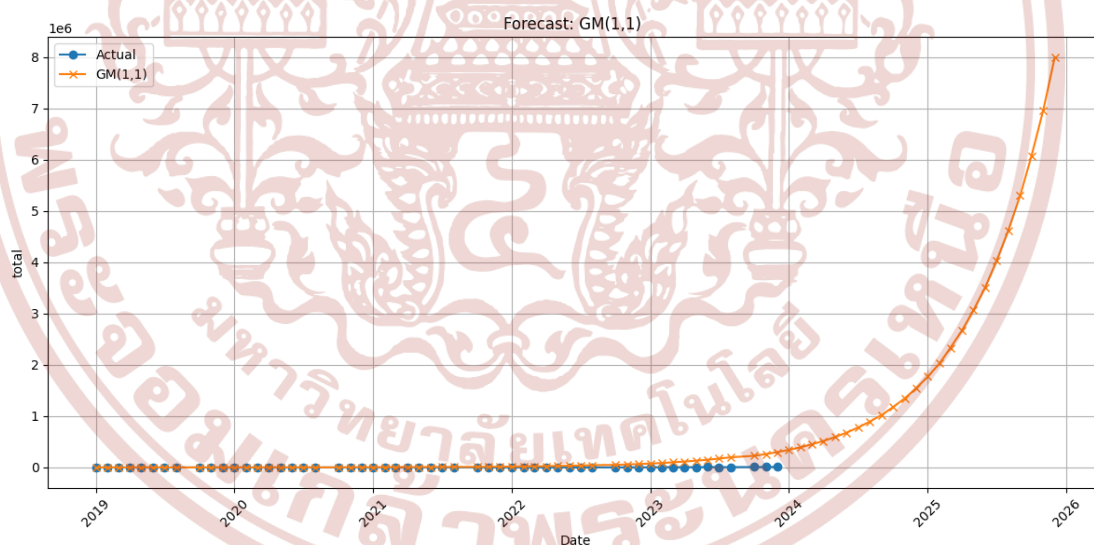


Figure 4-4 Graph of the forecast 2-year results using the GM algorithm

Figure 4-4 presents the forecasted number of electric vehicles (EVs) generated using the Grey Model (GM). The horizontal axis represents the year used for forecasting, while the vertical axis indicates the total number of EVs. The figure shows that the number of EVs remains relatively stable during the initial years, reflecting the early stage of EV adoption in Thailand. In the subsequent years, and through the forecast horizon up to 2024, the results indicate an increasing growth trend. This upward pattern is consistent with the expanding interest in and adoption of EVs in

Thailand. Next are shown Figure 4-5 presents the advance forecast of electric vehicles generated using the genetic algorithm (GA) model.

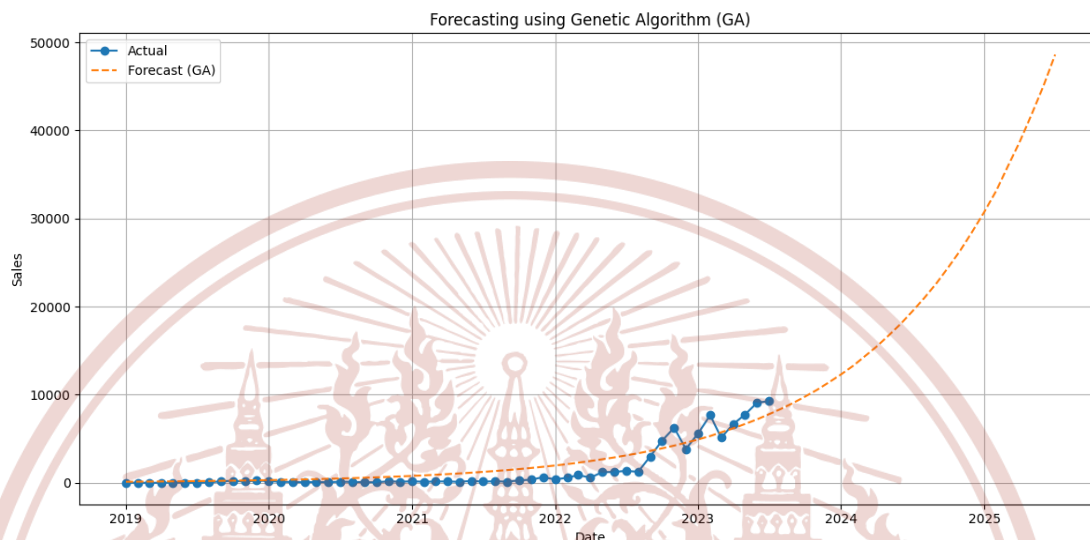


Figure 4-5 Graph of the forecast 2-year results using the GA algorithm

Figure 4-5 presents the forecasted number of electric vehicles (EVs) generated using the genetic algorithm (GA) model. The horizontal axis represents the year used for forecasting, while the vertical axis indicates the total number of EVs. The figure shows that the number of EVs remains relatively stable during the initial years, reflecting the early stage of EV adoption in Thailand. In the later years, and through the forecast period up to 2024, the results indicate a steadily increasing growth trend. This upward trend is consistent with the growing interest in and adoption of EVs in Thailand. The next are shown Figure 4-6 presents the forecasted number of electric vehicles generated using the GM-GA model.

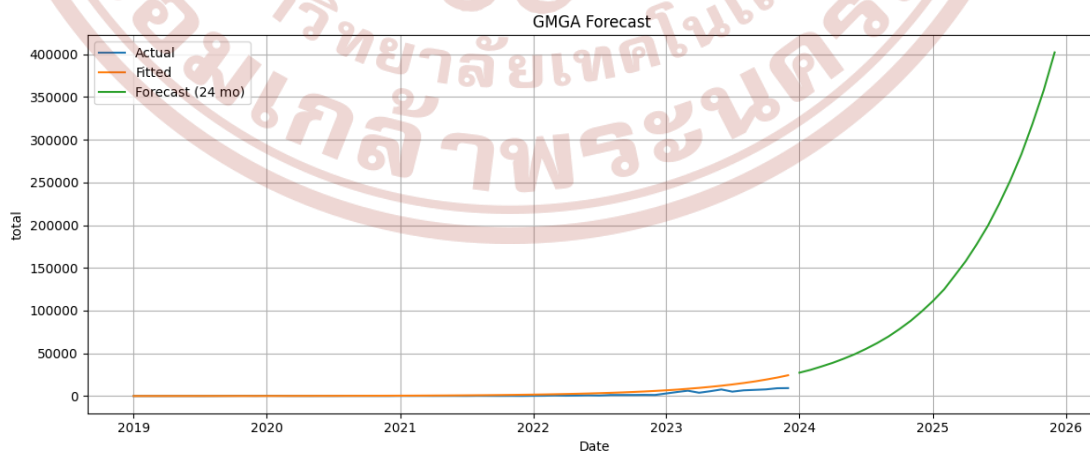


Figure 4-6 Graph of the forecast 2-year results using the GM-GA algorithm

Figure 4-6 presents the forecasted number of electric vehicles (EVs) generated using the Grey Model–Genetic Algorithm (GM–GA) model. The horizontal axis represents the year used for forecasting, while the vertical axis indicates the total number of EVs. The figure shows that the number of EVs remains relatively stable during the initial years, reflecting the early stage of EV adoption in Thailand. In the subsequent years, and throughout the forecast horizon up to 2024, the results demonstrate an increasing trend. This upward pattern is consistent with the growing interest in and adoption of EVs in Thailand. The next are shown in Figure 4-7, which presents the forecasted number of electric vehicles generated using the residual revised grey model combined with a genetic algorithm (R2GM–GA). This model, as investigated by researchers, has not been previously applied in studies on electric vehicle forecasting.

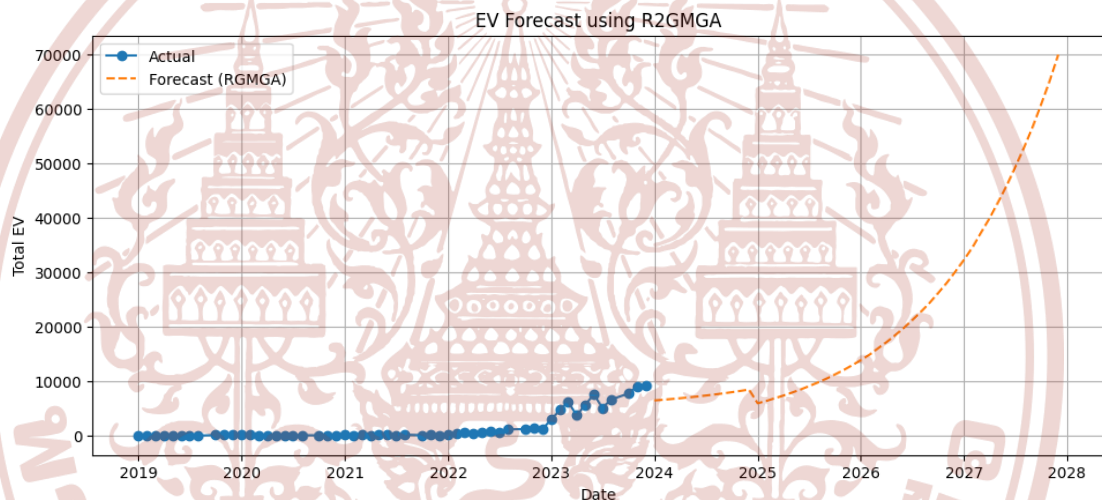


Figure 4-7 Graph of the forecast 2-year results using the R2GMGA algorithm

Figure 4-7 presents the forecasted number of electric vehicles (EVs) generated using the residual revised grey model combined with a genetic algorithm (R2GM–GA). The horizontal axis represents the year used for forecasting, while the vertical axis indicates the total number of EVs. The figure shows that the number of EVs remains relatively stable during the initial years, reflecting the early stage of EV adoption in Thailand. In the later years, and throughout the forecast horizon up to 2025, the results indicate an increasing trend, which is consistent with the growing interest in and adoption of EVs in Thailand.

After obtaining the forecasted values for 2024, the researchers further evaluated the performance of all models by comparing the predicted results with the actual number of registered EVs in 2024, as collected by the researchers. Model accuracy was assessed using the mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). The comparative error metrics for all models are shown in Table 4-2.

Table 4-2 Check error table

Model	MAE	RMSE	MAPE(%)
ANN	17,490.08	29,851.96	179.39
RNN	24,653.11	41,953.17	247.01
LSTM	17,162.79	29,212.71	173.59
GM	1,23,542.00	2,156,868.00	11,790.06
GA	29,574.49	51,589.75	292.88
GMGA	77,257.10	135,362.51	749.64
R2GMGA	1,027.22	2172.95	158.07

From Table 4-2 presents the forecasting error results for predicting the number of electric vehicle (EV) registrations using several forecasting models. The models examined in this study include Artificial Neural Network (ANN), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Grey Model (GM), Genetic Algorithm (GA), Grey Model optimized by Genetic Algorithm (GMGA), and the proposed model developed in this research, namely the Residual Revised Grey Model optimized by Genetic Algorithm (R2GMGA). The forecasting performance of each model was evaluated by comparing the predicted values with the actual EV registration data presented in Table 4-3. Subsequently, the forecasting errors were calculated using three statistical indicators: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). These indicators were used to systematically assess and compare the forecasting accuracy of the different models. The results show that the ANN model produced an MAE of 17,490.08, an RMSE of 29,851.96, and a MAPE of 179.39. These findings indicate that the ANN model was able to capture the general trend of the data; however, the magnitude of forecasting errors remained relatively high compared with some other models. For the RNN model, the MAE was 24,653.11, the RMSE was 41,953.17, and the MAPE was 247.01. The results indicate that the forecasting performance of the RNN model was lower than that of the ANN model, as reflected by higher error values across all evaluation indicators. The LSTM model, which is specifically designed for time-series data analysis, demonstrated improved forecasting performance. The LSTM model produced an MAE of 17,162.79, an RMSE of 29,212.71, and a MAPE of 173.59. These results suggest that the LSTM model was more effective in capturing temporal patterns in EV registration data compared with the ANN and RNN models, resulting in lower forecasting errors. In contrast, the Grey Model (GM) produced the highest forecasting errors among all models examined in this study. The GM model resulted in an MAE of 123,542.00, an RMSE of 2,156,868.00, and a MAPE of 11,790.06. These extremely large error values indicate that the GM model was not suitable for forecasting EV registration data, which exhibits considerable fluctuations and nonlinear characteristics. For the Genetic Algorithm (GA) model, the MAE was 29,574.49, the RMSE was 51,589.75, and the MAPE was 292.88. Although the GA model performed better than

the GM model, its forecasting accuracy was still lower than those of the neural network-based models. The GMGA model, which integrates the Grey Model with parameter optimization using a Genetic Algorithm, showed improved performance compared with the original GM model. However, the forecasting errors remained relatively high, with an MAE of 77,257.10, an RMSE of 135,362.51, and a MAPE of 749.64. This finding indicates that although Genetic Algorithm optimization can improve the Grey Model to some extent, the improvement was insufficient for achieving high forecasting accuracy for the dataset used in this study. In contrast, the proposed R2GMGA model demonstrated the best forecasting performance among all evaluated models. The R2GMGA model produced the lowest error values, with an MAE of 1,027.22, an RMSE of 2,172.95, and a MAPE of 158.07. These results indicate that incorporating residual correction together with Genetic Algorithm optimization significantly improved the forecasting accuracy of the Grey Model. When the models were ranked according to the lowest error values for each performance indicator, the R2GMGA model consistently achieved the best performance. For the MAE indicator, the ranking from best to worst was R2GMGA, LSTM, ANN, RNN, GA, GMGA, and GM. A similar ranking pattern was also observed for the RMSE and MAPE indicators. Further analysis was conducted by comparing the forecasting values presented in Table 4-5 with the error results summarized in Table 4-6. The percentage error calculation showed that the R2GMGA model achieved the lowest error rate, with a value of 2.84%. This error level is considered highly reliable, as it is lower than 5%. Based on these findings, it can be concluded that the R2GMGA model provides the highest forecasting accuracy for predicting the number of electric vehicle registrations in Thailand during the forecasting period. The model produced forecasting values that were closest to the actual observations and demonstrated the lowest error levels across all evaluation criteria. Therefore, this study suggests that the R2GMGA model is the most suitable and effective forecasting approach for predicting future trends in electric vehicle registrations in Thailand.

4.2 Relationship between electric vehicles and electric vehicle charging stations

After we have the EV registration figures, we wonder if there are enough charging stations for the new EVs ordered each year. Will each station be sufficient for the number of charging stations? We haven't conducted any research and collected data on how many EV charging stations there are, the number of charging heads at each station, what types they are, and what operating hours they have. However, there is currently no data on EV charging times, the number of vehicles in use, and how long each vehicle takes to charge. Therefore, we use a hypothetical formula and average values to determine how increasing EV purchases affect the construction of EV charging stations, using the following three formulas (Ewan Pritchard, 2023).

$$\text{Vehicles per charger per day} = \frac{\text{Operational hours per day} \times \text{utilization}}{\text{Average session hours}} \quad (4-3)$$

Where:

Vehicles per charger per day = The charging capacity of the charger that can charge a vehicle per day.

Operational Hours per Day = The number of hours the station is open.

Utilization = The actual usage rate.

Average Session Hours = The average time required for a single charge.

Where the formula for finding the actual utilization rate is as follows: (Mahlberg et al., 2023)

$$utilization = \frac{charginghours}{availablehours} \quad (4-4)$$

The variables are defined as follows:

Charging hours are the actual time used.

Available hours refer to the time slots during which service is available at each station.

After calculations, it was found that there were 1,590 EV charging stations nationwide from 2019 to 2023, with a total of 5,062 charging heads. Each charging head can fully charge 8.4 EVs per day, (Thailand, 2023) and the total capacity nationwide is 42,521 EVs per day. The following table displays the annual volume of EV purchases and the total number of EVs sold. As shown in Table 4-3

Table 4-3 EV Charging Station Performance Table

Years	Total	Total all	Charge per Day
2019	1,381		-
2020	1,400	2,781	-
2021	1,973	4,754	-
2022	9,892	14,646	-
2023	76,025	90,671	42,521

The table presents the annual number of registered electric vehicles (EVs) and the cumulative total for each year. The variable “Charge per day” represents the daily charging service capacity of EV charging stations, while “Minus EV” denotes the difference between the cumulative charging capacity and the cumulative number of registered EVs for each year. The results indicate that in 2023, the calculated difference was negative (−70,018 vehicles), implying that the existing charging infrastructure capacity was insufficient to support the cumulative number of EVs in that year. This finding reflects a disparity between the growth rate of EV registrations and the expansion of charging station capacity. To further examine this situation, data on the number of EV charging station users across Thailand during the period 2019–2023 were collected for comparative analysis. The objective was to assess whether the trend in charging station usage corresponds with the increase in registered EVs. Table 4-4 presents a comparison between the total number of registered EVs and the number of EV charging station users in Thailand, in order to evaluate the consistency of trends between vehicle growth and infrastructure utilization.

Table 4-4 Population EV Charging stations

Years	Total	Users
2019	1,381	1,690
2020	1,400	13,185
2021	1,973	46,355
2022	9,892	117,595
2023	76,025	420,390

Table 4-4 presents the annual number of registered electric vehicles (EVs) and the number of users of EV charging stations in Thailand. The “Year” column covers the period from 2019 to 2023. The “Total” column indicates the number of EVs registered in each year, while the “Users” column represents the recorded number of charging station users during the same period. It should be noted that the charging station user data were obtained from selected charging station providers operating between 2019 and 2023. Not all service providers disclosed their data due to confidentiality considerations. As a result, the available dataset does not cover all brands in the market. To address this limitation, it was assumed that providers who did not disclose their data exhibited usage patterns comparable to those who did. The aggregated data were then compared with the EV registration figures presented in Table 4-4. The comparative analysis indicates that the number of registered EVs is lower than the reported number of charging station users. Based on this observation, forecasting techniques were applied to assess whether the current charging infrastructure would be sufficient to accommodate future demand. The forecasting results are presented in Table 4-5, which compares the projected number of users from each model with the existing charging station capacity.

Table 4-5 Model Forecasting with Users charging stations

Model	total	Users
ANN	156,974	599,215
GM-GA	488,649	599,215
GM	6,837,219	599,215
GA	224,434	599,215
LSTM	154,673	599,215
RNN	193,156	599,215
R2GMGA	71,864	599,215

Table 4-5 presents a comparison between the forecasting results generated by each model and the total number of EV charging station users. The “Model” column identifies the forecasting techniques applied in this study. The “Total” column reports the 10-month projected number of electric vehicles derived from each model, while the “Users” column represents the cumulative number of charging station users recorded during the period 2019–2023. The comparison indicates that the GM-GA model provides the closest estimate to the observed number of users (488,649), followed by

GA (224,434), RNN (193,156), ANN (156,974), LSTM (154,673), and R2GMGA(71,864). In contrast, the GM model produces a substantially higher estimate (6,837,219), which deviates considerably from the recorded number of users. Although the R2GMGA model does not yield values closely aligned with the total number of charging station users, it should be noted that the number of electric vehicles used in this analysis is partially estimated. Due to confidentiality constraints, some companies did not disclose complete data, and certain values were therefore derived under specified assumptions. Furthermore, the results demonstrate that the total number of charging station users exceeds the number of registered EV purchases. This finding suggests that individual vehicles access charging stations multiple times within a given period, resulting in usage frequency greater than one charge per vehicle per day on average. Table 4-6 further examines this relationship by presenting the estimated number of usage days required for the total charging station usage to correspond to the total number of registered EV purchases.

Table 4-6 EV Charging Cars Per Day Table

Day	EV Charging Cars
1	42,521
2	85,042
3	127,563
4	170,084
5	212,605
7	297,647

Table 4-4 presents an analysis of the charging behavior of electric vehicles (EVs) based on their battery capacity and average daily travel distance. The results indicate that when an EV battery is fully charged to its maximum capacity, the vehicle can typically operate for approximately 3–4 days before requiring another charge. This finding corresponds with the battery capacity of EVs currently available in Thailand, which generally ranges between 40 and 80 kWh and provides an estimated driving range of approximately 250–500 kilometers, depending on driving conditions and driver behavior. In the context of urban commuting, particularly in Bangkok and its surrounding metropolitan areas, the average daily travel distance of EV users is estimated to be around 40–60 kilometers per day. Based on this travel pattern, a fully charged EV battery can support approximately 4–5 days of commuting before recharging is required. However, the duration may vary depending on vehicle type and battery capacity. Smaller EV models may require charging more frequently, whereas medium and large EVs with larger battery capacities are generally capable of operating for up to five days per charge. For users who travel longer distances exceeding 50–60 kilometers per day, the effective driving duration per charge is reduced. Under such conditions, a single full charge may support only about two days of travel before the next charging session is required. Therefore, variations in travel distance and vehicle specifications significantly influence charging frequency. Based on the total number of registered electric vehicles, which currently stands at 90,761 vehicles, the analysis summarized in Table 4-6 indicates that an EV typically requires recharging every 3–4

days on average. To estimate the number of electric vehicles that can be supported by the existing charging infrastructure within a single day, this study applies a charging capacity estimation approach. The objective is to determine the maximum number of EVs that can be charged per day (Kingdom, 2026) based on the operational (Agency, 2024) The capability of charging connectors. The calculation framework used to estimate the daily charging capacity is presented in the following equation. (Borlaug et al., 2023)

$$EV = \frac{T}{CH} \quad (4-4)$$

The variables are defined as follows:

EV = the number of charging stations required to support electric vehicle charging demand per day.

T = the total number of electric vehicles considered in the study.

CH = the average number of electric vehicles that can be served by one charging connector per day.

From the proposed formula, the variables required for the calculation were obtained from the dataset used in this study (Xylia et al., 2025). The variable T represents the total number of electric vehicles (EVs) projected in this research to examine the relationship between EV adoption and the development of EV charging infrastructure. The projection period covers 2024–2026, with a total estimated number of 449,757 EVs during this period. The variable CH represents the average number of electric vehicles that a single charging connector can serve per day. In this study, the value of CH was calculated using Equations (4-3) and (4-4), which estimate the operational charging capacity of a connector. (Agency, 2024) The calculation results indicate that one connector can serve approximately 8.4 EVs per day. By substituting the obtained values of T and CH into the equation, the required charging capacity to support the projected number of EVs can be estimated. (Borlaug et al., 2023) The calculation is performed by dividing the total number of EVs (T) by the charging capacity per connector (CH). The calculation result is presented as follows.

$$EV = \frac{449,757}{8.4} \quad (4-5)$$

$$EV = 53,542.5$$

Based on the calculation results, the estimated number of electric vehicle (EV) charging connectors required to support the projected EV (Xylia et al., 2025) demand is 53,542.5 connectors nationwide. For practical interpretation, this value was rounded to 53,543 connectors. This estimated requirement was then compared with the current EV charging infrastructure in Thailand, which consists of 1,590 charging stations with a total of 5,062 charging connectors. To estimate the number of charging stations required to support the projected number of EVs, a proportional analysis was applied based on the existing ratio between charging stations and charging connectors. The analysis indicates that, to accommodate approximately 53,543 charging connectors nationwide, Thailand would require approximately 16,818 charging stations. This

estimated number of charging stations is considered sufficient to support the projected growth of electric vehicles by 2026



CHAPTER 5

DISCUSSION AND CONCLUSIONS

The results of the study, analysis and forecasting of demand for electric cars affects the construction of electric car charging stations, by the author, have created a website to display the results effectively by summarizing and discussing the results as follows.

- 5.1 Summary of Operation Results
- 5.2 Discussion of Operation Results
- 5.3 Recommendations and Development Guidelines

5.1 Operation Results

5.1.1 The article entitled Forecast of the Total Volume of Import–Export Trade Based on Grey Modelling Optimized by Genetic Algorithm applies the Grey Model (GM (1,1))(Xie and Xie, 2009b) and enhances its forecasting performance through the integration of a Genetic Algorithm (GA). The incorporation of GA aims to optimize the parameters of the grey model in order to improve forecasting accuracy. The findings reported in that article indicate that the GM-GA model substantially reduces forecasting errors when compared with the conventional GM (1,1), particularly in medium- and long-term forecasting horizons. These results confirm the suitability of GM-GA for time series data characterized by limited sample sizes and relatively simple monotonic trends. By contrast, the research presented in this thesis addresses a forecasting problem with markedly different characteristics. The analytical focus is placed on forecasting electric vehicle demand and examining its implications for the development of charging station infrastructure. Such a system is inherently complex and nonlinear, as it is influenced by multiple interrelated factors, including technological progress, consumer adoption behavior, and government policy interventions. Compared with the macroeconomic trade data employed in the referenced article, the data analyzed here exhibit greater variability and more pronounced nonlinear dynamics, thereby increasing the difficulty of accurate forecasting. Moreover, the methodological framework adopted in this thesis is not confined to the application of GM and GM-GA models. The analysis is extended to include artificial intelligence and deep learning approaches, namely Artificial Neural Networks (ANN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) networks. These models are well-suited to capturing nonlinear relationships and temporal dependencies within complex datasets. In addition, an enhanced hybrid approach, referred to as the Residual Revised Grey Model optimized by Genetic Algorithm (R2GM-GA), is introduced to address the limitations of the conventional GM-GA model. By incorporating residual error analysis and revision, this approach further improves forecasting performance. With regard to model evaluation, the referenced article primarily employs Relative Error as the main indicator of forecasting accuracy. In contrast, forecasting performance in this thesis is assessed using several widely accepted metrics, including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE). The use of multiple evaluation criteria allows for a more comprehensive and transparent comparison of model performance across different forecasting techniques. In summary, while the referenced article demonstrates the effectiveness of the GM-GA model for

forecasting relatively simple time series data, the results presented in this thesis suggest that forecasting problems characterized by higher complexity, such as electric vehicle demand, require more advanced hybrid methodologies. The combination of improved grey models with deep learning techniques yields superior forecasting accuracy and provides greater practical relevance for long-term policy formulation and infrastructure planning.

5.1.2 Enhanced grey forecasting models that incorporate optimization techniques demonstrate improved predictive performance when compared with the traditional GM(1,1). (Xie and Xie, 2009a) The integration of Genetic Algorithms contributes to more effective parameter estimation, leading to reduced forecasting error and increased model robustness across different evaluation metrics. These findings support the broader applicability of hybrid grey-based forecasting approaches. Distinct characteristics of electric vehicle demand, including policy-driven fluctuations and rapid technological change, introduce greater uncertainty and nonlinearity than those typically observed in conventional economic time-series data. Under such conditions, standard grey models and basic optimization frameworks may struggle to adequately capture dynamic patterns, limiting their forecasting accuracy. The Residual Revised Grey Model optimized by a Genetic Algorithm (R2GM-GA) addresses these challenges through the incorporation of residual correction into the modeling process. Systematic forecasting errors are identified and modeled separately, allowing the revised framework to enhance stability and accuracy. Lower MAE, RMSE, and MAPE values obtained by R2GM-GA in comparison with ANN, RNN, LSTM, GM, GA, and GM-GA indicate a clear improvement in predictive capability. Comparative evaluation with neural network-based models further highlights the advantages of the proposed approach. Although ANN, RNN, and LSTM are effective in capturing nonlinear relationships, their performance is often dependent on large datasets and extensive parameter tuning. The R2GM-GA model, by contrast, maintains strong predictive performance under conditions of limited data availability, reinforcing its suitability for forecasting problems characterized by small sample sizes and uncertain information. Overall, the integration of residual revision and Genetic Algorithm optimization extends the applicability of grey forecasting models to complex and rapidly evolving domains. This approach offers a reliable analytical framework for supporting decision-making in emerging energy and transportation sectors, particularly in contexts where data limitations and structural uncertainty remain significant challenges.

5.1.3 The Grey Model optimized by a Genetic Algorithm (GM-GA) (Jia et al., 2008) is primarily designed to enhance the forecasting performance of the conventional GM(1,1) through parameter optimization of its differential equation. Genetic Algorithms are employed as a systematic search mechanism to identify optimal parameter values under conditions of limited data availability. While this approach effectively mitigates certain limitations of the traditional Grey Model, (Jia et al., 2008) the underlying model structure remains unchanged. Consequently, systematic residual patterns arising from data volatility and nonlinear dynamics may persist in the forecasting outcomes. In contrast, the Residual Revised Grey Model optimized by a Genetic Algorithm (R2GM-GA) represents a structural enhancement of the grey forecasting framework. This approach explicitly incorporates residual analysis and correction into the forecasting process. Forecasting errors generated by the initial Grey Model are modeled separately and subsequently integrated into the final prediction.

When combined with Genetic Algorithm optimization, this residual revision mechanism substantially improves the model's ability to capture complex and dynamic data behavior. In essence, the fundamental distinction between GM-GA and R2 GM-GA lies in the scope of model enhancement. GM-GA focuses on parameter optimization within an unchanged model structure, whereas R2GM-GA simultaneously refines both the model structure and the treatment of residual information. This integrated approach results in greater flexibility, stability, and forecasting accuracy, particularly when applied to datasets characterized by high uncertainty and dynamic growth patterns, such as long-term electric vehicle demand. Research employing the Sequential-GMC(1,n)-GA model (Sapnken et al., 2022) primarily aims to enhance the predictive accuracy of Grey-based models through the application of a Genetic Algorithm (GA) for parameter optimization. This approach improves model performance, particularly in situations characterized by limited and incomplete data. Nevertheless, the methodology remains largely dependent on the original Grey Model structure and places limited emphasis on addressing the residual errors that persist after the initial forecasting process. (Sapnken et al., 2022) In contrast, the present study extends the GM-GA framework by introducing a Residual Revised Grey Model optimized by a Genetic Algorithm (R2GM-GA). By incorporating a residual revision mechanism alongside GA-based optimization, the proposed model is able to systematically learn and correct remaining forecasting errors, thereby improving its ability to capture complex and nonlinear patterns in time-series data. Furthermore, the performance of the R2GM-GA model is evaluated through comprehensive comparisons with several advanced forecasting approaches using multiple error metrics, including MAE, RMSE, and MAPE. These results demonstrate the suitability and robustness of the proposed model for practical applications involving real-world data.

5.1.4 The GM-GA model represents an extension of the conventional Grey Model (Yao et al., 2022) through the integration of a Genetic Algorithm to optimize the model parameters. This hybridization enhances the flexibility of the grey forecasting framework and improves predictive accuracy, particularly in situations where the available data are limited. (Yao et al., 2022) By employing GA, the parameter estimation process becomes more adaptive than that of the traditional GM, leading to improved forecasting performance. However, the GM-GA approach primarily concentrates on optimizing the parameters within the core forecasting stage and does not explicitly account for the structural patterns of residual errors that remain after the initial prediction process. In contrast, the R2GM-GA model further advances the GM-GA framework by incorporating a residual revision mechanism as an integral component of the model structure. This additional process enables the model to systematically learn from and correct the remaining forecasting errors after the first prediction stage. Moreover, in the R2GM-GA model, the Genetic Algorithm is not only utilized to optimize the grey model parameters but also to enhance the effectiveness of the residual correction process. As a result, the model is better equipped to capture complex and nonlinear characteristics inherent in real-world data. Empirical evaluation using MAE, RMSE, and MAPE demonstrates that R2GM-GA consistently outperforms the GM-GA model, indicating its superior capability in reducing forecasting errors and improving overall predictive accuracy.

5.1.5 The GM-GA model extends the conventional Grey Model by integrating a Genetic Algorithm (GA) to optimize the parameters of the core forecasting model.

(Ou, 2012) This integration enhances the adaptability of the model and leads to improved forecasting accuracy compared with the traditional Grey Model, particularly when the available data are limited. Nevertheless, the GM-GA approach primarily focuses on parameter optimization within the main forecasting stage and does not systematically address the structure or patterns of residual errors that remain after the initial prediction. In contrast, the R2GM-GA model represents a further advancement of the GM-GA framework by incorporating a residual revision mechanism as a fundamental component of the model. This mechanism enables the model to identify and correct systematic errors that persist after the first forecasting stage. Moreover, the application of GA in the R2GM-GA model is not confined to optimizing the parameters of the grey model alone but also extends to enhancing the residual adjustment process. Consequently, the model demonstrates an improved ability to capture complex and nonlinear data characteristics and exhibits clearly superior forecasting performance compared with GM-GA, as evidenced by lower MAE, RMSE, and MAPE values.

5.2 Discussion

From the results of the analysis and forecasting of demand for electric cars affecting the construction of electric car charging stations, the following conclusions were drawn:

5.2.1 For the forecasting of electric vehicle sales, the organizers were able to efficiently forecast the number of electric vehicles by developing a forecasting model called R2GMGA. The forecast used data from electric vehicle registrations from 2019-2023 and compared it with the actual data for the first 10 months of 2024. A graph was plotted to show and compare the yearly differences in the volume of electric vehicles. The actual forecasted numbers were also displayed to compare the closest and most efficient values. The model validation error, MAPE, MAE, and RMSE errors were 0.0000001362122 and 0.0000004890965. The above model error, which was compared to the actual values 10 months in advance, had an MAE is 13,908, RMSE is 23,191, and MAPE is 141.90, with the best error.

5.2.2 A study was conducted to determine the relationship between electric vehicle purchases and the construction of charging stations, focusing on the current charging stations available. This study examined how many vehicles a charging station can currently accommodate per day, how many days an electric vehicle can last on a full charge, and the hours each charging station is open. It also examined the distance traveled by electric vehicles per day. The results indicated that electric vehicle purchases influence charging station construction. In 2023, data showed a total of 1,587 electric vehicle charging stations and 5,062 charging stations, which could accommodate 45,521 electric vehicles. However, the current number of electric vehicles from 2019-2023 is 90,671, making it insufficiently capable of supporting electric vehicles, thus impacting charging station construction.

5.3 Recommendations and Development Guidelines

5.3.1 Development Guidelines:

Electric vehicle (EV) sales data play a crucial role in analyzing battery characteristics, energy consumption behavior, and EV classification. In this research, only Battery Electric Vehicles (BEVs) were included in the forecasting process, as they represent the most widely adopted EV type in Thailand and operate solely on electricity

without reliance on internal combustion engines. Plug-in Hybrid Electric Vehicles (PHEVs) were excluded from the analysis because they still partially depend on conventional fuel systems and currently represent a relatively small proportion of the EV market in Thailand. However, if PHEVs become more prevalent in the future, they may be incorporated into EV classification frameworks to improve forecasting accuracy through more comprehensive segmentation of vehicle types.

5.3.2 Development Guidelines:

It is recommended that data on the duration and frequency of EV charging station usage across the country be systematically collected and recorded. The availability of such detailed operational data would significantly enhance the accuracy of demand forecasting and support more effective long-term infrastructure planning.

5.3.3 Development Guidelines:

Future datasets on EV charging infrastructure should include spatial information, particularly the distance between charging stations within the same region or province. This would enable more precise analysis of station density, usage patterns, and the optimal allocation of charging points and stations in each area. In addition, the current availability of data remains limited, as some companies and organizations do not disclose detailed information regarding the number of charging stations and usage statistics due to confidentiality concerns. If more comprehensive data becomes accessible in the future, it would further improve the accuracy of forecasting models and strengthen the analysis of the relationship between EV adoption and charging infrastructure development.

5.3.4 Development Guidelines:

To improve the accessibility and usability of the results for decision-makers, a dashboard has been developed to present the findings in a clear and comprehensive format. However, since the current platform is web-based, it may present limitations in accessibility for some users. Therefore, future development could include the creation of a mobile application to enhance convenience, allowing executives and stakeholders to access the information more easily across various electronic devices.

REFERENCES

- Agency, I. E. (2024). Global EV Outlook 2024 Moving towards increased affordability. Retrieved from <https://shorturl.asia/K2AM6>
- Aishwarya. (2024). Introduction to Recurrent Neural Network. Retrieved from <https://shorturl.asia/CDmIL>
- Akanbi, K., Jones, Y., Oluwadare, S., & Nti, I. K. (2024, 13-14 April 2024). *Predicting Happiness Index Using Machine Learning*. Paper presented at the 2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI).
- Albadr, M. A., Tiun, S., Ayob, M., & AL-Dhief, F. (2020). Genetic Algorithm Based on Natural Selection Theory for Optimization Problems. *Symmetry*, 12(11), 1758. Retrieved from <https://www.mdpi.com/2073-8994/12/11/1758>
- Aldosari, H. (2024, 15-16 March 2024). *A Hybrid Support Vector Machine with Grasshopper Optimization Algorithm based Feature Selection for Load Forecasting*. Paper presented at the 2024 International Conference on Distributed Computing and Optimization Techniques (ICDCOT).
- Alkhafaij, M. A., Al-luhiby, H. A., Al-Hameed, M. R., Saleem, M., Habelalmateen, M. I., & Mohammed, E. A. (2023, 15-16 July 2023). *Hybrid Metaheuristics with Sparse-Trained Deep Learning for Sustainable Electric Vehicle Charging Demand Forecasting*. Paper presented at the 2023 6th International Conference on Engineering Technology and its Applications (IICETA).
- Anurakjaturong, T. (2020). Main components and parts of electric vehicles (EVs), including production costs. Retrieved from <https://shorturl.asia/KS8st>
- ArunKumar, K. E., Kalaga, D. V., Kumar, C. M. S., Kawaji, M., & Brenza, T. M. (2021). Forecasting of COVID-19 using deep layer Recurrent Neural Networks (RNNs) with Gated Recurrent Units (GRUs) and Long Short-Term Memory (LSTM) cells. *Chaos, Solitons & Fractals*, 146, 110861. doi:<https://doi.org/10.1016/j.chaos.2021.110861>
- Authority, M. E. (2023). What are the charging rates at PEA charging stations and what are the rates? Retrieved from <https://shorturl.asia/vYhT0>
- Baratsas, S. G., Niziolek, A. M., Onel, O., Matthews, L. R., Floudas, C. A., Hallermann, D. R., . . . Pistikopoulos, E. N. (2022). A novel quantitative forecasting framework in energy with applications in designing energy-intelligent tax policies. *Applied Energy*, 305, 117790. doi:<https://doi.org/10.1016/j.apenergy.2021.117790>
- Barus, E. S., Zarlis, M., Nasution, Z., & Sutarman. (2019, 28-29 Nov. 2019). *Forecasting Plant Growth Using Neural Network Time Series*. Paper presented at the 2019 International Conference of Computer Science and Information Technology (ICoSNiKOM).
- Borlaug, B., Yang, F., Pritchard, E., Wood, E., & Gonder, J. (2023). Public electric vehicle charging station utilization in the United States. *Transportation Research Part D: Transport and Environment*, 114, 103564. doi:<https://doi.org/10.1016/j.trd.2022.103564>

- Buzna, L., De Falco, P., Ferruzzi, G., Khormali, S., Proto, D., Refa, N., . . . van der Poel, G. (2021). An ensemble methodology for hierarchical probabilistic electric vehicle load forecasting at regular charging stations. *Applied Energy*, 283, 116337. doi:<https://doi.org/10.1016/j.apenergy.2020.116337>
- Cavallaro, C., Cutello, V., Pavone, M., & Zito, F. (2024). Machine Learning and Genetic Algorithms: A case study on image reconstruction. *Knowledge-Based Systems*, 284, 111194. doi:<https://doi.org/10.1016/j.knosys.2023.111194>
- Chaaban, A. K., & Alfadl, N. (2024). A comparative study of machine learning approaches for an accurate predictive modeling of solar energy generation. *Energy Reports*, 12, 1293-1302. doi:<https://doi.org/10.1016/j.egy.2024.07.010>
- Chaidee, N. (2023). Prepare the garage in front of your house. to support the future of electric cars. Retrieved from <https://shorturl.asia/Imdez>
- Chen, G. (2016). A Gentle Tutorial of Recurrent Neural Network with Error Backpropagation. *ArXiv*, abs/1610.02583.
- Chen, Y. (2023). *A Comparative Study on the Forecast of Tourist Numbers in Huanggang City Based on Linear Regression Model and Grey GM (1,1) Model*. Paper presented at the Proceedings of the 2022 5th International Conference on E-Business, Information Management and Computer Science, Hong Kong, Hong Kong. <https://doi.org/10.1145/3584748.3584807>
- Chenchen, S., Militaru, I., & Folcut, B. (2023). *New energy generation forecast based on the grey GM (1,1) model*. Paper presented at the Proceedings of the 2023 8th International Conference on Intelligent Information Processing, <conf-loc>, <city>Bucharest</city>, <country>Romania</country>, </conf-loc>. <https://doi.org/10.1145/3635175.3635193>
- Chonlawanthon, T. (2024). Get to know 8 important components of electric cars. Retrieved from <https://shorturl.asia/n8olf>
- Chou, C.-C., Wang, C.-N., & Hsu, H.-P. (2022). A novel quantitative and qualitative model for forecasting the navigational risks of Maritime Autonomous Surface Ships. *Ocean Engineering*, 248, 110852. doi:<https://doi.org/10.1016/j.oceaneng.2022.110852>
- Chureungsakun, C. (2014). The birth of electric vehicles. Retrieved from <https://shorturl.asia/3SRaM>
- Chureungsakun, C. (2024). World Automotive Legends (The Birth of Electric Vehicles). Retrieved from <https://shorturl.asia/Top31>
- De Myttenaere, A., Golden, B., Le Grand, B., & Rossi, F. (2016). Mean Absolute Percentage Error for regression models. *Neurocomputing*, 192, 38-48. doi:<https://doi.org/10.1016/j.neucom.2015.12.114>
- Ding, S., & Li, R. (2021). Forecasting the sales and stock of electric vehicles using a novel self-adaptive optimized grey model. *Engineering Applications of Artificial Intelligence*, 100, 104148. doi:<https://doi.org/10.1016/j.engappai.2020.104148>
- Doğan, Y., Örüçü, F., Kut, A., & Radevski, V. (2011, 27-30 June 2011). *Optimizing the equation for a dataset with corresponding attributes by hybrid genetic algorithm*. Paper presented at the Proceedings of the ITI 2011, 33rd International Conference on Information Technology Interfaces.

- Draper, C., Reichle, R., de Jeu, R., Naeimi, V., Parinussa, R., & Wagner, W. (2013). Estimating root mean square errors in remotely sensed soil moisture over continental scale domains. *Remote Sensing of Environment*, 137, 288-298. doi:<https://doi.org/10.1016/j.rse.2013.06.013>
- Ewan Pritchard, B. B., Fan Yang, and Jeff Gonder. (2023). Evaluating Electric Vehicle Public Charging Utilization in the United States using the EV WATTS Dataset Retrieved from <https://shorturl.asia/NGx1K>
- Feng, J., Yang, J., Li, Y., Wang, H., Ji, H., Yang, W., & Wang, K. (2021). Load forecasting of electric vehicle charging station based on grey theory and neural network. *Energy Reports*, 7, 487-492. doi:<https://doi.org/10.1016/j.egy.2021.08.015>
- Frías-Paredes, L., Mallor, F., Gastón-Romeo, M., & León, T. (2018). Dynamic mean absolute error as new measure for assessing forecasting errors. *Energy Conversion and Management*, 162, 176-188. doi:<https://doi.org/10.1016/j.enconman.2018.02.030>
- Gani, M. M., Ritu, N. S., Mahi, F. I., Chakrabarty, J., & Mohammad, N. (2024, 25-27 April 2024). *Short-Term Residential Load Forecasting for Microgrid: Enhancing Performance through PCA and Clustering Techniques*. Paper presented at the 2024 3rd International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE).
- Gao, J., & Yang, H. (2024). An artificial neural network method for probabilistic life prediction of corroded reinforced concrete. *International Journal of Fatigue*, 186, 108418. doi:<https://doi.org/10.1016/j.ijfatigue.2024.108418>
- Geng, D., Ma, J., & Zhao, Y. (2021). *Electricity Generation Forecasting with Wind Power Data: A Grey Model Approach*. Paper presented at the 2021 2nd International Conference on Artificial Intelligence and Information Systems, Chongqing, China. <https://doi.org/10.1145/3469213.3469230>
- Guo, X., Li, D., & Zhang, A. (2012). Improved Support Vector Machine Oil Price Forecast Model Based on Genetic Algorithm Optimization Parameters. *AASRI Procedia*, 1, 525-530. doi:<https://doi.org/10.1016/j.aasri.2012.06.082>
- Hao, M., Wei, S., Yu, L., Li, W., Wu, M., Liu, J., . . . Li, Y. (2023, 6-8 Dec. 2023). *Artificial Neural Network-Based Approach to Modeling Energy Bands of GaN-Based Heterojunction Materials*. Paper presented at the 2023 International Conference on High Performance Big Data and Intelligent Systems (HDIS).
- He, C. (2023, 21-23 April 2023). *A Hybrid Model Based on Multi-LSTM and ARIMA for Time Series Forecasting*. Paper presented at the 2023 8th International Conference on Intelligent Computing and Signal Processing (ICSP).
- Hermawanto, D. (2013). Genetic Algorithm for Solving Simple Mathematical Equality Problem. *ArXiv, abs/1308.4675*.
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735-1780. doi:10.1162/neco.1997.9.8.1735
- Huang, L., He, J., Li, A., & Jingtao, C. (2023). *Population forecasting analysis based on Grey Model under Three-child Policy*. Paper presented at the Proceedings of the 7th International Conference on Intelligent Information Processing, Bucharest, Romania. <https://doi.org/10.1145/3570236.3570298>

- Huang, X., Wu, D., & Boulet, B. (2020, 9-10 Nov. 2020). *Ensemble Learning for Charging Load Forecasting of Electric Vehicle Charging Stations*. Paper presented at the 2020 IEEE Electric Power and Energy Conference (EPEC).
- Hyndman, R. J. (2014). *Forecasting: Principles & Practice*. Retrieved from robjhyndman.com/uwa
- Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4), 679-688. doi:<https://doi.org/10.1016/j.ijforecast.2006.03.001>
- Jia, Z., Li, W., & Han, Z. H. (2008, 12-14 Oct. 2008). *An Improved GM(1,1)-Genetic Algorithm to Short-Term Forecasting in Power System*. Paper presented at the 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing.
- Jin, C. (2022, 28-30 Oct. 2022). *Application and Optimization of Long Short-term Memory in Time Series Forecasting*. Paper presented at the 2022 International Communication Engineering and Cloud Computing Conference (CECCC).
- Jin, G., & Yang, C. (2023). *Forecasting Human Resources Demand of Port Enterprises Based on Grey BP Neural Network*. Paper presented at the Proceedings of the 2023 9th International Conference on Industrial and Business Engineering, <conf-loc>, <city>Beijing</city>, <country>China</country>, </conf-loc>. doi:<https://doi.org/10.1145/3629378.3629410>
- Johnsson, J., Björnsson, O., Andersson, P., Jakobsson, A., Cronberg, T., Lilja, G., . . . Frigyesi, A. (2020). Artificial neural networks improve early outcome prediction and risk classification in out-of-hospital cardiac arrest patients admitted to intensive care. *Critical Care*, 24(1), 474. doi:10.1186/s13054-020-03103-1
- Jonathan, B., Rahim, Z., Barzani, A., & Oktavega, W. (2019, 24-25 Oct. 2019). *Evaluation of Mean Absolute Error in Collaborative Filtering for Sparsity Users and Items on Female Daily Network*. Paper presented at the 2019 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS).
- Joshi, G., & Krishna, M. B. (2014, 24-27 Sept. 2014). *Solving system of non-linear equations using Genetic Algorithm*. Paper presented at the 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI).
- Karunasingha, D. S. K. (2022). Root mean square error or mean absolute error? Use their ratio as well. *Information Sciences*, 585, 609-629. doi:<https://doi.org/10.1016/j.ins.2021.11.036>
- Kayacan, E., Ulutas, B., & Kaynak, O. (2010). Grey system theory-based models in time series prediction. *Expert Systems with Applications*, 37(2), 1784-1789. doi:<https://doi.org/10.1016/j.eswa.2009.07.064>
- Khamraksa, C. (2023). What types of electric vehicle (EV) charging patterns are there in Thailand? Retrieved from <https://shorturl.asia/2OTDg>
- Kingdom, G. o. t. U. (2026). Electric vehicle public charging infrastructure statistics: January 2026 Retrieved from <https://shorturl.asia/svzql>

- Kluabwang, J. (2018). *Electricity Consumption Forecasting in Thailand Using Grey Prediction Models*. Paper presented at the Proceedings of the 2018 International Conference on Mechatronic Systems and Robots, Singapore, Singapore. <https://doi.org/10.1145/3230876.3230891>
- Kritanu. (2022). What is the difference between AC and DC Fast charging for electric vehicles? Retrieved from <https://shorturl.asia/XnUVs>
- Kumar, M. G., Kolla, N., Kotagi, V., & Mathapati, M. (2023, 2-3 Nov. 2023). *Fleet Load Charge Forecasting in Electric Vehicle Using Hybrid Deep Learning Model: LSTM-AQOA*. Paper presented at the 2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIIE).
- Kushwaha, R. C., Singh, D. R., & Prasad, A. K. (2024). An effective genetic algorithm for solving traveling salesman problem with group theory. In *Advances in Computers*: Elsevier.
- Lee, Y.-S., & Tong, L.-I. (2011). Forecasting energy consumption using a grey model improved by incorporating genetic programming. *Energy Conversion and Management*, 52(1), 147-152.
doi:<https://doi.org/10.1016/j.enconman.2010.06.053>
- Leenatham, A., & Khemavuk, P. (2020, 25-26 Sept. 2020). *Demand Forecasting Using Artificial Neural Network Based on Quantitative and Qualitative Data*. Paper presented at the 2020 1st International Conference on Big Data Analytics and Practices (IBDAP).
- Li, G.-q., Xu, S.-w., & Li, Z.-m. (2010). Short-Term Price Forecasting For Agro-products Using Artificial Neural Networks. *Agriculture and Agricultural Science Procedia*, 1, 278-287.
doi:<https://doi.org/10.1016/j.aaspro.2010.09.035>
- Li, J., Zhang, Q., Zhang, Z., Wang, F., & Li, X. (2021). *Valuation Analysis of Chinese and American Listed Companies Based on Multiple Linear Regression and Grey Forecasting Model*. Paper presented at the Proceedings of the 4th International Conference on Control and Computer Vision, Macau, China.
<https://doi.org/10.1145/3484274.3484308>
- Li, S. (2022). *Predicting the Total Population Development of China Based on Logistic Blocking Growth Model and Improved Grey GM (1,1) Prediction Model*. Paper presented at the Proceedings of the 6th International Conference on Information Systems Engineering, Shanghai, China.
<https://doi.org/10.1145/3503928.3503934>
- Li, Y. L., Zhu, Z. A., Chang, Y. K., & Chiang, C. K. (2020, 13-16 Nov. 2020). *Short-Term Wind Power Forecasting by Advanced Machine Learning Models*. Paper presented at the 2020 International Symposium on Computer, Consumer and Control (IS3C).
- Liu, B., Song, C., Liang, X., Lai, M., Yu, Z., & Ji, J. (2023). Regional differences in China's electric vehicle sales forecasting: Under supply-demand policy scenarios. *Energy Policy*, 177, 113554.
doi:<https://doi.org/10.1016/j.enpol.2023.113554>

- Liu, X., Peng, H., Bai, Y., Zhu, Y., & Liao, L. (2014). Tourism Flows Prediction based on an Improved Grey GM(1,1) Model. *Procedia - Social and Behavioral Sciences*, 138, 767-775. doi:<https://doi.org/10.1016/j.sbspro.2014.07.256>
- Lutsey, N., & Nicholas, M. (2019). *Update on electric vehicle costs in the United States through 2030*.
- Lye, G. (2017). France announces ban on sales of gasoline and diesel cars by 2040. Retrieved from <https://shorturl.asia/7X89l>
- MacNeil, P. E., & Schultz, S. R. (2010, 18-21 March 2010). *A genetic algorithm approach to the solution of a differential equation*. Paper presented at the Proceedings of the IEEE SoutheastCon 2010 (SoutheastCon).
- Mahlberg, J. A., Desai, J., & Bullock, D. M. (2023). Evaluation of Electric Vehicle Charging Usage and Driver Activity. *World Electric Vehicle Journal*, 14(11), 308. Retrieved from <https://www.mdpi.com/2032-6653/14/11/308>
- Mahmoudinazlou, S., & Kwon, C. (2024). A hybrid genetic algorithm for the min-max Multiple Traveling Salesman Problem. *Computers & Operations Research*, 162, 106455. doi:<https://doi.org/10.1016/j.cor.2023.106455>
- Master, w. e. (2023). What is an EV charger? Retrieved from <https://shorturl.asia/Q40te>
- Medium. (2024). Artificial Neural Network (ANN) with Practical Implementation. Retrieved from <https://shorturl.asia/izyU0>
- Mendo, L. (2009). Estimation of a probability with guaranteed normalized mean absolute error. *IEEE Communications Letters*, 13(11), 817-819. doi:10.1109/LCOMM.2009.091128
- Michańków, J., Sakowski, P., & Ślepaczuk, R. (2024). Mean Absolute Directional Loss as a new loss function for machine learning problems in algorithmic investment strategies. *Journal of Computational Science*, 81, 102375. doi:<https://doi.org/10.1016/j.jocs.2024.102375>
- Nayak, G. H. H., Alam, M. W., Singh, K. N., Avinash, G., Kumar, R. R., Ray, M., & Deb, C. K. (2024). Exogenous variable driven deep learning models for improved price forecasting of TOP crops in India. *Scientific Reports*, 14(1), 17203. doi:10.1038/s41598-024-68040-3
- Niu, D. x., Li, W., Han, Z. h., & Yuan, X. e. (2008, 18-20 Oct. 2008). *Power Load Forecasting Based on Improved Genetic Algorithm-GM(1,1) Model*. Paper presented at the 2008 Fourth International Conference on Natural Computation.
- Olawumi, M. A., Oladapo, B. I., & Olawale, R. A. (2024). Revolutionising waste management with the impact of Long Short-Term Memory networks on recycling rate predictions. *Waste Management Bulletin*, 2(3), 266-274. doi:<https://doi.org/10.1016/j.wmb.2024.08.006>
- Ou, S.-L. (2012). Forecasting agricultural output with an improved grey forecasting model based on the genetic algorithm. *Computers and Electronics in Agriculture*, 85, 33-39. doi:<https://doi.org/10.1016/j.compag.2012.03.007>

- Pan, D. (2022). *Forecasting CO2 Emission from Thermal Power Production in Beijing-Tianjin-Hebei by Using GM (1,1) Model*. Paper presented at the 2022 2nd International Conference on Bioinformatics and Intelligent Computing, <conf-loc>, <city>Harbin</city>, <country>China</country>, </conf-loc>. <https://doi.org/10.1145/3523286.3524593>
- Phromwong, S. (2018). The Department of Land Transport is continuously enhancing vehicle safety standards across all dimensions, covering both conventional and electric vehicles. Manufacturers, assemblers, and importers are required to obtain model certification in accordance with the Department's specified conditions. Retrieved from <https://shorturl.asia/A6TQD>
- Pornpatcharapong, W. (2008). Artificial Neural Networks (ANN). Retrieved from <https://shorturl.asia/YQruL>
- Pu, T., Huang, M., & Yang, J. (2022). *Forecasting international migrants using grey model with heat label*. Paper presented at the Proceedings of the 5th International Conference on Computer Science and Software Engineering, <conf-loc>, <city>Guilin</city>, <country>China</country>, </conf-loc>. <https://doi.org/10.1145/3569966.3571179>
- Rashid, M., Elfouly, T., & Chen, N. (2023, 24-27 Oct. 2023). *Travel Motif-Based Learning Scheme for Electric Vehicle Charging Demand Forecasting*. Paper presented at the 2023 IEEE Vehicle Power and Propulsion Conference (VPPC).
- Rayudu, D. V., & Roseline, J. F. (2023, 5-7 Jan. 2023). *Accurate Weather Forecasting for Rainfall Prediction Using Artificial Neural Network Compared with Deep Learning Neural Network*. Paper presented at the 2023 International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF).
- Roy, S., & Tanveer, S. (2023, 20-21 April 2023). *Stock Price Forecasting using DeepNet*. Paper presented at the 2023 International Conference on Computational Intelligence, Communication Technology and Networking (CICTN).
- Samavati, F. C., Mesgari, H., Ali, S., & Moosavian, A. (2011, 17-19 Sept. 2011). *A comparison between MATLAB optimization toolbox and GA in cooperative grasp planning*. Paper presented at the 2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM).
- Sapnken, F. E., Acyl, A. K., Boukar, M., Nyobe, S. L. B., & Tamba, J. G. (2023). A technique for improving petroleum products forecasts using grey convolution models and genetic algorithms. *MethodsX*, 10, 102097. doi:<https://doi.org/10.1016/j.mex.2023.102097>
- Sapnken, F. E., Ahmat, K. A., Boukar, M., Biobiongono Nyobe, S. L., & Tamba, J. G. (2022). Forecasting petroleum products consumption in Cameroon's household sector using a sequential GMC(1,n) model optimized by genetic algorithms. *Heliyon*, 8(12), e12138. doi:<https://doi.org/10.1016/j.heliyon.2022.e12138>

- Semenova, N. G., Vlatskaya, L. A., & Semenov, A. M. (2020, 27-30 Oct. 2020). *Application of genetic algorithms in problems of compensating devices placement optimization*. Paper presented at the 2020 International Conference on Electrotechnical Complexes and Systems (ICOECS).
- Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. *Physica D: Nonlinear Phenomena*, 404, 132306. doi:<https://doi.org/10.1016/j.physd.2019.132306>
- Shrinivisan, L., Verma, R., & Tekade, K. (2023, 2-4 Nov. 2023). *Forecasting Battery State of Charge for Electric Vehicles Through Deep Neural Network*. Paper presented at the 2023 7th International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS).
- Stekler, H., & Symington, H. (2016). Evaluating qualitative forecasts: The FOMC minutes, 2006–2010. *International Journal of Forecasting*, 32(2), 559-570. doi:<https://doi.org/10.1016/j.ijforecast.2015.02.003>
- Thailand, E. G. A. o. (2023). EGAT EV Information (2023). Retrieved from <https://shorturl.asia/Wymfk>
- Tsai, Y. T., Zeng, Y. R., & Chang, Y. S. (2018, 12-15 Aug. 2018). *Air Pollution Forecasting Using RNN with LSTM*. Paper presented at the 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech).
- Van Houdt, G., Mosquera, C., & Nápoles, G. (2020). A Review on the Long Short-Term Memory Model. *Artificial Intelligence Review*, 53. doi:10.1007/s10462-020-09838-1
- Wang, C.-N., Dang, T.-T., Nguyen, N.-A.-T., & Le, T.-T.-H. (2020). Supporting Better Decision-Making: A Combined Grey Model and Data Envelopment Analysis for Efficiency Evaluation in E-Commerce Marketplaces. *Sustainability*, 12(24), 10385. Retrieved from <https://www.mdpi.com/2071-1050/12/24/10385>
- Wang, S. H., Wan, J. R., Chang, Y. J., & Cai, C. C. (2007, 19-22 Aug. 2007). *Medium-Long Term Load Forecasting Based on Improved Grey Model*. Paper presented at the 2007 International Conference on Machine Learning and Cybernetics.
- Wang, X. (2011). Application Research on Electricity Demand Forecasting Based on Gaussian Quadrature Formula. *Procedia Engineering*, 15, 5574-5578. doi:<https://doi.org/10.1016/j.proeng.2011.08.1034>
- Wang, Y. (2014, 26-28 April 2014). *Short-term wind power forecasting by genetic algorithm of wavelet neural network*. Paper presented at the 2014 International Conference on Information Science, Electronics and Electrical Engineering.
- Wang, Y., Dang, Y., Li, Y., & Liu, S. (2010). An approach to increase prediction precision of GM(1,1) model based on optimization of the initial condition. *Expert Systems with Applications*, 37(8), 5640-5644. doi:<https://doi.org/10.1016/j.eswa.2010.02.048>

- Wu-yong, Q., Yao-guo, D., & Si-feng, L. (2013, 15-17 Nov. 2013). *Study on grey prediction model GM(1,1) and its application based on amplitude compression transformation*. Paper presented at the Proceedings of 2013 IEEE International Conference on Grey systems and Intelligent Services (GSIS).
- Wu, L., Yu, B., Ju, X., & Jiang, L. (2021). *Forecasting Method of Electricity Charge based on Gray algorithm*. Paper presented at the The 2nd International Conference on Computing and Data Science, Stanford, CA, USA.
<https://doi.org/10.1145/3448734.3450826>
- Wu, Z., & Li, C. (2019). *An Adaptive Grey Forecasting Model NGM(1, 1, k2) and Its Application for Short-term Traffic Flow Prediction*. Paper presented at the Proceedings of the 3rd International Conference on Computer Science and Application Engineering, Sanya, China.
<https://doi.org/10.1145/3331453.3361318>
- Xie, Q., & Xie, Y. (2009a, 5-6 Dec. 2009). *Forecast of Regional Gross National Product Based on Grey Modelling Optimized by Genetic Algorithm*. Paper presented at the 2009 International Conference on E-Learning, E-Business, Enterprise Information Systems, and E-Government.
- Xie, Q., & Xie, Y. (2009b, 21-22 Nov. 2009). *Forecast of the Total Volume of Import-export Trade Based on Grey Modelling Optimized by Genetic Algorithm*. Paper presented at the 2009 Third International Symposium on Intelligent Information Technology Application.
- Xu, Y., Lin, T., Du, P., & Wang, J. (2024). The research on a novel multivariate grey model and its application in carbon dioxide emissions prediction. *Environmental Science and Pollution Research*, 31(14), 21986-22011. doi:10.1007/s11356-024-32262-9
- Xydias, E. S., Marmaras, C. E., Cipcigan, L. M., Hassan, A. S., & Jenkins, N. (2013, 6-7 Nov. 2013). *Electric vehicle load forecasting using data mining methods*. Paper presented at the IET Hybrid and Electric Vehicles Conference 2013 (HEVC 2013).
- Xylia, M., Olsson, E., Macura, B., & Nykvist, B. (2025). Estimating charging infrastructure demand for electric vehicles: a systematic review. *Energy Strategy Reviews*, 59, 101753. doi:<https://doi.org/10.1016/j.esr.2025.101753>
- Yang, X., Wang, S., Peng, Y., Chen, J., & Meng, L. (2023). Short-term photovoltaic power prediction with similar-day integrated by BP-AdaBoost based on the Grey-Markov model. *Electric Power Systems Research*, 215, 108966. doi:<https://doi.org/10.1016/j.epsr.2022.108966>
- Yao, R., Jin, S., Luo, Z., & Fang, L. (2022, 26-28 Aug. 2022). *An improved GM(1,1) model based on iterative weighted least square criterion parameter estimation*. Paper presented at the 2022 International Conference on Computers, Information Processing and Advanced Education (CIPAE).
- Yu, Z., & Zhao, C. (2017, 21-23 April 2017). *A Combination Forecasting Model of Extreme Learning Machine Based on Genetic Algorithm Optimization*. Paper presented at the 2017 International Conference on Computing Intelligence and Information System (CIIS).
- Zhang, G. P. (2000). Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 30(4), 451-462. doi:10.1109/5326.897072

- Zhang, R., Mao, S., & Kang, Y. (2023). A novel traffic flow prediction model: Variable order fractional grey model based on an improved grey evolution algorithm. *Expert Systems with Applications*, 224, 119943. doi:<https://doi.org/10.1016/j.eswa.2023.119943>
- Zhang, T., Huang, Y., Liao, H., & Liang, Y. (2023). A hybrid electric vehicle load classification and forecasting approach based on GBDT algorithm and temporal convolutional network. *Applied Energy*, 351, 121768. doi:<https://doi.org/10.1016/j.apenergy.2023.121768>
- Zhao, W., Tao, T., & Zio, E. (2015). System reliability prediction by support vector regression with analytic selection and genetic algorithm parameters selection. *Applied Soft Computing*, 30, 792-802. doi:<https://doi.org/10.1016/j.asoc.2015.02.026>
- Zhou, J. J. (2013, 17-19 Dec. 2013). *The application of grey forecasting model based on excel modeling and solving in logistics demand forecast*. Paper presented at the 2013 10th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP).
- Zhou, W., & He, J.-M. (2013). Generalized GM (1,1) model and its application in forecasting of fuel production. *Applied Mathematical Modelling*, 37(9), 6234-6243. doi:<https://doi.org/10.1016/j.apm.2013.01.002>
- Zhou, Y., Wang, X., & Zhang, D. (2022). *Forecasting of Extreme Weather Intensity Based on Grey Model*. Paper presented at the Proceedings of the 2022 2nd International Conference on Control and Intelligent Robotics, Nanjing, China. <https://doi.org/10.1145/3548608.3559193>



APPENDIX A

Data Set

DATASET

1. Volume of newly registered small passenger vehicles (new license plates)

This study utilized secondary data obtained from the Automotive Information Center of Thailand. The data comprise monthly statistics on the number of newly registered small passenger vehicles from January 2019 to October 2024. A summary of the data is shown in Table A-1.

Table A-1 Raw Data

NO.	Years	Month	Brand	<=2000 CC	>2000C C	BEV	Not Specific	Total
1.	2024	January	Aion	0	0	274	0	0
2.	2024	January	Audi	66	18	3	0	87
3.	2024	January	BMW	765	137	166	0	1,068
4.	2024	January	BYD	0	0	7880	0	7,880
5.	2024	January	HONDA	11,064	9	0	0	11,073
6.	2024	January	TOYOT A	14,044	12,806	66	1	26,917
7.	2024	January	VOLVO	194	0	188	0	382
8.	2024	January	MG	1,300	0	1,948	0	3,248
9.	2024	January	TESLA	0	0	52	0	52

Table A-1 displays raw secondary data sourced from the Automotive Information Center of Thailand. The dataset is organized by car brand ranking, year, and month of registration. The recorded variables include car brand, newly registered vehicles with engine capacities of $\leq 2,000$ cc, vehicles with engine capacities exceeding 2,000 cc, battery electric vehicles (BEVs), and non-specific vehicles, which primarily refer to older models or registrations lacking detailed specifications. The total column summarizes the number of registered vehicles for each brand on a monthly basis, whereas the sum across all brands reflects the total vehicle registrations for each month.

2. Number of Electric Vehicle Charging Stations Nationwide (Raw Data)

This appendix provides raw secondary data on the number of electric vehicle charging stations by brand in Thailand during the period 2019–2024. The data presentation starts with the Evolt brand, which is shown in Figure A-1, while data for other brands are displayed in the following figures.



Figure A-1 Raw data from the Evolt app.

Figure A-1 depicts the user interface of the Evolt application, which displays the locations of electric vehicle charging stations. By selecting a station, users can access detailed information regarding station location, the number of available charging points, and the categorization of charging point types. The updated application version further includes data on electricity usage and charging costs per point, as well as access to customer support via an integrated contact icon. Figure A-2 shows the raw data extracted from the PTT application.

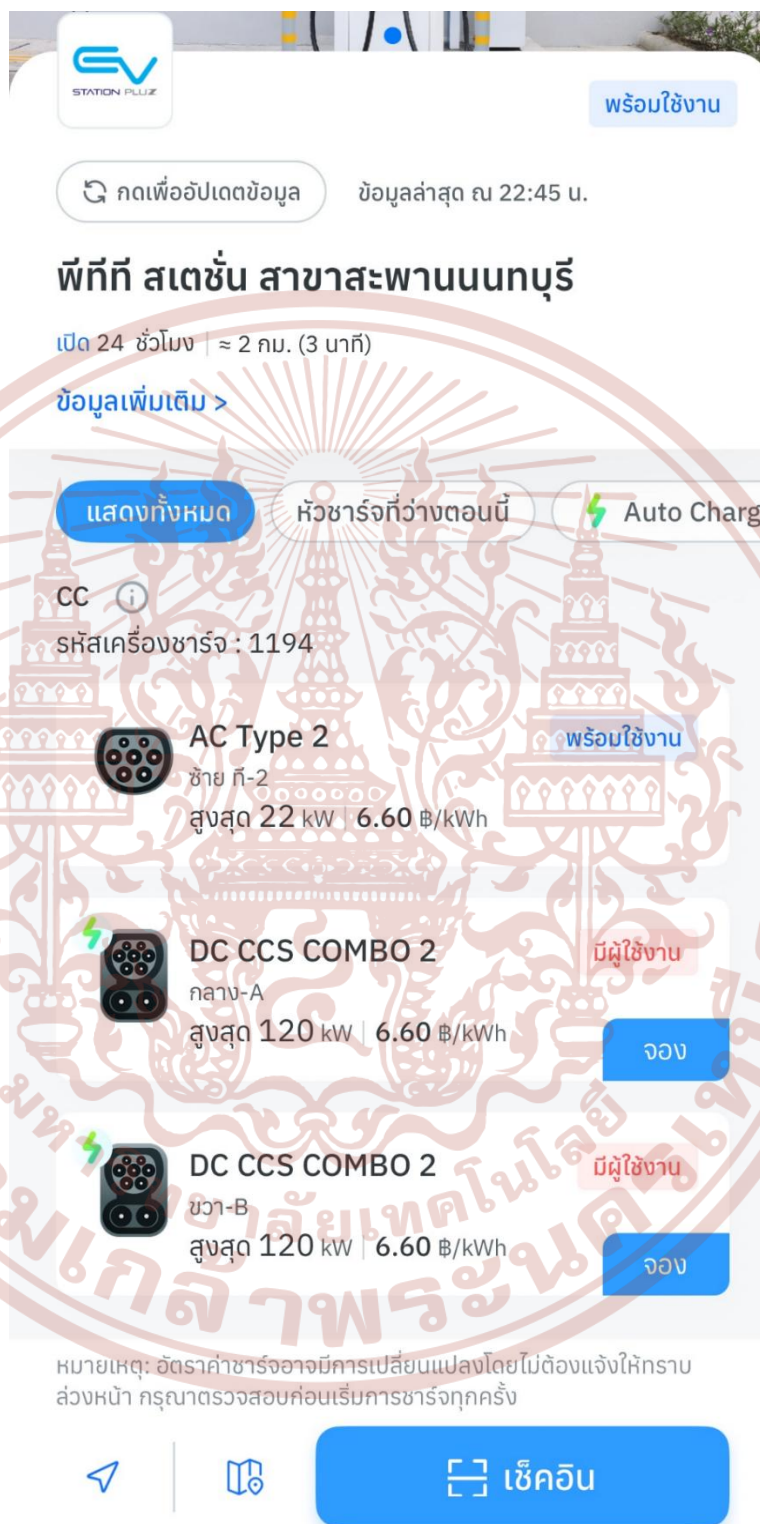


Figure A-2 Raw data from the PTT app

Figure A-2 depicts the user interface of the PTT application, which presents the locations of electric vehicle charging stations. Upon selecting a station, the application displays information on station location, the number of charging points available, and the categorization of charging point types. The updated version further includes data on electricity output and station operating hours. User assistance for application-related issues is provided via the official PTT EV Station Facebook page. Figure A-3 shows the raw data extracted from the MEA application.

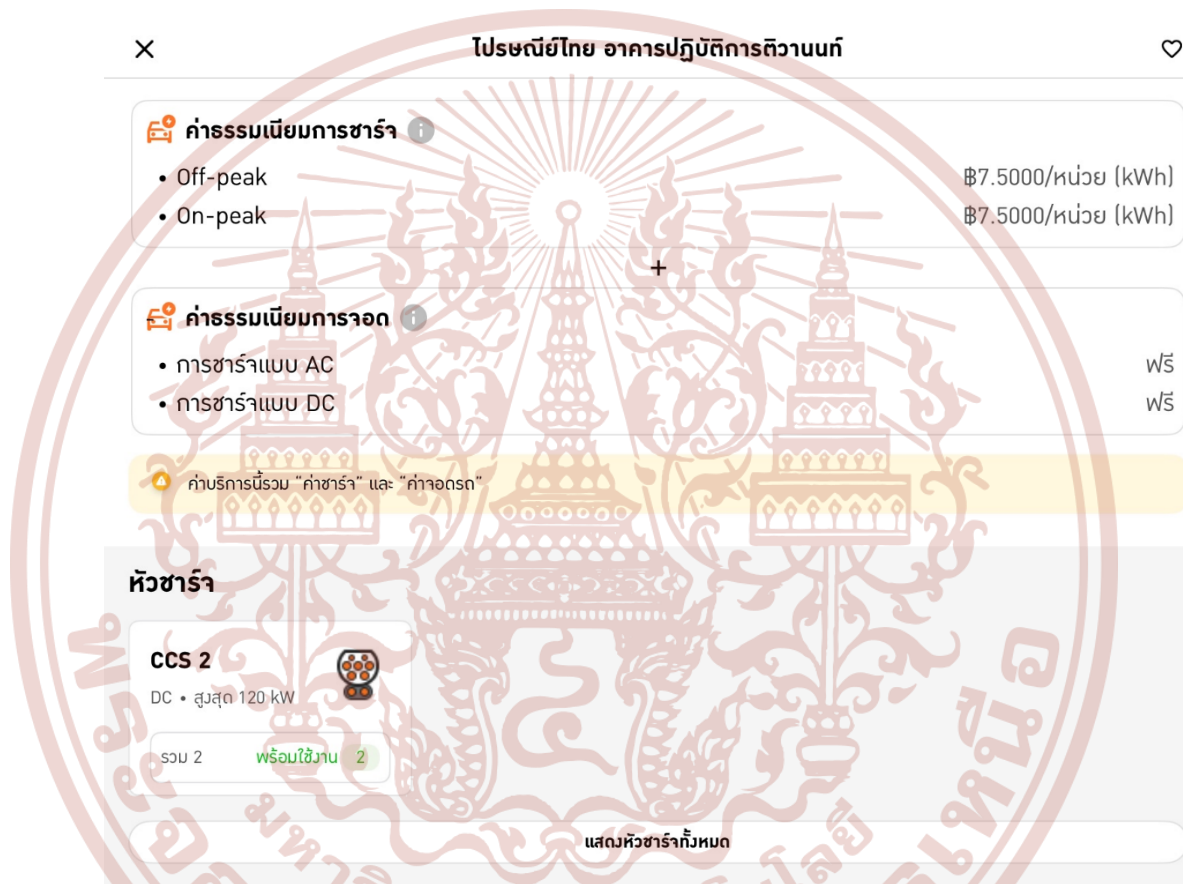


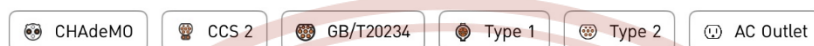
Figure A-3 Raw data from the MEA app

Figure A-3 depicts the user interface of the MEA application, which presents the locations of electric vehicle charging stations operated by the Metropolitan Electricity Authority. Upon selecting a station, the application displays information regarding station location and available charging units. Although the application does not categorize charging units by electric vehicle charging type, the updated version includes data on electricity output, station operating hours, and charging costs per unit. User assistance for application-related issues is provided through the MEA contact number (1130). Figure A-4 shows the detailed information for all charging units associated with the selected station.



หัวชาร์จ

เลือกหัวชาร์จของคุณ



MEA Quick Charger 120 kW

ตู้ชาร์จ



Figure A-4 Raw data from the MEA All head app

Figure A-4 illustrates detailed information on all charging units corresponding to the charging stations depicted in Figure A-3, including their charging capacity and operational status. Figure A-5 shows data on the number of electric vehicle charging stations extracted from the EA Anywhere application.

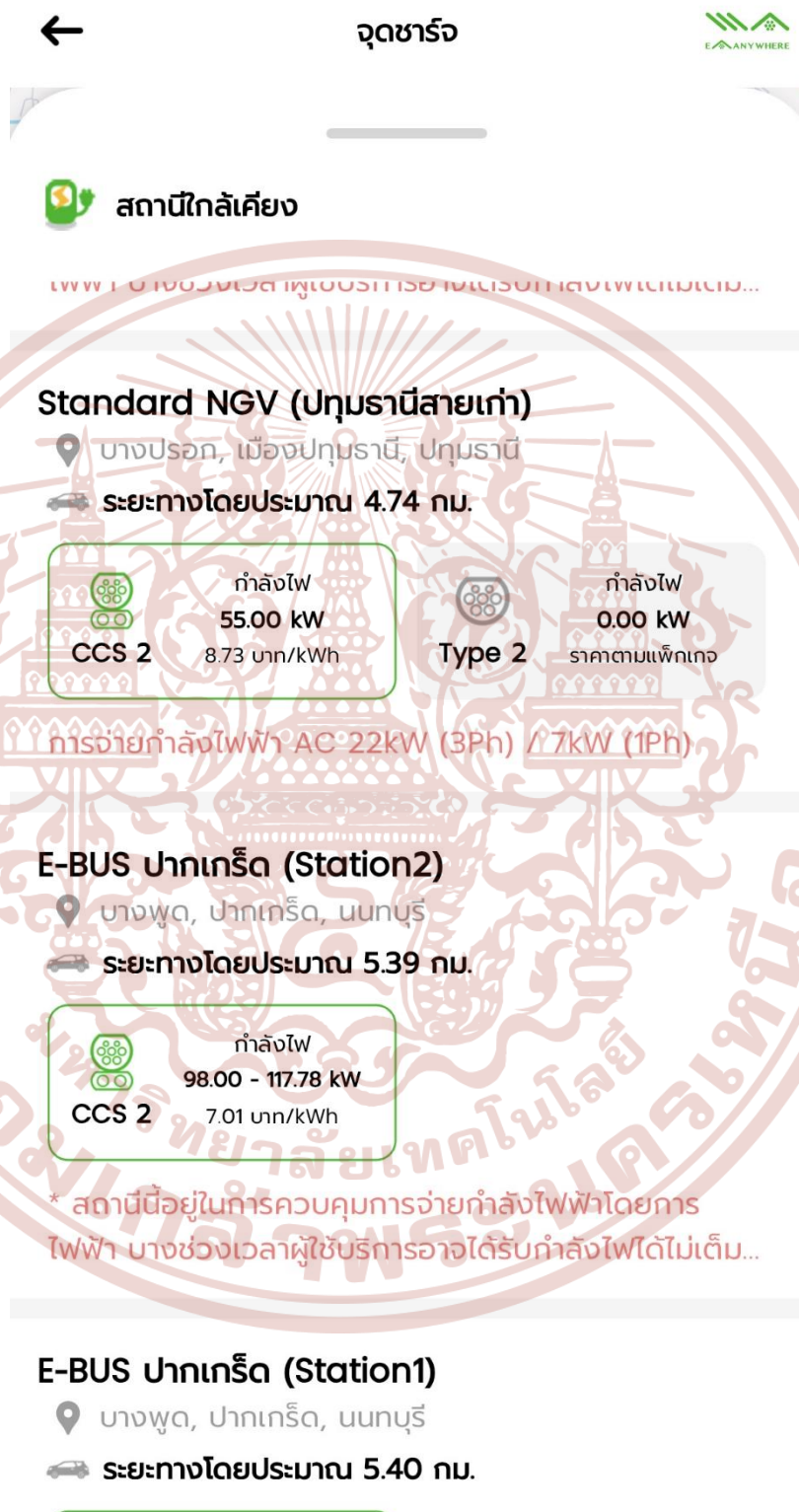


Figure A-5 Raw data from the EA Anywhere app

Figure A-5 depicts the dataset presented in the EA Anywhere application, which compiles information on electric vehicle charging stations operated by EA Anywhere as well as those provided by other organizations, such as the Provincial Electricity Authority, the Electricity Generating Authority of Thailand, the Metropolitan Electricity Authority, and POR. Upon selecting a station, the interface displays information on station location, available charging units, and the categorization of charging units according to electric vehicle type. The updated version additionally reports charging capacity, operating hours, contact information, and charging prices for individual units. Figure 6 shows the data source associated with the establishment of MG electric vehicle charging stations.

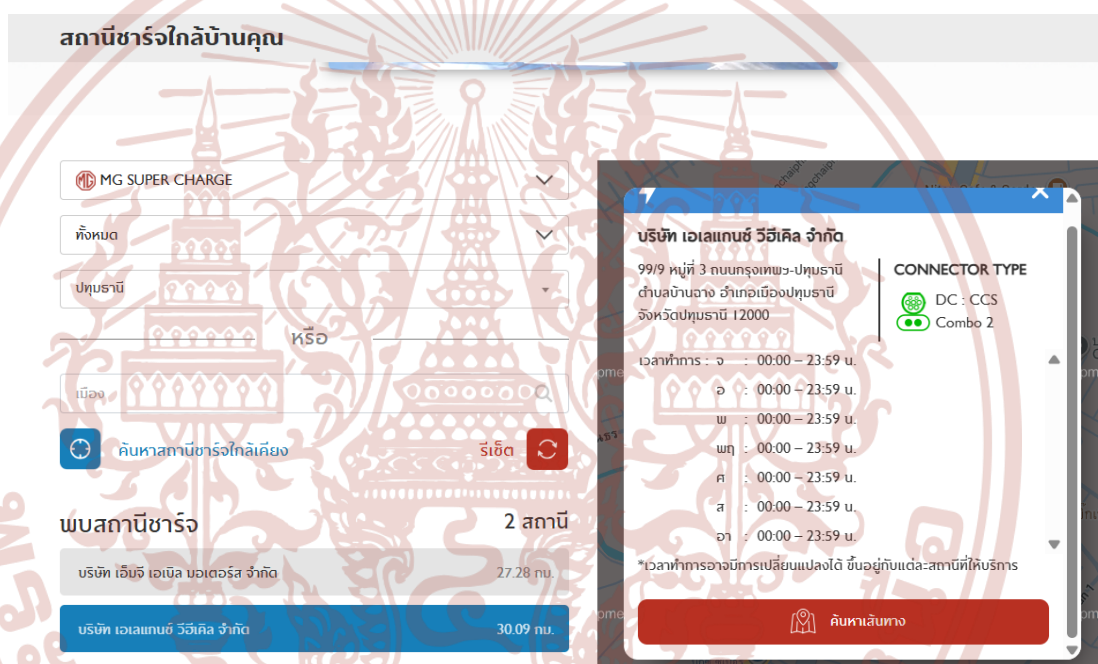


Figure A-6 Raw data from the MG Website

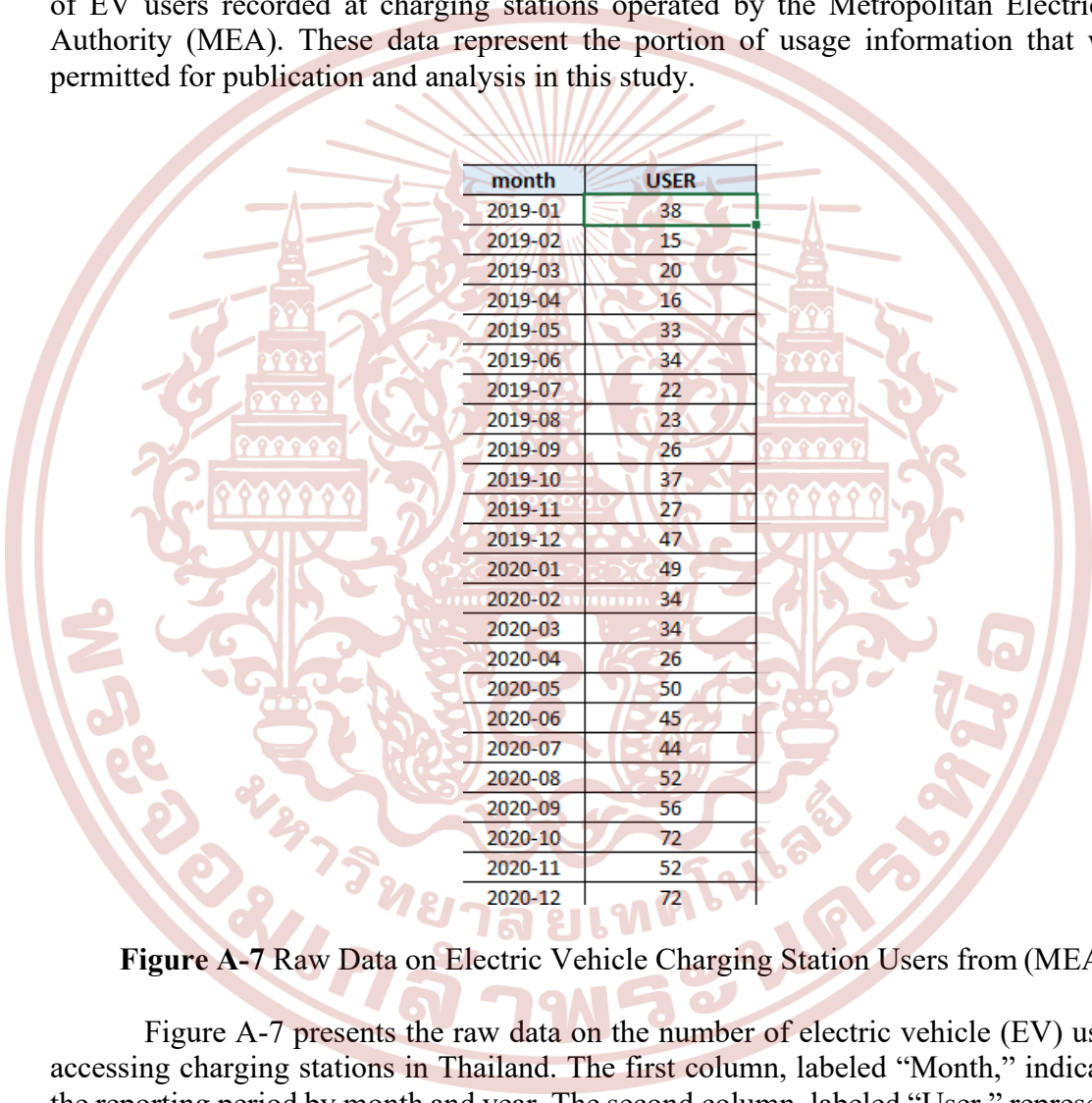
Figure A-6 illustrates raw secondary data sourced from the MG official website (<https://shorturl.asia/liAuR>). The data summarizes the total number of MG-operated charging stations and charging points, categorized by charging point type. The dataset also presents general information on charging stations from other service providers and their operating hours. Nevertheless, the data are limited by the unavailability of charging power capacity details and the absence of explicit cost information. Additionally, no contact details, including telephone numbers or social media references, are reported.

3. Dataset of Electric Vehicle Charging Station Users

The appendix presents the raw data on the number of electric vehicle (EV) users who accessed EV charging stations in Thailand. These data were obtained through formal requests submitted to organizations involved in the development and operation of EV charging infrastructure within the country. To ensure consistency with the EV

registration records analyzed in this study, data were specifically requested from charging station providers operating during the period 2019–2023.

However, complete datasets were not available from all providers. Several organizations indicated that detailed usage information was confidential and could not be fully disclosed for research purposes. Consequently, only the data approved for academic use are included in this appendix. The following figure illustrates the number of EV users recorded at charging stations operated by the Metropolitan Electricity Authority (MEA). These data represent the portion of usage information that was permitted for publication and analysis in this study.



month	USER
2019-01	38
2019-02	15
2019-03	20
2019-04	16
2019-05	33
2019-06	34
2019-07	22
2019-08	23
2019-09	26
2019-10	37
2019-11	27
2019-12	47
2020-01	49
2020-02	34
2020-03	34
2020-04	26
2020-05	50
2020-06	45
2020-07	44
2020-08	52
2020-09	56
2020-10	72
2020-11	52
2020-12	72

Figure A-7 Raw Data on Electric Vehicle Charging Station Users from (MEA)

Figure A-7 presents the raw data on the number of electric vehicle (EV) users accessing charging stations in Thailand. The first column, labeled “Month,” indicates the reporting period by month and year. The second column, labeled “User,” represents the total number of EV users who utilized charging stations during each corresponding month. Figure A-8 displays additional raw data on EV charging station usage in Thailand obtained from the Provincial Electricity Authority (PEA). These data provide supplementary information on monthly user activity from another major charging infrastructure provider.

month	Brand	Model	count
2020-08	Zeekr	9	2
2020-08	BMW	300e	1
2020-08	AUDI	55 quattro	4
2020-08	BYD	ATTO 3	22
2020-08	TOYOTA	bZ4X	1
2020-08	VOLVO	C40 Recharge	1
2020-08	MINI	Cooper SE	1
2020-08	VOLVO	CX40 Recharge	1
2020-08	CHANGAN	DEEPAL S7	1
2020-08	CHANGAN	DEEPAL S7	7
2020-08	CHANGAN	Deepal S7	2
2020-08	BYD	Denza D9	2
2020-08	BYD	Dolphin	4
2020-08	BYD	E6	30
2020-08	HONDA	e:NP1	1
2020-08	MINI	Electric	7
2020-08	MG	EP Plus	2
2020-08	MERCEDES	EQE SUV	1
2020-08	MG	ES	2
2020-08	MG	ES5	1
2020-08	KIA	EV5	1
2020-08	GEELY	EX5	1
2020-08	FOMM	Fomm One	7
2020-08	XPENG	G6	1
2020-08	MERCEDES	GLC 300e	1
2020-08	GAC AION	Hyper GT	3
2020-08	BMW	i 3s	10
2020-08	JAGUAR	i-pace	4
2020-08	BMW	i5	2
2020-08	BMW	i7	1

Figure A-8 Raw Data on Electric Vehicle Charging Station Users from (PEA)

Figure A-8 presents the raw dataset of electric vehicle (EV) users who utilized charging stations operated by the Provincial Electricity Authority in Thailand. The dataset provides disaggregated information on charging station usage based on reporting period, vehicle brand, and vehicle model. The first column, labeled “Month,” specifies the reporting period in terms of month and year, indicating the duration during which the charging stations were in operation and actively used. The “Brand” column identifies the manufacturers of the electric vehicles recorded in the dataset, thereby allowing for analysis of usage patterns across different EV brands. The “Model” column further categorizes the data by specifying the individual vehicle models associated with each brand. This level of detail enables a more refined examination of

charging behavior according to vehicle type. The final column, labeled “Count,” represents the total number of recorded charging sessions or users corresponding to each brand and model within the specified reporting period. These data contribute to a comprehensive understanding of user distribution and charging demand patterns within the PEA charging network during the study timeframe.





APPENDIX B

Electric car sales forecasting error prediction algorithm

ELECTRIC CAR SALES FORECASTING ERROR PREDICTION ALGORITHM

1. A set of instructions for forecasting the deviation of electric car sales using an ANN model

```

pip install keras
df = pd.read_excel('data.xlsx')
data = df['จำนวน'].values.reshape(-1, 1)
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt

# Generate some example data
df = pd.read_excel('data.xlsx')
data = df['จำนวน'].values.reshape(-1, 1)

# Normalize the data
scaler = MinMaxScaler()
data_normalized = scaler.fit_transform(data.reshape(-1, 1)).flatten()

# Create the dataset with a window size for the time series
def create_dataset(data, window_size):
    X, Y = [], []
    for i in range(len(data) - window_size):
        X.append(data[i:(i + window_size)])
        Y.append(data[i + window_size])
    return np.array(X), np.array(Y)

window_size = 10
X, Y = create_dataset(data_normalized, window_size)

# Split the data into training and testing sets
split = int(0.8 * len(data))
X_train, X_test, Y_train, Y_test = X[:split], X[split:], Y[:split], Y[split:]

# Build the model
model = Sequential([
    Dense(64, input_shape=(window_size,), activation='relu'),
    Dense(1)
])

```

```

model.compile(optimizer='adam', loss='mse')

# Train the model
model.fit(X_train, Y_train, epochs=100, batch_size=32, verbose=0)

# Make predictions
Y_pred = model.predict(X_test)

# Inverse transform the predictions and actual values
Y_pred_inv = scaler.inverse_transform(Y_pred)
Y_test_inv = scaler.inverse_transform(Y_test.reshape(-1, 1))

# Calculate MAPE, MAE, RMSE
def calculate_metrics(y_true, y_pred):
    mape = np.mean(np.abs((y_true - y_pred) / y_true)) * 100
    mae = mean_absolute_error(y_true, y_pred)
    rmse = np.sqrt(mean_squared_error(y_true, y_pred))
    return mape, mae, rmse

mape, mae, rmse = calculate_metrics(y_true_inv, y_pred_inv)

print("MAPE:", mape)
print("MAE:", mae)
print("RMSE:", rmse)

# Plot the results
plt.figure(figsize=(14, 7))
plt.plot(Y_test_inv, label='Actual')
plt.plot(Y_pred_inv, label='Predicted')
plt.xlabel('Date')
plt.ylabel('total')
plt.legend()
plt.show()

```

2.A set of instructions for forecasting the deviation of electric car sales using RNN models

```

pip install tensorflow
import numpy as np
import matplotlib.pyplot as plt
from keras.models import Sequential
from keras.layers import LSTM, Dense
from sklearn.preprocessing import MinMaxScaler

# Generate some example data
np.random.seed(0)
df = pd.read_excel('data.xlsx')

```

```

data = df['จำนวน'].values.reshape(-1, 1)
# Normalize the data

scaler = MinMaxScaler(feature_range=(0, 1))
data_normalized = scaler.fit_transform(data.reshape(-1, 1))

# Prepare the data for training the RNN
look_back = 10
X, y = [], []
for i in range(len(data_normalized) - look_back):
    X.append(data_normalized[i:(i + look_back), 0])
    y.append(data_normalized[i + look_back, 0])
X, y = np.array(X), np.array(y)

# Reshape X to be [samples, time steps, features]
X = np.reshape(X, (X.shape[0], X.shape[1], 1))

# Build the RNN model
model = Sequential()
model.add(LSTM(units=50, input_shape=(X.shape[1], 1)))
model.add(Dense(units=1))
model.compile(optimizer='adam', loss='mean_squared_error')

# Train the model
model.fit(X, y, epochs=100, batch_size=32)

# Make predictions
predicted_data_normalized = model.predict(X)
predicted_data = scaler.inverse_transform(predicted_data_normalized)

# Calculate MAPE, MAE, RMSE
def evaluate_forecasts(actual, predicted):
    mape = np.mean(np.abs((actual - predicted) / actual)) * 100
    mae = mean_absolute_error(actual, predicted)
    rmse = np.sqrt(mean_squared_error(actual, predicted))
    return mape, mae, rmse

train_mape, train_mae, train_rmse = evaluate_forecasts(trainY[0], trainPredict[:,0])
test_mape, test_mae, test_rmse = evaluate_forecasts(testY[0], testPredict[:,0])

print("Train MAPE: %.3f, MAE: %.3f, RMSE: %.3f % (train_mape, train_mae, train_rmse))
print("Test MAPE: %.3f, MAE: %.3f, RMSE: %.3f % (test_mape, test_mae, test_rmse))

# Plot the results
plt.figure(figsize=(10, 6))

```

```
plt.plot(data, label='True Data')
plt.plot(np.arange(look_back, len(predicted_data) + look_back), predicted_data,
label='Predicted Data')
plt.xlabel('Date')
plt.ylabel('total')
plt.legend()
plt.show()
```

3. A set of instructions for forecasting the deviation of electric vehicle sales using LSTM models

```
import numpy as np
import pandas as pd
from sklearn.metrics import mean_absolute_error, mean_squared_error
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
import matplotlib.pyplot as plt

# Example data
df = pd.read_excel('data.xlsx')
data = df['จำนวน'].values.reshape(-1, 1)

# Normalize the data
scaler = MinMaxScaler(feature_range=(0, 1))
data_normalized = scaler.fit_transform(data.reshape(-1, 1))

# Prepare the data for LSTM
def prepare_data(data, n_features):
    X, y = [], []
    for i in range(len(data)):
        end_ix = i + n_features
        if end_ix > len(data) - 1:
            break
        seq_x, seq_y = data[i:end_ix], data[end_ix]
        X.append(seq_x)
        y.append(seq_y)
    return np.array(X), np.array(y)

n_steps = 3
X, y = prepare_data(data_normalized, n_steps)

# Reshape data for LSTM
n_features = 1
X = X.reshape((X.shape[0], X.shape[1], n_features))

# Split data into training and test sets
```

```

split = int(0.8 * len(X))
X_train, X_test, y_train, y_test = X[:split], X[split:], y[:split], y[split:]

# Define the LSTM model
model = Sequential()
model.add(LSTM(50, activation='relu', input_shape=(n_steps, n_features)))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse')

# Fit the model
model.fit(X_train, y_train, epochs=200, verbose=0)

# Make predictions
y_pred = model.predict(X_test, verbose=0)

# Inverse transform predictions and actual values
y_pred_inv = scaler.inverse_transform(y_pred)
y_test_inv = scaler.inverse_transform(y_test.reshape(-1, 1))

# Calculate MAPE, MAE, RMSE
mape = np.mean(np.abs((y_test_inv - y_pred_inv) / y_test_inv)) * 100
mae = mean_absolute_error(y_test_inv, y_pred_inv)
rmse = np.sqrt(mean_squared_error(y_test_inv, y_pred_inv))

print(f'MAPE: (mape:.2f %)')
print(f'MAE: (mae:.2f)')
print(f'RMSE: (rmse:.2f)')
')

# Plot the results
plt.plot(y_test_inv, label='Actual')
plt.plot(y_pred_inv, label='Predicted')
plt.xlabel('Date')
plt.ylabel('total')
plt.legend()
plt.show()

```

4. Forecasting algorithm for electric vehicle sales using the GM model.

```

pip install grey-model
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt

```

```

import numpy as np
import matplotlib.pyplot as plt

# Define the Grey Model (1,1) function
def grey_model_11(data):
    n = len(data)
    x0 = data[0]
    x1 = np.cumsum(data)

    z = np.zeros(n-1)
    for i in range(1, n):
        z[i-1] = -0.5 * (x1[i-1] + x1[i])

    B = np.vstack((np.ones(n-1), z)).T
    Y = data[1:]

    # Solve for the coefficients
    a, b = np.linalg.lstsq(B, Y, rcond=None)[0]

    # Forecast the next value
    x_next = (x0 - b/a) * np.exp(-a)

    return x_next

# Example time series data
df = pd.read_excel('data.xlsx')
data = df['จำนวน']

# Forecast using Grey Model (1,1)
forecast = grey_model_11(data)
def grey_forecast(data, steps):
    # Extracting the input sequence
    X = data.values

    # Generate accumulated series
    X1 = np.cumsum(X)

    # Build the first-order grey model
    Z = (X1[:-1] + X1[1:]) / 2

    # Generate the coefficient matrix
    B = np.vstack((-Z, np.ones(len(Z)))).T

    # Solve the coefficients
    a, b = np.linalg.lstsq(B, X[1:], rcond=None)[0]

    # Forecasting

```

```

forecast = (X[0] - b / a) * (1 - np.exp(a)) * np.exp(-a * (len(X) + steps - 1 -
np.arange(steps)))

return forecast

# Define the number of steps ahead to forecast
forecast_steps = 10 # You can adjust this value as needed

# Perform forecasting
forecast_values = grey_forecast(data['จำนวน'], forecast_steps)

# Actual values for comparison
actual_values = data['จำนวน'].values[-forecast_steps:]

# Calculate MAPE, MAE, and RMSE
mape = np.mean(np.abs((actual_values - forecast_values) / actual_values)) * 100
mae = mean_absolute_error(actual_values, forecast_values)
rmse = np.sqrt(mean_squared_error(actual_values, forecast_values))

print("MAPE:", mape)
print("MAE:", mae)
print("RMSE:", rmse)
# Plot the original data and the forecasted value
plt.figure(figsize=(10, 6))
plt.plot(data, label='Original Data', marker='o')
plt.plot(len(data), forecast, 'red', label='Forecasted Value')
plt.legend()
plt.xlabel('Time')
plt.ylabel('Value')
plt.title('Grey Model (1,1) Forecasting')
plt.show()

```

5. A set of instructions for forecasting the variance of electric vehicle sales using a GA model

```
pip install deap numpy pandas
```

```
pip install geneticalgorithm
```

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from deap import base, creator, tools, algorithms

```

```

# Step 1: Load and preprocess data
data = pd.read_excel('data.xlsx')
series = data['จำนวน'].values

```

```

# Split data into train and test sets
train_size = int(len(series) * 0.8)
train, test = series[:train_size], series[train_size:]

# Step 2: Define the fitness function
def fitness_function(individual):
    # Decode the individual to get the parameters
    # For simplicity, assume the individual is the forecasted values directly
    forecast = individual

    # Calculate error (e.g., RMSE)
    error = np.sqrt(np.mean((forecast - test[:len(forecast)])**2))
    return error,

# Step 3: Set up the Genetic Algorithm
creator.create("FitnessMin", base.Fitness, weights=(-1.0,))
creator.create("Individual", list, fitness=creator.FitnessMin)

toolbox = base.Toolbox()
toolbox.register("attr_float", np.random.rand)
toolbox.register("individual", tools.initRepeat, creator.Individual, toolbox.attr_float,
n=len(test))
toolbox.register("population", tools.initRepeat, list, toolbox.individual)

toolbox.register("evaluate", fitness_function)
toolbox.register("mate", tools.cxBlend, alpha=0.5)
toolbox.register("mutate", tools.mutGaussian, mu=0, sigma=1, indpb=0.2)
toolbox.register("select", tools.selTournament, tournsize=3)

population = toolbox.population(n=50)
ngen = 40
cxpb = 0.5
mutpb = 0.2

# Step 4: Run the GA and optimize
best_ind = algorithms.eaSimple(population, toolbox, cxpb, mutpb, ngen,
stats=None, halloffame=None, verbose=True)[1]

best_individual = tools.selBest(population, k=1)[0]
forecast = np.array(best_individual)
best_individual = hof[0]
n_forecast = 10
train_data = values[:-n_forecast]
test_data = values[-n_forecast:]
forecasted_values = forecast_model(best_individual, train_data, n_forecast)

```

```
mape = mean_absolute_percentage_error(test_data, forecasted_values)
mae = mean_absolute_error(test_data, forecasted_values)
rmse = mean_squared_error(test_data, forecasted_values, squared=False)
```

```
# Step 5: Plot the results
plt.figure(figsize=(10, 6))
plt.plot(range(len(train)), train, label='Training Data')
plt.plot(range(len(train), len(series)), test, label='Actual Test Data')
plt.plot(range(len(train), len(train) + len(forecast)), forecast, label='Forecasted Data',
linestyle='dashed')
plt.xlabel('Time')
plt.ylabel('Value')
plt.legend()
plt.show()
print(f'Best Individual: (best_individual)')
print(f'MAPE: (mape)')
print(f'MAE: (mae)')
print(f'RMSE: (rmse)')
```

6 Forecasting instructions for finding the deviation of electric vehicle sales using the GMGA model

```
pip install pandas numpy matplotlib openpyxl xlswriter
```

```
pip install pandas numpy scikit-learn deap openpyxl
```

```
# -*- coding: utf-8 -*-
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
# ===== 1) Load and prepare data =====
```

```
path = "/content/test_data1.xlsx" # <<<== เปลี่ยนเป็น path ไฟล์คุณ
```

```
df = pd.read_excel(path)
```

```
# แปลงชื่อเดือนภาษาไทย -> ตัวเลขเดือน
```

```
thai_months = {
```

```
    "มกราคม":1, "กุมภาพันธ์":2, "มีนาคม":3, "เมษายน":4, "พฤษภาคม":5, "มิถุนายน":6,
    "กรกฎาคม":7, "สิงหาคม":8, "กันยายน":9, "ตุลาคม":10, "พฤศจิกายน":11, "ธันวาคม":12,
    "ม.ค.":1, "ก.พ.":2, "มี.ค.":3, "เม.ย.":4, "พ.ค.":5, "มิ.ย.":6,
    "ก.ค.":7, "ส.ค.":8, "ก.ย.":9, "ต.ค.":10, "พ.ย.":11, "ธ.ค.":12
```

```
}
```

```
df["month_num"] = df["Month"].map(thai_months)
```

```
df["date"] = pd.to_datetime(dict(year=df["Year"], month=df["month_num"], day=1))
```

```

# รวมรายเดือน (ทุกแบรนต์รวมกัน)
monthly = df.groupby("date")["จำนวน"].sum().sort_index()
monthly = monthly.asfreq("MS").interpolate("linear") # ให้เป็น series รายเดือนครบ

series = monthly.astype(float)

# ===== 2) Function GM(1,1) =====
def gm11_fit_forecast(x0, alpha=0.5, m=24):
    """
    GM(1,1) with adjustable alpha (background coefficient)
    """
    x0 = np.asarray(x0, dtype=float)
    n = len(x0)
    if n < 4:
        raise ValueError("ต้องมีข้อมูลอย่างน้อย 4 จุด")
    # 1-AGO
    x1 = np.cumsum(x0)
    z1 = alpha * x1[1:] + (1 - alpha) * x1[:-1]
    B = np.column_stack((-z1, np.ones(n-1)))
    Y = x0[1:]
    theta, _, _ = np.linalg.lstsq(B, Y, rcond=None)
    a, b = theta
    def x1_hat(k):
        return (x0[0] - b/a) * np.exp(-a*(k-1)) + b/a
    x1_hat_vals = np.array([x1_hat(k) for k in range(1, n+m+1)])
    x0_hat_vals = np.empty(n+m)
    x0_hat_vals[0] = x0[0]
    for k in range(1, n+m):
        x0_hat_vals[k] = x1_hat_vals[k] - x1_hat_vals[k-1]
    return ("a":a, "b":b, "fitted":x0_hat_vals[:n], "forecast":x0_hat_vals[n:n+m])

# ===== 3) Genetic Algorithm find alpha =====
rng = np.random.default_rng(123)
pop_size, gens = 50, 80
cx_rate, mut_rate = 0.85, 0.25
horizon = 24

def rmse(y_true, y_pred):
    return np.sqrt(np.nanmean((y_true - y_pred)**2))

def fitness(alpha, x0):
    try:
        out = gm11_fit_forecast(x0, alpha=float(alpha), m=horizon)
        return rmse(x0[1:], out["fitted"][1:])
    except Exception:
        return np.inf

```

```

# shift ถ้ามีค่าติดลบ/ศูนย์
shift = 0
if (series <= 0).any():
    shift = abs(series.min()) + 1e-6
x0 = (series + shift).values

# GA loop
pop = rng.random(pop_size)
for g in range(gens):
    scores = np.array([fitness(a, x0) for a in pop])
    def tournament():
        idx = rng.choice(pop_size, 3, replace=False)
        return pop[idx[np.argmin(scores[idx])]]
    new_pop = []
    while len(new_pop) < pop_size:
        if rng.random() < cx_rate:
            p1, p2 = tournament(), tournament()
            w = rng.random()
            c1 = w*p1 + (1-w)*p2
            c2 = w*p2 + (1-w)*p1
            new_pop.extend([c1, c2])
        else:
            new_pop.append(tournament())
    pop = np.clip(np.array(new_pop[:pop_size]), 0.0, 1.0)
    # mutation
    mut_mask = rng.random(pop_size) < mut_rate
    pop[mut_mask] = np.clip(pop[mut_mask] + rng.normal(0, 0.08, mut_mask.sum()),
0.0, 1.0)

best_alpha = float(pop[np.argmin([fitness(a, x0) for a in pop])])

# ===== 4) Fit model & Forecast =====
out = gml1_fit_forecast(x0, alpha=best_alpha, m=horizon)
fitted = out["fitted"] - shift
forecast = out["forecast"] - shift

dates_hist = series.index
dates_fore = pd.date_range(dates_hist[-1] + pd.offsets.MonthBegin(1),
periods=horizon, freq="MS")

results = pd.DataFrame({
    "Date": list(dates_hist) + list(dates_fore),
    "Actual": list(series.values) + [np.nan]*horizon,
    "Fitted(GMGA)": list(fitted) + [np.nan]*horizon,
    "Forecast(GMGA)": [np.nan]*len(dates_hist) + list(forecast)
})

```

```
# ===== 5) Error Metrics =====
actual_insample = results.loc[results["Actual"].notna(), "Actual"].values
fitted_insample = results.loc[results["Actual"].notna(), "Fitted(GMGA)"].values

MAE = np.mean(np.abs(actual_insample[1:] - fitted_insample[1:]))
RMSE = np.sqrt(np.mean((actual_insample[1:] - fitted_insample[1:]**2)))
MAPE = np.mean(np.abs((actual_insample[1:] - fitted_insample[1:]) /
actual_insample[1:])) * 100

print("Best alpha:", best_alpha)
print("MAE:", MAE)
print("RMSE:", RMSE)
print("MAPE(%):", MAPE)

# ===== 6) Plot =====
plt.figure(figsize=(12,5))
plt.plot(results["Date"], results["Actual"], label="Actual")
plt.plot(results["Date"], results["Fitted(GMGA)"], label="Fitted")
plt.plot(results["Date"], results["Forecast(GMGA)"], label="Forecast (24 mo)")
plt.legend()
plt.grid(True)
plt.title("GMGA Forecast ")
plt.xlabel("Date")
plt.ylabel("total")
plt.tight_layout()
plt.show()

# ===== 7) Save Table =====
results.to_excel("gmga_forecast_24mo.xlsx", index=False)
results.to_csv("gmga_forecast_24mo.csv", index=False)
```

7. Forecasting instructions for finding the deviation of electric vehicle sales using the R2GMGA model

```
pip install deap
```

```
pip install numpy pandas matplotlib deap openpyxl
```

```
import numpy as np
import pandas as pd
from scipy.optimize import minimize
from sklearn.metrics import mean_absolute_error, mean_squared_error
```

```
# Define the residual revised grey model
def residual_revised_grey_model(params, X):
    # Extract model parameters
```

```

a, b = params

# Calculate the predicted values
y_pred = a * X + b

return y_pred

# Define the objective function to minimize using GA
def objective_function(params, X_train, y_train):
    # Predict using the grey model
    y_pred = residual_revised_grey_model(params, X_train)

    # Calculate the mean squared error
    mse = mean_squared_error(y_train, y_pred)

    return mse

# Genetic Algorithm optimization
def optimize_ga(X_train, y_train):
    # Define bounds for parameters
    bounds = [(-10, 10), (-10, 10)] # Example bounds for parameters a and b

    # Run genetic algorithm
    result = minimize(objective_function, x0=np.random.rand(2), args=(X_train,
y_train), bounds=bounds)

    return result.x

# Import data
# Assuming you have your data in a CSV file named 'data.csv'
data = pd.read_excel('test_data1.xlsx')

# Assuming your data has two columns: 'X' and 'y'
X = data['จำนวน'].values
y = data['จำนวน'].values

# Split data into training and testing sets
# Depending on your dataset size, you may choose a different splitting strategy
X_train, X_test = X[:int(0.8*len(X))], X[int(0.8*len(X)):]
y_train, y_test = y[:int(0.8*len(y))], y[int(0.8*len(y)):]

# Optimize the model parameters using GA
optimized_params = optimize_ga(X_train, y_train)

# Evaluate the optimized model
y_pred_train = residual_revised_grey_model(optimized_params, X_train)
y_pred_test = residual_revised_grey_model(optimized_params, X_test)

```

```
# Calculate evaluation metrics
mape_train = np.mean(np.abs((y_train - y_pred_train) / y_train)) * 100
mae_train = mean_absolute_error(y_train, y_pred_train)
rmse_train = np.sqrt(mean_squared_error(y_train, y_pred_train))

mape_test = np.mean(np.abs((y_test - y_pred_test) / y_test)) * 100
mae_test = mean_absolute_error(y_test, y_pred_test)
rmse_test = np.sqrt(mean_squared_error(y_test, y_pred_test))

print("Training Set Metrics:")
print("MAPE:", mape_train)
print("MAE:", mae_train)
print("RMSE:", rmse_train)
print("\nTesting Set Metrics:")
print("MAPE:", mape_test)
print("MAE:", mae_test)
print("RMSE:", rmse_test)
```





APPENDIX C

Electric vehicle sales forecasting commands a set of 2 years in advance

ELECTRIC VEHICLE SALES FORECASTING COMMANDS A SET OF 2 YEARS IN ADVANCE

1.A command set for forecasting electric vehicle sales data in the next 2 years using an ANN model

```

pip install deap openpyxl

pip install deap openpyxl matplotlib pandas scikit-learn

pip install pandas numpy scikit-learn matplotlib tensorflow openpyxl

pip install tensorflow scikit-learn matplotlib pandas openpyxl

pip install pandas openpyxl numpy scikit-learn tensorflow

pip install pandas openpyxl numpy scikit-learn

pip install pandas numpy scikit-learn openpyxl

pip install pandas openpyxl numpy scikit-learn tensorflow

pip install pandas numpy matplotlib scipy openpyxl

pip install pandas openpyxl numpy

pip install geneticalgorithm

pip install deap

import pandas as pd
import numpy as np
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
import matplotlib.pyplot as plt

# Load data
df = pd.read_excel("test_data1.xlsx")

# Convert Thai month names to numbers
thai_months = {
    'มกราคม': 1, 'กุมภาพันธ์': 2, 'มีนาคม': 3, 'เมษายน': 4,
    'พฤษภาคม': 5, 'มิถุนายน': 6, 'กรกฎาคม': 7, 'สิงหาคม': 8,
    'กันยายน': 9, 'ตุลาคม': 10, 'พฤศจิกายน': 11, 'ธันวาคม': 12

```

```

}
df['Month_num'] = df['Month'].map(thai_months)

# Create a date column
df['Date'] = pd.to_datetime(df.rename(columns=('Year': 'year', 'Month_num':
'month'))[['year', 'month']].assign(day=1))

# Monthly data collection
monthly = df.groupby('Date')['จำนวน'].sum().reset_index().sort_values('Date')

# Prepare data
data = monthly['จำนวน'].values.astype(float).reshape(-1, 1)
scaler = MinMaxScaler()
data_scaled = scaler.fit_transform(data)

# Create a timeline
def create_dataset(data, look_back=12):
    X, y = [], []
    for i in range(len(data) - look_back):
        X.append(data[i:i+look_back, 0])
        y.append(data[i + look_back, 0])
    return np.array(X), np.array(y)

look_back = 12
X, y = create_dataset(data_scaled, look_back)

# Build and train models
model = Sequential([
    Dense(64, input_dim=look_back, activation='relu'),
    Dense(32, activation='relu'),
    Dense(1)
])
model.compile(loss='mse', optimizer='adam')
model.fit(X, y, epochs=300, verbose=0)

# 24 month forecast
forecast_scaled = list(data_scaled.flatten())
for _ in range(24):
    input_data = np.array(forecast_scaled[-look_back:]).reshape(1, -1)
    next_pred = model.predict(input_data, verbose=0)[0][0]
    forecast_scaled.append(next_pred)

# Restore scale
forecast = scaler.inverse_transform(np.array(forecast_scaled).reshape(-1, 1)).flatten()

# Actual data date + future date
dates_all = list(monthly['Date'])

```

```

last_date = dates_all[-1]
future_dates = [last_date + pd.DateOffset(months=i+1) for i in range(24)]
dates_all.extend(future_dates)

# Result table
actual_all = np.concatenate([data.flatten(), [np.nan]*24])
forecast_all = forecast

result_df = pd.DataFrame({
    'Date': dates_all,
    'Actual': actual_all,
    'Forecast_ANN': np.round(forecast_all, 2)
})

# Evaluation (only real data)
valid_mask = ~np.isnan(actual_all)
rmse = np.sqrt(mean_squared_error(actual_all[valid_mask],
forecast_all[valid_mask]))
mae = mean_absolute_error(actual_all[valid_mask], forecast_all[valid_mask])
mape = np.mean(np.abs((actual_all[valid_mask] - forecast_all[valid_mask]) /
actual_all[valid_mask])) * 100

# Show results
print("\n tableANN:")
print(result_df.tail(30).to_string(index=False))
print(f"\nError ANN:")
print(f" RMSE : (rmse:.2f)")
print(f" MAE : (mae:.2f)")
print(f" MAPE : (mape:.2f)%")

# Graph
plt.figure(figsize=(12, 6))
plt.plot(result_df['Date'], result_df['Actual'], label='Actual', marker='o')
plt.plot(result_df['Date'], result_df['Forecast_ANN'], label='Forecast', marker='x')
plt.title('Actual vs Forecast (ANN)')
plt.xlabel('Date')
plt.ylabel('total')
plt.xticks(rotation=45)
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```

2. A command set for forecasting electric vehicle sales data in the next 2 years using an RNN and LSTM model

pip install pandas numpy matplotlib scikit-learn tensorflow

```

pip install pandas numpy matplotlib scikit-learn tensorflow
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LSTM, SimpleRNN
from tensorflow.keras.optimizers import Adam

df = pd.read_excel("test_data1.xlsx")

month_order = [
    "มกราคม": 1, "กุมภาพันธ์": 2, "มีนาคม": 3, "เมษายน": 4, "พฤษภาคม": 5,
    "มิถุนายน": 6, "กรกฎาคม": 7, "สิงหาคม": 8, "กันยายน": 9,
    "ตุลาคม": 10, "พฤศจิกายน": 11, "ธันวาคม": 12
]
df['Month_num'] = df['Month'].map(month_order)
monthly_data = df.groupby(['Year', 'Month_num'])['จำนวน'].sum().reset_index()
monthly_data['Date'] = pd.to_datetime(dict(year=monthly_data['Year'],
month=monthly_data['Month_num'], day=1))
monthly_data = monthly_data.sort_values('Date')

series = monthly_data[['Date', 'จำนวน']].set_index('Date')
values = series['จำนวน'].values.reshape(-1, 1)
scaler = MinMaxScaler()
scaled = scaler.fit_transform(values)

def create_sequences(data, seq_len):
    X, y = [], []
    for i in range(len(data) - seq_len):
        X.append(data[i:i+seq_len])
        y.append(data[i+seq_len])
    return np.array(X), np.array(y)

seq_len = 12
X, y = create_sequences(scaled, seq_len)

model_lstm = Sequential([
    LSTM(50, activation='relu', input_shape=(seq_len, 1)),
    Dense(1)
])
model_lstm.compile(optimizer=Adam(learning_rate=0.01), loss='mse')
model_lstm.fit(X, y, epochs=100, verbose=0)

model_rnn = Sequential([
    SimpleRNN(50, activation='relu', input_shape=(seq_len, 1)),

```

```

Dense(1)
])
model_rnn.compile(optimizer=Adam(learning_rate=0.01), loss='mse')
model_rnn.fit(X, y, epochs=200, verbose=0)

def forecast(model, last_seq, n_steps):
    output = []
    seq = last_seq.copy()
    for _ in range(n_steps):
        pred = model.predict(seq.reshape(1, seq_len, 1), verbose=0)
        output.append(pred[0][0])
        seq = np.append(seq[1:], pred)
    return np.array(output)

last_seq = scaled[-seq_len:]
lstm_pred = forecast(model_lstm, last_seq, 24)
rnn_pred = forecast(model_rnn, last_seq, 24)

lstm_actual = scaler.inverse_transform(lstm_pred.reshape(-1,1)).flatten()
rnn_actual = scaler.inverse_transform(rnn_pred.reshape(-1,1)).flatten()

dates_future = pd.date_range(start=series.index[-1] + pd.offsets.MonthBegin(1),
periods=24, freq='MS')
result_df = pd.DataFrame( {
    "Date": list(series.index) + list(dates_future),
    "Actual": list(series['จำนวน']) + [np.nan]*24,
    "LSTM_Forecast": list([np.nan]*len(series)) + list(lstm_actual),
    "RNN_Forecast": list([np.nan]*len(series)) + list(rnn_actual)
})

print(result_df.tail(30))

plt.figure(figsize=(12,6))
plt.plot(result_df['Date'], result_df['Actual'], label="Actual")
plt.plot(result_df['Date'], result_df['LSTM_Forecast'], label="LSTM Forecast")
plt.plot(result_df['Date'], result_df['RNN_Forecast'], label="RNN Forecast")
plt.axvline(series.index[-1], color='red', linestyle=':', label="Forecast Start")
plt.title("EV Forecast (LSTM vs RNN)")
plt.xlabel("Date")
plt.ylabel("Total EV")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```

3. A command set for forecasting electric vehicle sales data in the next 2 years using a GM model

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error, mean_absolute_error
from scipy.optimize import minimize
from deap import base, creator, tools, algorithms
import random

df = pd.read_excel("test_data1.xlsx")

thai_months = {
    'มกราคม': 1, 'กุมภาพันธ์': 2, 'มีนาคม': 3, 'เมษายน': 4,
    'พฤษภาคม': 5, 'มิถุนายน': 6, 'กรกฎาคม': 7, 'สิงหาคม': 8,
    'กันยายน': 9, 'ตุลาคม': 10, 'พฤศจิกายน': 11, 'ธันวาคม': 12
}
df['Month_num'] = df['Month'].map(thai_months)
df['Date'] = pd.to_datetime(df.rename(columns=('Year': 'year', 'Month_num':
'month'))[['year', 'month']].assign(day=1))
monthly = df.groupby('Date')['จำนวน'].sum().reset_index().sort_values('Date')
data = monthly['จำนวน'].values
dates = monthly['Date'].tolist()

def gm11(x0, n_forecast):
    x1 = np.cumsum(x0)
    z1 = np.array([(x1[i] + x1[i-1]) / 2 for i in range(1, len(x1))])
    B = np.vstack([-z1, np.ones(len(z1))]).T
    Y = x0[1:]
    a_b = np.linalg.lstsq(B, Y, rcond=None)[0]
    a, b = a_b
    f = lambda k: (x0[0] - b / a) * np.exp(-a * k) + b / a
    forecast = [f(i) for i in range(len(x0) + n_forecast)]
    return np.array(forecast), a, b

def evaluate_forecast(actual, forecast):
    mask = ~np.isnan(actual)
    rmse = np.sqrt(mean_squared_error(actual[mask], forecast[mask]))
    mae = mean_absolute_error(actual[mask], forecast[mask])
    mape = np.mean(np.abs((actual[mask] - forecast[mask]) / actual[mask])) * 100
    return rmse, mae, mape

forecast_gm, a_gm, b_gm = gm11(data, 24)

```

```

def gm_model(a, b, x0, steps):
    f = lambda k: (x0[0] - b / a) * np.exp(-a * k) + b / a
    return np.array([f(i) for i in range(len(x0) + steps)])

def fitness(individual, x0, steps):
    a, b = individual
    pred = gm_model(a, b, x0, 0)
    return np.sqrt(mean_squared_error(x0, pred[:len(x0)])),

creator.create("FitnessMin", base.Fitness, weights=(-1.0,))
creator.create("Individual", list, fitness=creator.FitnessMin)
toolbox = base.Toolbox()
toolbox.register("attr_float", random.uniform, -1, 1)
toolbox.register("individual", tools.initCycle, creator.Individual,
                 (toolbox.attr_float, toolbox.attr_float), n=1)
toolbox.register("population", tools.initRepeat, list, toolbox.individual)
toolbox.register("evaluate", fitness, x0=data, steps=24)
toolbox.register("mate", tools.cxBlend, alpha=0.5)
toolbox.register("mutate", tools.mutGaussian, mu=0, sigma=0.1, indpb=0.2)
toolbox.register("select", tools.selTournament, tournsize=3)

pop = toolbox.population(n=50)
algorithms.eaSimple(pop, toolbox, cxpb=0.6, mutpb=0.2, ngen=50, verbose=False)

best_ind = tools.selBest(pop, 1)[0]
a_ga, b_ga = best_ind[0], best_ind[1]
forecast_ga = gm_model(a_ga, b_ga, data, 24)

forecast_len = len(data) + 24
actual_all = np.concatenate([data, [np.nan]*24])
future_dates = [dates[-1] + pd.DateOffset(months=i+1) for i in range(24)]
all_dates = dates + future_dates

result_df = pd.DataFrame((
    'Date': all_dates,
    'Actual': actual_all,
    'GM(1,1)': np.round(forecast_gm, 2),
    'GmGA': np.round(forecast_ga, 2)
))

rmse_gm, mae_gm, mape_gm = evaluate_forecast(actual_all, forecast_gm)
rmse_ga, mae_ga, mape_ga = evaluate_forecast(actual_all, forecast_ga)

print(result_df.tail(30).to_string(index=False))
print("\n 📊 ความแม่นยำ:")
print(f"GM(1,1):          RMSE=[rmse_gm:.2f],          MAE=[mae_gm:.2f],
MAPE=[mape_gm:.2f]%)")

```

```
print(f'GmGA          :      RMSE=[rmse_ga:.2f],      MAE=[mae_ga:.2f],
MAPE=[mape_ga:.2f]%%')
```

```
plt.figure(figsize=(12, 6))
plt.plot(result_df['Date'], result_df['Actual'], label='Actual', marker='o')
plt.plot(result_df['Date'], result_df['GM(1,1)'], label='GM(1,1)', marker='x')
plt.xticks(rotation=45)
plt.xlabel("Date")
plt.ylabel("total")
plt.title("Forecast: GM(1,1) ")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

4. A command set for forecasting electric vehicle sales data in the next 2 years using an GA model

```
import pandas as pd
import numpy as np
from datetime import datetime
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error
from deap import base, creator, tools, algorithms
import random

df = pd.read_excel('test_data1.xlsx')

thai_months = [
    'มกราคม': 1, 'กุมภาพันธ์': 2, 'มีนาคม': 3, 'เมษายน': 4,
    'พฤษภาคม': 5, 'มิถุนายน': 6, 'กรกฎาคม': 7, 'สิงหาคม': 8,
    'กันยายน': 9, 'ตุลาคม': 10, 'พฤศจิกายน': 11, 'ธันวาคม': 12
]
df['Month_num'] = df['Month'].map(thai_months)

monthly_total = df.groupby(['Year', 'Month_num'])['จำนวน'].sum().reset_index()
monthly_total['Month_num'] = monthly_total['Month_num'].astype(int)
monthly_total['Date'] = pd.to_datetime(monthly_total['Year'].astype(str) + '-' +
monthly_total['Month_num'].astype(str) + '-01')
monthly_total = monthly_total.sort_values('Date').reset_index(drop=True)

sales = monthly_total['จำนวน'].values
t = np.arange(len(sales))

def model(params, t):
    a, b = params
```

```

result = a * np.exp(b * t)
return np.clip(result, 0, 1e6)

def evaluate(individual):
    pred = model(individual, t)
    rmse = np.sqrt(mean_squared_error(sales, pred))
    return (rmse,)

creator.create("FitnessMin", base.Fitness, weights=(-1.0,))
creator.create("Individual", list, fitness=creator.FitnessMin)

toolbox = base.Toolbox()
toolbox.register("a_param", random.uniform, 0.1, 500)
toolbox.register("b_param", random.uniform, -0.1, 0.1)
toolbox.register("individual", tools.initCycle, creator.Individual,
                (toolbox.a_param, toolbox.b_param), n=1)
toolbox.register("population", tools.initRepeat, list, toolbox.individual)

toolbox.register("evaluate", evaluate)
toolbox.register("mate", tools.cxBlend, alpha=0.5)
toolbox.register("mutate", tools.mutGaussian, mu=0, sigma=1.0, indpb=0.2)
toolbox.register("select", tools.selTournament, tournsize=3)

pop = toolbox.population(n=50)
hof = tools.HallOfFame(1)
stats = tools.Statistics(lambda ind: ind.fitness.values)
stats.register("avg", np.mean)
stats.register("min", np.min)

algorithms.eaSimple(pop, toolbox, cxpb=0.5, mutpb=0.3, ngen=100,
                    stats=stats, halloffame=hof, verbose=False)

best_params = hof[0]
print(f'Best Parameters (a, b): {best_params}')

t_all = np.arange(len(sales) + 24)
forecast_values = model(best_params, t_all)

future_dates = pd.date_range(start=monthly_total['Date'].iloc[0], periods=len(t_all),
                              freq='MS')
results = pd.DataFrame([
    'Date': future_dates,
    'Actual': np.append(sales, [np.nan]*24),
    'Forecast': forecast_values
])

print(results.tail(24))

```

```

plt.figure(figsize=(12,6))
plt.plot(results['Date'], results['Actual'], label='Actual', marker='o')
plt.plot(results['Date'], results['Forecast'], label='Forecast (GA)', linestyle='--')
plt.title('Forecasting using Genetic Algorithm (GA)')
plt.xlabel('Date')
plt.ylabel('total')
plt.grid(True)
plt.legend()
plt.tight_layout()
plt.show()

results.to_excel('GA_forecast_results.xlsx', index=False)

```

5. A command set for forecasting electric vehicle sales data in the next 2 years using a GMGA model

```

pip install pandas numpy matplotlib openpyxl xlswriter

```

```

pip install pandas numpy scikit-learn deap openpyxl

```

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

```

```

path = "/content/test_data1.xlsx"
df = pd.read_excel(path)

```

```

thai_months = [
    "มกราคม":1, "กุมภาพันธ์":2, "มีนาคม":3, "เมษายน":4, "พฤษภาคม":5, "มิถุนายน":6,
    "กรกฎาคม":7, "สิงหาคม":8, "กันยายน":9, "ตุลาคม":10, "พฤศจิกายน":11, "ธันวาคม":12,
    "ม.ค.":1, "ก.พ.":2, "มี.ค.":3, "เม.ย.":4, "พ.ค.":5, "มิ.ย.":6,
    "ก.ค.":7, "ส.ค.":8, "ก.ย.":9, "ต.ค.":10, "พ.ย.":11, "ธ.ค.":12
]
df["month_num"] = df["Month"].map(thai_months)
df["date"] = pd.to_datetime(dict(year=df["Year"], month=df["month_num"], day=1))

```

```

monthly = df.groupby("date")["จำนวน"].sum().sort_index()
monthly = monthly.asfreq("MS").interpolate("linear")

```

```

series = monthly.astype(float)

```

```

def gm11_fit_forecast(x0, alpha=0.5, m=24):
    """

```

GM(1,1) with adjustable alpha (background coefficient)

```

"""
x0 = np.asarray(x0, dtype=float)
n = len(x0)
if n < 4:
    raise ValueError("ต้องมีข้อมูลอย่างน้อย 4 จุด")

x1 = np.cumsum(x0)
z1 = alpha * x1[1:] + (1 - alpha) * x1[:-1]
B = np.column_stack((-z1, np.ones(n-1)))
Y = x0[1:]
theta, _, _ = np.linalg.lstsq(B, Y, rcond=None)
a, b = theta
def x1_hat(k):
    return (x0[0] - b/a) * np.exp(-a*(k-1)) + b/a
x1_hat_vals = np.array([x1_hat(k) for k in range(1, n+m+1)])
x0_hat_vals = np.empty(n+m)
x0_hat_vals[0] = x0[0]
for k in range(1, n+m):
    x0_hat_vals[k] = x1_hat_vals[k] - x1_hat_vals[k-1]
return [{"a":a, "b":b, "fitted":x0_hat_vals[:n], "forecast":x0_hat_vals[n:n+m]]

rng = np.random.default_rng(123)
pop_size, gens = 50, 80
cx_rate, mut_rate = 0.85, 0.25
horizon = 24

def rmse(y_true, y_pred):
    return np.sqrt(np.nanmean((y_true - y_pred)**2))

def fitness(alpha, x0):
    try:
        out = gml1_fit_forecast(x0, alpha=float(alpha), m=horizon)
        return rmse(x0[1:], out["fitted"][1:])
    except Exception:
        return np.inf

shift = 0
if (series <= 0).any():
    shift = abs(series.min()) + 1e-6
x0 = (series + shift).values

pop = rng.random(pop_size)
for g in range(gens):
    scores = np.array([fitness(a, x0) for a in pop])
    def tournament():
        idx = rng.choice(pop_size, 3, replace=False)
        return pop[idx[np.argmin(scores[idx])]]

```

```

new_pop = []
while len(new_pop) < pop_size:
    if rng.random() < cx_rate:
        p1, p2 = tournament(), tournament()
        w = rng.random()
        c1 = w*p1 + (1-w)*p2
        c2 = w*p2 + (1-w)*p1
        new_pop.extend([c1, c2])
    else:
        new_pop.append(tournament())
pop = np.clip(np.array(new_pop[:pop_size]), 0.0, 1.0)
mut_mask = rng.random(pop_size) < mut_rate
pop[mut_mask] = np.clip(pop[mut_mask] + rng.normal(0, 0.08, mut_mask.sum()),
0.0, 1.0)

best_alpha = float(pop[np.argmin([fitness(a, x0) for a in pop])])

out = gm11_fit_forecast(x0, alpha=best_alpha, m=horizon)
fitted = out["fitted"] - shift
forecast = out["forecast"] - shift

dates_hist = series.index
dates_fore = pd.date_range(dates_hist[-1] + pd.offsets.MonthBegin(1),
periods=horizon, freq="MS")

results = pd.DataFrame({
    "Date": list(dates_hist) + list(dates_fore),
    "Actual": list(series.values) + [np.nan]*horizon,
    "Fitted(GMGA)": list(fitted) + [np.nan]*horizon,
    "Forecast(GMGA)": [np.nan]*len(dates_hist) + list(forecast)
})

actual_insample = results.loc[results["Actual"].notna(), "Actual"].values
fitted_insample = results.loc[results["Actual"].notna(), "Fitted(GMGA)"].values

MAE = np.mean(np.abs(actual_insample[1:] - fitted_insample[1:]))
RMSE = np.sqrt(np.mean((actual_insample[1:] - fitted_insample[1:])**2))
MAPE = np.mean(np.abs((actual_insample[1:] - fitted_insample[1:]) /
actual_insample[1:])) * 100

print("Best alpha:", best_alpha)
print("MAE:", MAE)
print("RMSE:", RMSE)
print("MAPE(%)", MAPE)

plt.figure(figsize=(12,5))
plt.plot(results["Date"], results["Actual"], label="Actual")

```

```

plt.plot(results["Date"], results["Fitted(GMGA)"], label="Fitted")
plt.plot(results["Date"], results["Forecast(GMGA)"], label="Forecast (24 mo)")
plt.legend()
plt.grid(True)
plt.title("GMGA Forecast")
plt.xlabel("Date")
plt.ylabel("total")
plt.tight_layout()
plt.show()
results.to_excel("gmga_forecast_24mo.xlsx", index=False)
results.to_csv("gmga_forecast_24mo.csv", index=False)

```

6. A command set for forecasting electric vehicle sales data in the next 2 years using an R2GMGA model

```
pip install deap
```

```
pip install numpy pandas matplotlib deap openpyxl
```

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from datetime import timedelta
from deap import base, creator, tools, algorithms

# ---- STEP 1: Load and preprocess data ----

df = pd.read_excel('test_data1.xlsx')

thai_months = [
    'มกราคม': 1, 'กุมภาพันธ์': 2, 'มีนาคม': 3, 'เมษายน': 4,
    'พฤษภาคม': 5, 'มิถุนายน': 6, 'กรกฎาคม': 7, 'สิงหาคม': 8,
    'กันยายน': 9, 'ตุลาคม': 10, 'พฤศจิกายน': 11, 'ธันวาคม': 12
]

df['Month'] = df['Month'].map(thai_months)
df['Date'] = pd.to_datetime(df[['Year', 'Month']].assign(DAY=1))
monthly_data = df.groupby('Date')['จำนวน'].sum().sort_index()

# ---- STEP 2: Define Grey Model GM(1,1) ----

def gm11(x0, forecast_horizon=12):
    """Grey Model GM(1,1) with adjustable forecast horizon."""
    X1 = np.cumsum(x0)
    B = np.array([-0.5*(x1[i]+x1[i+1]) for i in range(len(x0)-1)]).reshape(-1,1)
    B = np.hstack((B, np.ones((len(B), 1))))
    Y = x0[1:].reshape(-1,1)
    [[a], [b]] = np.linalg.inv(B.T @ B) @ B.T @ Y

```

```

x0_hat = [x0[0]]
for k in range(1, len(x0) + forecast_horizon): # dynamic steps ahead
    x0_hat.append((x0[0] - b/a)*(1 - np.exp(a)) * np.exp(-a*k))
x0_hat = np.array(x0_hat)
return x0_hat, a, b

# ---- STEP 3: Define GA Optimization ----

def evaluate(individual, data):
    alpha = individual[0]
    x0 = data.copy()
    residual = x0 - gm11(x0[:len(x0)])[0][:len(x0)]
    x0_corr = x0 + alpha * residual
    pred, _, _ = gm11(x0_corr[:len(x0)])
    rmse = np.sqrt(np.mean((pred[:len(x0)] - x0) ** 2))
    return rmse,

def optimize_rgmga(data):
    if 'FitnessMin' not in creator.__dict__:
        creator.create("FitnessMin", base.Fitness, weights=(-1.0,))
    if 'Individual' not in creator.__dict__:
        creator.create("Individual", list, fitness=creator.FitnessMin)

    toolbox = base.Toolbox()
    toolbox.register("attr_float", np.random.rand)
    toolbox.register("individual", tools.initRepeat, creator.Individual, toolbox.attr_float,
1)
    toolbox.register("population", tools.initRepeat, list, toolbox.individual)
    toolbox.register("evaluate", evaluate, data=data)
    toolbox.register("mate", tools.cxBlend, alpha=0.5)
    toolbox.register("mutate", tools.mutGaussian, mu=0, sigma=0.1, indpb=0.2)
    toolbox.register("select", tools.selTournament, tournsize=3)

    pop = toolbox.population(n=30)
    result, _ = algorithms.eaSimple(pop, toolbox, cxpb=0.5, mutpb=0.2, ngen=30,
verbose=False)
    best = tools.selBest(pop, k=1)[0]
    return best[0]

# ---- STEP 4: Forecast using optimized RGMGA ----

data = monthly_data.values.astype(float)
forecast_horizon = 18

alpha = optimize_rgmga(data)
residual = data - gm11(data[:len(data)])[0][:len(data)]
data_corr = data + alpha * residual

```

```

forecast, a, b = gm11(data_corr[:len(data)], forecast_horizon=forecast_horizon)

# ---- STEP 5: Create Forecast Output ----

dates = monthly_data.index
forecast_index = pd.date_range(dates[-1] + timedelta(days=31),
                                periods=forecast_horizon, freq='MS')
forecast_values = forecast[len(data):len(data)+forecast_horizon]

forecast_df = pd.DataFrame(['Forecast': forecast_values], index=forecast_index)
print("\n=== Forecast Results ===")
print(forecast_df)

# ---- STEP 6: Plot ----

plt.figure(figsize=(12, 5))
plt.plot(dates, data, label='Actual', color='blue')
plt.plot(forecast_df.index, forecast_df['Forecast'], label=f'Forecast', linestyle='-',
color='red')
plt.title('R2GMGA Forecast')
plt.xlabel('Date')
plt.ylabel('total')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

```



APPENDIX D

A set of instructions to find the relationship between electric vehicle purchases and the creation of electric vehicle charging stations.

A SET OF INSTRUCTIONS TO FIND THE RELATIONSHIP BETWEEN ELECTRIC VEHICLE PURCHASES AND THE CREATION OF ELECTRIC VEHICLE CHARGING STATIONS

1. A set of instructions to find the relationship between electric vehicle purchases and the creation of electric vehicle charging stations (department store type).

Import pandas as pd

```

ev = pd.read_excel("Malls.xlsx", sheet_name=None)
test = pd.read_excel("test_data1.xlsx", sheet_name="Sheet1")
ev_all = pd.concat(ev.values(), ignore_index=True)
ev_all["จำนวนหัว"] = pd.to_numeric(ev_all["จำนวนหัว"],
errors="coerce").fillna(0).astype(int)
total_chargers = ev_all["จำนวนหัว"].sum()
test["จำนวน"] = pd.to_numeric(test["จำนวน"], errors="coerce").fillna(0).astype(int)
ev_by_year = test.groupby("Year", as_index=False)[
"จำนวน"].sum().sort_values("Year")
avg_session_hr = 1.0
operational_hours = 12.0
utilization = 0.5
vehicles_per_charger_per_day = (operational_hours * utilization) / avg_session_hr
vehicles_supported_per_day = total_chargers * vehicles_per_charger_per_day
vehicles_supported_per_year = vehicles_supported_per_day * 365
summary = pd.DataFrame({
    "total_chargers": [total_chargers],
    "vehicles_per_charger_per_day": [vehicles_per_charger_per_day],
    "vehicles_supported_per_day": [int(round(vehicles_supported_per_day))],
    "vehicles_supported_per_year": [int(round(vehicles_supported_per_year))],
    "assumptions": [f'avg_session_hr={avg_session_hr},
operational_hours={operational_hours}, utilization={utilization}']
})

print("\n=== Summary ===")
print(summary)
freq_days = [1, 2, 3, 7]
fleet_support = []
for d in freq_days:
    fleet_support.append(int(round(vehicles_supported_per_day * d)))

freq_df = pd.DataFrame([
    "days_between_charges": freq_days,
    "supported_fleet_size": fleet_support
])

```

```

print("\n=== Supported Fleet Size by Charging Frequency ===")
print(freq_df)
comparison = ev_by_year.copy()
comparison["capacity_per_year"] = int(round(vehicles_supported_per_year))
comparison["capacity_minus_ev"] = comparison["capacity_per_year"] -
comparison["จำนวน"]

print("\n=== EV count vs Capacity ===")
print(comparison)

```

2. A set of instructions to find the relationship between electric vehicle purchases and the creation of electric vehicle charging stations (gas stations)

```

import pandas as pd

ev = pd.read_excel("Stations.xlsx", sheet_name=None)
test = pd.read_excel("test_data1.xlsx", sheet_name="Sheet1")
ev_all = pd.concat(ev.values(), ignore_index=True)
ev_all["จำนวนหัว"] = pd.to_numeric(ev_all["จำนวนหัว"],
errors="coerce").fillna(0).astype(int)
total_chargers = ev_all["จำนวนหัว"].sum()
test["จำนวน"] = pd.to_numeric(test["จำนวน"], errors="coerce").fillna(0).astype(int)
ev_by_year = test.groupby("Year", as_index=False)[
"จำนวน"].sum().sort_values("Year")
avg_session_hr = 1.0
operational_hours = 24.0
utilization = 0.1
vehicles_per_charger_per_day = (operational_hours * utilization) / avg_session_hr
vehicles_supported_per_day = total_chargers * vehicles_per_charger_per_day
vehicles_supported_per_year = vehicles_supported_per_day * 365
summary = pd.DataFrame({
    "total_chargers": [total_chargers],
    "vehicles_per_charger_per_day": [vehicles_per_charger_per_day],
    "vehicles_supported_per_day": [int(round(vehicles_supported_per_day))],
    "vehicles_supported_per_year": [int(round(vehicles_supported_per_year))],
    "assumptions": [f'avg_session_hr={avg_session_hr},
operational_hours={operational_hours}, utilization={utilization}']
}))

print("\n=== Summary ===")
print(summary)
freq_days = [1, 2, 3, 7]
fleet_support = []
for d in freq_days:
    fleet_support.append(int(round(vehicles_supported_per_day * d)))

```

```

freq_df = pd.DataFrame({
    "days_between_charges": freq_days,
    "supported_fleet_size": fleet_support
})

print("\n=== Supported Fleet Size by Charging Frequency ===")
print(freq_df)
comparison = ev_by_year.copy()
comparison["capacity_per_year"] = int(round(vehicles_supported_per_year))
comparison["capacity_minus_ev"] = comparison["capacity_per_year"] -
comparison["จำนวน"]

print("\n=== EV count vs Capacity ===")
print(comparison)

```



APPENDIX E

Website showing research results

WEBSITE SHOWING RESEARCH RESULTS

1. Entering the website

Press to enter <https://app.site123.com/?w=11256440>. Log in to the website
As shown in picture D-1

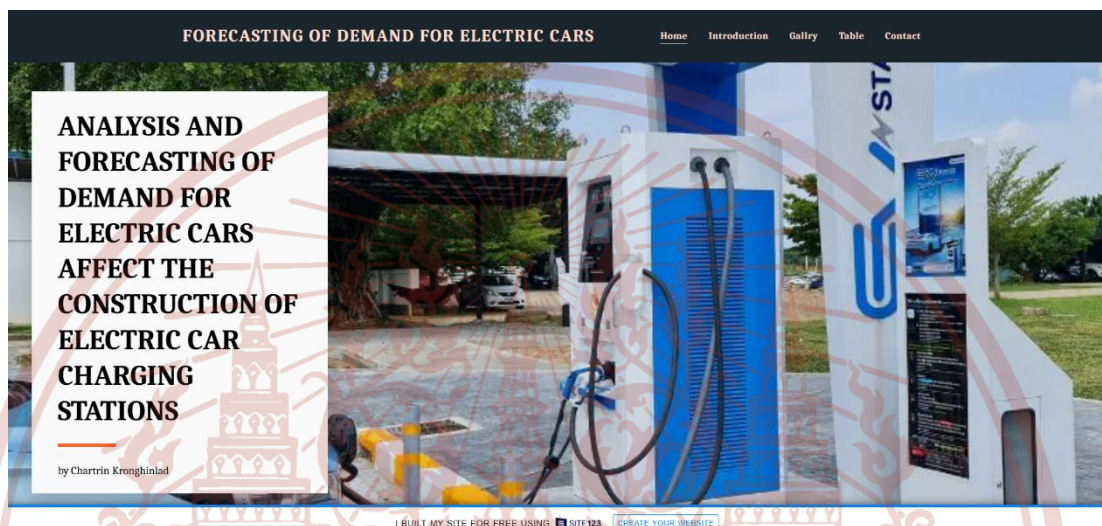


Figure E-1 Web main page

Figure E-1 presents the website developed by the researchers to display information related to this study. The primary purpose of the website is to present the research results in a clear and accessible manner, including graphical visualizations, data tables comparing actual values, error analysis tables, tables illustrating the relationship between the number of electric vehicles (EVs) and the rate of EV charging station construction, and tables showing daily EV charging rates.

2. Options menu bar

You can select the options that you want to be displayed, as shown in Figure E-2.

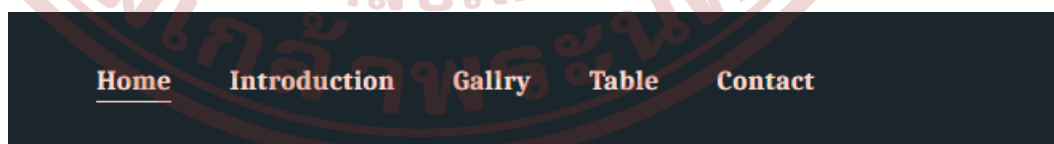


Figure E-2 Options Website

Figure E-2 illustrates the menu bar of the website developed by researchers. The menu consists of Home, which serves as the homepage; Introduction, which briefly explains the purpose and scope of the thesis; Gallery, which presents the results of all models in graphical form; Table, which displays the research results in tabular format, including comparisons with actual values, error analysis tables, tables illustrating the relationship between the number of electric vehicles (EVs) and the rate of EV charging station construction, and tables showing daily EV charging rates; and Contact, which

allows visitors to provide their information and contact details to communicate with the website developer.

3.Introduction

Show the window showing the Introduction information related to this research. AS shown in Figure E-3.



Figure E-3 Introductions pages

Figure E-3 presents the Introduction window, which provides an overview of the purpose and subject matter of this thesis.

4. Gallery

4. Show a window showing Gallery data related to this research. As shown in Figure E-4

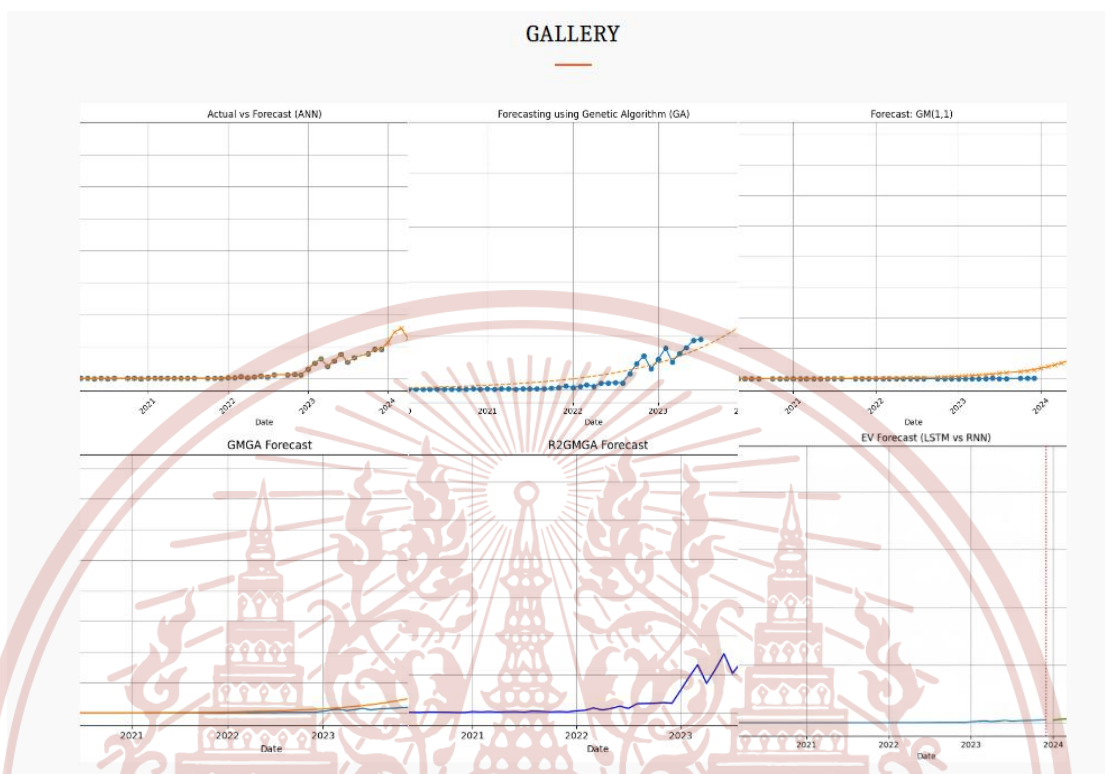


Figure E-4 Gallery pages

Figure E-4 presents the Gallery window, which displays graphical plots of the projected electric vehicle (EV) sales generated by each forecasting model, including ANN, RNN, LSTM, GM, GA, GM–GA, and R2GM–GA. The horizontal axis represents the time period in years, while the vertical axis indicates the total number of EVs predicted by each model.

5. Table

Shows a window showing information about the actual forecast and the electric vehicle charging station construction relationship, as shown in Figure E-5.

Date	Actual	ANN	RNN	LSTM	GM	GA	GMGA	R2-GMGA
1/1/2567	13,974	10,696	7,486	10,200	34,199	11,695	27,303	6,500
1/2/2567	4,128	13,665	12,173	11,087	392,215	13,314	30,690	6,625
1/3/2567	5,224	14,100	14,259	12,024	449,811	14,078	34,498	6,759
1/4/2567	5,677	12,826	10,130	13,013	515,861	15,397	38,778	6,903
1/5/2567	7,692	14,616	17,867	14,078	591,606	16,860	43,589	7,056
1/6/2567	5,888	16,147	19,476	15,397	678,469	18,480	48,997	7,225
1/7/2567	6,020	15,799	21,250	16,860	778,082	20,550	55,075	7,399
1/8/2567	8,354	17,580	25,771	18,480	892,316	22,983	61,908	7,590
1/9/2567	6,288	19,628	28,858	20,550	1,023,318	25,929	69,589	7,794
1/10/2567	6,631	21,918	36,064	22,983	1,173,549	29,560	78,222	8,013
TOTAL	69,876	156,974	193,156	154,673	6,837,219	224,434	488,649	71,864

Figure E-5 Table pages

Figure E-5 presents the results table displayed on the website interface. This section provides the research findings in tabular form. The table illustrates the forecasting results for electric vehicle demand generated by the ANN, RNN, LSTM, GM, GA, GM-GA, and R2GM-GA models, alongside the actual observed values collected by the researchers for error evaluation. The comparison indicates that the R2GM-GA model achieved the lowest forecasting error among all forecasting models examined in this study. In addition, the website contains two other result figures, resulting in a total of three result displays presented on the website.

VITA

Name Mr. Chartrin Kronghinlad

Thesis Title Electric Vehicle Demand Forecasting Using Machine Learning and Charging Infrastructure Suitability Analysis in Thailand

Major Field Mathematics with Computer Science

Biography Chartrin Kronghinlad received a Bachelor's degree in Applied Mathematics from King Mongkut's University of Technology North Bangkok and is currently pursuing a Master's degree in Mathematics with Computer Science at the same university. He is currently working as a Data Analyst at FUJIFILM Business Innovation (Thailand). His research interests include machine learning, data analytics, and data science. He can be contacted via email at Chartrinfirst@gmail.com.

