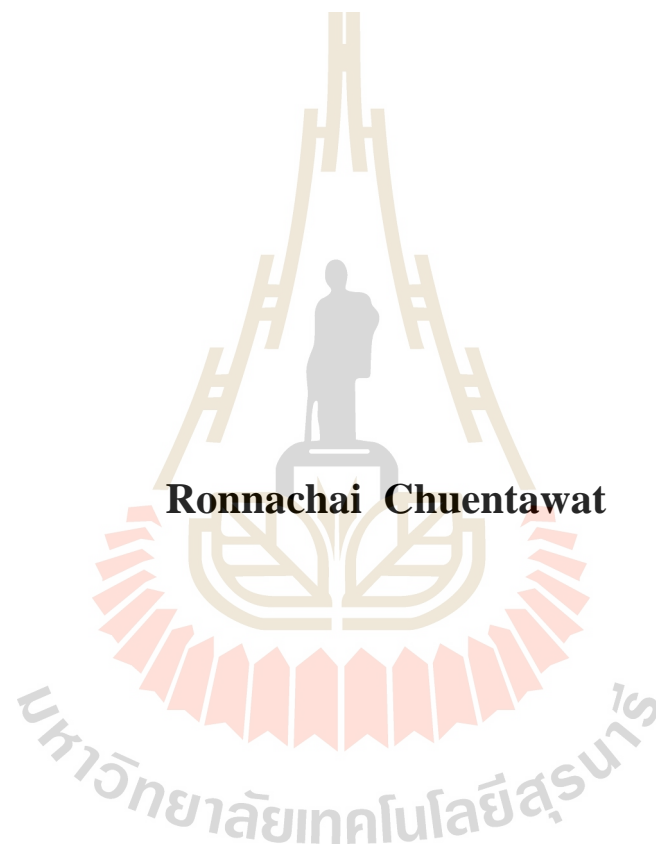


# การพยากรณ์อนุกรมเวลาด้วยเทคนิคอัจฉริยะและการวิเคราะห์อนุกรมเวลา



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต  
สาขาวิชาวิศวกรรมคอมพิวเตอร์  
มหาวิทยาลัยเทคโนโลยีสุรนารี  
ปีการศึกษา 2559

**THE TIME SERIES FORECASTING WITH  
INTELLIGENT TECHNIQUES AND  
TIME SERIES ANALYSIS**



**Ronnachai Chuentawat**

**A Thesis Submitted in Partial Fulfillment of the Requirements for the  
Degree of Doctor of Philosophy in Computer Engineering**

**Suranaree University of Technology**

**Academic Year 2016**

## การพยากรณ์อนุกรมเวลาด้วยเทคนิคอัจฉริยะและการวิเคราะห์อนุกรมเวลา

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้บัณฑิตวิทยาลัยรับนี้เป็นส่วนหนึ่งของการศึกษา  
ตามหลักสูตรปริญญาคุุณบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

\_\_\_\_\_

(รศ. ดร. นิตยา เกิดประสพ)

ประธานกรรมการ

\_\_\_\_\_

(รศ. ดร. กิตติศักดิ์ เกิดประสพ)

กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

\_\_\_\_\_

(รศ. ดร. มนต์ทิพย์ภา อุฑารสกุล)

กรรมการ

\_\_\_\_\_

(ผศ. ดร. ประเมศวร์ ห่อแก้ว)

กรรมการ

\_\_\_\_\_

(ผศ. ดร. สายสุนีย์ จัปโจร)

กรรมการ

\_\_\_\_\_

(ศ. ดร. ชูกิจ ลิ้มปีจ่างค์)

รองอธิการบดีฝ่ายวิชาการและนวัตกรรม

\_\_\_\_\_

(รศ. ร.อ. ดร. กนต์ธร ชำนิประศาสน์)

คณบดีสำนักวิชาวิศวกรรมศาสตร์

รณชัย ชื่นขวัญ : การพยากรณ์อนุกรมเวลาด้วยเทคนิคอัจฉริยะและการวิเคราะห์อนุกรมเวลา  
(THE TIME SERIES FORECASTING WITH INTELLIGENT TECHNIQUES  
AND TIME SERIES ANALYSIS)

อาจารย์ที่ปรึกษา : รองศาสตราจารย์ ดร.กิตติศักดิ์ เกิดประสพ, 242 หน้า.

การพยากรณ์อนุกรมเวลา หมายถึง การพยากรณ์ค่าสังเกตที่สนใจล่วงหน้า โดยนำข้อมูลที่เกิดขึ้นในอดีต มาสร้างตัวแบบพยากรณ์ สำหรับพยากรณ์ข้อมูลในอนาคต ซึ่งการสร้างตัวแบบพยากรณ์ แบ่งได้เป็น 2 ประเภท ได้แก่ การสร้างตัวแบบด้วยวิธีการวิเคราะห์อนุกรมเวลาแบบดั้งเดิม ที่เป็นการวิเคราะห์เชิงปริมาณด้วยหลักสถิติ และการสร้างตัวแบบด้วยเทคนิคอัจฉริยะ ที่เป็นวิธีการด้านแมชชีนเลิร์นนิง ซึ่งการพยากรณ์อนุกรมเวลา เป็นการพยากรณ์เชิงปริมาณ ผลที่ได้จากการพยากรณ์ คือ ข้อมูลเชิงปริมาณที่คาดว่าจะเกิดขึ้นในอนาคต โดยความแม่นยำของการพยากรณ์ขึ้นอยู่กับ การเลือกตัวแบบที่เหมาะสมกับลักษณะของอนุกรมเวลาที่นำมาพยากรณ์ และการพยากรณ์ที่แม่นยำ ก่อให้เกิดประโยชน์ในการวางแผนการดำเนินการ เพื่อลดต้นทุนด้านเวลา ด้านทรัพยากร หรือบรรเทาความเสียหายที่อาจเกิดขึ้นในอนาคตได้ ดังนั้น งานวิจัยนี้ จึงนำเสนอ การพยากรณ์หน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง และการไฟฟ้าส่วนภูมิภาค เขต 3 ซึ่งเป็นอนุกรมเวลาที่มีคาบเวลาเป็นรายเดือน ด้วยวิธีการสร้างตัวแบบพยากรณ์แบบผสมผสาน (Hybrid model) ระหว่างวิธีการวิเคราะห์อนุกรมเวลาแบบดั้งเดิมและเทคนิคอัจฉริยะเข้าด้วยกัน โดยใช้รูปแบบออโตเรกเรสซีฟ (Autoregressive) ของตัวแบบ ARIMA ซึ่งเป็นตัวแบบพยากรณ์จากวิธีการวิเคราะห์อนุกรมเวลาแบบดั้งเดิม เป็นตัวกำหนดอินพุตเวกเตอร์ (Input vector) ให้กับชุดข้อมูลฝึกสอน (Training data set) สำหรับนำไปสร้างตัวแบบพยากรณ์ด้วยเทคนิคอัจฉริยะ จำนวน 2 เทคนิค ได้แก่ เทคนิคเครือข่ายประสาทเทียม (Artificial neural network) และเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชัน (Support vector regression) จากนั้นประเมินผลการทดลอง ด้วยการเปรียบเทียบความแม่นยำในการพยากรณ์ของตัวแบบผสมผสานที่สร้างจากเทคนิคอัจฉริยะทั้ง 2 เทคนิค กับตัวแบบพยากรณ์ที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาแบบดั้งเดิม ผลที่ได้จากงานวิจัยนี้ จะทำให้ทราบขั้นตอนในการพัฒนาตัวแบบพยากรณ์ และได้ตัวแบบพยากรณ์ที่มีความเหมาะสมที่สุด สำหรับพยากรณ์หน่วยจำหน่ายไฟฟ้า ซึ่งเป็นข้อมูลที่สะท้อนถึงปริมาณการใช้ไฟฟ้าในประเทศไทย และนำไปสู่การวางแผน พัฒนาการผลิตและจำหน่ายไฟฟ้า ให้มีความเหมาะสมต่อความต้องการของผู้บริโภคในประเทศไทยได้เป็นอย่างดี

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ปีการศึกษา 2559

ลายมือชื่อนักศึกษา \_\_\_\_\_

ลายมือชื่ออาจารย์ที่ปรึกษา \_\_\_\_\_

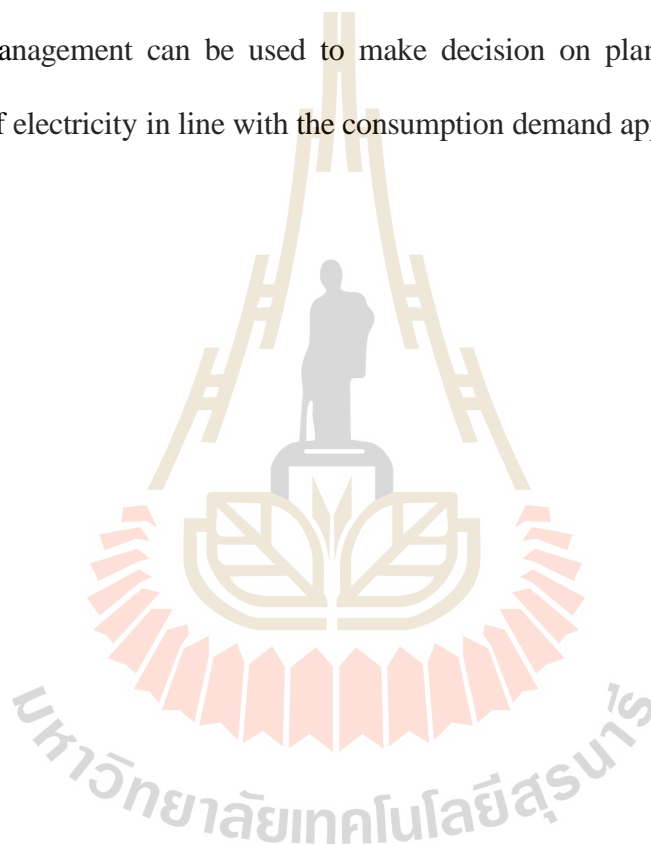


RONNACHAI CHUENTAWAT: THE TIME SERIES FORECASTING  
WITH INTELLIGENT TECHNIQUES AND TIME SERIES ANALYSIS.  
THESIS ADVISOR : ASSOC. PROF. KITTISAK KERDPRASOP, Ph.D.,  
242 PP.

TIME SERIES FORECASTING/AUTOREGRESSIVE INTEGRATED MOVING  
AVERAGE/ARTIFICIAL NEURAL NETWORK/SUPPORT VECTOR REGRESSION  
/HYBRID MODEL

The time series forecasting is a prediction of future observed value of the time series by using a historical data to create a forecast model and using this model forecasts the future values. There are two types of forecasting model, the first type creates by the traditional time series analysis method, and the second type creates by the intelligent technique. The intelligent technique is commonly known as the machine learning method. Time series forecasting is a quantitative prediction of numerical data and a result of such prediction will provide future quantitative values. The accuracy of a forecasting results depends on an appropriate selection of a forecast model. Selection must be based upon characteristic of the time series. Forecast results accuracy will assist management for better planning, decreases a cost of operation, and possible reduces management mistake. This research aims to forecast the electrical power distribution units (EPDU) by using historical data from the Metropolitan Electrical Authority (MEA) and the Provincial Electrical Authority (PEA) of Thailand. The data of EPDU are categorized as time series in monthly time period format. This data are being applied to generate a unique hybrid model that combines the autoregressive of the ARIMA model with the intelligent techniques. This research also uses two intelligence

techniques so called artificial neural network and support vector regression to create this hybrid forecast model. Also the autoregressive of the ARIMA model is applied for selecting input vector of the training data set, then evaluate the experiment by measuring the forecast error of the hybrid model compare with the traditional time series model. The result of this research reveals step by step for generating the forecast models and creates the forecast EPDU model which can predict the amount of electrical usage that management can be used to make decision on planning, generating, and distributing of electricity in line with the consumption demand appropriately.



School of Computer Engineering

Academic Year 2016

Student's Signature\_\_\_\_\_

Advisor's Signature\_\_\_\_\_

## กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณ บุคคล และกลุ่มบุคคลต่าง ๆ ที่ได้กรุณาให้คำปรึกษา แนะนำ ช่วยเหลืออย่างดียิ่ง ทั้งในด้านวิชาการ และด้านการดำเนินงานวิจัย ดังต่อไปนี้

รองศาสตราจารย์ ดร.กิตติศักดิ์ เกิดประสพ อาจารย์ที่ปรึกษาวิทยานิพนธ์ และรองศาสตราจารย์ ดร.นิตยา เกิดประสพ ที่ให้คำปรึกษาในการทำงานวิจัย การจัดรูปแบบ และช่วยตรวจทานความถูกต้องของวิทยานิพนธ์

คุณสายฝน สิบพลกรัง เลขานุการสาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่ให้ความช่วยเหลือในการประสานงานด้านเอกสารระหว่างศึกษา และนักศึกษาระดับบัณฑิตศึกษา สาขาวิชาวิศวกรรมคอมพิวเตอร์ทุกท่าน ที่ให้คำปรึกษา

มหาวิทยาลัยราชภัฏนครราชสีมา ที่ช่วยสนับสนุนทุนการศึกษา ทุนวิจัย และค่าใช้จ่ายในระหว่างศึกษา

นอกจากนี้ ขอขอบคุณครูอาจารย์ ทั้งในอดีตและปัจจุบัน ที่ให้ความรู้แก่ผู้วิจัย ทำให้ได้รับความสำเร็จในชีวิตการศึกษา

ท้ายที่สุด ขอกราบขอบพระคุณ บิดา มารดา ที่ให้กำเนิด อบรม เลี้ยงดู และส่งเสริมการศึกษาเป็นอย่างดี ทำให้ผู้วิจัยประสบความสำเร็จในชีวิตทุกด้าน

รณชัย ชื่นธวัช

# สารบัญ

หน้า

บทคัดย่อ (ภาษาไทย) .....	ก
บทคัดย่อ (ภาษาอังกฤษ) .....	ข
กิตติกรรมประกาศ .....	ง
สารบัญ .....	จ
สารบัญตาราง .....	ฉ
สารบัญรูป .....	ฉ
<b>บทที่</b>	
<b>1 บทนำ .....</b>	<b>1</b>
1.1 ความสำคัญและที่มาของปัญหาการวิจัย .....	1
1.2 วัตถุประสงค์ของการวิจัย .....	4
1.3 ขอบเขตของการวิจัย .....	5
1.4 ประโยชน์ที่จะได้รับ .....	5
<b>2 ปรัชญาวรรณกรรม .....</b>	<b>7</b>
2.1 หลักการวิเคราะห์อนุกรมเวลา .....	7
2.1.1 กระบวนการพยากรณ์ข้อมูล .....	8
2.1.2 ส่วนประกอบของอนุกรมเวลา .....	9
2.1.3 การแสดงข้อมูลอนุกรมเวลาด้วยการสร้างกราฟ .....	10
2.1.4 อนุกรมเวลาที่มีลักษณะนิ่ง .....	11
2.1.5 การปรับอนุกรมเวลาให้มีลักษณะนิ่ง .....	13
2.1.6 การกำหนดตัวแบบอนุกรมเวลาเชิงเดี่ยว .....	17
2.1.7 การวัดความคลาดเคลื่อนของการพยากรณ์ .....	19

## สารบัญ (ต่อ)

หน้า

2.2	วิธีการสร้างตัวแบบอนุกรมเวลาเชิงเดี่ยว .....	22
2.2.1	วิธีการวิเคราะห์ความถดถอย .....	22
2.2.2	วิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล .....	27
2.2.3	วิธีการของบอชซ์และเจนกินส์ .....	31
2.3	การพยากรณ์อนุกรมเวลาด้วยเครือข่ายประสาทเทียม .....	42
2.3.1	แนวคิดและหลักการทำงานของเครือข่ายประสาทเทียม .....	43
2.3.2	ตัวแบบของนิเวรอน .....	44
2.3.3	การเรียนรู้ของเครือข่ายเพอร์เซปตรอน .....	48
2.3.4	การเรียนรู้แบบแพร่กลับ .....	54
2.3.5	การพยากรณ์อนุกรมเวลาด้วยเครือข่ายประสาทเทียม .....	62
2.4	การพยากรณ์อนุกรมเวลาด้วยซัพพอร์ตเวกเตอร์แมชชีนและจินตริกอัลกอริทึม ....	64
2.4.1	ไฮเปอร์เพลนและมาร์จิน .....	65
2.4.2	ซัพพอร์ตเวกเตอร์แมชชีนกับกรณีที่เป็นการแบ่งแยกเชิงเส้น .....	68
2.4.3	ซัพพอร์ตเวกเตอร์แมชชีนแบบซอฟต์มาร์จิน .....	72
2.4.4	เคอร์เนลฟังก์ชันกับซัพพอร์ตเวกเตอร์แมชชีน .....	73
2.4.5	ตัวอย่างการจำแนกข้อมูลด้วยซัพพอร์ตเวกเตอร์แมชชีน .....	76
2.4.6	ซัพพอร์ตเวกเตอร์รีเกรสชันกับการพยากรณ์อนุกรมเวลา .....	79
2.4.7	จินตริกอัลกอริทึมกับการสร้างซัพพอร์ตเวกเตอร์รีเกรสชัน ที่เหมาะสมที่สุด .....	82
2.5	งานวิจัยที่เกี่ยวข้อง .....	85
3	วิธีดำเนินการวิจัย .....	94
3.1	กรอบแนวคิดของการวิจัย .....	94
3.1.1	ชุดข้อมูล .....	96

## สารบัญ (ต่อ)

หน้า

3.1.2	การสร้างตัวแบบการพยากรณ์ด้วยวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม .....	98
3.1.3	การสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ร่วมกับเทคนิคอัจฉริยะ .....	102
3.2	เครื่องมือที่ใช้ในการวิจัย .....	111
4	การทดสอบและอภิปรายผล .....	112
4.1	ผลการทดสอบของตัวแบบพยากรณ์จากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม .....	112
4.2	ผลการทดสอบของตัวแบบผสมผสาน .....	118
4.2.1	ผลการวิเคราะห์รูปแบบออโตรีเกรสซีฟจากตัวแบบ ARIMA .....	118
4.2.2	ผลการทดสอบสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟกับเทคนิคเครือข่ายประสาทเทียม .....	121
4.2.3	ผลการทดสอบสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟกับเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชันร่วมกับจินตคณิตอัลกอริทึม .....	127
4.2.4	ผลการทดสอบวัดความทนทานของตัวแบบผสมผสาน .....	132
4.3	อภิปรายผล .....	135
4.3.1	คุณลักษณะของข้อมูล .....	135
4.3.2	อภิปรายผลการพยากรณ์ของตัวแบบที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม .....	136
4.3.3	อภิปรายผลการพยากรณ์ของตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟกับเทคนิคเครือข่ายประสาทเทียม .....	137

## สารบัญ (ต่อ)

หน้า

4.3.4	อภิปรายเปรียบเทียบผลการพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR .....	139
4.3.5	อภิปรายผลการทดสอบวัดความทนทานของตัวแบบผสมผสาน .....	142
5	สรุปผลการวิจัยและข้อเสนอแนะ .....	144
5.1	สรุปผลการวิจัย .....	145
5.1.1	ชุดข้อมูล .....	146
5.1.2	สรุปผลการทดสอบตัวแบบพยากรณ์จากวิธีวิเคราะห์ อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม .....	146
5.1.3	สรุปผลการวิเคราะห์รูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA .....	147
5.1.4	สรุปผลการทดสอบตัวแบบผสมผสาน ANN .....	147
5.1.5	สรุปผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR .....	148
5.2	ปัญหาและข้อเสนอแนะ .....	149
	รายการอ้างอิง .....	150
	ภาคผนวก	
	ภาคผนวก ก. รหัสต้นฉบับของโปรแกรม .....	154
	ภาคผนวก ข. บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ .....	165
	ประวัติผู้เขียน .....	242

## สารบัญตาราง

ตารางที่	หน้า
2.1	รูปแบบการถดถอยและสมการถดถอย .....24
2.2	รูปแบบ AR(p, q) ของ $\{Z_t\}$ และรูปแบบ ARIMA(p, d, q) ของ $\{Y_t\}$ .....36
2.3	ลักษณะรูปร่างของกราฟ ACF และกราฟ PACF สำหรับกำหนดรูปแบบ ARIMA .....41
2.4	ค่าของอินพุตเวกเตอร์และทาร์เก็ตเวกเตอร์ สำหรับรอบการคำนวณที่ 1 และ 2 .....60
2.5	ค่าสัญญาณของเครือข่าย สำหรับรอบการทำงานที่ 1 .....60
2.6	เคอร์เนลฟังก์ชันที่นิยมใช้ร่วมกับซัพพอร์ตเวกเตอร์แมชชีน .....76
2.7	อินพุตเวกเตอร์และทาร์เก็ตของปัญหา XOR .....77
2.8	การเปรียบเทียบงานวิจัยที่เกี่ยวข้องกับการพยากรณ์อนุกรมเวลา และงานวิจัยของวิทยานิพนธ์ฉบับนี้ สำหรับประเด็นเปรียบเทียบด้านลักษณะของอนุกรมเวลาและเทคนิคที่ใช้ .....91
2.9	การเปรียบเทียบงานวิจัยที่เกี่ยวข้องกับการพยากรณ์อนุกรมเวลา และงานวิจัยของวิทยานิพนธ์ฉบับนี้ สำหรับประเด็นเปรียบเทียบด้านมาตรวัดความคลาดเคลื่อนและตัวแบบที่เหมาะสมที่สุด .....92
4.1	ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบพยากรณ์จากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม สำหรับชุดข้อมูลที่ 1 ..... 115
4.2	ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบพยากรณ์จากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม สำหรับชุดข้อมูลที่ 2 ..... 115
4.3	ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 1 .....123
4.4	ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 2 .....126
4.5	ค่าพยากรณ์และค่าคลาดเคลื่อนของตัวแบบผสมผสาน GASVR ของชุดข้อมูลทั้ง 2 ชุด...131
4.6	ผลการทดสอบความทนทานของค่าพยากรณ์ สำหรับชุดข้อมูลที่ 1 ..... 133
4.7	ผลการทดสอบความทนทานของค่าพยากรณ์ สำหรับชุดข้อมูลที่ 2 ..... 134



## สารบัญตาราง (ต่อ)

ตารางที่	หน้า
4.8 ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR สำหรับชุดข้อมูลที่ 1 .....	139
4.9 ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR สำหรับชุดข้อมูลที่ 2 .....	140



## สารบัญรูป

รูปที่	หน้า
2.1	กระบวนการพยากรณ์ .....8
2.2	กราฟของจำนวนผู้โดยสารของสนามบินแห่งหนึ่ง (หน่วย 1000 คน) ตั้งแต่ปีค.ศ. 1949-1960 .....11
2.3	กราฟของจำนวนผู้โดยสาร เมื่อกำจัดแนวโน้มด้วยการล่าหลัง 1 ช่วงเวลา ( $k=1$ ) .....14
2.4	กราฟของจำนวนผู้โดยสาร เมื่อแปลงข้อมูลด้วยการหาค่าลอการิทึมฐาน $e$ .....15
2.5	กราฟของจำนวนผู้โดยสาร เมื่อกำจัดฤดูกาลด้วยการล่าหลัง 12 ช่วงเวลา ( $k=12$ ) .....16
2.6	กราฟของจำนวนผู้โดยสาร เมื่อทำให้ความแปรปรวนคงที่ กำจัดแนวโน้ม และกำจัดฤดูกาล .....17
2.7	เซลล์ประสาท .....44
2.8	ตัวแบบของนิวรอน .....45
2.9	ตัวแบบของนิวรอนเมื่อรวมไบแอสไว้ในน้ำหนักซินแนปส์ .....46
2.10	ประเภทของแอกติเวชันฟังก์ชัน .....48
2.11	เพอร์เซปตรอนแบบชั้นเดียว .....49
2.12	กราฟสัญญาณการไหลของเพอร์เซปตรอน .....50
2.13	เส้นแบ่งเขตการตัดสินใจเชิงเส้น 2 มิติ .....51
2.14	โครงสร้างของเครือข่ายเพอร์เซปตรอนเพื่อหาฟังก์ชันแทนตัวปฏิบัติการ OR .....53
2.15	เครือข่ายประสาทเทียมที่มีโครงสร้างไปข้างหน้าแบบหลายชั้น .....55
2.16	เครือข่ายประสาทเทียมที่มีโครงสร้างไปข้างหน้าแบบหลายชั้นและแพร์กลับ .....56
2.17	เครือข่ายไปข้างหน้าแบบ 3 ชั้น ด้วยการเรียนรู้แบบแพร์กลับ .....56
2.18	อัลกอริทึมแบบแพร์กลับ .....57
2.19	ตัวอย่างเครือข่ายไปข้างหน้า 3 ชั้น แบบแพร์กลับ .....59

## สารบัญรูป (ต่อ)

รูปที่	หน้า
2.20	โครงสร้างของตัวอย่างเครือข่ายหลังจากผ่านการปรับน้ำหนักซินแนปส์ในรอบแรก .....62
2.21	โครงสร้างของเครือข่ายประสาทเทียมสำหรับการพยากรณ์อนุกรมเวลาเชิงเดี่ยว .....63
2.22	ไฮเปอร์เพลนเชิงเส้น 2 มิติ .....66
2.23	มาร์จิ้นของไฮเปอร์เพลน .....68
2.24	ซอฟต์แวร์มาร์จิ้นไฮเปอร์เพลน .....72
2.25	การแปลงจุดข้อมูลจากอินพุตสเปซไปยังพีเจอร์สเปซ .....74
2.26	การจำแนกข้อมูลไม่เป็นเชิงเส้นของ SVM .....75
2.27	ปัญหาการจำแนกข้อมูล XOR .....76
2.28	การสร้างท่อเปิดชนิดอนแบบมีตัวแปรสเกล รอบสมการความถดถอยเชิงเส้นของ SVR ....80
2.29	วิธีการคัดเลือกโครโมโซมแบบวงล้อสุ่ม .....83
3.1	กรอบแนวคิดของการวิจัย .....96
3.2	ตัวอย่างหน่วยจำหน่ายไฟฟ้าประจำเดือนธันวาคม 2554 ของข้อมูลชุดที่ 1 .....97
3.3	หน่วยจำหน่ายไฟฟ้า ตั้งแต่เดือนมกราคม 2553 ถึงเดือนพฤษภาคม 2558 ของข้อมูลชุดที่ 2 .....98
3.4	ขั้นตอนการสร้างตัวแบบการพยากรณ์ด้วยวิธีการวิเคราะห์อนุกรมเวลาแบบดั้งเดิม .....99
3.5	ข้อมูลอนุกรมเวลา .....100
3.6	ชุดคำสั่งของภาษาอาร์ สำหรับสร้างตัวแบบ SLR ตัวแบบ Holt-Winter และตัวแบบ ARIMA .....102
3.7	ตัวอย่างชุดข้อมูลฝึกสอนที่ได้จากการวิเคราะห์รูปแบบออโตรีเกรสซีฟ ของตัวแบบ ARIMA .....104
3.8	ขั้นตอนการสร้างตัวแบบ Hybrid ANN ที่มีโครงสร้างเหมาะสมที่สุด .....105
3.9	โครงสร้างของตัวแบบ Hybrid ANN ที่ใช้ดำเนินการทดลอง .....105
3.10	ขั้นตอนการสร้างตัวแบบ Hybrid GASVR .....108

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.11	โครงสร้างของโครโมโซม ..... 109
4.1	คำสั่งและค่าพยากรณ์ข้อมูลของตัวแบบ SLR ..... 113
4.2	คำสั่งและค่าพยากรณ์ข้อมูลของตัวแบบ Holt-Winters ..... 113
4.3	คำสั่งและค่าพยากรณ์ข้อมูลของตัวแบบ ARIMA ..... 114
4.4	ชุดคำสั่งคำนวณค่าคลาดเคลื่อนของตัวแบบ SLR, Holt-Winters และ ARIMA สำหรับชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง ..... 114
4.5	กราฟคำสั่งเกณฑ์ที่ใช้สร้างตัวแบบพยากรณ์ของวิธีวิเคราะห์อนุกรมเวลาแบบดั้งเดิม ..... 116
4.6	กราฟเปรียบเทียบค่าพยากรณ์ของตัวแบบที่สร้างจากวิธีวิเคราะห์ อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม และคำสั่งเกณฑ์จริงของชุดข้อมูลที่ 1 ..... 116
4.7	กราฟเปรียบเทียบค่าพยากรณ์ของตัวแบบที่สร้างจากวิธีวิเคราะห์ อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม และคำสั่งเกณฑ์จริงของชุดข้อมูลที่ 2 ..... 117
4.8	ผลการทดสอบลักษณะหนึ่งของอนุกรมเวลาทั้ง 2 ชุด ..... 117
4.9	ผลการวิเคราะห์รูปแบบที่เหมาะสมที่สุดของตัวแบบ ARIMA ของอนุกรมเวลาทั้ง 2 ชุด ..... 118
4.10	ตัวอย่างชุดข้อมูลที่ฝึกสอนของอนุกรมเวลาชุดที่ 1 ..... 119
4.11	ตัวอย่างชุดข้อมูลที่ฝึกสอนของอนุกรมเวลาชุดที่ 2 ..... 120
4.12	โครงสร้างของตัวแบบผสมผสาน ANN ที่ใช้ดำเนินการทดสอบ ของชุดข้อมูลที่ 1 ..... 121
4.13	ชุดคำสั่งสร้างตัวแบบผสมผสาน ANN2 ของชุดข้อมูลที่ 1 ..... 122
4.14	ตัวอย่างข้อมูลที่ใช้ในชุดคำสั่งสร้างตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 1 ..... 123
4.15	กราฟค่าพยากรณ์ของตัวแบบผสมผสาน ANN และคำสั่งเกณฑ์จริงของชุดข้อมูลที่ 1 ..... 124
4.16	โครงสร้างของตัวแบบผสมผสาน ANN ที่ใช้ดำเนินการทดสอบของชุดข้อมูลที่ 2 ..... 125
4.17	ตัวอย่างข้อมูลที่ใช้ในชุดคำสั่งสร้างตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 2 ..... 125

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.18 กราฟค่าพยากรณ์ของตัวแบบผสมผสาน ANN และค่าสังเกตจริงของชุดข้อมูลที่ 2 .....	126
4.19 ชุดคำสั่งค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินตคณิตอรรถิธิม ของข้อมูลฝึกสอนชุดที่ 1 .....	127
4.20 ชุดคำสั่งค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินตคณิตอรรถิธิม ของข้อมูลฝึกสอนชุดที่ 2 .....	128
4.21 ผลการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินตคณิตอรรถิธิม ของข้อมูลฝึกสอนชุดที่ 1 .....	128
4.22 ผลการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินตคณิตอรรถิธิม ของข้อมูลฝึกสอนชุดที่ 2 .....	129
4.23 ชุดคำสั่งสร้างตัวแบบผสมผสาน GASVR ของข้อมูลฝึกสอนชุดที่ 1 .....	129
4.24 ชุดคำสั่งสร้างตัวแบบผสมผสาน GASVR ของข้อมูลฝึกสอนชุดที่ 2 .....	130
4.25 กราฟค่าพยากรณ์ของตัวแบบผสมผสาน GASVR และค่าสังเกตจริง ของข้อมูลทั้ง 2 ชุด .....	131
4.26 โครงสร้างของตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด สำหรับชุดข้อมูลที่ 1 .....	138
4.27 โครงสร้างของตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด สำหรับชุดข้อมูลที่ 2 .....	138
4.28 กราฟเปรียบเทียบค่าสังเกตจริงและค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR สำหรับชุดข้อมูลที่ 1 .....	140
4.29 กราฟเปรียบเทียบค่าสังเกตจริงและค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR สำหรับชุดข้อมูลที่ 2 .....	141

# บทที่ 1

## บทนำ

### 1.1 ความสำคัญและที่มาของปัญหาการวิจัย

งานวิจัยนี้ เป็นการศึกษาการพยากรณ์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า ด้วยเทคนิค อัจฉริยะและการวิเคราะห์อนุกรมเวลา ซึ่งการพยากรณ์อนุกรมเวลา เป็นการพยากรณ์เชิงปริมาณ ที่หมายถึง การใช้ข้อมูลเชิงปริมาณ ที่อยู่ในรูปแบบตัวเลข มีการเก็บรวบรวมอย่างเป็นระบบ และ ต่อเนื่องในอดีต มาวิเคราะห์ด้วยวิธีการเชิงสถิติ เพื่อพยากรณ์ข้อมูลที่จะเกิดขึ้นในอนาคต ภายใต้ สมมุติฐานว่า ข้อมูลในอนาคตมีรูปแบบการเคลื่อนไหว หรือการเปลี่ยนแปลงที่คล้ายกับข้อมูลในอดีต ซึ่งข้อดีของการพยากรณ์เชิงปริมาณ เมื่อเปรียบเทียบกับพยากรณ์เชิงคุณภาพ มีหลายประการ เช่น มีค่าใช้จ่ายในการดำเนินการที่ต่ำกว่า มีความถูกต้องสูงกว่า เมื่อข้อมูลที่นำมาใช้ในการพยากรณ์มี ความโน้มเอียงต่ำ เนื่องจากพยากรณ์ข้อมูลด้วยวิธีการเชิงสถิติ ที่มีการวัดค่าความถูกต้องอย่างชัดเจน

อนุกรมเวลา (Time series) หมายถึง กลุ่มของข้อมูลเชิงสถิติ ที่มีการเก็บรวบรวมอย่างเป็น ระบบ และต่อเนื่อง โดยมีการแบ่งช่วงเวลากลับกันที่ข้อมูล ที่มีช่วงห่างของเวลาเท่ากันอย่าง สม่ำเสมอและเป็นปกติ เช่น รายชั่วโมง รายวัน รายสัปดาห์ รายเดือน รายไตรมาส หรือรายปี ซึ่งอนุกรมเวลาเป็นลำดับของค่าสังเกตของตัวแปรที่สนใจ ถ้าอนุกรมเวลามีตัวแปรที่สนใจ 1 ตัว เรียกว่า อนุกรมเวลาเชิงเดี่ยว (Univariate time series) แต่ถ้าอนุกรมเวลามีตัวแปรที่สนใจมากกว่า 1 ตัว เรียกว่า อนุกรมเวลาเชิงพหุ (Multivariate time series) โดยค่าสังเกตของตัวแปรมีค่าเป็น ตัวเลข และมีการวัดค่า หรือบันทึกค่าตามคาบเวลาที่แบ่งไว้อย่างต่อเนื่อง ตามลำดับของคาบเวลา และรูปแบบของข้อมูลอนุกรมเวลา จะถูกนำมาศึกษา หรือทำความเข้าใจว่า อนุกรมเวลามีรูปแบบ การเคลื่อนไหว หรือการเปลี่ยนแปลงเป็นอย่างไร เพื่อทำการค้นหาวิธีการพยากรณ์ที่เหมาะสมกับ อนุกรมเวลานั้น ๆ ซึ่งตัวอย่างของข้อมูลอนุกรมเวลาที่เกิดขึ้นอย่างเป็นธรรมชาติ ปรากฏใน หลากหลายลักษณะ เช่น ข้อมูลการใช้ไฟฟ้า ข้อมูลราคาน้ำมัน ข้อมูลราคาทองคำ ข้อมูลสภาพ อากาศ ข้อมูลน้ำฝน น้ำท่า หรือข้อมูลสารเจือปนที่เป็นมลพิษทางอากาศ เป็นต้น

การวิเคราะห์อนุกรมเวลา (Time series analysis) เป็นการศึกษารูปแบบการเคลื่อนไหวของอนุกรมเวลา ภายใต้สมมติฐานว่า รูปแบบการเคลื่อนไหวของอนุกรมเวลาในอนาคต ไม่แตกต่างจากรูปแบบการเคลื่อนไหวในอดีต โดยมีวัตถุประสงค์ เพื่อศึกษารูปแบบการเคลื่อนไหวของข้อมูล ด้วยการพิจารณาจากลำดับของข้อมูลที่บันทึกตามคาบเวลาในอดีต และเพื่อพยากรณ์ค่าข้อมูลในอนาคตด้วยตัวแบบพยากรณ์ หรือสมการคณิตศาสตร์ที่เหมาะสมกับข้อมูลอนุกรมเวลา โดยตัวแบบที่ใช้ในการพยากรณ์ ได้จากการศึกษารูปแบบการเคลื่อนไหวของข้อมูลอนุกรมเวลาดังนั้น วิธีการวิเคราะห์อนุกรมเวลา คือ การสร้างตัวแบบพยากรณ์ที่เหมาะสม จากข้อมูลอนุกรมเวลาที่มีการเก็บบันทึกไว้ในอดีต จากนั้น นำตัวแบบพยากรณ์ มาใช้พยากรณ์ข้อมูลในอนาคต ซึ่งมีประโยชน์ต่อการวางแผน การจัดการ และควบคุมการดำเนินการในด้านต่าง ๆ ได้เป็นอย่างดี โดยตัวอย่างของตัวแบบพยากรณ์ที่สร้างจากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยว ได้แก่ ตัวแบบ SLR (Simple linear regression) ที่สร้างจากวิธีการวิเคราะห์ความถดถอยอย่างง่าย ตัวแบบ Holt-Winters ที่สร้างจากวิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล หรือตัวแบบ ARIMA (Autoregressive integrated moving average) ที่สร้างจากวิธีการของบ็อกซ์และเจนกินส์ (Box et al., 2008)

สำหรับเทคนิคอัจฉริยะที่นำมาใช้กับการวิเคราะห์อนุกรมเวลา เพื่อสร้างตัวแบบการพยากรณ์อนุกรมเวลา เป็นวิธีการด้านแมชชีนเลิร์นนิง (Machine learning) ซึ่งเกี่ยวข้องกับการศึกษาและการสร้างขั้นตอนวิธี ที่ทำให้คอมพิวเตอร์สามารถเรียนรู้จากข้อมูลในอดีตและทำนายข้อมูลในอนาคตได้ โดยขั้นตอนวิธีของแมชชีนเลิร์นนิง จะดำเนินการสร้างตัวแบบจากชุดข้อมูลตัวอย่างกลุ่มหนึ่ง เรียกว่า ชุดข้อมูลฝึกสอน (Training data set) ที่ประกอบด้วยข้อมูลอินพุต (Input) และข้อมูลเอาต์พุต (Output) ที่เรียกว่า ทาร์เก็ต (Target) และเมื่อได้ตัวแบบการพยากรณ์ จะใช้ตัวแบบเพื่อทำนายข้อมูลเอาต์พุตที่ยังไม่ทราบค่า จากข้อมูลอินพุตที่ทราบค่าได้ ซึ่งเทคนิคอัจฉริยะที่สำคัญสำหรับการสร้างตัวแบบการพยากรณ์ ได้แก่ เครือข่ายประสาทเทียม (Artificial neural network) เป็นเทคนิคที่มีแนวคิดจากวิธีการทำงานของเครือข่ายประสาทในสมองของมนุษย์ โดยมีวัตถุประสงค์พื้นฐาน คือ การสร้างตัวแบบทางคณิตศาสตร์ ที่จำลองการทำงานของเครือข่ายประสาทในสมองมนุษย์ให้กับเครื่องจักร เพื่อให้เครื่องจักรมีความสามารถในการเรียนรู้การจดจำรูปแบบ (Pattern recognition) และการสกัดความรู้ใหม่ (Knowledge extraction) เช่นเดียวกับความสามารถที่มีในสมองมนุษย์ และเทคนิคอัจฉริยะอีกเทคนิคหนึ่ง คือ ซัพพอร์ตเวกเตอร์แมชชีน (Support vector machine) ซึ่งในกรณีที่ข้อมูลเป็นตัวเลขทั้งหมดจะเรียกว่า ซัพพอร์ตเวกเตอร์รีเกรสชัน (Support vector regression) โดยมีแนวคิด คือ การสร้างเส้นไฮเปอร์เพลนที่ดีที่สุด (Optimal



hyperplane) เพื่อใช้จำแนกข้อมูล ซึ่งเส้นไฮเปอร์เพลนที่ดีที่สุด คือ เส้นที่มีมาร์จิน (Margin) ทั้ง 2 ด้านกว้างมากที่สุด นอกจากนี้ ยังมีขั้นตอนวิธีที่ใช้ช่วยค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดของการสร้างตัวแบบด้วยเทคนิคซัพพอร์ตเวกเตอร์แมชชีน นั่นคือ จีเนติกอัลกอริทึม (Genetic algorithm) ซึ่งเป็นขั้นตอนวิธีการค้นหาค่าตอบที่มีแนวคิดจากทฤษฎีวิวัฒนาการของสิ่งมีชีวิต

ข้อมูลที่นำมาใช้ศึกษาในงานวิจัยนี้ มีลักษณะเป็นอนุกรมเวลาเชิงเดี่ยว จำนวน 2 ชุด ข้อมูลชุดที่ 1 มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้า (The Electrical Power Distribution Units) ของการไฟฟ้านครหลวงแห่งประเทศไทย (Metropolitan Electricity Authority of Thailand) และข้อมูลชุดที่ 2 มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้าของการไฟฟ้าส่วนภูมิภาคแห่งประเทศไทย (Provincial Electricity Authority of Thailand) เขต 3 ซึ่งครอบคลุมพื้นที่จำหน่ายไฟฟ้า 4 จังหวัด ได้แก่ นครราชสีมา ชัยภูมิ บุรีรัมย์ และสุรินทร์ โดยหน่วยจำหน่ายไฟฟ้า 1 หน่วย หมายถึง ปริมาณการใช้ไฟฟ้า 1 กิโลวัตต์ต่อชั่วโมง (kW/h) ดังนั้น หน่วยจำหน่ายไฟฟ้าจึงเป็นข้อมูลที่สะท้อนถึงปริมาณการใช้ไฟฟ้าของผู้บริโภค ซึ่งมีลักษณะเป็นอนุกรมเวลาเชิงเดี่ยว ที่บันทึกข้อมูลเป็นรายเดือน ตามลำดับเวลา โดยความสำคัญของการพยากรณ์ปริมาณการใช้ไฟฟ้าของผู้บริโภคอย่างแม่นยำ จะสามารถช่วยวางแผนด้านทรัพยากรจัดสรรงบประมาณ และลดค่าใช้จ่ายในการดำเนินการลงได้ ซึ่งสอดคล้องกับงานวิจัยในอดีต ที่กล่าวถึงความสำคัญของการพยากรณ์ปริมาณการใช้ไฟฟ้า เช่น Bunn and Farmer (1985) ศึกษาการพยากรณ์ความต้องการไฟฟ้า พบว่า ค่าใช้จ่ายการดำเนินการจะเพิ่มขึ้น 10 ล้านปอนด์ ทุกปี เมื่อมีการพยากรณ์ผิดพลาดไปทุก 1% หรืองานวิจัยในอดีตที่กล่าวว่าการพยากรณ์ปริมาณการใช้ไฟฟ้ามากกว่าค่าจริง อาจทำให้ผลิตพลังงานไฟฟ้ามากกว่าความต้องการจริง เป็นผลให้สูญเสียค่าใช้จ่ายโดยไม่จำเป็น แต่ถ้าพยากรณ์ปริมาณการใช้ไฟฟ้าน้อยกว่าค่าจริง อาจทำให้ผลิตพลังงานไฟฟ้าไม่เพียงพอต่อความต้องการ เป็นผลให้เกิดความสูญเสียทางเศรษฐกิจ การดำเนินธุรกิจ สูญเสียเวลาและภาพลักษณ์ (Hong, 2009; Ismail et al., 2009; Kaytez et al., 2015)

จากการศึกษาแหล่งข้อมูลที่นำมาใช้ในงานวิจัยนี้ ทั้ง 2 แห่ง ได้แก่ รายงานสถานการณ์การใช้ไฟฟ้าของการไฟฟ้านครหลวง และการสัมภาษณ์หัวหน้าแผนกข้อมูลสถิติและวิเคราะห์การไฟฟ้าส่วนภูมิภาคเขต 3 ภาค 2 นครราชสีมา พบว่า ไม่มีการพยากรณ์หน่วยจำหน่ายไฟฟ้า เพื่อนำมาช่วยวางแผนการผลิตและจำหน่ายไฟฟ้า ให้สอดคล้องกับความต้องการไฟฟ้าของผู้บริโภค ดังนั้น จากความสำคัญและปัญหาที่ได้กล่าวมา จึงเป็นที่มาของงานวิจัยนี้ ที่ศึกษาการพยากรณ์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้าจำนวน 2 ชุดข้อมูล ด้วยเทคนิคอัจฉริยะและการวิเคราะห์อนุกรมเวลา เพื่อนำเสนอเทคนิคการพยากรณ์ที่ให้ความแม่นยำมากที่สุด สำหรับการพยากรณ์หน่วย



จำหน่ายไฟฟ้าในประเทศไทย ซึ่งจะช่วยให้หน่วยงานที่เกี่ยวข้อง สามารถนำเทคนิคการพยากรณ์ที่นำเสนอในงานวิจัยนี้ ไปใช้พยากรณ์หน่วยจำหน่ายไฟฟ้าล่วงหน้า เพื่อวางแผนการผลิตและจำหน่ายไฟฟ้า ได้อย่างมีประสิทธิภาพ โดยงานวิจัยนี้ จะนำเสนอเทคนิคใหม่ สำหรับการสร้างตัวแบบพยากรณ์อนุกรมเวลาเชิงเดี่ยว เรียกว่า ตัวแบบผสมผสาน (Hybrid model) ซึ่งเป็นการผสมผสานการวิเคราะห์รูปแบบออโตรีเกรสซีฟ (Autoregressive) ของตัวแบบ ARIMA เพื่อนำมา กำหนดอินพุตของชุดข้อมูลฝึกสอน ที่ใช้ในการสร้างตัวแบบพยากรณ์จากเทคนิคอัจฉริยะ ภายใต้สมมุติฐานว่า ถ้ากำหนดอินพุตของตัวแบบพยากรณ์ที่มีความสัมพันธ์กับเอาต์พุต จะทำให้ได้ตัวแบบพยากรณ์ที่สามารถพยากรณ์ค่าเอาต์พุตจากค่าอินพุตได้อย่างแม่นยำ นอกจากนี้ ตัวแบบผสมผสานดังกล่าว เป็นการผสมผสานรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ซึ่งค่าพยากรณ์มีความสัมพันธ์แบบเชิงเส้นกับค่าสังเกตในอดีต จึงสามารถตรวจจบบรูปแบบเชิงเส้นได้ดี (Wang and Meng, 2012) ร่วมกับเทคนิคอัจฉริยะที่สามารถตรวจจบบรูปแบบไม่เป็นเชิงเส้นได้ดี (Wang and Meng, 2012; Zhang et al., 2016) ดังนั้น จึงคาดหวังได้ว่า ตัวแบบผสมผสาน จะสามารถตรวจจบบรูปแบบเชิงเส้นและรูปแบบไม่เป็นเชิงเส้นที่แฝงอยู่ในอนุกรมเวลาได้ทั้ง 2 รูปแบบ ทำให้ค่าคลาดเคลื่อนต่ำกว่าตัวแบบพยากรณ์ที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลา ที่ตรวจจบบรูปแบบเชิงเส้นได้ดี แต่ตรวจจบบรูปแบบไม่เป็นเชิงเส้นได้ไม่ดี โดยประเมินผลการทดสอบด้วยการเปรียบเทียบความแม่นยำของตัวแบบผสมผสานกับตัวแบบที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลา เพื่อพิสูจน์ว่า วิธีการสร้างตัวแบบผสมผสานที่งานวิจัยนี้ได้นำเสนอ เป็นวิธีการใหม่ ในการสร้างตัวแบบพยากรณ์ นอกเหนือจากวิธีการสร้างตัวแบบพยากรณ์ด้วยวิธีการวิเคราะห์อนุกรมเวลา ซึ่งเป็นวิธีการทางสถิติแบบดั้งเดิม และสามารถนำตัวแบบผสมผสาน มาใช้พยากรณ์อนุกรมเวลาที่มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้าในประเทศไทย ได้อย่างเหมาะสม

## 1.2 วัตถุประสงค์ของการวิจัย

1. เพื่อศึกษาและพัฒนาตัวแบบผสมผสาน (Hybrid model) ระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ร่วมกับเทคนิคอัจฉริยะ สำหรับพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าของประเทศไทย จำนวน 2 ชุดข้อมูล
2. เพื่อเปรียบเทียบความแม่นยำในการพยากรณ์ของตัวแบบผสมผสาน กับตัวแบบที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยว

### 1.3 ขอบเขตของการวิจัย

1. ข้อมูลหน่วยจำหน่ายไฟฟ้า ประกอบด้วยข้อมูล 2 ชุด เท่านั้น ได้แก่ ชุดที่ 1 คือ ข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวงแห่งประเทศไทย ที่จำหน่ายให้ผู้บริโภคประเภทบ้านพักอาศัย เท่านั้น โดยเป็นข้อมูลรายเดือน ตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 และชุดที่ 2 คือ ข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้าส่วนภูมิภาคแห่งประเทศไทย เขต 3 โดยเป็นข้อมูลหน่วยจำหน่ายไฟฟ้ารวมทุกประเภท ยกเว้นหน่วยจำหน่ายไฟฟ้าที่เกิดจากการใช้ไฟฟ้าสาธารณะ ซึ่งเป็นข้อมูลรายเดือน ตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558

2. เทคนิคอัจฉริยะที่นำมาใช้สร้างตัวแบบผสมผสาน มีจำนวน 2 เทคนิค ได้แก่ เทคนิคเครือข่ายประสาทเทียม (Artificial neural network) และเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชันร่วมกับการกำหนดค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินเนติกอัลกอริทึม (Support vector regression with optimal parameters by genetic algorithm)

3. การศึกษาและพัฒนาตัวแบบพยากรณ์หน่วยจำหน่ายไฟฟ้าด้วยเทคนิคการวิเคราะห์อนุกรมเวลาเชิงเดี่ยว เพื่อนำไปเปรียบเทียบความแม่นยำกับตัวแบบผสมผสาน มีจำนวน 3 เทคนิค ได้แก่ เทคนิคการวิเคราะห์ความถดถอยเชิงเส้นอย่างง่าย (Simple linear regression) เทคนิคการปรับให้เรียบแบบเอกซ์โพเนนเชียลของ โฮลท์และวินเตอร์ (Holt-Winters exponential smoothing) และเทคนิคการสร้างตัวแบบ ARIMA

4. การเปรียบเทียบความแม่นยำของตัวแบบพยากรณ์ ใช้การวัดค่าคลาดเคลื่อนด้วยมาตรวัดความคลาดเคลื่อน 4 มาตรวัด ได้แก่ SSE (Sum square error), MAE (Mean absolute error), RMSE (Root mean square error) และ MAPE (Mean absolute percentage error) เนื่องจากเป็นมาตรวัดมาตรฐานที่นิยมใช้ในการเปรียบเทียบความแม่นยำของตัวแบบพยากรณ์

### 1.4 ประโยชน์ที่จะได้รับ

1. ศึกษาวิธีการพัฒนาตัวแบบพยากรณ์หน่วยจำหน่ายไฟฟ้าด้วยเทคนิคอัจฉริยะ ได้แก่ เทคนิคเครือข่ายประสาทเทียม และเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชันร่วมกับการกำหนดค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินเนติกอัลกอริทึม

2. ศึกษาวิธีการพัฒนาตัวแบบการพยากรณ์หน่วยจำหน่ายไฟฟ้าด้วยเทคนิคการวิเคราะห์อนุกรมเวลาแบบดั้งเดิม ได้แก่ เทคนิคการวิเคราะห์ความถดถอยเชิงเส้นอย่างง่าย เทคนิคการปรับให้เรียบแบบเอกซ์โพเนนเชียลของโฮลท์และวินเตอร์ และเทคนิคการสร้างตัวแบบ ARIMA
3. สามารถพัฒนาตัวแบบพยากรณ์ด้วยเทคนิคใหม่ เรียกว่า ตัวแบบผสมผสาน ซึ่งเป็นตัวแบบพยากรณ์ที่ผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟ ของตัวแบบ ARIMA ร่วมกับเทคนิคอัจฉริยะ สำหรับพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าของประเทศไทย
4. ได้ตัวแบบการพยากรณ์หน่วยจำหน่ายไฟฟ้า ที่แม่นยำมากที่สุด ซึ่งสามารถนำไปใช้พยากรณ์หน่วยจำหน่ายไฟฟ้า ในเขตพื้นที่ของการไฟฟ้านครหลวง และการไฟฟ้าส่วนภูมิภาค เขต 3 โดยค่าพยากรณ์ที่ได้จากตัวแบบการพยากรณ์ สามารถนำไปใช้วางแผนดำเนินการผลิตพลังงานไฟฟ้าให้เหมาะสมต่อความต้องการที่แท้จริงของผู้บริโภค
5. สามารถนำวิธีการพัฒนาตัวแบบการพยากรณ์หน่วยจำหน่ายไฟฟ้าในงานวิจัยนี้ ไปใช้พยากรณ์ความต้องการไฟฟ้าของผู้บริโภค ให้กับหน่วยงานอื่น ที่รับผิดชอบการผลิตพลังงานไฟฟ้าเพื่อจำหน่ายให้กับผู้บริโภคในประเทศไทย เช่น การไฟฟ้าฝ่ายผลิตแห่งประเทศไทย หรือการไฟฟ้าส่วนภูมิภาค ในเขตต่าง ๆ ได้อย่างแม่นยำ เนื่องจาก ข้อมูลมีคุณลักษณะคล้ายคลึงกับข้อมูลที่นำมาใช้ในงานวิจัยนี้

## บทที่ 2

### ปริทัศน์วรรณกรรม

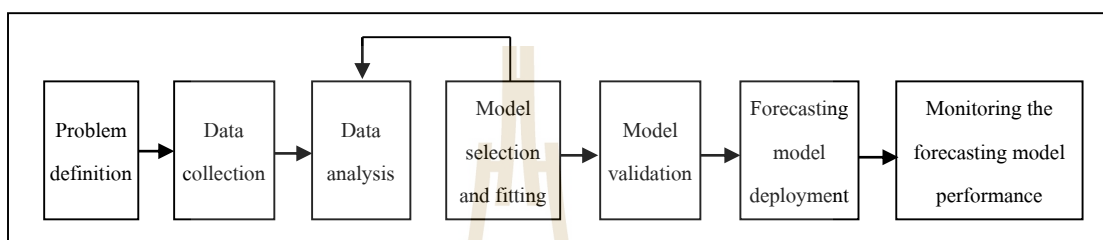
เนื้อหาในบทนี้ ประกอบด้วย การทบทวนวรรณกรรมและงานวิจัยที่เกี่ยวข้อง โดยมีวัตถุประสงค์ของงานวิจัย คือ การศึกษาและพัฒนาตัวแบบผสมผสาน (Hybrid model) ซึ่งเป็นตัวแบบพยากรณ์ที่ผสมผสานระหว่างรูปแบบของโครงข่ายประสาทเทียมของตัวแบบ ARIMA ร่วมกับเทคนิคอัจฉริยะ จากนั้น เปรียบเทียบความแม่นยำของตัวแบบผสมผสาน กับตัวแบบพยากรณ์ที่สร้างด้วยวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ดังนั้น การทบทวนวรรณกรรม ประกอบด้วยรายละเอียด 5 หัวข้อ ได้แก่ หลักการวิเคราะห์อนุกรมเวลา วิธีการสร้างตัวแบบการพยากรณ์อนุกรมเวลาเชิงเดี่ยว การพยากรณ์อนุกรมเวลาด้วยเครือข่ายประสาทเทียม การพยากรณ์อนุกรมเวลาด้วยซัพพอร์ตเวกเตอร์แมชชีนร่วมกับจินตคณิต และงานวิจัยที่เกี่ยวข้อง

#### 2.1 หลักการวิเคราะห์อนุกรมเวลา

อนุกรมเวลา (Time series) หมายถึง ลำดับของค่าสังเกตของตัวแปรที่สนใจ โดยมีการเก็บบันทึกค่าสังเกตตามลำดับเวลา ซึ่งค่าสังเกตที่อยู่ติดกันตามลำดับเวลา จะมีความสัมพันธ์กัน ทำให้ไม่สามารถสลับลำดับของข้อมูลได้ เพราะจะทำให้ความสัมพันธ์เปลี่ยนไป ตัวอย่างของชุดข้อมูลที่มีลักษณะเป็นอนุกรมเวลา ได้แก่ ปริมาณการส่งสินค้ารายเดือน จำนวนอุบัติเหตุบนท้องถนนรายสัปดาห์ หรือค่าสังเกตรายชั่วโมงที่เป็นผลผลิตจากระบบการทางเคมี เป็นต้น (Box et al., 2008, p.1) สำหรับหลักการวิเคราะห์อนุกรมเวลา เป็นการศึกษาความสัมพันธ์ระหว่างค่าสังเกตของอนุกรมเวลา โดยนำอนุกรมเวลามาวิเคราะห์ กำหนดตัวแบบ และใช้ตัวแบบพยากรณ์ค่าสังเกตในอนาคต เพื่อนำค่าพยากรณ์ไปใช้ในการวางแผน จัดการ และควบคุมการดำเนินการในด้านต่าง ๆ ให้มีประสิทธิภาพเพิ่มขึ้น ดังนั้น เนื้อหาในหัวข้อนี้ จะกล่าวถึงพื้นฐานทางสถิติที่จำเป็นสำหรับการกำหนดตัวแบบ วิเคราะห์ และพยากรณ์ข้อมูลอนุกรมเวลา ประกอบด้วย กระบวนการพยากรณ์ข้อมูล ส่วนประกอบของอนุกรมเวลา การแสดงผลด้วยกราฟ อนุกรมเวลาที่มีลักษณะนิ่ง (Stationary time series) การปรับอนุกรมเวลาให้มีลักษณะนิ่ง การกำหนดตัวแบบอนุกรมเวลาเชิงเดี่ยว และการวัดความคลาดเคลื่อนของการพยากรณ์

### 2.1.1 กระบวนการพยากรณ์ข้อมูล

Montgomery, Jennings, and Kulahci (2008) กล่าวว่า กระบวนการ คือ ลำดับของกิจกรรมต่อเนื่อง ซึ่งแปลงอินพุต (Input) ให้เป็นเอาต์พุต (Output) และการทำงานทุกชนิดต่างก็กระทำในรูปแบบของกระบวนการทั้งสิ้น การพยากรณ์ก็เช่นเดียวกัน โดยกิจกรรมในกระบวนการพยากรณ์สามารถแสดงได้ดังรูปที่ 2.1



รูปที่ 2.1 กระบวนการพยากรณ์ (Montgomery et al. 2008, p.12).

จากรูปที่ 2.1 สามารถอธิบายกิจกรรมต่าง ๆ ในกระบวนการพยากรณ์ ได้ดังนี้

1. Problem definition คือ ขั้นตอนการทำความเข้าใจว่าการพยากรณ์เป็นอย่างไร ซึ่งเป็นการกำหนดค่าสังเกตที่สนใจ การแบ่งคาบเวลา เช่น รายวัน รายเดือน หรือรายไตรมาส เป็นต้น และระดับความถูกต้องของการพยากรณ์ที่ยอมรับได้
2. Data collection คือ ขั้นตอนการเก็บรวบรวมและบันทึกข้อมูล ซึ่งเป็นค่าสังเกตของตัวแปรที่สนใจ ตามลำดับเวลา โดยวิธีการเก็บบันทึกข้อมูล อาจมาจากแหล่งข้อมูลปฐมภูมิ หรือแหล่งข้อมูลทุติยภูมิ ซึ่งแหล่งข้อมูลมีผลต่อความน่าเชื่อถือ และความถูกต้องของข้อมูลที่นำมาใช้ในกระบวนการพยากรณ์
3. Data analysis คือ ขั้นตอนการวิเคราะห์และสำรวจข้อมูลเบื้องต้น ซึ่งเป็นขั้นตอนสำคัญก่อนการคัดเลือกตัวแบบการพยากรณ์ โดยการสร้างกราฟข้อมูล เพื่อแสดงให้เห็นถึงรูปแบบของข้อมูล เช่น แนวโน้ม ฤดูกาล หรือวัฏจักร และคำนวณค่าสถิติของข้อมูล เช่น ค่าเฉลี่ย ค่าเบี่ยงเบนมาตรฐาน หรือเปอร์เซ็นต์ไทล์
4. Model selection and fitting คือ ขั้นตอนการเลือกตัวแบบการพยากรณ์ที่เข้ากับชุดข้อมูล (Fitting model) ซึ่งอาจเลือกตัวแบบที่เข้ากับชุดข้อมูลได้มากกว่า 1 ตัวแบบ โดยการ

กำหนดตัวแบบที่เข้ากับชุดข้อมูล ทำได้ด้วยการประมาณค่าพารามิเตอร์ที่ไม่ทราบค่า และวิธีที่นิยมใช้ในการประมาณค่า คือ วิธีกำลังสองน้อยที่สุด (The least squares method) (Montgomery et al., 2008, p.13)

5. Model validation คือ ขั้นตอนการประเมินเพื่อเลือกตัวแบบการพยากรณ์ที่ให้ค่าพยากรณ์แม่นยำมากที่สุด จากตัวแบบที่ได้เลือกไว้ในขั้นตอนที่ 4 โดยทำการวัดค่าผิดพลาดเชิงสถิติระหว่างค่าพยากรณ์กับค่าสังเกตที่แบ่งไว้เป็นชุดข้อมูลทดสอบ (Test data set) เช่น ค่า Sum square error (SSE) ค่า Root mean square error (RMSE) หรือ ค่า Mean absolute percentage error (MAPE) เป็นต้น

6. Forecasting model deployment คือ ขั้นตอนการนำตัวแบบการพยากรณ์ที่ได้จากขั้นตอนที่ 5 ไปพยากรณ์ค่าสังเกตล่วงหน้า ตามช่วงเวลาที่ต้องการ

7. Monitoring the forecasting model performance คือ ขั้นตอนติดตามผลหลังจากการนำตัวแบบการพยากรณ์ไปใช้พยากรณ์ค่าสังเกตล่วงหน้า โดยตรวจสอบว่า ตัวแบบการพยากรณ์ให้ค่าพยากรณ์ ที่ยังมีความแม่นยำอยู่ในช่วงที่ยอมรับได้อยู่หรือไม่

### 2.1.2 ส่วนประกอบของอนุกรมเวลา

ส่วนประกอบของอนุกรมเวลา ประกอบด้วยส่วนประกอบหลัก 4 ส่วน ได้แก่ แนวโน้ม (Trend) วัฏจักร (Cycle) การเปลี่ยนแปลงตามฤดูกาล (Seasonal variations) และความผันผวนผิดปกติ (Irregular fluctuations) (Bowerman et al., 2005, p.5)

1. แนวโน้ม (Trend) หมายถึง ทิศทางการเคลื่อนไหวของอนุกรมเวลา ในทิศทางเพิ่มขึ้นหรือลดลง ตลอดคาบเวลาที่สนใจ ซึ่งการเคลื่อนไหวของแนวโน้ม อาจเกิดจากปัจจัยที่หลากหลาย เช่น ยอดขายของสินค้าในอุตสาหกรรมการผลิตสินค้า อาจเกิดจากปัจจัยต่าง ๆ ได้แก่ การเปลี่ยนแปลงเทคโนโลยีอุตสาหกรรม การเปลี่ยนแปลงของรสนิยมผู้บริโภค การเพิ่มขึ้นของรายได้ประชากร การเพิ่มขึ้นของประชากร หรือการเปลี่ยนแปลงของราคาสินค้า เป็นต้น

2. วัฏจักร (Cycle) หมายถึง รอบการเคลื่อนไหวของอนุกรมเวลา ในทิศทางขึ้นหรือลงบริเวณรอบระดับแนวโน้ม ซึ่งความผันผวนของวัฏจักร มีช่วงระยะเวลาหลายปี อาจตั้งแต่ 2 ปี ถึง 10 ปี หรือยาวนานกว่านั้น แต่ละช่วงของวัฏจักรมีความแตกต่างกัน และช่วงของวัฏจักรมีความสั้นยาวแตกต่างกันด้วย ตัวอย่างของความผันผวนของวัฏจักรที่พบได้ในอนุกรมเวลา คือ วัฏจักรทางธุรกิจ (Business cycle) ที่สามารถแบ่งช่วงวัฏจักรได้หลายรูปแบบ เช่น วัฏจักรที่แบ่งเป็น 3 ช่วง ได้แก่ ช่วงรุ่งเรือง (Growth) ช่วงคงที่ (Maturity) และช่วงตกต่ำ (Decline) หรือวัฏจักรที่แบ่งเป็น 6 ช่วง ได้แก่

ช่วงเติบโต (Growth) ช่วงรุ่งเรือง (Prosperity) ช่วงเตือน (Warning) ช่วงซบเซา (Recession) ช่วงตกต่ำ (Depression) และช่วงฟื้นตัว (Recovery) (ทรงศิริ แต่สมบัติ, 2549, หน้า 31)

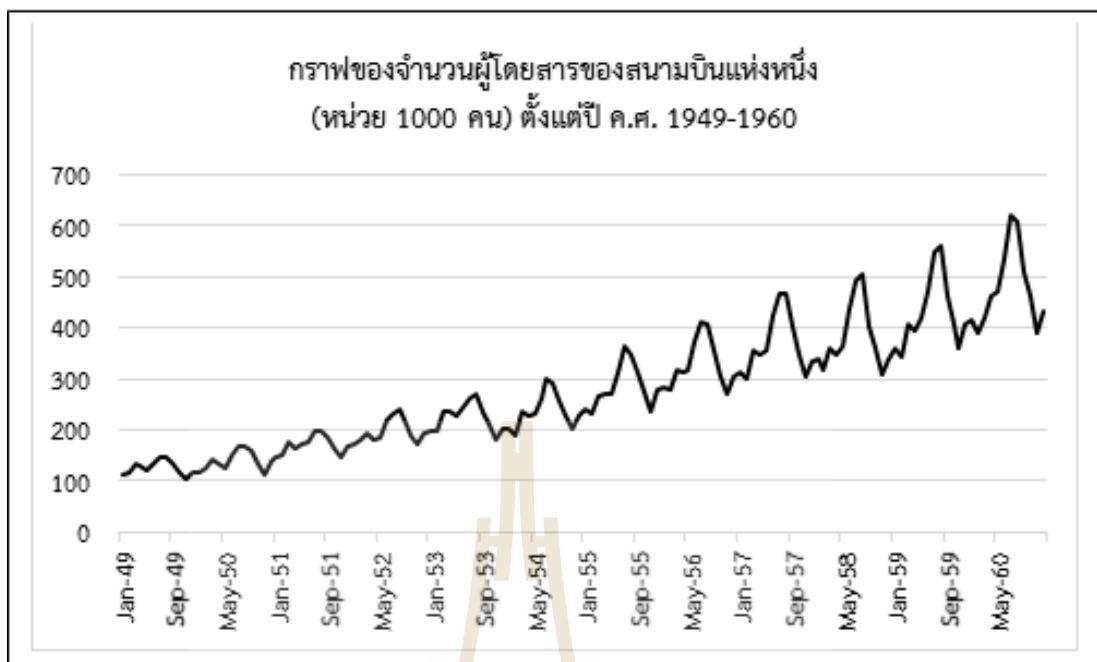
3. การเปลี่ยนแปลงตามฤดูกาล (Seasonal variations) เป็นรูปแบบเชิงคาบเวลาของอนุกรมเวลาที่มีความสมบูรณ์ในตัวเองภายในรอบปีปฏิทิน และมีลักษณะการเกิดซ้ำเป็นรายปี ซึ่งการเปลี่ยนแปลงตามฤดูกาล มักมีสาเหตุจากปัจจัยด้านสภาพภูมิอากาศ เช่น ค่าเฉลี่ยอุณหภูมิรายเดือน ยอดขายเครื่องปรับอากาศรายเดือน หรือปริมาณการใช้ไฟฟ้ารายเดือน มีลักษณะเป็นฤดูกาลโดยธรรมชาติ เพราะค่าสังเกตในฤดูร้อนมีค่าสูงกว่าฤดูหนาว และเกิดซ้ำเป็นรายปี สำหรับประเทศเขตร้อน เป็นต้น

4. ความผันผวนผิดปกติ (Irregular fluctuations) เป็นการเคลื่อนไหวที่ผิดปกติในอนุกรมเวลา ซึ่งเกิดจากปัจจัยอื่น นอกเหนือจากแนวโน้ม วัฏจักร และฤดูกาล โดยมีสาเหตุจากการเกิดเหตุการณ์ผิดปกติ ที่ไม่สามารถคาดการณ์ล่วงหน้าได้ เช่น แผ่นดินไหว พายุไต้ฝุ่น สงคราม จลาจล หรือการนัดหยุดงาน เป็นต้น และความผันผวนผิดปกติ เป็นสาเหตุที่ทำให้เกิดความคลาดเคลื่อนในการวิเคราะห์อนุกรมเวลา

### 2.1.3 การแสดงข้อมูลอนุกรมเวลาด้วยการสร้างกราฟ

การพัฒนาตัวแบบการพยากรณ์ควรเริ่มจากการวิเคราะห์ข้อมูลด้วยการสร้างกราฟ เพราะส่วนประกอบหลัก 4 ส่วน ของอนุกรมเวลาสามารถพบเห็นและเข้าใจได้ง่าย เมื่อนำข้อมูลมาสร้างกราฟ ซึ่งพื้นฐานการแสดงผลด้วยกราฟสำหรับข้อมูลอนุกรมเวลา คือ การสร้างกราฟอนุกรมเวลาด้วยการนำค่าสังเกต  $Y_t$  และคาบเวลา  $t$  เมื่อ  $t = 1, 2, \dots, T$  มาสร้างเป็นกราฟ โดยให้ค่าสังเกต  $Y_t$  เป็นแกนตั้ง และคาบเวลา  $t$  เป็นแกนนอน เมื่อทำการสร้างกราฟแล้ว ส่วนประกอบของอนุกรมเวลา เช่น แนวโน้ม หรือฤดูกาล จะพบเห็นได้จากกราฟ ซึ่งสามารถสร้างกราฟได้หลายรูปแบบ เช่น กราฟเส้น (Line chart) กราฟแท่ง (Bar chart) หรือกราฟแจกแจงความถี่ (Histogram) ตัวอย่างเช่น การสร้างกราฟเส้นของจำนวนผู้โดยสารของสนามบินแห่งหนึ่ง (หน่วย 1000 คน) ตั้งแต่ปี ค.ศ. 1956-1960 ดังรูปที่ 2.2





รูปที่ 2.2 กราฟของจำนวนผู้โดยสารของสนามบินแห่งหนึ่ง (หน่วย 1000 คน) ตั้งแต่ปี ค.ศ. 1949-1960 (พงศ์เทพ วิวัชรณะเดช และสมชาย จาคศรี, 2550, หน้า 4).

จากรูปที่ 2.2 เมื่อสร้างกราฟอนุกรมเวลา พบว่า อนุกรมเวลาเกิดการเคลื่อนไหวเนื่องจากแนวโน้มและฤดูกาลแบบบวก และจากการสร้างกราฟอนุกรมเวลา ทำให้สามารถคาดการณ์รูปแบบการเคลื่อนไหวของอนุกรมเวลาได้เบื้องต้น ซึ่งนำไปสู่การเลือกตัวแบบการพยากรณ์ที่มีความเหมาะสมกับอนุกรมเวลา เพื่อให้เกิดความแม่นยำในการพยากรณ์ได้มากยิ่งขึ้น

#### 2.1.4 อนุกรมเวลาที่มีลักษณะนิ่ง

อนุกรมเวลาที่มีลักษณะนิ่ง (Stationary time series) เป็นอนุกรมเวลาที่มีความเหมาะสมกับการนำไปใช้สร้างตัวแบบการพยากรณ์ เพราะมีความสมดุลเชิงสถิติ หรือข้อมูลมีความเสถียร เป็นผลให้การประมาณค่าพารามิเตอร์ที่ไม่ทราบค่า มีความถูกต้องน่าเชื่อถือ และเมื่อเกิดความผันผวนผิดปกติ ทำให้ค่าคลาดเคลื่อนมีค่าสูง ตัวแบบการพยากรณ์ที่สร้างจากอนุกรมเวลาที่มีลักษณะนิ่ง จะให้ค่าพยากรณ์ที่มีการปรับตัวเข้าสู่สภาพปกติได้อย่างรวดเร็ว นอกจากนี้ การนำอนุกรมเวลาที่มีลักษณะไม่นิ่ง (Non-stationary time series) ไปใช้ประมาณค่าพารามิเตอร์ในตัวแบบต่าง ๆ อาจส่งผลให้เกิดปรากฏการณ์ของความสัมพันธ์ปลอม (Spurious regression)



อนุกรมเวลาที่มีลักษณะหนึ่ง หมายความว่า คุณสมบัติของอนุกรมเวลาไม่มีผลกระทบจากการเลื่อนจุดกำเนิดของเวลา กล่าวคือ การแจกแจงความน่าจะเป็นของค่าสังเกต  $Y_t, Y_{t+1}, \dots, Y_{t+n}$  มีลักษณะเช่นเดียวกับการแจกแจงความน่าจะเป็นของค่าสังเกต  $Y_{t+k}, Y_{t+k+1}, \dots, Y_{t+k+n}$  และถ้ากำหนดให้  $n=0$  จะกล่าวได้ว่า การแจกแจงความน่าจะเป็นของ  $Y_t$  มีลักษณะเหมือนกันตลอดคาบเวลา หรือหมายความว่า อนุกรมเวลาที่มีลักษณะหนึ่ง คือ อนุกรมเวลาที่มีค่าเฉลี่ย (Mean) และความแปรปรวน (Variance) คงที่ ตลอดคาบเวลา

จากที่กล่าวมา ก่อนนำอนุกรมเวลาไปกำหนดตัวแบบการพยากรณ์ ควรมีการทดสอบความนิ่งของข้อมูล (Unit root test) ซึ่งการทดสอบลักษณะหนึ่งของอนุกรมเวลา มีหลายวิธี โดยวิธีที่ได้รับความนิยมและเป็นที่ยอมรับ ได้แก่ วิธี Augmented Dickey-Fuller (ADF) test ที่เสนอโดย Dickey and Fuller ในปี ค.ศ. 1979 ถ้าผลการทดสอบแสดงให้เห็นว่าอนุกรมเวลาที่มีลักษณะไม่นิ่ง นั่นคือ อนุกรมเวลาที่มีการเคลื่อนไหวตามแนวโน้มของเวลา (Time trend) และความแปรปรวนมีค่าเพิ่มขึ้นตามแนวโน้มของระยะเวลาที่เพิ่มขึ้น ดังนั้น เพื่อให้การประมาณค่าพารามิเตอร์มีความถูกต้องน่าเชื่อถือ จะต้องปรับข้อมูลให้มีลักษณะหนึ่ง ด้วยการหาผลต่างของค่าสังเกตลำดับที่ 1 (First difference) หรือลำดับที่สูงขึ้นไปจนกว่าอนุกรมเวลาจะมีลักษณะหนึ่ง โดยใช้วิธีหาผลต่างของค่าสังเกต ดังสมการที่ 2.1

$$\Delta Y_t = \alpha + \beta + \gamma Y_{t-1} + \delta_1 \Delta Y_{t-1} + \dots + \delta_{p-1} \Delta Y_{t-p+1} + \varepsilon_t \quad \text{----- (2.1)}$$

เมื่อ  $\Delta Y_t = Y_t - Y_{t-1}$  = ผลต่างของค่าสังเกตลำดับแรก ณ เวลา  $t$

$\alpha$  = ค่าคงที่

$\beta$  = ค่าสัมประสิทธิ์บนแนวโน้มเวลา (The coefficient on a time trend)

$\gamma$  = ค่าสัมประสิทธิ์การทดสอบสมมุติฐานลักษณะหนึ่ง

$\delta$  = ค่าสัมประสิทธิ์ของอนุพันธ์ล่าช้า (The coefficient of lagged derivatives)

$\varepsilon_t$  = ค่าคลาดเคลื่อน (Error term) ณ เวลา  $t$  ที่วัดด้วยการประมาณแบบกำลังสองน้อยที่สุด (The least squares estimate)

ภายใต้สมมุติฐานที่ใช้ในการทดสอบ

$H_0: \gamma = 0$  (Non-stationary)

$H_1: \gamma \neq 0$  (Stationary)

สำหรับการทดสอบสมมุติฐาน จะกำหนดระดับความเชื่อมั่น 95 % (P-value = 0.05) ถ้าการทดสอบยอมรับสมมุติฐานว่าง หมายความว่า อนุกรมเวลามีลักษณะไม่นิ่ง แต่ถ้าการทดสอบปฏิเสธสมมุติฐานว่าง และยอมรับสมมุติฐานทางเลือก หมายความว่า อนุกรมเวลามีลักษณะนิ่ง

### 2.1.5 การปรับอนุกรมเวลาให้มีลักษณะนิ่ง

เมื่อทำการทดสอบลักษณะนิ่งของอนุกรมเวลา ถ้าผลการทดสอบปรากฏว่า อนุกรมเวลามีลักษณะไม่นิ่ง หมายความว่า อนุกรมเวลาไม่คงที่ มีความแปรปรวนไม่คงที่ และอาจมีการเปลี่ยนแปลงตามฤดูกาล ดังนั้น การปรับอนุกรมเวลาให้มีลักษณะนิ่ง จะต้องกำจัดแนวโน้ม ทำให้ความแปรปรวนคงที่ และกำจัดกาเปลี่ยนแปลงตามฤดูกาล

1. การกำจัดแนวโน้ม เป็นการทำให้ค่าเฉลี่ยคงที่ ทำได้ด้วยการสร้างอนุกรมเวลาชุดใหม่จากการหาผลต่างของค่าสังเกตที่เวลา  $t$  กับค่าสังเกตที่เวลา  $t-k$  เรียกว่า การหาผลต่างของค่าสังเกตลำดับที่  $k$  ซึ่งค่า  $k$  จะเริ่มจาก 1 เรียกว่า การหาผลต่างของค่าสังเกตลำดับที่ 1 จากนั้นทดสอบความนิ่ง ถ้าข้อมูลยังไม่นิ่ง ให้เพิ่มค่า  $k$  ขึ้นอีก 1 จนกว่าอนุกรมเวลาจะมีลักษณะนิ่ง ดังนั้น การกำจัดแนวโน้ม คือ การสร้างอนุกรมเวลาชุดใหม่ จากการหาผลต่างของค่าสังเกตที่เวลา  $t$  กับค่าสังเกตที่เวลา  $t-k$  เริ่มจาก  $k = 1$  ดังสมการที่ 2.2

$$Z_t = Y_t - Y_{t-k} \quad \text{----- (2.2)}$$

เมื่อ  $k$  = การล่าหลัง (Lag)  $k$  ช่วงเวลา โดย  $k$  เริ่มจาก 1

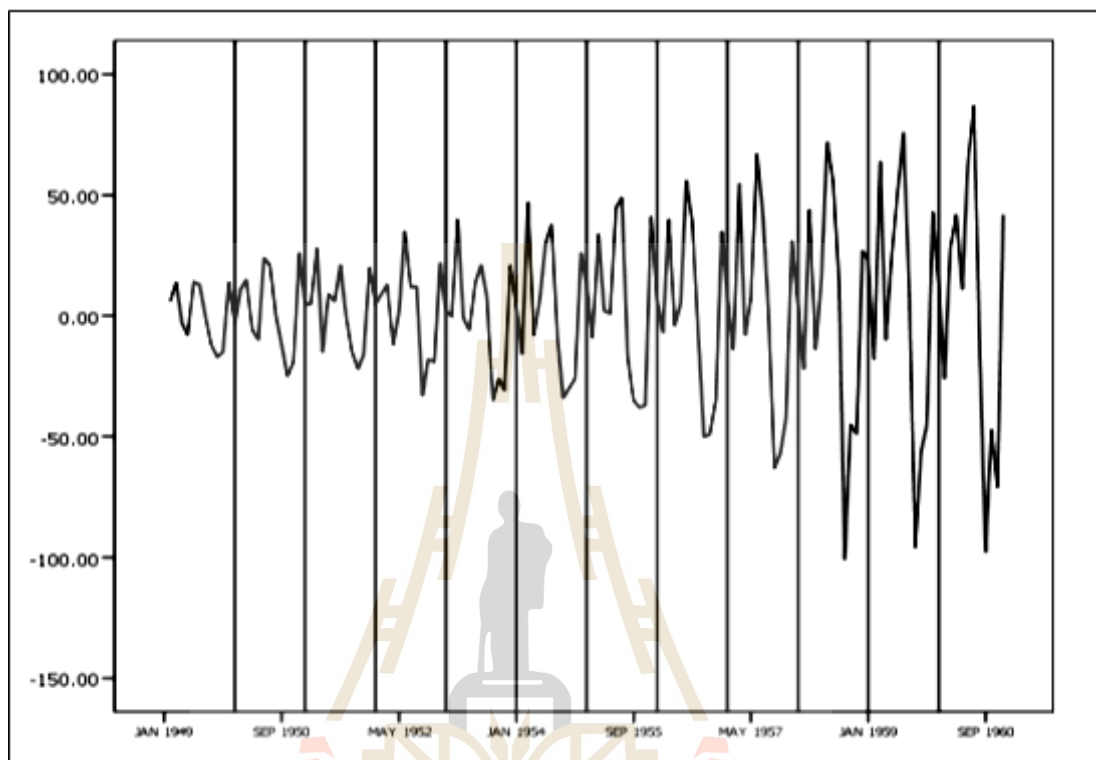
$Y_t$  = ค่าสังเกต ณ เวลา  $t$

$Y_{t-k}$  = ค่าสังเกต ณ เวลา  $t-k$

$Z_t$  = ค่าสังเกตใหม่ ณ เวลา  $t$  ซึ่งเป็นผลต่างของค่าสังเกตเดิมที่ล่าหลังกัน

$k$  ช่วงเวลา

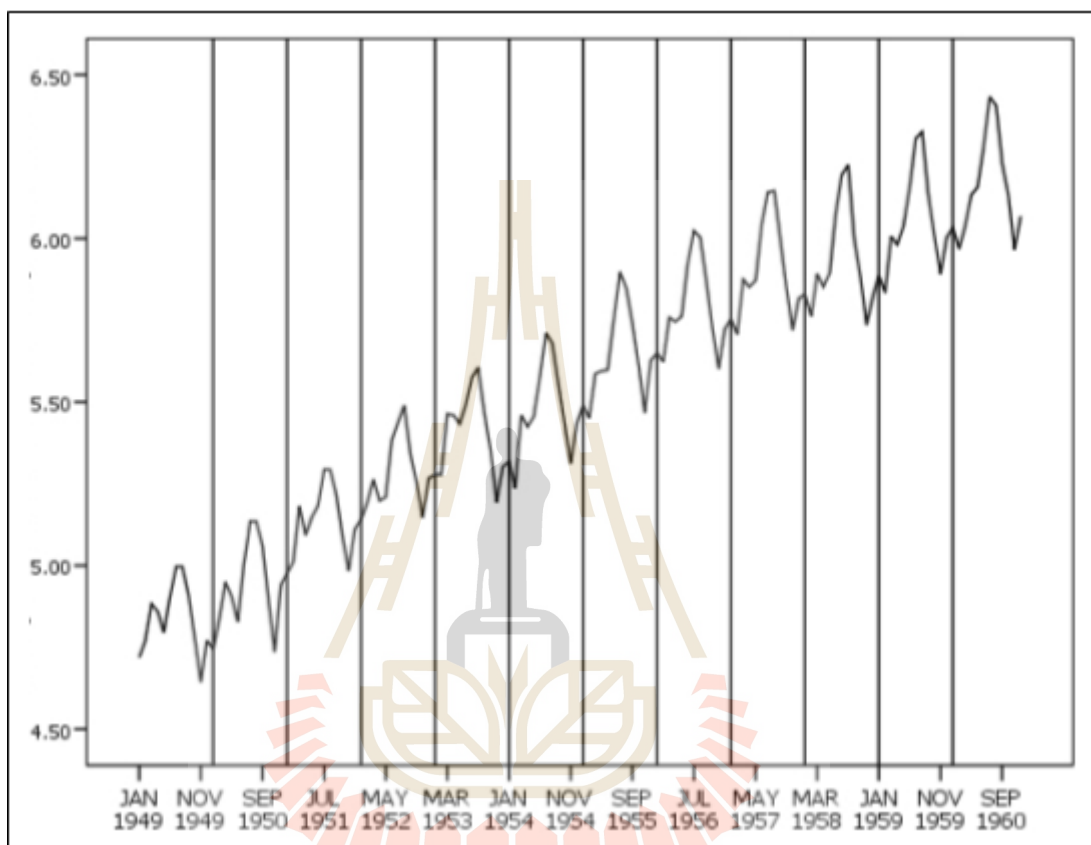
จากกราฟจำนวนผู้โดยสารของสนามบินแห่งหนึ่ง (รูปที่ 2.2) เมื่อกำจัดแนวโน้มด้วยการหาผลต่างค่าล่าหลังที่  $k=1$  จะได้กราฟดังรูปที่ 2.3



รูปที่ 2.3 กราฟของจำนวนผู้โดยสาร เมื่อกำจัดแนวโน้มด้วยการล่าหลัง 1 ช่วงเวลา ( $k=1$ )  
(พงศ์เทพ วิวรรณระเดช และสมชาย จาดศรี, 2550, หน้า 8).

2. การทำให้ความแปรปรวนคงที่ สามารถทำได้โดยการแปลงค่าสังเกตด้วยการหาค่าลอการิทึมฐาน  $e$  (Natural logarithm) หรือการหาค่ารากที่สอง (Square root) ของค่าสังเกต ในกรณีที่ใช้วิธีหาค่ารากที่สองกับค่าสังเกตที่มีค่าเป็นลบ ซึ่งค่าลบไม่สามารถนำไปหาค่ารากที่สองได้ ก็อาจแปลงค่าสังเกตก่อนนำไปหาค่ารากที่สอง ด้วยการบวกเพิ่มทุกค่าสังเกตด้วยค่าคงที่ที่เหมาะสม เพื่อให้ไม่เกิดค่าลบ แล้วจึงนำไปหาค่ารากที่สอง และเมื่อนำค่าสังเกตที่ผ่านการแปลงด้วยการหาค่าลอการิทึมหรือรากที่สอง ไปกำหนดตัวแบบการพยากรณ์ และได้ตัวแบบการพยากรณ์ที่เหมาะสมที่สุดแล้ว เมื่อนำตัวแบบพยากรณ์มาพยากรณ์ข้อมูลล่วงหน้า จะต้องนำข้อมูลมาทำการแปลงกลับดั้งเดิมด้วย เช่น ถ้าแปลงค่าโดยใช้ลอการิทึมฐาน  $e$  เมื่อแปลงกลับ ให้นำค่าที่ได้มาเป็นเลขยกกำลังที่

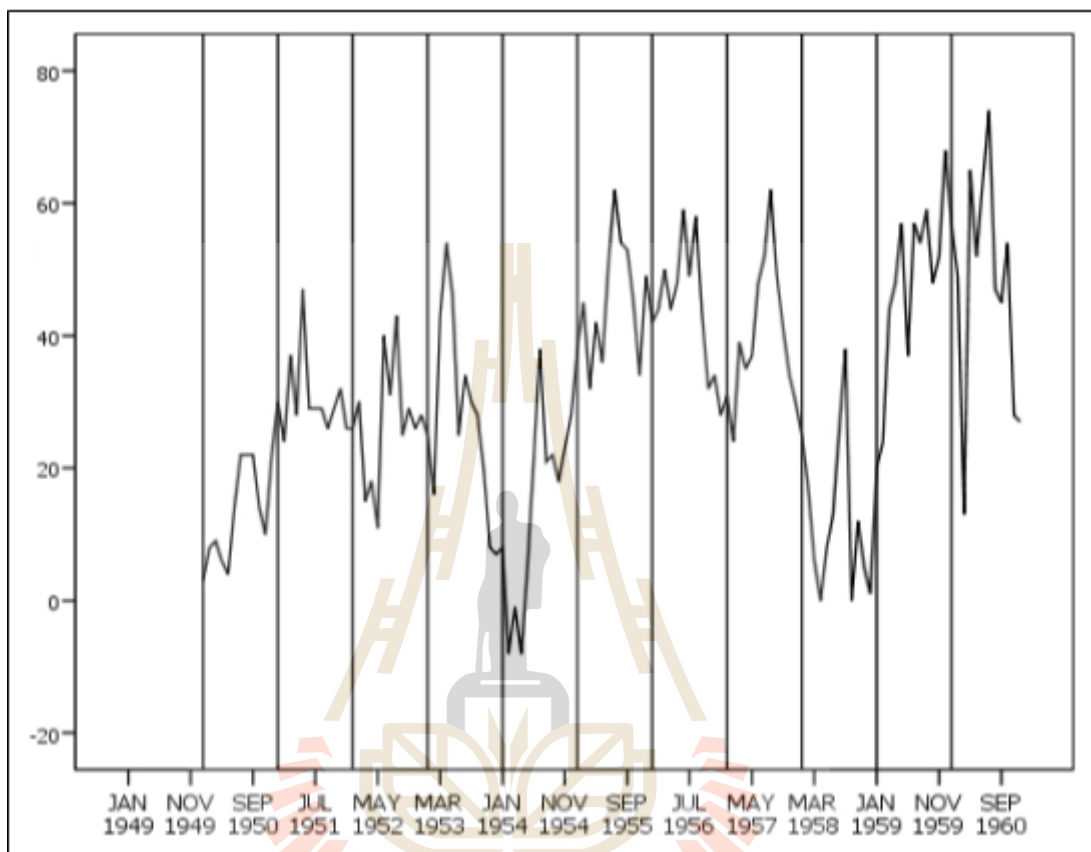
มีฐานเป็นค่า  $e$  หรือถ้าแปลงค่าโดยใช้รากที่สอง เมื่อแปลงกลับ ให้นำค่าที่ได้มายกกำลังสอง เป็นต้น ซึ่งจากกราฟจำนวนผู้โดยสารของสนามบินแห่งหนึ่ง (รูปที่ 2.2) เมื่อทำให้ความแปรปรวนคงที่ โดยการแปลงข้อมูลด้วยการหาค่าลอการิทึมฐาน  $e$  (Natural logarithm) จะได้กราฟดังรูปที่ 2.4



รูปที่ 2.4 กราฟของจำนวนผู้โดยสาร เมื่อแปลงข้อมูลด้วยการหาค่าลอการิทึมฐาน  $e$   
(พงศ์เทพ วิวรรณระเดช และสมชาย จาคศรี, 2550, หน้า 10).

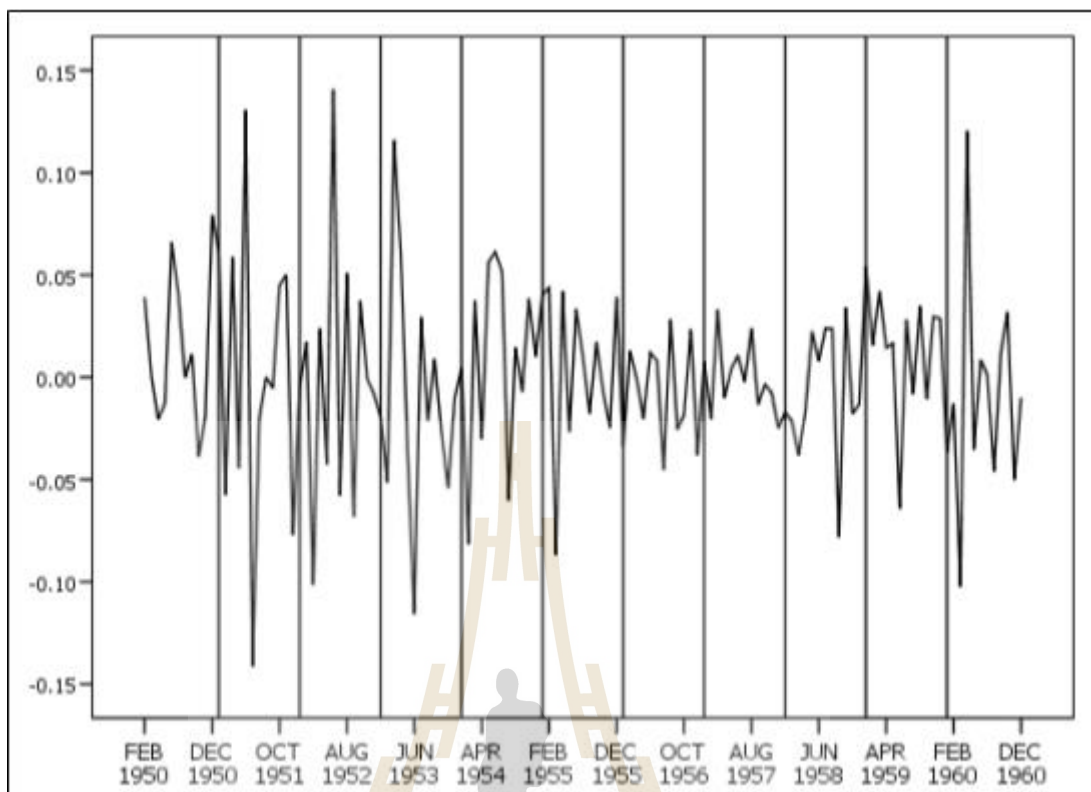
3. การกำจัดการเปลี่ยนแปลงตามฤดูกาล สามารถทำได้ด้วยวิธีการเดียวกับการกำจัดแนวโน้ม ซึ่งอนุกรมเวลาที่มีการเปลี่ยนแปลงตามฤดูกาล เป็นอนุกรมเวลาที่มีคาบเวลาเป็นรายเดือน ดังนั้นการกำจัดการเปลี่ยนแปลงตามฤดูกาล ทำได้ด้วยการหาผลต่างของค่าสังเกต ที่ล่าหลัง 12 ช่วงเวลา ( $k = 12$ ) แล้วนำผลต่างที่ได้ มาสร้างเป็นอนุกรมเวลาชุดใหม่ จากนั้น จึงนำไปใช้ประมาณค่าพารามิเตอร์ ในขั้นตอนการกำหนดคัวแบบ และจากกราฟจำนวนผู้โดยสารของสนามบินแห่ง

หนึ่ง (รูปที่ 2.2) เมื่อทำการเปลี่ยนแปลงตามฤดูกาล ด้วยการหาผลต่างค่าล่าหลังที่  $k=12$  จะได้กราฟดังรูปที่ 2.5



รูปที่ 2.5 กราฟของจำนวนผู้โดยสาร เมื่อกำจัดฤดูกาลด้วยการล่าหลัง 12 ช่วงเวลา ( $k=12$ ) (พงศเทพ วิวรรณระเดช และสมชาย ขาดศรี, 2550, หน้า 11).

4. การกำจัดแนวโน้ม การเปลี่ยนแปลงตามฤดูกาล และการทำให้ความแปรปรวนคงที่ สามารถทำได้ด้วยการผสมผสานวิธีการที่ได้กล่าวไว้ข้างต้น ซึ่งควรเริ่มจากการทำให้ความแปรปรวนคงที่ ด้วยการแปลงข้อมูล โดยใช้การหาค่าลอการิทึม หรือการหาค่ารากที่สอง จากนั้น นำชุดข้อมูลที่ผ่านการแปลงข้อมูล มากำจัดแนวโน้ม และการเปลี่ยนแปลงตามฤดูกาลตามลำดับ ซึ่งจากกราฟจำนวนผู้โดยสารของสนามบินแห่งหนึ่ง (รูปที่ 2.2) เมื่อทำให้ความแปรปรวนคงที่ โดยการแปลงข้อมูลด้วยลอการิทึม กำจัดแนวโน้มด้วยการหาผลต่างค่าล่าหลังที่  $k=1$  และกำจัดการเปลี่ยนแปลงตามฤดูกาลด้วยการหาผลต่างค่าล่าหลังที่  $k=12$  จะได้กราฟดังรูปที่ 2.6



รูปที่ 2.6 กราฟของจำนวนผู้โดยสาร เมื่อทำให้ความแปรปรวนคงที่ กำจัดแนวโน้ม และกำจัดฤดูกาล (พงศเทพ วิวรรณะเดช และสมชาย จาคศรี, 2550, หน้า 12).

### 2.1.6 การกำหนดตัวแบบอนุกรมเวลาเชิงเดี่ยว

อนุกรมเวลาสามารถแบ่งได้ 2 ประเภท ได้แก่ ประเภทที่ 1 คือ อนุกรมเวลาเชิงเดี่ยว (Univariate time series) หมายถึง อนุกรมเวลาที่มีค่าสังเกตที่สนใจ เพียงค่าเดียว ในแต่ละช่วงเวลา ส่วนประเภทที่ 2 คือ อนุกรมเวลาเชิงพหุ (Multivariate time series) หมายถึง อนุกรมเวลาที่มีค่าสังเกตที่สนใจ มากกว่า 1 ค่า ในแต่ละช่วงเวลา ซึ่งเนื้อหาในส่วนนี้ กล่าวถึงการกำหนดตัวแบบอนุกรมเวลาเชิงเดี่ยว เท่านั้น โดยเป็นการหาสมการแสดงความสัมพันธ์ระหว่างค่าสังเกตกับเวลาที่เปลี่ยนแปลงไป และวิธีการกำหนดตัวแบบอนุกรมเวลาเชิงเดี่ยวมีหลายวิธี ดังนี้

1. วิธีการวิเคราะห์ความถดถอย (Regression analysis) เป็นวิธีการที่ทำการหาความสัมพันธ์ระหว่างตัวแปร โดยมีตัวแปร 1 ตัวแปร หรือมากกว่า 1 ตัวแปร เป็นตัวแปรอิสระ (Independent variable) ส่วนอีกตัวแปรหนึ่งเป็นตัวแปรตาม (Dependent variable) ในกรณีที่มีตัวแปรอิสระมากกว่า 1 ตัวแปร เรียกว่า การวิเคราะห์การถดถอยแบบพหุ (Multiple regression analysis) แต่

เนื้อหาในส่วนนี้ เป็นการวิเคราะห์ความถดถอย โดยมีตัวแปรอิสระ 1 ตัวแปร นั่นคือ เวลา ส่วนตัวแปรตาม คือ ค่าสังเกตของตัวแปรที่สนใจ นอกจากนี้ การวิเคราะห์ความถดถอย สามารถแบ่งตามรูปแบบความสัมพันธ์ได้เป็น 2 รูปแบบ คือ ความถดถอยเชิงเส้น (Linear regression) หมายถึง ค่าสังเกตและเวลา มีความสัมพันธ์กันในรูปแบบของสมการเชิงเส้น อีกประเภทหนึ่ง คือ ความถดถอยไม่เป็นเชิงเส้น (Non-linear regression) หมายถึง ค่าสังเกตและเวลา มีความสัมพันธ์กันในรูปแบบที่ไม่ใช่สมการเชิงเส้น เช่น สมการโพลีโนเมียล หรือสมการเอกซ์โพเนนเชียล เป็นต้น

2. วิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล (Exponential smoothing) เป็นวิธีการกำหนดตัวแบบอนุกรมเวลา โดยนำค่าสังเกตแต่ละช่วงเวลาในอดีต มาใช้ในการพยากรณ์ค่าสังเกตในอนาคต ซึ่งมีการให้น้ำหนักค่าสังเกตแต่ละช่วงเวลาไม่เท่ากัน โดยให้น้ำหนักค่าสังเกตล่าสุดมากที่สุด ส่วนค่าสังเกตก่อนหน้า จะมีน้ำหนักลดหลั่นลง ไปแบบเอกซ์โพเนนเชียล และวิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล แบ่งได้ 3 ประเภท ตามจำนวนของพารามิเตอร์ปรับให้เรียบ (Smoothing parameter) ได้แก่

2.1 Single exponential smoothing (SES) มีพารามิเตอร์ปรับให้เรียบ 1 ตัว คือ ค่าปรับให้เรียบของน้ำหนัก ( $\alpha$ ) เป็นค่าคงที่ ที่มีค่าอยู่ระหว่าง 0 ถึง 1

2.2 Double exponential smoothing (DES) มีพารามิเตอร์ปรับให้เรียบ 2 ตัว คือ ค่าปรับให้เรียบของน้ำหนัก ( $\alpha$ ) และค่าปรับให้เรียบของแนวโน้ม ( $\beta$ ) ทั้ง 2 ตัว เป็นค่าคงที่ ที่มีค่าอยู่ระหว่าง 0 ถึง 1

2.3 Triple exponential smoothing (TES) มีพารามิเตอร์ปรับให้เรียบ 3 ตัว คือ ค่าปรับให้เรียบของน้ำหนัก ( $\alpha$ ) ค่าปรับให้เรียบของแนวโน้ม ( $\beta$ ) และค่าปรับให้เรียบของฤดูกาล ( $\gamma$ ) ทั้ง 3 ตัว เป็นค่าคงที่ ที่มีค่าอยู่ระหว่าง 0 ถึง 1

3. วิธีการของบ็อกซ์และเจนกินส์ (Box-Jenkins Method) เป็นวิธีการวิเคราะห์อนุกรมเวลาที่อาศัยกระบวนการเชิงสุ่ม (Stochastic process) โดยถือว่าค่าสังเกตที่เกิดขึ้นตามเวลาที่เปลี่ยนแปลงไป มีลักษณะการเกิดค่าสังเกตที่เป็นไปตามกฎความน่าจะเป็น ซึ่งวิธีการของบ็อกซ์และเจนกินส์ จะพยากรณ์ค่าสังเกตในอนาคต จากรูปแบบหลัก 2 รูปแบบ ได้แก่ รูปแบบ Autoregressive คือ รูปแบบที่กำหนดว่า ค่าพยากรณ์ที่เวลาใด ๆ ขึ้นอยู่กับค่าสังเกตก่อนหน้า และรูปแบบ Moving average คือ รูปแบบที่กำหนดว่า ค่าพยากรณ์ที่เวลาใด ๆ ขึ้นอยู่กับค่าคลาดเคลื่อนก่อนหน้า



### 2.1.7 การวัดความคลาดเคลื่อนของการพยากรณ์

การวัดความคลาดเคลื่อนของการพยากรณ์ (Measuring forecast errors) คือ การวัดค่าความคลาดเคลื่อนระหว่างค่าพยากรณ์ ( $\hat{Y}_t$ ) กับค่าสังเกตจริง ( $Y_t$ ) ณ เวลา  $t$  ใด ๆ ซึ่งค่าคลาดเคลื่อนแต่ละค่า บ่งบอกถึงการเบี่ยงเบนของค่าพยากรณ์ออกจากค่าสังเกตจริง โดยสามารถแสดงสมการหาค่าคลาดเคลื่อนของการพยากรณ์ ( $\varepsilon_t$ ) ณ เวลา  $t$  ใด ๆ ได้ดังสมการที่ 2.3

$$\varepsilon_t = Y_t - \hat{Y}_t \quad \text{----- (2.3)}$$

จากสมการที่ 2.3 เมื่อหาผลรวมของค่าความคลาดเคลื่อนตลอดช่วงเวลาที่วัดค่าสังเกต ตั้งแต่เวลา  $t=1$  ถึง  $t=n$  เมื่อ  $n$  คือ จำนวนของค่าสังเกตที่นำมาใช้กำหนดตัวแบบการพยากรณ์ ดังนี้  $\sum_{t=1}^n (Y_t - \hat{Y}_t)$  ซึ่งผลรวมของค่าคลาดเคลื่อนที่เกิดขึ้นอาจเข้าใกล้ศูนย์ แต่ไม่สามารถกล่าวได้ว่า การพยากรณ์มีค่าคลาดเคลื่อนต่ำ เพราะสาเหตุที่ผลรวมของค่าคลาดเคลื่อนเข้าใกล้ศูนย์ เนื่องจากค่าคลาดเคลื่อนแต่ละช่วงเวลาที่ค่าเป็นบวกและลบ สลับกันไปมา ทำให้เมื่อนำมาหาผลรวมแล้ว ค่าคลาดเคลื่อนที่เป็นบวกและลบ หักล้างกันเอง ดังนั้น ในการวัดค่าคลาดเคลื่อน เพื่อใช้บ่งชี้ความแม่นยำของการพยากรณ์ จึงต้องมีการกำจัดเครื่องหมายบวกหรือลบที่เกิดขึ้น ด้วยการใส่ค่าสัมบูรณ์ (Absolute) หรือการยกกำลังสอง (Square) ก่อนนำมาหาผลรวม ซึ่งวิธีการวัดค่าคลาดเคลื่อนที่นิยมใช้ สรุปได้ดังนี้

1. Sum of Squared Error (SSE) เป็นการวัดค่าคลาดเคลื่อนด้วยการนำค่าคลาดเคลื่อนแต่ละค่ายกกำลังสอง แล้วหาผลรวมทั้งหมด โดยค่า SSE บ่งบอกถึงขนาดของความคลาดเคลื่อนรวม แต่มิได้บ่งบอกถึงทิศทางของความคลาดเคลื่อน และมีความไวต่อการเปลี่ยนแปลงของขนาดและการแปลงข้อมูล ซึ่งการวัดค่า SSE เป็นวิธีการวัดความคลาดเคลื่อนรวมที่ง่าย แต่ยากต่อการแปลผลค่าที่เกิดขึ้น โดยสมการหาค่า SSE แสดงได้ดังสมการที่ 2.4 (Hamzacebi, 2008)

$$SSE = \sum_{t=1}^n (Y_t - \hat{Y}_t)^2 \quad \text{----- (2.4)}$$



2. Mean Squared Error (MSE) เป็นการวัดค่าคลาดเคลื่อนด้วยการหาค่าเฉลี่ยของผลรวมความคลาดเคลื่อน โดยที่ ค่า MSE มีคุณสมบัติเช่นเดียวกับค่า SSE และสมการหาค่า MSE แสดงได้ดังสมการที่ 2.5 (Hamzacebi, 2008)

$$MSE = \frac{1}{n} \sum_{t=1}^n (Y_t - \hat{Y}_t)^2 \quad \text{----- (2.5)}$$

3. Root Mean Squared Error (RMSE) เป็นการวัดค่าคลาดเคลื่อนมาตรฐาน ที่มีการใช้กันอย่างกว้างขวาง โดยเป็นการวัดขนาดของความคลาดเคลื่อน ซึ่งไม่สามารถใช้เปรียบเทียบและเฉลี่ยความคลาดเคลื่อนข้ามอนุกรมเวลาต่างชนิดได้ และสมการหาค่า RMSE แสดงได้ดังสมการที่ 2.6 (Bergmeir and Benítez, 2012)

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (Y_t - \hat{Y}_t)^2} \quad \text{----- (2.6)}$$

4. Mean Absolute Error (MAE) เป็นการวัดค่าคลาดเคลื่อนด้วยการหาค่าเฉลี่ยของผลรวมค่าสัมบูรณ์ของค่าคลาดเคลื่อนแต่ละค่า โดยค่า MAE บ่งบอกถึงขนาดของความคลาดเคลื่อนรวม แต่ไม่ได้บ่งบอกถึงทิศทางของความคลาดเคลื่อน นอกจากนี้ ค่า MAE ขึ้นอยู่กับขนาดของการวัดและได้รับผลกระทบเมื่อมีการแปลงข้อมูล ซึ่งสมการหาค่า MAE แสดงได้ดังสมการที่ 2.7 (Hamzacebi, 2008)

$$MAE = \frac{1}{n} \sum_{t=1}^n |Y_t - \hat{Y}_t| \quad \text{----- (2.7)}$$

5. Mean Absolute Percentage Error (MAPE) เป็นการวัดค่าคลาดเคลื่อนด้วยการหารค่าอ้างอิง และกำหนดเป็นเปอร์เซ็นต์ของความคลาดเคลื่อน และสมการหาค่า MAPE แสดงได้ดังสมการที่ 2.8 (Bergmeir and Benítez, 2012)

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| 100 \frac{Y_t - \hat{Y}_t}{Y_t} \right| \quad \text{----- (2.8)}$$

สำหรับการวัดค่า MAPE ใช้การหาค่าสัมบูรณ์ เพื่อกำจัดเครื่องหมายบวกลบ ไม่ให้ค่าคลาดเคลื่อนหักล้างกันเอง แต่ปัญหาหลักของการวัดด้วยวิธีนี้ คือ การหารด้วย  $Y_t$  ทำให้กรณีที่  $Y_t = 0$  ไม่สามารถนิยามค่า MAPE ได้ นอกจากนี้ ถ้าค่า  $Y_t$  เข้าใกล้ศูนย์ จะทำให้ค่า MAPE มีค่าสูงมาก และอาจไม่สะท้อนความคลาดเคลื่อนที่เกิดจากการพยากรณ์ได้

6. Mean Relative Absolute Error (MRAE) เป็นการวัดค่าคลาดเคลื่อนด้วยการหารค่าอ้างอิงเช่นเดียวกับการวัดค่า MAPE แต่ใช้ความคลาดเคลื่อนจากการเปรียบเทียบแบบคู่เทียบ (Benchmark method) และสมการหาค่า MRAE แสดงได้ดังสมการที่ 2.9 (Bergmeir and Benítez, 2012)

$$\text{MRAE} = \frac{1}{n} \sum_{t=1}^n \left| \frac{Y_t - \hat{Y}_t}{Y_t - \hat{Y}_{tB}} \right| \quad \text{----- (2.9)}$$

เมื่อ  $\hat{Y}_{tB}$  เป็นค่าพยากรณ์ของ  $Y_t$  ที่ได้จากวิธีการแบบคู่เทียบ อย่างไรก็ตามวิธีการนี้ เกิดปัญหาหากส่วนหารเป็นศูนย์ หรือมีค่าเข้าใกล้ศูนย์ เช่นเดียวกับการวัดค่า MAPE

7. Relative Mean Average Error (RELMAE) เป็นวิธีการวัดค่าคลาดเคลื่อน ด้วยการวัดความคลาดเคลื่อนเฉลี่ย (Mean average error : MAE) เปรียบเทียบกับการวัดความคลาดเคลื่อนเฉลี่ยที่ได้จากวิธีแบบคู่เทียบ ( $\text{MAE}_B$ ) และสมการหาค่า RELMAE แสดงได้ดังสมการที่ 2.10 (Bergmeir and Benítez, 2012)

$$\text{RELMAE} = \frac{\text{MAE}}{\text{MAE}_B} \quad \text{----- (2.10)}$$

วิธีการวัดค่า RELMAE มีข้อดี คือ ไม่เกิดปัญหาส่วนหารเป็นศูนย์ หรือเข้าใกล้ศูนย์ แต่มีข้อด้อย คือ ไม่สามารถคำนวณค่าได้โดยตรงจากอนุกรมเวลา นอกจากนี้ การใช้วิธีการแบบคู่เทียบ อาจชักนำให้เกิดพฤติกรรมของการวัดค่า ที่ไม่พึงปรารถนาบางประการ เช่น ความโน้มเอียงในการวัดค่า เป็นต้น

สรุป วิธีการวัดความคลาดเคลื่อนแต่ละวิธี ต่างมีข้อเด่นและข้อด้อย การเลือกใช้วิธีการวัดค่าความคลาดเคลื่อน จึงอยู่ที่วัตถุประสงค์ของการวัดความคลาดเคลื่อน เช่น วัดความ

แข็งแกร่งของตัวแบบการพยากรณ์ วัดขนาดของความคลาดเคลื่อน วัดค่าความคลาดเคลื่อนเป็นเปอร์เซ็นต์ ต้องการวิธีการวัดที่คำนวณง่าย หรือใช้งานง่าย เป็นต้น

## 2.2 วิธีการสร้างตัวแบบอนุกรมเวลาเชิงเดี่ยว

การสร้างตัวแบบอนุกรมเวลาเชิงเดี่ยว (Univariate time series modeling) หมายถึง การค้นหาหรือสร้างสมการคณิตศาสตร์ เพื่อแสดงความสัมพันธ์ระหว่างค่าสังเกตของตัวแปร 1 ตัว กับเวลาที่เปลี่ยนแปลงไป ซึ่งการบันทึกค่าสังเกตของตัวแปร มีระยะห่างของเวลาเท่ากัน และเป็นไปตามลำดับเวลา จากนั้น ทำการวิเคราะห์รูปแบบของค่าสังเกตที่บันทึกไว้ เพื่อทำความเข้าใจการเคลื่อนไหวตามเวลาของอนุกรมเวลา และสร้างสมการที่เป็นความสัมพันธ์ระหว่างค่าสังเกตกับเวลา ภายใต้สมมุติฐานว่า รูปแบบการเคลื่อนไหวของค่าสังเกตในอนาคต เป็นเช่นเดียวกับรูปแบบการเคลื่อนไหวของค่าสังเกตในอดีต สำหรับวิธีการสร้างสมการ คือ การประมาณค่าพารามิเตอร์ที่ไม่ทราบค่าในสมการ ด้วยการใช้หลักทางสถิติ ทำให้สมการที่สร้างขึ้น สามารถใช้ประมาณการ หรืออธิบายการเคลื่อนไหวของอนุกรมเวลา และใช้พยากรณ์ค่าสังเกตในอนาคตได้ จากที่กล่าวมา คือ วิธีการสร้างตัวแบบอนุกรมเวลาเชิงเดี่ยว สำหรับพยากรณ์ค่าสังเกตล่วงหน้า ตามช่วงเวลาที่ต้องการ ซึ่งมีหลายวิธี ในหัวข้อนี้ จะกล่าวถึงวิธีที่สำคัญ และนิยมใช้อย่างแพร่หลาย 3 วิธี ได้แก่ วิธีการวิเคราะห์ความถดถอย วิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล และวิธีการของบอซซ์และเจนกินส์

### 2.2.1 วิธีการวิเคราะห์ความถดถอย

การวิเคราะห์ความถดถอย (Regression analysis) เป็นการศึกษาความสัมพันธ์ระหว่างปัจจัย หรือตัวแปรตั้งแต่ 2 ตัวแปรขึ้นไป โดยเป็นความสัมพันธ์ของตัวแปรอิสระ ที่มีค่าของตัวแปรเกิดขึ้นอย่างอิสระ และตัวแปรอิสระมีตั้งแต่ 1 ตัวแปรขึ้นไป กับตัวแปรตาม 1 ตัวแปร ซึ่งค่าของตัวแปรตามขึ้นอยู่กับรูปแบบความสัมพันธ์กับตัวแปรอิสระ วัตถุประสงค์ของการวิเคราะห์ความถดถอย คือ เพื่อให้ทราบถึงลักษณะของความสัมพันธ์ ขนาดของความสัมพันธ์ และทิศทางของความสัมพันธ์ระหว่างตัวแปรอิสระกับตัวแปรตาม โดยการวิเคราะห์ความถดถอย แบ่งได้เป็น 2 ประเภท ได้แก่ การวิเคราะห์ความถดถอยอย่างง่าย (Simple regression analysis) และการวิเคราะห์ความถดถอยเชิงพหุ (Multiple regression analysis) โดยการวิเคราะห์ความถดถอยอย่างง่าย เป็นการศึกษาความสัมพันธ์ระหว่างตัวแปรอิสระ 1 ตัวแปร กับตัวแปรตาม 1 ตัวแปร ส่วนการ

วิเคราะห์ความถดถอยเชิงพหุ เป็นการศึกษาความสัมพันธ์ระหว่างตัวแปรอิสระมากกว่า 1 ตัวแปร กับตัวแปรตาม 1 ตัวแปร

สำหรับอนุกรมเวลาเชิงเดี่ยว คือ ค่าสังเกตของตัวแปร 1 ตัวแปร ที่มีค่าเปลี่ยนแปลงหรือเคลื่อนไหวไปตามลำดับเวลา ดังนั้น การสร้างตัวแบบอนุกรมเวลาเชิงเดี่ยว ด้วยวิธีการวิเคราะห์ความถดถอย จึงเป็นการวิเคราะห์ความถดถอยอย่างง่าย โดยมีตัวแปรอิสระเป็นเวลา ใช้สัญลักษณ์  $t$  แทนตัวแปรเวลา และ  $t$  มีค่าตั้งแต่ 1 ถึง  $n$  เมื่อ  $n$  คือ จำนวนของค่าสังเกตที่ใช้ในการสร้างตัวแบบอนุกรมเวลา ส่วนตัวแปรตาม คือ ตัวแปรที่มีค่าเป็นค่าสังเกตที่สนใจ เมื่อได้สมการจากวิธีการวิเคราะห์ความถดถอย หรือเรียกว่า สมการถดถอย จะใช้สมการถดถอยพยากรณ์ค่าสังเกตล่วงหน้าได้ตามต้องการ จากที่กล่าวมา สิ่งที่น่าสนใจ คือ การวิเคราะห์ความถดถอยอย่างง่าย ดังนั้น เนื้อหาที่จะกล่าวถึงต่อไปในหัวข้อนี้ เป็นวิธีการวิเคราะห์ความถดถอยอย่างง่าย เท่านั้น

1. รูปแบบการถดถอย เป็นการอธิบายลักษณะความสัมพันธ์ระหว่างตัวแปรอิสระ (เวลา) กับตัวแปรตาม (ค่าสังเกต) โดยรูปแบบการถดถอย ประกอบด้วย ฟังก์ชันความสัมพันธ์และความคลาดเคลื่อน ซึ่งฟังก์ชันความสัมพันธ์ในรูปแบบการถดถอย มีหลายลักษณะ ได้แก่ ฟังก์ชันขนานแกนนอน ฟังก์ชันเชิงเส้น ฟังก์ชันเอกซ์โพเนนเชียล ฟังก์ชันกำลังสอง ฟังก์ชันไฮเปอร์โบล่า ฟังก์ชันเอกซ์โพเนนเชียลผกผัน ฟังก์ชันเอกซ์โพเนนเชียลดัดแปลง ฟังก์ชันกอมเพอร์ส และฟังก์ชัน โลจิสติก จากรูปแบบการถดถอย จะมีพารามิเตอร์ที่ไม่ทราบค่า ใช้สัญลักษณ์  $\beta_j$  ดังนั้น จึงต้องประมาณค่าพารามิเตอร์ที่ไม่ทราบค่า ด้วยตัวประมาณ  $b_j$  ได้เป็นสมการถดถอยที่จะนำไปใช้ในการพยากรณ์ค่าสังเกตล่วงหน้า ซึ่งฟังก์ชันความสัมพันธ์ รูปแบบการถดถอย และสมการถดถอย สามารถสรุปได้ดังตารางที่ 2.1

ตารางที่ 2.1 รูปแบบการถดถอยและสมการถดถอย (ทรงศิริ แต่สมบัติ, 2549, หน้า 76).

ฟังก์ชันความสัมพันธ์	รูปแบบการถดถอย	สมการถดถอย
ขนานแกนนอน (Constant)	$Y = \beta_0 + \varepsilon$	$\hat{Y} = b_0$
เชิงเส้น (Linear)	$Y = \beta_0 + \beta_1 X + \varepsilon$	$\hat{Y} = b_0 + b_1 X$
กำลังสอง (Quadratic)	$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \varepsilon$	$\hat{Y} = b_0 + b_1 X + b_2 X^2$
ไฮเพอร์โบลิก (Hyperbolic)	$Y = \beta_0 + \beta_1 \frac{1}{X} + \varepsilon$	$\hat{Y} = b_0 + b_1 \frac{1}{X}$
เอกซ์โพเนนเชียล (Exponential)	$Y = \beta_0 \beta_1^X + \varepsilon$	$\hat{Y} = b_0 b_1^X$
ยกกำลัง (Polynomial)	$Y = \beta_0 X^{\beta_1} + \varepsilon$	$\hat{Y} = b_0 X^{b_1}$
ไฮเพอร์โบลา (Hyperbola)	$Y = \frac{X}{\beta_0 X + \beta_1} + \varepsilon$	$\hat{Y} = \frac{X}{b_0 X + b_1}$
ไฮเพอร์โบลิกเอกซ์โพเนนเชียล (Hyperbolic exponential)	$Y = \beta_0 e^{\beta_1/X} + \varepsilon$	$\hat{Y} = b_0 e^{b_1/X}$
เอกซ์โพเนนเชียลดัดแปลง (Modified exponential)	$Y = \beta_0 + \beta_1 \beta_2^X + \varepsilon$	$\hat{Y} = b_0 + b_1 b_2^X$
Gompertz	$Y = \beta_0 \beta_1^{\beta_2^X} + \varepsilon$	$\hat{Y} = b_0 b_1^{b_2^X}$
Logistic	$Y = \frac{1}{\beta_0 + \beta_1 \beta_2^X} + \varepsilon$	$\hat{Y} = \frac{1}{b_0 + b_1 b_2^X}$

2. สมการถดถอยเชิงเส้น เป็นสมการถดถอยที่ใช้พยากรณ์ค่าสังเกตล่วงหน้าได้ตามต้องการ โดยการอธิบายวิธีการวิเคราะห์ความถดถอย จะเลือกรูปแบบที่มีฟังก์ชันความสัมพันธ์เชิงเส้น ซึ่งเป็นรูปแบบการถดถอยอย่างง่าย มาใช้ในการอธิบาย โดยรูปแบบการถดถอยเชิงเส้น  $Y = \beta_0 + \beta_1 X + \varepsilon$  มีพารามิเตอร์ที่ไม่ทราบค่า 2 ตัว คือ  $\beta_0$  กับ  $\beta_1$  จึงต้องทำการประมาณค่าพารามิเตอร์ ด้วยตัวประมาณ  $b_0$  กับ  $b_1$  ตามลำดับ ได้เป็นสมการถดถอยเชิงเส้น  $\hat{Y} = b_0 + b_1 X$  ซึ่งตัวประมาณ  $b_0$  กับ  $b_1$  ประมาณค่าด้วยวิธีกำลังสองน้อยที่สุด (Least squared method) โดยประมาณค่า  $b_0$  กับ  $b_1$  ที่ทำให้ผลรวมกำลังสองของความคลาดเคลื่อน (Sum Squared Errors: SSE) มีค่าน้อยที่สุด และทำได้ด้วยการกำหนดอนุพันธ์ย่อยของ SSE เทียบกับ  $b_0$  และ  $b_1$  ให้มีค่าเท่ากับ 0 ดังต่อไปนี้

$$\frac{\partial SSE}{\partial b_0} = \frac{\partial \sum (Y - b_0 - b_1 X)^2}{\partial b_0} = -2 \sum (Y - b_0 - b_1 X) = 0$$

$$\frac{\partial SSE}{\partial b_1} = \frac{\partial \sum (Y - b_0 - b_1 X)^2}{\partial b_1} = -2 \sum X(Y - b_0 - b_1 X) = 0$$

ทำให้ได้สมการหาค่า  $b_0$  และ  $b_1$  ดังสมการที่ 2.11 และ 2.12 ต่อไปนี้

$$b_1 = \frac{\sum xy}{\sum x^2} \quad \text{----- (2.11)}$$

$$b_0 = \bar{Y} - b_1 \bar{X} \quad \text{----- (2.12)}$$

เมื่อ  $\bar{X}$  คือค่าเฉลี่ยของ  $X$ ,  $\bar{Y}$  คือค่าเฉลี่ยของ  $Y$ ,  $x = X - \bar{X}$  และ  $y = Y - \bar{Y}$

ดังนั้น สามารถสร้างสมการถดถอย ที่แสดงความสัมพันธ์ระหว่างตัวแปรอิสระ  $X$  และตัวแปรตาม  $Y$  ด้วยการประมาณค่าตัวประมาณ  $b_0$  และ  $b_1$  ด้วยวิธีกำลังสองน้อยที่สุด นอกจากนี้ การวิเคราะห์ความถดถอยเชิงเส้น สามารถนำไปปรับใช้กับการวิเคราะห์ความถดถอยที่ไม่เป็นเชิงเส้นได้ เช่น รูปแบบไฮเปอร์โบล่า เมื่อไม่รวมค่าคลาดเคลื่อน ( $\epsilon$ ) จะอยู่ในรูป  $Y = \frac{X}{\beta_0 X + \beta_1}$  แปลงเป็น  $\frac{1}{Y} = \beta_0 + \beta_1 \frac{1}{X}$  ถ้าให้  $\frac{1}{Y} = Y'$  และ  $\frac{1}{X} = X'$  จะสามารถปรับรูปแบบสมการได้เป็น  $Y' = \beta_0 + \beta_1 X'$  ซึ่งเป็นสมการเชิงเส้น เป็นต้น

จากที่กล่าวมา สามารถใช้สมการถดถอยเชิงเส้น เป็นตัวแบบการพยากรณ์ค่าสังเกตล่วงหน้าได้ โดยมีตัวแปรอิสระ  $X$  เป็นเวลา และตัวแปรตาม  $Y$  เป็นค่าสังเกต ซึ่งความแม่นยำของการพยากรณ์ ขึ้นกับการเลือกฟังก์ชันความสัมพันธ์ให้สอดคล้องกับรูปแบบความสัมพันธ์ระหว่างค่าสังเกตกับเวลา ถ้ากรณีที่ รูปแบบความสัมพันธ์ไม่เป็นเชิงเส้น สามารถปรับให้อยู่ในรูปแบบเชิงเส้นได้ ตามที่กล่าวไว้ข้างต้น

3. สมมุติฐานการถดถอย เป็นการกำหนดสมมุติฐานของความคลาดเคลื่อน เนื่องจากรูปแบบการถดถอย มีค่าคลาดเคลื่อน ( $\epsilon$ ) เกิดขึ้น ดังนั้น ต้องกำหนดสมมุติฐานของความคลาดเคลื่อน เพราะค่าความคลาดเคลื่อนเป็นตัวอธิบายการแปรเปลี่ยนของตัวแปรตาม ( $Y$ ) ที่สังเกตได้ เมื่อตัวแปรอิสระเป็น  $X$  โดยจะทำการวิเคราะห์ความถดถอย ภายใต้สมมุติฐานการถดถอยที่ตั้งไว้ และสมมุติฐานการถดถอยมี 4 ข้อ ดังนี้ (Bowerman et al., 2005, p.97)

3.1 ณ ค่าของตัวแปรอิสระ  $X$  ใด ๆ ค่าคลาดเคลื่อนของประชากร มีค่าเฉลี่ยเท่ากับ 0

3.2 สมมติฐานความแปรปรวนคงที่ หมายถึง ณ ค่าของตัวแปรอิสระ  $X$  ใด ๆ ค่าคลาดเคลื่อนของประชากร มีความแปรปรวนที่ไม่ขึ้นกับค่าของ  $X$  หรือผลต่างค่าคลาดเคลื่อนของประชากรที่สัมพันธ์กับผลต่างของ  $X$  มีค่าความแปรปรวนเท่ากัน และกำหนดสัญลักษณ์ความแปรปรวนคงที่เป็น  $\sigma^2$

3.3 สมมติฐานการแจกแจงปกติ หมายถึง ณ ค่าของตัวแปรอิสระ  $X$  ใด ๆ ค่าคลาดเคลื่อนของประชากร มีการแจกแจงแบบปกติ

3.4 สมมติฐานความเป็นอิสระ หมายถึง แต่ละค่าความคลาดเคลื่อน มีความเป็นอิสระต่อกัน หรือแต่ละค่าความคลาดเคลื่อน สัมพันธ์กับค่าของตัวแปรตามของมันเท่านั้น และเป็นอิสระจากค่าความคลาดเคลื่อนอื่น

4. ค่าวัดความเหมาะสมของรูปแบบ เป็นตัววัดความเหมาะสมของรูปแบบการถดถอย โดยการวัดความเหมาะสมของรูปแบบการถดถอย สามารถพิจารณาได้จากการวัดค่าสถิติซึ่งแบ่งได้ 2 กลุ่ม ดังนี้

4.1 ค่าสถิติวัดขนาดของค่าคลาดเคลื่อน ได้แก่ ค่าผลรวมกำลังสองของความคลาดเคลื่อน (SSE), ค่าเฉลี่ยกำลังสองของความคลาดเคลื่อน (MSE), รากที่สองของค่าเฉลี่ยกำลังสองของความคลาดเคลื่อน (RMSE) และรูปแบบการถดถอยที่เหมาะสม คือ รูปแบบที่ให้ค่า SSE, MSE และ RMSE น้อยที่สุด โดยค่า SSE, MSE และ RMSE เป็นดังสมการที่ 2.13-2.15

$$SSE = \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad \text{----- (2.13)}$$

เมื่อ  $n$ =จำนวนข้อมูล และ  $\hat{y}_i = b_0 + b_1x_i$

$$MSE = \frac{SSE}{n-2} \quad \text{----- (2.14)}$$

$$RMSE = \sqrt{MSE} \quad \text{----- (2.15)}$$



4.2 ค่าสถิติวัดความผันแปร ได้แก่ ค่าสัมประสิทธิ์ตัวกำหนด ใช้สัญลักษณ์  $R^2$  และค่าสัมประสิทธิ์ตัวกำหนดปรับแล้ว ใช้สัญลักษณ์  $R_a^2$  โดยค่า  $R^2$  เป็นค่าสถิติที่วัดว่าตัวแปรอิสระ  $X$  มีส่วนในการอธิบายความผันแปรของตัวแปรตาม  $Y$  มากน้อยเพียงใด หรือกล่าวได้ว่า ค่า  $R^2$  เป็นตัววัดความแข็งแกร่งของความสัมพันธ์ระหว่าง  $X$  กับ  $Y$  ซึ่งค่า  $R^2$  มีค่าระหว่าง 0 ถึง 1 ถ้าค่า  $R^2$  เข้าใกล้ 1 แสดงว่า ตัวแปร  $X$  สามารถอธิบายความผันแปรของตัวแปร  $Y$  ได้ดี หรือกล่าวได้ว่า รูปแบบการถดถอยมีความเหมาะสมมาก แต่ถ้าค่า  $R^2$  เข้าใกล้ 0 แสดงว่า ตัวแปร  $X$  ไม่สามารถอธิบายความผันแปรของตัวแปร  $Y$  หรือกล่าวได้ว่า รูปแบบการถดถอยไม่มีความเหมาะสม ส่วนค่า  $R_a^2$  เป็นค่าวัดความเหมาะสมของรูปแบบ ที่ได้จากการแบ่งส่วนของความแปรปรวนรวม โดยที่ ความแปรปรวนรวม = ความแปรปรวนที่อธิบายได้ + ความแปรปรวนที่อธิบายไม่ได้ และค่า  $R_a^2$  แปรผกผันกับค่า  $MSE$  ซึ่ง ค่า  $R_a^2$  อาจมีค่าน้อยกว่า 0 ได้ โดยค่า  $R^2$  และ  $R_a^2$  สามารถคำนวณได้ดังสมการที่ 2.16 และ 2.17 ตามลำดับ

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST} \quad \text{----- (2.16)}$$

$$R_a^2 = 1 - \frac{(n-1)SSE}{(n-2)SST} \quad \text{----- (2.17)}$$

เมื่อ  $SST$  คือ ผลรวมกำลังสองของความผันแปร ส่วน  $SSR$  คือ ผลรวมกำลังสองของความผันแปรเนื่องจากการถดถอย และสามารถคำนวณค่า  $SST$  และ  $SSR$  ได้ดังสมการที่ 2.18 และ 2.19 ตามลำดับ

$$SST = \sum_{i=1}^n (Y_i - \bar{Y})^2 \quad \text{----- (2.18)}$$

$$SSR = SST - SSE \quad \text{----- (2.19)}$$

## 2.2.2 วิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล

การปรับให้เรียบแบบเอกซ์โพเนนเชียล (Exponential smoothing) เป็นวิธีการพยากรณ์ที่มีประสิทธิภาพ เมื่อองค์ประกอบของอนุกรมเวลา ได้แก่ แนวโน้ม และฤดูกาล มีการเปลี่ยนแปลงตลอดช่วงเวลาที่สนใจ สำหรับวิธีการนี้ เป็นวิธีการให้น้ำหนักกับค่าสังเกตของอนุกรมเวลา



ที่นำมาใช้ในการพยากรณ์ไม่เท่ากัน โดยให้น้ำหนักค่าสังเกตที่เกิดขึ้นล่าสุด มากที่สุด ส่วนน้ำหนักของค่าสังเกตก่อนหน้า จะมีค่าลดหลั่นไปตามลำดับเวลา หมายความว่า ยิ่งค่าสังเกตล้ำหลังมาก ยิ่งมีน้ำหนักน้อย และน้ำหนักของค่าสังเกต จะลดหลั่นลงแบบเอกซ์โพเนนเชียลตามลำดับเวลา ซึ่งใช้ค่าคงที่ 1 ตัวหรือมากกว่า มากำหนดน้ำหนักให้กับค่าสังเกตแต่ละค่า และเรียกค่าคงที่นี้ว่า ค่าคงที่ปรับให้เรียบ (Smoothing constant)

วิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล มีหลายวิธี แต่ละวิธี มีคุณสมบัติร่วมกัน คือ การให้น้ำหนักค่าสังเกตที่เกิดขึ้นล่าสุด มากที่สุด โดยการให้น้ำหนักค่าสังเกต จะใช้ค่าคงที่ปรับให้เรียบ ที่อาจมี 1 ตัว หรือมากกว่า 1 ตัว ดังนั้น การแบ่งประเภทของวิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล จะแบ่งตามจำนวนของค่าคงที่ปรับให้เรียบ ได้ดังนี้

1. ซิงเกิลเอกซ์โพเนนเชียลสมูทติง (Single exponential smoothing) หรือเรียกโดยย่อว่า SES เป็นวิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียลอย่างง่าย (Simple exponential smoothing) ด้วยการให้น้ำหนักกับค่าสังเกตแต่ละค่า โดยน้ำหนักจะลดหลั่นลงแบบเอกซ์โพเนนเชียลตามเวลาล้ำหลัง ซึ่งค่าสังเกตล่าสุด จะมีน้ำหนักมากที่สุด และการให้น้ำหนักกับค่าสังเกตแต่ละค่า จะใช้ค่าคงที่ปรับให้เรียบ 1 ตัว โดยมีสัญลักษณ์  $\alpha$  (Alpha) และมีสมการสำหรับตัวแบบพยากรณ์ซิงเกิลเอกซ์โพเนนเชียลสมูทติง ดังสมการที่ 2.20 (Prema and Rao, 2015)

$$F_{t+1} = \alpha Y_t + (1 - \alpha)F_t \quad \text{----- (2.20)}$$

เมื่อ  $F_{t+1}$  คือ ค่าพยากรณ์ล่วงหน้า ณ เวลา  $t+1$

$F_t$  คือ ค่าพยากรณ์ ณ เวลา  $t$

$Y_t$  คือ ค่าสังเกต ณ เวลา  $t$

$\alpha$  คือ ค่าคงที่ปรับให้เรียบ มีค่าระหว่าง 0 กับ 1

จากสมการที่ 2.20 แสดงได้ว่า ค่าพยากรณ์ในอนาคต ล่วงหน้า 1 ช่วงเวลา (ณ เวลา  $t+1$ ) ขึ้นอยู่กับค่าสังเกต ณ เวลาปัจจุบัน (ณ เวลา  $t$ ) และค่าพยากรณ์ ณ เวลาปัจจุบัน (ณ เวลา  $t$ ) ที่ได้มาจากการพยากรณ์ในอดีตถอยหลังไป 1 ช่วงเวลา (ณ เวลา  $t-1$ ) และสามารถปรับสมการที่ 2.20 ให้มีลักษณะเป็นกระบวนการเรียกตัวเองแบบอัตโนมัติ ได้ดังสมการที่ 2.21

$$F_{t+1} = \alpha Y_t + \alpha(1 - \alpha)Y_{t-1} + \alpha(1 - \alpha)^2 Y_{t-2} + \dots + \alpha(1 - \alpha)^{t-1} Y_1 + (1 - \alpha)^t F_1 \quad \text{----- (2.21)}$$

จากสมการที่ 2.21 แสดงว่า น้ำหนักของค่าสังเกต ลดลงแบบเอกซ์โพเนนเชียลตามลำดับเวลาแล้ว จึงเรียกรูปแบบนี้ว่า วิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล และแสดงได้ว่า ถ้ากำหนด  $\alpha$  ให้มีค่ามาก (เข้าใกล้ 1) ค่าน้ำหนักของค่าสังเกต จะลดลงเร็ว แต่ถ้ากำหนดค่า  $\alpha$  ให้มีค่าน้อย (เข้าใกล้ 0) ค่าน้ำหนัก จะลดลงช้า นั่นคือ จะกำหนดค่า  $\alpha$  ให้มีค่ามาก เมื่อต้องการถ่วงน้ำหนักให้กับค่าสังเกตล่าสุดมากกว่าค่าก่อนหน้า

2. ดับเบิลเอกซ์โพเนนเชียลสมูทติง (Double exponential smoothing) หรือเรียกโดยย่อว่า DES เป็นวิธีการปรับให้เรียบ ที่นำมาใช้กับข้อมูลที่มีส่วนประกอบของแนวโน้ม ซึ่งสมการพยากรณ์ด้วยวิธี DES คล้ายสมการพยากรณ์ด้วยวิธี SES ต่างกันที่ วิธี DES มีส่วนประกอบ 2 ส่วน คือ ระดับ (Level) และแนวโน้ม (Trend) โดยสามารถแสดงสมการสำหรับส่วนประกอบระดับ ดังสมการที่ 2.22 ส่วนสมการสำหรับส่วนประกอบแนวโน้ม แสดงได้ดังสมการที่ 2.23 และสมการพยากรณ์ด้วยวิธี DES ที่ได้จากส่วนประกอบระดับและแนวโน้ม แสดงได้ดังสมการที่ 2.24 (Prema and Rao, 2015)

$$\text{Level: } L_t = \alpha Y_t + (1 - \alpha)(L_t - b_{t-1}) \quad \text{----- (2.22)}$$

$$\text{Trend: } b_t = \beta(L_t - L_{t-1}) + (1 - \beta)b_{t-1} \quad \text{----- (2.23)}$$

$$\text{Forecast: } F_{t+m} = L_t + b_{tm} \quad \text{----- (2.24)}$$

เมื่อ  $\alpha$  คือ ค่าคงที่ปรับให้เรียบของส่วนประกอบระดับ มีค่าระหว่าง 0 ถึง 1

$\beta$  คือ ค่าคงที่ปรับให้เรียบของส่วนประกอบแนวโน้ม มีค่าระหว่าง 0 ถึง 1

$L_t$  คือ ค่าประมาณระดับของอนุกรมเวลา ณ เวลา  $t$

$Y_t$  คือ ค่าสังเกต ณ เวลา  $t$

$b_t$  คือ ค่าประมาณความชันของอนุกรมเวลา ณ เวลา  $t$

$m$  คือ จำนวนคาบเวลาล่วงหน้าที่ต้องการพยากรณ์

$F_{t+m}$  คือ ค่าพยากรณ์ล่วงหน้า  $m$  คาบเวลา

3. ทริปเปิลเอกซ์โพเนนเชียลสมูทติง (Triple exponential smoothing) หรือเรียกโดยย่อว่า TES เป็นวิธีการปรับให้เรียบที่นำมาใช้สร้างตัวแบบการพยากรณ์กับข้อมูลที่มีส่วนประกอบของแนวโน้ม และการเปลี่ยนแปลงตามฤดูกาล ซึ่งบางครั้งเรียกวิธีการนี้ว่า วิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียลของโฮลท์และวินเทอร์ (Holt-Winters exponential smoothing)

ในการจัดการกับการเปลี่ยนแปลงตามฤดูกาล จะต้องเพิ่มพารามิเตอร์อีก 1 ตัว ดังนั้น เมื่อเทียบกับวิธี DES ทำให้วิธี TES มีสมการเพิ่มขึ้น 1 สมการ เพื่อจัดการกับส่วนประกอบฤดูกาล และสมการทั้งหมด สำหรับพยากรณ์ค่าสังเกตในอนาคต ด้วยวิธีการปรับให้เรียบแบบ ทริปเปิลเอกซ์โพเนนเชียลสมูทติง แสดงได้ดังสมการที่ 2.25-2.28 (Prema and Rao, 2015)

$$\text{Level: } L_t = \alpha(Y_t - S_{t-s}) + (1 - \alpha)(L_{t-1} + b_{t-1}) \quad \text{----- (2.25)}$$

$$\text{Trend: } b_t = \beta(L_t - L_{t-1}) + (1 - \beta)b_{t-1} \quad \text{----- (2.26)}$$

$$\text{Seasonal: } S_t = \gamma(Y_{t-1} - L_t) + (1 - \gamma)S_{t-s} \quad \text{----- (2.27)}$$

$$\text{Forecast: } F_{t+m} = L_t + b_{tm} + S_{t-s+m} \quad \text{----- (2.28)}$$

เมื่อ  $\alpha$  คือ ค่าคงที่ปรับให้เรียบของส่วนประกอบระดับ มีค่าระหว่าง 0 ถึง 1

0 ถึง 1

$\beta$  คือ ค่าคงที่ปรับให้เรียบของส่วนประกอบแนวโน้ม มีค่าระหว่าง

0 ถึง 1

$\gamma$  คือ ค่าคงที่ปรับให้เรียบของส่วนประกอบฤดูกาล มีค่าระหว่าง

$L_t$  คือ ค่าประมาณระดับของอนุกรมเวลา ณ เวลา  $t$

$Y_t$  คือ ค่าสังเกต ณ เวลา  $t$

$b_t$  คือ ค่าประมาณความชันของอนุกรมเวลา ณ เวลา  $t$

$S_t$  คือ ส่วนประกอบฤดูกาล

$S$  คือ ความยาวของแต่ละฤดูกาล เช่น ฤดูกาลละ 12 เดือน ( $S = 12$ )

$m$  คือ จำนวนคาบเวลาล่วงหน้าที่ต้องการพยากรณ์

$F_{t+m}$  คือ ค่าพยากรณ์ล่วงหน้า  $m$  คาบเวลา

### 2.2.3 วิธีการของบ็อกซ์และเจนกินส์

วิธีการของบ็อกซ์และเจนกินส์ (Box and Jenkins method) เป็นวิธีการวิเคราะห์อนุกรมเวลาที่เกี่ยวข้องกระบวนการเชิงสุ่ม (Stochastic process) โดยถือว่าข้อมูลที่เกิดขึ้นตามเวลาที่เปลี่ยนแปลงไป มีลักษณะการเกิดค่าของข้อมูลที่เป็นไปตามกฎความน่าจะเป็น ซึ่งวิธีของบ็อกซ์และเจนกินส์ จะพยากรณ์ข้อมูลจากรูปแบบหลัก 2 รูปแบบ ได้แก่ รูปแบบ Autoregressive (AR) คือรูปแบบที่กำหนดว่า ค่าพยากรณ์ที่เวลาใด ๆ ขึ้นอยู่กับค่าสังเกตก่อนหน้า และรูปแบบ Moving Average (MA) คือ รูปแบบที่กำหนดว่า ค่าพยากรณ์ที่เวลาใด ๆ ขึ้นอยู่กับค่าคลาดเคลื่อนก่อนหน้า โดยเรียกรูปแบบที่เกิดจากรูปแบบ AR ผสานกับรูปแบบ MA ได้ว่า รูปแบบ ARMA ซึ่งเป็นรูปแบบที่อธิบายลักษณะการเคลื่อนไหวของอนุกรมเวลาที่ค่าสังเกตมีสหสัมพันธ์กัน และมีความคงตัว หรือเรียกว่า สเตชันนารี แต่โดยทั่วไป อนุกรมเวลาที่เก็บรวบรวมมา จะมีการเคลื่อนไหวเนื่องจากแนวโน้ม หรือฤดูกาล ซึ่งเป็นลักษณะของอนุกรมเวลาที่ไม่เป็นสเตชันนารี ดังนั้น ก่อนนำอนุกรมเวลาที่ไม่เป็นสเตชันนารีมาวิเคราะห์ด้วยรูปแบบ ARMA จะต้องแปลงอนุกรมเวลาให้เป็นสเตชันนารี และเรียกรูปแบบที่ใช้วิเคราะห์อนุกรมเวลาที่แปลงให้เป็นสเตชันนารี ได้ว่า รูปแบบ ARIMA (Autoregressive Integrated Moving Average) และรายละเอียดในแต่ละรูปแบบ อธิบายได้ดังต่อไปนี้

1. รูปแบบ AR เป็นรูปแบบที่กำหนดว่า ค่าสังเกต  $Y_t$  ที่เวลา  $t$  ใด ๆ ขึ้นอยู่กับค่าสังเกตก่อนหน้าที่เวลา  $t-1, t-2, \dots, t-p$  โดยรูปแบบ AR ต้องระบุอันดับ  $p$  เพื่อใช้กำหนดสมการความสัมพันธ์ตามรูปแบบ AR ได้ว่า ค่าสังเกต  $Y_t$  ขึ้นอยู่กับค่าสังเกตก่อนหน้าที่ล่าช้าไป  $p$  คาบเวลา ดังนั้น ที่อันดับ  $p$  ใด ๆ สามารถกำหนดสมการตามรูปแบบ AR( $p$ ) ได้ดังนี้ (ทรงศิริ แต่สมบัติ, 2549, หน้า 252)

$$Y_t = \theta_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \varepsilon_t \quad \text{----- (2.29)}$$

เมื่อ  $Y_t$  คือค่าสังเกตที่เวลา  $t$  ใด ๆ  $\theta_0$  คือ เทอมที่เป็นค่าคงที่  $\phi_1, \phi_2, \dots, \phi_p$  คือเซตของพารามิเตอร์ถ่วงน้ำหนัก ในรูปแบบ AR(p) และ  $\varepsilon_t$  คือ ค่าคลาดเคลื่อน ณ เวลา  $t$  โดยทั่วไป รูปแบบ AR(p) จะมีอันดับ  $p$  เท่ากับ 1 หรือ 2 ดังนั้น สำหรับรูปแบบ AR(1) และ AR(2) เป็นดังนี้

$$Y_t = \theta_0 + \phi_1 Y_{t-1} + \varepsilon_t \quad \text{เมื่อ } p = 1$$

$$Y_t = \theta_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \varepsilon_t \quad \text{เมื่อ } p = 2$$

ถ้ากำหนดตัวดำเนินการย้อนกลับ (Backward shift operator) ใช้สัญลักษณ์  $B$  โดยที่  $B^p Y_t = Y_{t-p}$  จะสามารถเขียนสมการที่ 2.29 ในรูปแบบของตัวดำเนินการย้อนกลับ ได้ดังสมการที่ 2.30

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) Y_t = \theta_0 + \varepsilon_t \quad \text{----- (2.30)}$$

สมการ  $1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p = 0$  เป็นสมการโพลิโนเมียลลำดับที่  $p$  ของ  $B$  และคำตอบของสมการเป็นค่าของ  $B$  ที่ได้จากการแก้สมการจำนวน  $p$  ค่า โดยเลือกใช้ค่า  $B$  เพียงค่าเดียว เฉพาะค่า  $|B| > 1$  และเรียกเงื่อนไขสำหรับค่าพารามิเตอร์  $\phi_1, \phi_2, \dots, \phi_p$  ที่ทำให้  $|B| > 1$  ว่าเงื่อนไขของเสถียร (Stationary) ตัวอย่างเช่น สำหรับรูปแบบ AR(1) คำตอบของสมการ  $1 - \phi_1 B = 0$  ได้แก่  $B = 1/\phi_1$  หรือ  $|\phi_1| < 1$  เป็นต้น

2. รูปแบบ MA เป็นรูปแบบที่กำหนดว่า ค่าสังเกต  $Y_t$  ที่เวลา  $t$  ใด ๆ ขึ้นอยู่กับค่าคลาดเคลื่อนก่อนหน้าเป็นเวลา  $t-1, t-2, \dots, t-q$  โดยรูปแบบ MA ต้องระบุอันดับ  $q$  เพื่อใช้กำหนดสมการความสัมพันธ์ตามรูปแบบ MA ได้ว่า ค่าสังเกต  $Y_t$  ขึ้นอยู่กับค่าคลาดเคลื่อนก่อนหน้าที่ล่าช้าไป  $q$  คาบเวลา ดังนั้น ที่อันดับ  $q$  ใด ๆ สามารถกำหนดสมการตามรูปแบบ MA(q) ได้ดังนี้

$$Y_t = \theta_0 + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \dots - \theta_q \varepsilon_{t-q} \quad \text{----- (2.31)}$$

เมื่อ  $Y_t$  คือค่าสังเกตที่เวลา  $t$  ใด ๆ  $\theta_0$  คือ เทอมที่เป็นค่าคงที่  $\theta_1, \theta_2, \dots, \theta_q$  คือเซตของพารามิเตอร์ถ่วงน้ำหนัก ในรูปแบบ MA(q) และ  $\varepsilon_t$  คือ ค่าคลาดเคลื่อน ณ เวลา  $t$  โดยทั่วไป รูปแบบ MA(q) จะมีอันดับ  $q$  เท่ากับ 1 หรือ 2 ดังนั้น สำหรับรูปแบบ MA(1) และ MA(2) เป็นดังนี้

$$Y_t = \theta_0 + \varepsilon_t - \theta_1 \varepsilon_{t-1} \quad \text{เมื่อ } q = 1$$

$$Y_t = \theta_0 + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} \quad \text{เมื่อ } q = 2$$

ถ้ากำหนดตัวดำเนินการย้อนกลับ  $B$  เช่นเดียวกับรูปแบบ AR(p) จะสามารถเขียนสมการที่ 2.31 ในรูปแบบของตัวดำเนินการย้อนกลับ ได้ดังสมการที่ 2.32 ดังนี้

$$Y_t = \theta_0 + (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q) \varepsilon_t \quad \text{----- (2.32)}$$

สมการ  $1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q = 0$  เป็นสมการโพลิโนเมียลลำดับที่  $q$  ของ  $B$  ดังนั้น การหาคำตอบของสมการ ซึ่งเป็นค่าของ  $B$  ทำได้เช่นเดียวกับรูปแบบ AR(p) โดยมีเงื่อนไขว่า  $|B| > 1$  และเรียกเงื่อนไขสำหรับค่าพารามิเตอร์  $\theta_1, \theta_2, \dots, \theta_q$  ที่ทำให้  $|B| > 1$  ว่าเงื่อนไขของอินเวอร์ติเบิล (Invertible) ตัวอย่างเช่น สำหรับรูปแบบ MA(1) คำตอบของสมการ  $1 - \theta_1 B = 0$  ได้แก่  $B = 1/\theta_1$  หรือ  $|\theta_1| < 1$  เป็นต้น

3. รูปแบบ ARMA เป็นรูปแบบที่นำรูปแบบ AR(p) ผสานกับรูปแบบ MA(q) สร้างเป็นรูปแบบ ARMA(p, q) เมื่อกำหนดอันดับ  $p$  และ  $q$  สามารถกำหนดสมการตามรูปแบบ ARMA(p, q) ได้ดังสมการที่ 2.33

$$Y_t = \theta_0 + \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \quad \text{----- (2.33)}$$

โดยทั่วไป รูปแบบ ARMA(p, q) จะมีอันดับ  $p + q \leq 2$  เช่น รูปแบบ ARMA(1, 1) ที่มีรูปแบบ ดังต่อไปนี้

$$Y_t = \theta_0 + \phi_1 Y_{t-1} + \varepsilon_t - \theta_1 \varepsilon_{t-1} \quad \text{เมื่อ } p = 1, q = 1$$

เมื่อกำหนดตัวดำเนินการย้อนกลับ B จะสามารถเขียนสมการที่ 2.33 ในรูปแบบของตัวดำเนินการย้อนกลับ ได้ดังสมการที่ 2.34

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) Y_t = \theta_0 + (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q) \varepsilon_t \quad \text{----- (2.34)}$$

4. รูปแบบ ARIMA เป็นรูปแบบที่ใช้กับอนุกรมเวลาที่ไม่เป็นสเตชันนารี เนื่องจากแนวโน้ม เพราะรูปแบบ ARMA(p, q) เป็นรูปแบบที่ใช้กับอนุกรมเวลาที่เป็นสเตชันนารี เท่านั้น แต่ถ้าอนุกรมเวลาไม่เป็นสเตชันนารี เนื่องจากแนวโน้ม ต้องแปลงอนุกรมเวลาเดิม  $\{Y_t\}$  ที่ไม่เป็นสเตชันนารี ให้เป็นอนุกรมเวลาใหม่  $\{Z_t\}$  ที่เป็นสเตชันนารี ด้วยการหาผลต่างระหว่างค่าสังเกตในอนุกรมเวลาเดิม นั่นคือ  $Z_t = \nabla^d Y_t$  เมื่อ  $d =$  ระยะห่างของคาบเวลาที่ทำมาหาผลต่างระหว่างค่าสังเกต เช่น อนุกรมเวลาที่ไม่เป็นสเตชันนารี เนื่องจาก มีแนวโน้มเป็นเส้นตรง จะมีค่า  $d = 1$  และได้  $Z_t = \nabla Y_t = Y_t - Y_{t-1}$  หรืออนุกรมเวลาที่ไม่เป็นสเตชันนารี เนื่องจากมีแนวโน้มกำลังสอง จะมีค่า  $d = 2$  และได้  $Z_t = \nabla^2 Y_t = Y_t - 2Y_{t-1} + Y_{t-2}$  เป็นต้น เมื่อแปลงอนุกรมเวลาให้เป็นสเตชันนารี ดังนั้น อนุกรมเวลา  $\{Z_t\}$  จะมีรูปแบบ ARMA(p, q) หรือใช้สัญลักษณ์ว่า  $Z_t \sim \text{ARMA}(p, q)$  และสามารถใช้รูปแบบ ARMA(p, q) สร้างสมการพยากรณ์ให้กับอนุกรมเวลาใหม่  $\{Z_t\}$  แต่โดยทั่วไป การนำเสนอสมการพยากรณ์ จะเป็นสมการพยากรณ์ของอนุกรมเวลาเดิม  $\{Y_t\}$  ซึ่งกำหนดได้ด้วยการใช้รูปแบบ ARIMA(p, d, q) เมื่อ  $p$  คือ อันดับของรูปแบบ AR ส่วน  $q$  คือ อันดับของรูปแบบ MA และ  $d$  คือ ระยะห่างของคาบเวลาที่ทำมาหาผลต่างระหว่างค่าสังเกต โดยสามารถกำหนดสมการตามรูปแบบ ARIMA(p, d, q) ได้ดังสมการที่ 2.35 (Box et al., 2008, p.100)

$$\varphi(B) Y_t = \theta(B) \nabla^d Y_t = \theta_0 + \theta(B) \varepsilon_t \quad \text{----- (2.35)}$$

$$\begin{aligned} \text{เมื่อ} \quad \phi(B) &= 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \\ \theta(B) &= 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q \\ \theta_0 &= \text{ค่าคงที่} \end{aligned}$$

จากสมการที่ 2.35 แต่ละเทอมของสมการ มีความหมายดังนี้

1.  $\phi(B)$  เรียกว่า ตัวดำเนินการออโตรีเกรสซีฟ (Autoregressive operator) เมื่ออนุกรมเวลาเป็นสเตชันนารี นั่นคือ รากของสมการ  $\phi(B) = 0$

2.  $\varphi(B) = \phi(B)\nabla^d$  เรียกว่า ตัวดำเนินการเจนเนอรัลไลซ์ออโตรีเกรสซีฟ (Generalized autoregressive operator) เป็นตัวดำเนินการไม่เป็นสเตชันนารี เมื่อ  $d$  คือ ระยะห่างของคาบเวลาที่นำมาหาผลต่างระหว่างค่าสังเกต ที่ทำให้รากของสมการ  $\varphi(B) = 0$

3.  $\theta(B)$  เรียกว่า ตัวดำเนินการมูฟวิงเอเวอร์เรจ (Moving average operator) เมื่ออนุกรมเวลาเป็นอินเวอร์ติเบิล นั่นคือ รากของสมการ  $\theta(B) = 0$

เมื่อ  $d=0$  ตัวแบบตามสมการที่ 2.35 จะเป็นกระบวนการสเตชันนารี หมายความว่า อนุกรมเวลาเป็นสเตชันนารีและอินเวอร์ติเบิล นอกจากนี้ เมื่อ  $d=0$  รูปแบบ ARIMA(p, d, q) ดังสมการที่ 2.35 จะมีรูปแบบเหมือนกับรูปแบบ ARMA(p, q) ดังสมการที่ 2.34 หรือสมการที่ 2.35 เท่ากับสมการที่ 2.34 เมื่อ  $d=0$  และเพื่อให้เห็นความสัมพันธ์ของรูปแบบ ARMA(p, q) ของอนุกรมเวลาใหม่  $\{Z_t\}$  ที่แปลงให้เป็นสเตชันนารี กับรูปแบบ ARIMA(p, d, q) ของอนุกรมเวลาเดิม  $\{Y_t\}$  ที่ไม่เป็นสเตชันนารี ด้วยการแสดงตัวอย่างของรูปแบบ ARMA(p, q) ของ  $\{Z_t\}$  กับรูปแบบ ARIMA(p, d, q) ของ  $\{Y_t\}$  ที่สัมพันธ์กัน เมื่อ  $Z_t = \nabla Y_t = Y_t - Y_{t-1}$  หรือ  $d=1$  และ  $p+q \leq 2$  โดยไม่อยู่ในรูปของตัวดำเนินการย้อนกลับ ดังตารางที่ 2.2



ตารางที่ 2.2 รูปแบบ AR(p, q) ของ  $\{Z_t\}$  และรูปแบบ ARIMA(p, d, q) ของ  $\{Y_t\}$

(ทรงศิริ แต่สมบัติ, 2549, หน้า 284).

$Z_t$		$Y_t$	
รูปแบบ	รูปแบบ ARMA(p, q) ของ $\{Z_t\}$	รูปแบบ	รูปแบบ ARIMA(p, d, q) ของ $\{Y_t\}$
ARMA(0,0) หรือ White noise	$Z_t = \theta_0 + \varepsilon_t$	ARIMA(0,1,0) หรือ Random walk	$Y_t = \theta_0 + Y_{t-1} + \varepsilon_t$
AR(1)	$Z_t = \theta_0 + \phi_1 Z_{t-1} + \varepsilon_t$	ARI(1,1)	$Y_t = \theta_0 + (1 + \phi_1)Y_{t-1} - \phi_1 Y_{t-2} + \varepsilon_t$
AR(2)	$Z_t = \theta_0 + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \varepsilon_t$	ARI(2,1)	$Y_t = \theta_0 + (1 + \phi_1)Y_{t-1} + (\phi_2 - \phi_1)Y_{t-2} - \phi_2 Y_{t-3} + \varepsilon_t$
MA(1)	$Z_t = \theta_0 + \varepsilon_t - \theta_1 \varepsilon_{t-1}$	IMA(1,1)	$Y_t = \theta_0 + Y_{t-1} + \varepsilon_t - \theta_1 \varepsilon_{t-1}$
MA(2)	$Z_t = \theta_0 + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2}$	IMA(1,2)	$Y_t = \theta_0 + Y_{t-1} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2}$
ARMA(1,1)	$Z_t = \theta_0 + \phi_1 Z_{t-1} + \varepsilon_t - \theta_1 \varepsilon_{t-1}$	ARIMA(1,1,1)	$Y_t = \theta_0 + (1 + \phi_1)Y_{t-1} - \phi_1 Y_{t-2} + \varepsilon_t - \theta_1 \varepsilon_{t-1}$

5. รูปแบบ SARMA เป็นรูปแบบที่ใช้สำหรับอนุกรมเวลาที่มีการเคลื่อนไหวเนื่องจากฤดูกาล ตัวอย่างเช่น ข้อมูลปริมาณการใช้ไฟฟ้ารายเดือน โดยเป็นอนุกรมเวลาที่มีค่าสังเกตเป็นปริมาณการใช้ไฟฟ้าประจำเดือน ซึ่งปริมาณการใช้ไฟฟ้าเดือนเดียวกันของทุกปี มีค่าคล้ายคลึงกัน (ปริมาณการใช้ไฟฟ้าเดือนเมษายนจะมีค่าสูงกว่าค่าเฉลี่ยในทุก ๆ ปี) ลักษณะเช่นนี้ แสดงให้เห็นว่าข้อมูลมีฤดูกาล โดยมีคาบเวลาของฤดูกาลเท่ากับ 12 เดือน หรือข้อมูลปริมาณน้ำฝนรายไตรมาส ก็มีลักษณะการเคลื่อนไหวเนื่องจากฤดูกาล โดยมีคาบเวลาของฤดูกาลเท่ากับ 4 ไตรมาส เป็นต้น และถ้าให้ S หมายถึง จำนวนคาบเวลาใน 1 ฤดูกาล จะกำหนดรูปแบบ SARMA ด้วยรูปแบบ SARMA(P, Q)<sub>s</sub> โดยมีส่วนประกอบ 2 ส่วน ได้แก่ ส่วนที่ 1 คือ รูปแบบ AR(P)<sub>s</sub> ที่ค่าสังเกต  $Y_t$  ขึ้นอยู่กับค่าสังเกตก่อนหน้าที่ล่าหลังไป P\*S คาบเวลา เมื่อ อันดับ P หมายถึง จำนวนฤดูกาลล่าหลัง

เช่น ถ้า  $S=12$  และ  $P=1$  ค่าสังเกต  $Y_t$  ขึ้นอยู่กับค่าสังเกตก่อนหน้าที่ล่าช้าไป 12 คาบเวลา หรือ  $Y_t$  ขึ้นอยู่กับ  $Y_{t-12}$  เป็นต้น สำหรับส่วนที่ 2 คือ รูปแบบ  $MA(Q)_S$  ที่ค่าสังเกต  $Y_t$  ขึ้นอยู่กับค่าคลาดเคลื่อนก่อนหน้าที่ล่าช้าไป  $Q \cdot S$  คาบเวลา เมื่อ อันดับ  $Q$  หมายถึง จำนวนฤดูกาลล่าช้า เช่น ถ้า  $S=12$  และ  $Q=1$  ค่าสังเกต  $Y_t$  ขึ้นอยู่กับค่าคลาดเคลื่อนก่อนหน้าที่ล่าช้าไป 12 คาบเวลา หรือ  $Y_t$  ขึ้นอยู่กับ  $\varepsilon_{t-12}$  เป็นต้น ดังนั้น สามารถกำหนดรูปแบบ  $SARMA(P, Q)_S$  ในรูปของตัวดำเนินการย้อนกลับ ได้ดังสมการที่ 2.36 ดังนี้ (ทรงศิริ แต่สมบัติ, 2549, หน้า 282)

$$\phi_P(B^S)Y_t = \theta_0 + \Theta_Q(B^S)\varepsilon_t \quad \text{----- (2.36)}$$

$$\begin{aligned} \text{เมื่อ } \phi_P(B^S) &= 1 - \phi_S B^S - \phi_{2S} B^{2S} - \dots - \phi_{PS} B^{PS} \\ \Theta_Q(B^S) &= 1 - \Theta_S B^S - \Theta_{2S} B^{2S} - \dots - \Theta_{QS} B^{QS} \\ \theta_0 &= \text{ค่าคงที่} \end{aligned}$$

ตัวอย่างเช่น รูปแบบ  $SARMA(1,1)_{12}$  สามารถกำหนดเป็นตัวแบบ ได้ดังนี้

$$(1 - \phi_{12} B^{12})Y_t = \theta_0 + (1 - \Theta_{12} B^{12})\varepsilon_t$$

และกำหนดตัวแบบในรูปแบบเต็ม โดยไม่มีตัวดำเนินการย้อนกลับ ได้ดังนี้

$$Y_t = \theta_0 + \phi_{12} Y_{t-12} + \varepsilon_t - \Theta_{12} \varepsilon_{t-12}$$

สำหรับอนุกรมเวลาที่มีรูปแบบ  $ARMA(p, q)$  และมีการเคลื่อนไหวเนื่องจากฤดูกาล (มีรูปแบบ  $SARMA(P, Q)_S$ ) สามารถนำรูปแบบทั้ง 2 มาผสมกันด้วยรูปแบบการคูณ เป็นรูปแบบ  $ARMA(p, q) \times (P, Q)_S$  โดยมีรูปแบบทั่วไปเป็นดังสมการที่ 2.37

$$\phi(B)\phi_P(B^S)Y_t = \theta_0 + \theta(B)\Theta_Q(B^S)\varepsilon_t \quad \text{----- (2.37)}$$

$$\begin{aligned} \text{เมื่อ } \phi(B) &= 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \\ \theta(B) &= 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q \end{aligned}$$

$$\begin{aligned}\phi_p(B^S) &= 1 - \phi_S B^S - \phi_{2S} B^{2S} - \dots - \phi_{pS} B^{pS} \\ \Theta_Q(B^S) &= 1 - \Theta_S B^S - \Theta_{2S} B^{2S} - \dots - \Theta_{QS} B^{QS} \\ \theta_0 &= \text{ค่าคงที่}\end{aligned}$$

ตัวอย่างเช่น รูปแบบ ARMA(1,1)<sub>x(1,1)</sub><sub>12</sub> สามารถกำหนดเป็นตัวแบบ ได้ดังนี้

$$(1 - \phi_1 B)(1 - \phi_{12} B^{12})Y_t = \theta_0 + (1 - \theta_1 B)(1 - \Theta_{12} B^{12})\varepsilon_t$$

และกำหนดตัวแบบในรูปแบบเต็ม โดยไม่มีตัวดำเนินการย้อนกลับ ได้ดังนี้

$$Y_t = \theta_0 + \phi_{12} Y_{t-12} + \phi_1 Y_{t-1} - \phi_1 \phi_{12} Y_{t-13} + \varepsilon_t - \Theta_{12} \varepsilon_{t-12} - \theta_1 \varepsilon_{t-1} + \phi_1 \phi_{12} \varepsilon_{t-13}$$

6. รูปแบบ SARIMA เป็นรูปแบบที่ใช้สำหรับอนุกรมเวลาที่ไม่เป็นสเตชันนารี เนื่องจากฤดูกาล โดยต้องแปลงอนุกรมเวลาเดิม  $\{Y_t\}$  ที่ไม่เป็นสเตชันนารี ให้เป็นอนุกรมเวลาใหม่  $\{Z_t\}$  ที่เป็นสเตชันนารี ด้วยการหาผลต่างฤดูกาล ซึ่ง  $Z_t = \nabla_S^D Y_t = (1 - B^{DS})Y_t$  เมื่อ D คือ จำนวนครั้งของการหาผลต่างฤดูกาล เช่น เมื่อ D=1 และ S=12 จะได้  $Z_t = \nabla_{12} Y_t = (1 - B^{12})Y_t = Y_t - Y_{t-12}$  เป็นต้น โดยสามารถกำหนดรูปแบบ SARIMA ด้วยรูปแบบ SARIMA(P, D, Q)<sub>S</sub> ในรูปของตัวดำเนินการย้อนกลับ ได้ดังสมการที่ 2.38 ดังนี้ (ทรงศิริ แต่สมบัติ, 2549, หน้า 286)

$$\phi_p(B^S)(1 - B^{DS})Y_t = \theta_0 + \Theta_Q(B^S)\varepsilon_t \quad \text{----- (2.38)}$$

$$\begin{aligned}\text{เมื่อ} \quad \phi_p(B^S) &= 1 - \phi_S B^S - \phi_{2S} B^{2S} - \dots - \phi_{pS} B^{pS} \\ \Theta_Q(B^S) &= 1 - \Theta_S B^S - \Theta_{2S} B^{2S} - \dots - \Theta_{QS} B^{QS} \\ \theta_0 &= \text{ค่าคงที่}\end{aligned}$$

ตัวอย่างเช่น รูปแบบ SARIMA(1,1,1)<sub>12</sub> สามารถกำหนดเป็นตัวแบบ ได้ดังนี้

$$(1 - \phi_{12} B^{12})(1 - B^{12})Y_t = \theta_0 + (1 - \Theta_{12} B^{12})\varepsilon_t$$

หรือ

$$(1 - B^{12} - \phi_{12}B^{12} + \phi_{12}B^{24})Y_t = \theta_0 + (1 - \Theta_{12}B^{12})\varepsilon_t$$

และกำหนดตัวแบบในรูปแบบเต็ม โดยไม่มีตัวดำเนินการย้อนกลับ ได้ดังนี้

$$Y_t = \theta_0 + (1 + \phi_{12})Y_{t-12} - \phi_{12}Y_{t-24} + \varepsilon_t - \Theta_{12}\varepsilon_{t-12}$$

สำหรับอนุกรมเวลาที่มีรูปแบบ ARIMA(p, d, q) และรูปแบบ SARIMA(P, D, Q)<sub>s</sub> สามารถนำรูปแบบทั้ง 2 มาผสานกันด้วยรูปแบบการคูณ เป็นรูปแบบ ARIMA(p, d, q)x(P, D, Q)<sub>s</sub> โดยมีรูปแบบทั่วไปที่อยู่ในรูปของตัวดำเนินการย้อนกลับ และเพิ่มเทอมของค่าคงที่  $\theta_0$  เป็นดังสมการที่ 2.39 (Box, Jenkins and Reinsel, 2008, p.358) ดังนี้

$$\phi(B)\phi_p(B^S)\nabla^d\nabla_S^D Y_t = \theta_0 + \theta(B)\Theta_Q(B^S)\varepsilon_t \quad \text{----- (2.39)}$$

หรือ

$$\phi(B)\phi_p(B^S)(1 - B^d)(1 - B^{DS})Y_t = \theta_0 + \theta(B)\Theta_Q(B^S)\varepsilon_t$$

$$\text{เมื่อ } \phi(B) = 1 - \phi_1B - \phi_2B^2 - \dots - \phi_pB^p$$

$$\theta(B) = 1 - \theta_1B - \theta_2B^2 - \dots - \theta_qB^q$$

$$\phi_p(B^S) = 1 - \phi_S B^S - \phi_{2S} B^{2S} - \dots - \phi_{pS} B^{pS}$$

$$\Theta_Q(B^S) = 1 - \Theta_S B^S - \Theta_{2S} B^{2S} - \dots - \Theta_{QS} B^{QS}$$

$$\theta_0 = \text{ค่าคงที่}$$

ตัวอย่างเช่น รูปแบบ ARIMA(0,1,1)x(0,1,1)<sub>12</sub> สามารถกำหนดเป็นตัวแบบ ดังนี้

$$\nabla\nabla_{12}Y_t = \theta_0 + (1 - \theta_1B)(1 - \Theta_{12}B^{12})\varepsilon_t$$

หรือ

$$(1 - B)(1 - B^{12})Y_t = \theta_0 + (1 - \theta_1 B)(1 - \Theta_{12} B^{12})\varepsilon_t$$

และกำหนดตัวแบบในรูปแบบเต็ม โดยไม่มีตัวดำเนินการย้อนกลับ ดังนี้

$$Y_t - Y_{t-1} - Y_{t-12} + Y_{t-13} = \theta_0 + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \Theta_{12} \varepsilon_{t-12} + \phi_1 \phi_{12} \varepsilon_{t-13}$$

7. ขั้นตอนของวิธีการของบ็อกซ์และเจนกินส์ เป็นการนำวิธีการของบ็อกซ์และเจนกินส์ ไปใช้สำหรับการพยากรณ์ข้อมูล โดยมี 4 ขั้นตอน แต่ละขั้นตอนมีรายละเอียดดังต่อไปนี้ (Tseng and Tzeng, 2002)

ขั้นตอนที่ 1 การกำหนดรูปแบบ ARIMA(p, d, q)x(P, D, Q)<sub>s</sub> เป็นขั้นตอนที่ใช้เลือกอันดับ p, d, q และอันดับ P, D, Q (ในกรณีที่อนุกรมเวลามีการเคลื่อนไหวเนื่องจากฤดูกาล) โดยจะต้องทำการทดสอบความไม่เป็นสเตรชันนารีเนื่องจากแนวโน้ม เพื่อกำหนดอันดับ d และความไม่เป็นสเตรชันนารีเนื่องจากฤดูกาล เพื่อกำหนดอันดับ D ก่อนที่จะเลือกอันดับ p, q และอันดับ P, Q จากนั้น เลือกอันดับ p, q และอันดับ P, Q ด้วยการใชฟังค์ชันออโตคอร์รีเลชัน (Autocorrelation function: ACF) และ ฟังก์ชันพาร์เชียลออโตคอร์รีเลชัน (Partial autocorrelation function: PACF) โดยทำการสร้างกราฟออโตคอร์รีเลชัน และกราฟพาร์เชียลออโตคอร์รีเลชัน จากนั้นพิจารณารูปร่างของกราฟ เพื่อเลือกอันดับ p, q และอันดับ P, Q สำหรับกำหนดรูปแบบ ARIMA ซึ่งแนวทางการเลือกอันดับ p, q และอันดับ P, Q สามารถสรุปได้ ดังตารางที่ 2.3

ตารางที่ 2.3 ลักษณะรูปร่างของกราฟ ACF และกราฟ PACF สำหรับกำหนดรูปแบบ ARIMA (พงศ์เทพ วิวรรณเดชะ และสมชาย จาคศรี, 2550, หน้า 18).

รูปร่างของกราฟ ACF และกราฟ PACF	รูปแบบ ARIMA
ACF - ค่อย ๆ ลดลงแบบ Exponential PACF - ยอดที่ lag 1 และไม่พบสหสัมพันธ์ที่ lag อื่น	อันดับ Autoregressive(p) = 1
ACF - มีรูปร่างแบบ Sine wave หรือค่อย ๆ ลดลงแบบ Exponential PACF - ยอดที่ lag 1 และ 2 และไม่พบสหสัมพันธ์ที่ lag อื่น	อันดับ Autoregressive(p) = 2
ACF - ยอดที่ lag 1 และไม่พบสหสัมพันธ์ที่ lag อื่น PACF - ค่อย ๆ ลดลงแบบ Exponential	อันดับ Moving average(q) = 1
ACF - ยอดที่ lag 1 และ 2 และไม่พบสหสัมพันธ์ที่ lag อื่น PACF - มีรูปร่างแบบ Sine wave หรือค่อย ๆ ลดลงแบบ Exponential	อันดับ Moving average(q) = 2
ACF - ค่อย ๆ ลดลงแบบ Exponential โดยเริ่มที่ lag 1 PACF - ค่อย ๆ ลดลงแบบ Exponential โดยเริ่มที่ lag 1	อันดับ Autoregressive(p) = 1 และ อันดับ Moving average(q) = 1
ทั้งหมดเป็นศูนย์หรือเข้าใกล้ศูนย์	ข้อมูลเป็นแบบสุ่ม (Random)
มีค่าสูงในช่วงที่คงที่	ต้องรวมตัวแปร Seasonal autoregressive (P)
ไม่พบลักษณะของการค่อย ๆ ลดลงจนเป็นศูนย์	ชุดข้อมูลยังไม่คงที่

ขั้นตอนที่ 2 ประมาณค่าพารามิเตอร์ในรูปแบบ ARIMA (p, d, q)x(P, D, Q)<sub>s</sub> จากอนุกรมเวลาที่เป็นสเตชันนารี ถ้าอนุกรมเวลายังไม่เป็นสเตชันนารี ต้องแปลงให้เป็นสเตชันนารี จากนั้นจึงนำมาประมาณค่าพารามิเตอร์ ด้วยวิธีการประมาณค่าแบบง่าย หรือวิธีกำลังสองน้อยที่สุด หรือวิธีภาวะน่าจะเป็นสูงสุด

ขั้นตอนที่ 3 ตรวจสอบความเหมาะสมของรูปแบบ ARIMA (p, d, q)x(P, D, Q)<sub>s</sub> จากอนุกรมเวลาที่เป็นสเตชันนารี ถ้าอนุกรมเวลายังไม่เป็นสเตชันนารี ต้องแปลงให้เป็นสเตชันนารี

โดยพิจารณาจากค่าสถิติเป็นเกณฑ์ เช่น ค่าความแปรปรวนของค่าคลาดเคลื่อน ( $S^2$ ) ดังสมการที่ 2.40 (ทรงศิริ แต่สมบัติ, 2549, หน้า 96) ค่า AIC (Akaike Information Criterion) ดังสมการที่ 2.41 หรือค่า BIC (Bayesian Information Criterion) ดังสมการที่ 2.42 (Wei and Shou, 2010) ซึ่งรูปแบบที่เหมาะสมที่สุด คือ รูปแบบที่ให้ค่าสถิติที่กล่าวมา ต่ำที่สุด

$$S^2 = \frac{SSE}{n - 2} \quad \text{----- (2.40)}$$

เมื่อ  $SSE =$  ค่า Sum squared error ตามสมการที่ 2.4  
 $n =$  จำนวนข้อมูล

$$AIC = 2K - 2 \log(\hat{L}_{\max}) \quad \text{----- (2.41)}$$

$$BIC = K \log(N) - 2 \log(\hat{L}_{\max}) \quad \text{----- (2.42)}$$

เมื่อ  $K =$  จำนวนของพารามิเตอร์  
 $N =$  จำนวนข้อมูล  
 $\hat{L}_{\max} =$  ค่ามากที่สุดของฟังก์ชัน ไลค์ลิฮูด (Likelihood function)

ขั้นตอนที่ 4 สร้างสมการพยากรณ์จากรูปแบบที่กำหนดในขั้นตอนที่ 1 โดยใช้ค่าประมาณของพารามิเตอร์ที่ได้จากขั้นตอนที่ 2 และได้ผ่านการตรวจสอบความเหมาะสมในขั้นตอนที่ 3 จากนั้นใช้สมการพยากรณ์ ไปพยากรณ์ค่าสังเกตล่วงหน้า โดยสามารถวัดความแม่นยำของการพยากรณ์ ด้วยการวัดค่าคลาดเคลื่อน เช่น ค่า RMSE หรือ ค่า MAPE ดังสมการที่ 2.6 และ 2.8 ตามลำดับ

### 2.3 การพยากรณ์อนุกรมเวลาด้วยเครือข่ายประสาทเทียม

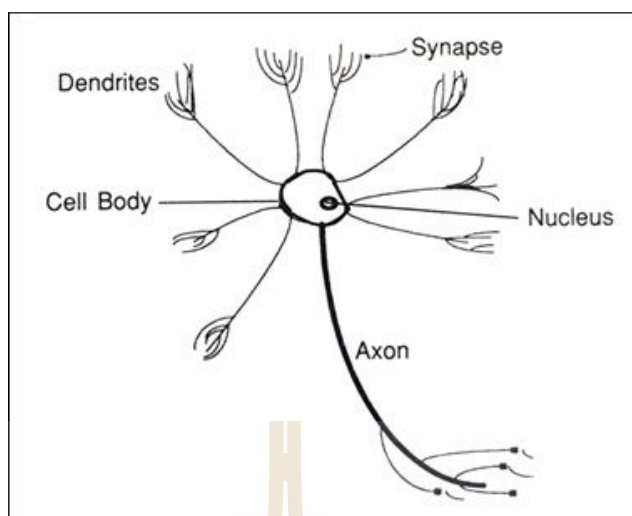
เครือข่ายประสาทเทียม (Artificial neural networks: ANN) เป็นวิธีการด้านแมชชีนเลิร์นนิง (Machine learning) ที่มีแนวคิดจากวิธีการทำงานของเครือข่ายประสาทในสมองของมนุษย์ และได้

นำวิธีการของเครือข่ายประสาทเทียมไปประยุกต์ใช้กับการทำเหมืองข้อมูล (Data mining) ในด้านต่าง ๆ เช่น การจดจำรูปแบบ (Pattern recognition) การจำแนก (Classification) การจัดกลุ่ม (Clustering) การประมาณค่าฟังก์ชัน (Function approximation) หรือการพยากรณ์ (Forecasting) โดยที่วัตถุประสงค์พื้นฐานของเครือข่ายประสาทเทียม คือ การสร้างตัวแบบทางคณิตศาสตร์ ที่จำลองการทำงานของเครือข่ายประสาทในสมองมนุษย์ให้กับเครื่องจักร เพื่อให้เครื่องจักรมีความสามารถในการเรียนรู้การจดจำรูปแบบ และการสกัดความรู้ใหม่ (Knowledge extraction) เช่นเดียวกับความสามารถที่มีในสมองมนุษย์ ซึ่งวิธีการเรียนรู้ เป็นการเรียนรู้จากประสบการณ์ที่ได้จากความรู้เดิม และมีการปรับเปลี่ยนตัวเองต่อการตอบสนองของข้อมูลนำเข้า (Input) ตามกฎของการเรียนรู้ (Learning rule) โดยเนื้อหาในบทนี้ จะกล่าวถึง แนวคิดและหลักการทำงานของเครือข่ายประสาทเทียม ตัวแบบของนิวรอน (Model of neurons) การเรียนรู้ของเครือข่ายเพอร์เซปตรอน (Perceptron learning) การเรียนรู้แบบแพร่กลับ (Back-propagation learning) และการพยากรณ์อนุกรมเวลาด้วยเครือข่ายประสาทเทียม

### 2.3.1 แนวคิดและหลักการทำงานของเครือข่ายประสาทเทียม

แนวคิดเริ่มต้นของเครือข่ายประสาทเทียม มาจากการศึกษาระบบประสาทชีวภาพในสมองของมนุษย์ โดยระบบประสาทชีวภาพมีองค์ประกอบพื้นฐานเป็นเซลล์ประสาท หรือ “นิวรอน” (Neurons) ซึ่งเป็นเซลล์ขนาดเล็ก เมื่อได้รับการกระตุ้นทางเคมีไฟฟ้าจากแหล่งกระตุ้น จะตอบสนองด้วยการสร้างกระแสไฟฟ้า และส่งต่อกระแสไฟฟ้าไปยังนิวรอนอื่น หรือเซลล์อื่น ๆ ต่อไป โดยนิวรอนเป็นเซลล์ที่มีความซับซ้อน ประกอบด้วย ตัวเซลล์ (Cell body) ที่มีนิวเคลียส (Nucleus) อยู่ภายใน และปลายรับกระแสประสาท เรียกว่า เดนไดรต์ (Dendrite) ซึ่งเป็นเสมือนอินพุตของเซลล์ ทำหน้าที่รับกระแสประสาทที่ถูกส่งมาจากเซลล์อื่นผ่านทางจุดประสานประสาท เรียกว่า ซินแนปส์ (Synapses) และปลายส่งกระแสประสาท เรียกว่า แอกซอน (Axon) ซึ่งเป็นเสมือนเอาต์พุตของเซลล์ ทำหน้าที่ส่งกระแสประสาทไปยังเซลล์อื่นผ่านทางปลายเชื่อม (Terminal link) และซินแนปส์ โดยภาพของเซลล์ประสาทแสดงได้ดังรูปที่ 2.7





รูปที่ 2.7 เซลล์ประสาท (Patterson, 1996, p.7).

จากการศึกษาระบบประสาทชีวภาพ ทำให้เกิดแนวคิดในการจำลองการทำงานของเครือข่ายเซลล์ประสาทในสมองมนุษย์ด้วยคอมพิวเตอร์ สร้างเป็นเครือข่ายประสาทเทียม โดยมีหน่วยประมวลผลข้อมูล เรียกว่า นิวรอน (Neurons) เป็นองค์ประกอบพื้นฐานในเครือข่ายประสาทเทียม แต่ละนิวรอน ประกอบด้วยอินพุตและเอาต์พุต โดยจำลองให้อินพุตแต่ละตัวมีน้ำหนัก (Weight) และนิวรอนแต่ละหน่วยจะมีค่าขีดเริ่ม (Threshold) เป็นตัวกำหนดว่า น้ำหนักรวมของอินพุตต้องมีค่าเท่าไร จึงจะสามารถส่งเอาต์พุตไปยังนิวรอนอื่นได้ เมื่อนำนิวรอนแต่ละหน่วยมาเชื่อมต่อกัน เพื่อให้ทำงานร่วมกัน จะมีการทำงานในเชิงตรรกะ คล้ายคลึงกับการรับส่งกระแสประสาทของเซลล์ประสาทในสมองของมนุษย์ แต่เครือข่ายประสาทเทียมในคอมพิวเตอร์ จะรับส่งข้อมูลที่อยู่ในรูปของตัวเลข

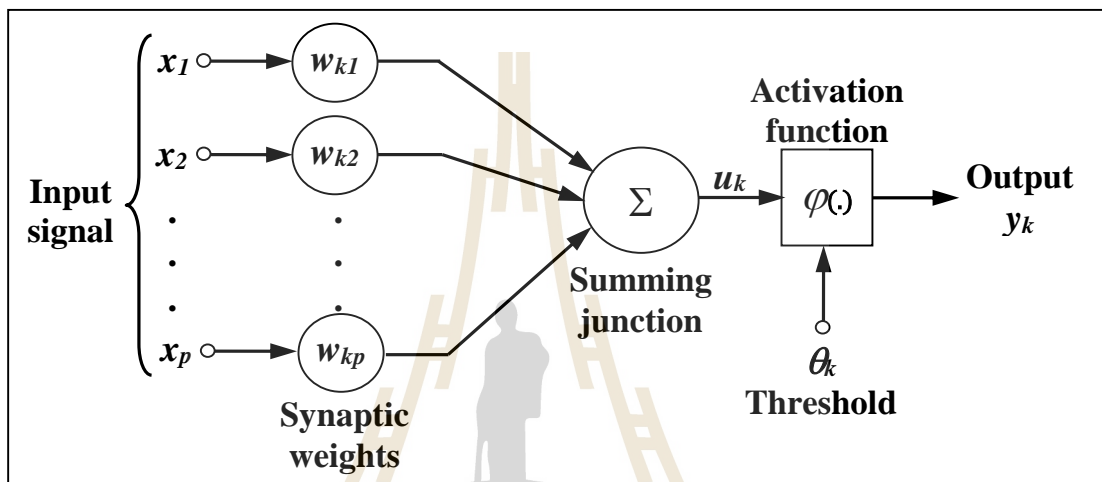
### 2.3.2 ตัวแบบของนิวรอน

ตัวแบบของนิวรอน (Model of neurons) สามารถแสดงได้ดังรูปที่ 2.8 โดยมีองค์ประกอบพื้นฐานที่สำคัญ 3 ส่วน แต่ละส่วนอธิบายได้ ดังนี้

1. กลุ่มของซินแนปส์ (Set of synapses) หรือส่วนต่อประสาน แต่ละซินแนปส์ มีการกำหนดน้ำหนัก ซึ่งสัญญาณอินพุต  $x_j$  ของซินแนปส์  $j$  ที่เชื่อมต่อกับนิวรอน  $k$  จะถูกคูณด้วยน้ำหนักซินแนปส์  $w_{kj}$  ซึ่งน้ำหนัก  $w_{kj}$  ใดๆ จะมีค่าเป็นบวก ถ้าเป็นซินแนปส์กระตุ้น (Excitatory) แต่จะมีค่าเป็นลบ ถ้าเป็นซินแนปส์ยับยั้ง (Inhibitory)

2. แอดเดอร์ (Adder) หรือซัมมิงจังชัน (Summing junction) ทำหน้าที่รวมสัญญาณอินพุตที่ถูกคูณด้วยน้ำหนักซินแนปส์ของนิวรอน ซึ่งเป็นการรวมแบบเชิงเส้น

3. แอคติเวชันฟังก์ชัน (Activation function) ทำหน้าที่จำกัดขนาดของเอาต์พุตของนิวรอน โดยทั่วไป ขนาดของเอาต์พุตเมื่อทำให้เป็นบรรทัดฐาน จะมีค่าอยู่ระหว่าง 0 ถึง 1 ( $[0, 1]$ ) หรือมีค่าอยู่ระหว่าง -1 ถึง 1 ( $[-1, 1]$ )



รูปที่ 2.8 ตัวแบบของนิวรอน (Haykin, 1994, p.8).

จากรูปที่ 2.8 สามารถอธิบายตัวแบบของนิวรอน ด้วยสมการคณิตศาสตร์ได้ ดังนี้ (Haykin, 1994, p.8)

$$u_k = \sum_{j=1}^p w_{kj} x_j \quad \text{----- (2.43)}$$

และ

$$y_k = \varphi(u_k - \theta_k) \quad \text{----- (2.44)}$$

เมื่อ  $x_1, x_2, \dots, x_p$  เป็นสัญญาณอินพุต  $w_{k1}, w_{k2}, \dots, w_{kp}$  เป็นน้ำหนักซินแนปส์ของนิวรอน  $u_k$  คือผลรวมเชิงเส้นของผลคูณระหว่าง  $w_{kj}$  และ  $x_j$  ส่วน  $\theta_k$  คือค่าขีดเริ่ม  $\varphi(\cdot)$  คือแอกติเวชันฟังก์ชัน และ  $y_k$  คือสัญญาณเอาต์พุตของนิวรอน

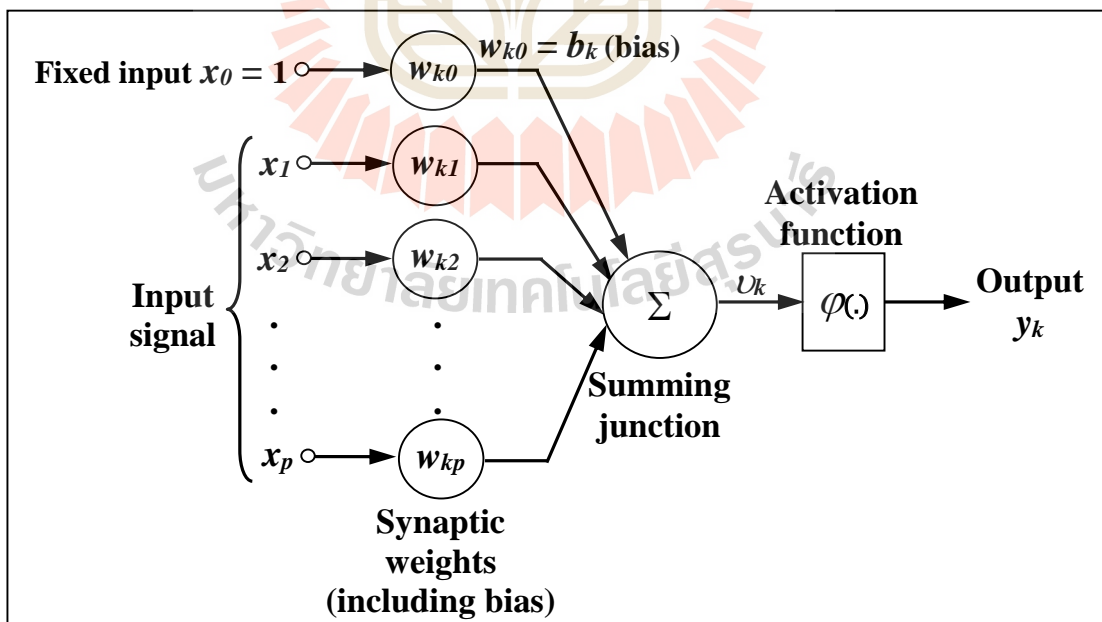
สำหรับค่า  $\theta_k$  ซึ่งเป็นพารามิเตอร์ภายนอกของนิวรอน  $k$  สามารถนำมารวมไว้ในสมการที่ 2.43 ได้ ด้วยการกำหนดเป็นอินพุต  $x_0 = 1$  และน้ำหนัก  $w_{k0} = b_k$  โดยที่  $b_k$  เรียกว่าค่าไบแอส (Bias) ทำให้สมการที่ 2.43 และ 2.44 เป็นดังนี้ (Haykin, 1994, p.9)

$$u_k = \sum_{j=0}^p w_{kj} x_j \quad \text{----- (2.45)}$$

และ

$$y_k = \varphi(u_k) \quad \text{----- (2.46)}$$

เมื่อเปลี่ยนค่า  $\theta_k$  เป็นค่า  $b_k$  ทำให้ตัวแบบของนิวรอนเปลี่ยนไปเป็นดังรูปที่ 2.9



รูปที่ 2.9 ตัวแบบของนิวรอนเมื่อรวมไบแอสไว้ในน้ำหนักซินแนปส์ (Haykin, 1994, p.10).

สำหรับแอกติเวชันฟังก์ชัน หรือฟังก์ชันถ่ายโอน (Transfer function) กำหนดด้วยสัญลักษณ์  $\varphi(\cdot)$  จะเป็นตัวกำหนดเอาต์พุตของนิวรอน โดยรูปแบบของเอาต์พุต ขึ้นกับประเภทของแอกติเวชันฟังก์ชัน ซึ่งแอกติเวชันฟังก์ชันพื้นฐาน แบ่งได้เป็น 3 ประเภท ดังนี้

1. เทรชโฮลด์ฟังก์ชัน (Threshold function) หรือฮาร์ดลิมิตฟังก์ชัน (Hard limit function) เป็นแอกติเวชันฟังก์ชัน ที่ให้ค่าเป็นสองระดับหรือสองค่า แสดงได้ดังรูปที่ 2.10a และนิยามฟังก์ชันได้ดังสมการที่ 2.47 (Haykin, 1994, p.10)

$$\varphi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases} \quad \text{----- (2.47)}$$

จากสมการที่ 2.47 เอาต์พุตของนิวรอน  $k$  ( $y_k$ ) แสดงได้ดังสมการที่ 2.48 (Haykin, 1994, p.10)

$$y_k = \begin{cases} 1 & \text{if } v_k \geq 0 \\ 0 & \text{if } v_k < 0 \end{cases} \quad \text{----- (2.48)}$$

เมื่อ  $v_k$  เป็นระดับแอกติวิตีภายใน (Internal activity level) ของนิวรอน และแสดงได้ดังสมการที่ 2.49 (Haykin, 1994, p.10)

$$v_k = \sum_{j=1}^p w_{kj} x_j - \theta_k \quad \text{----- (2.49)}$$

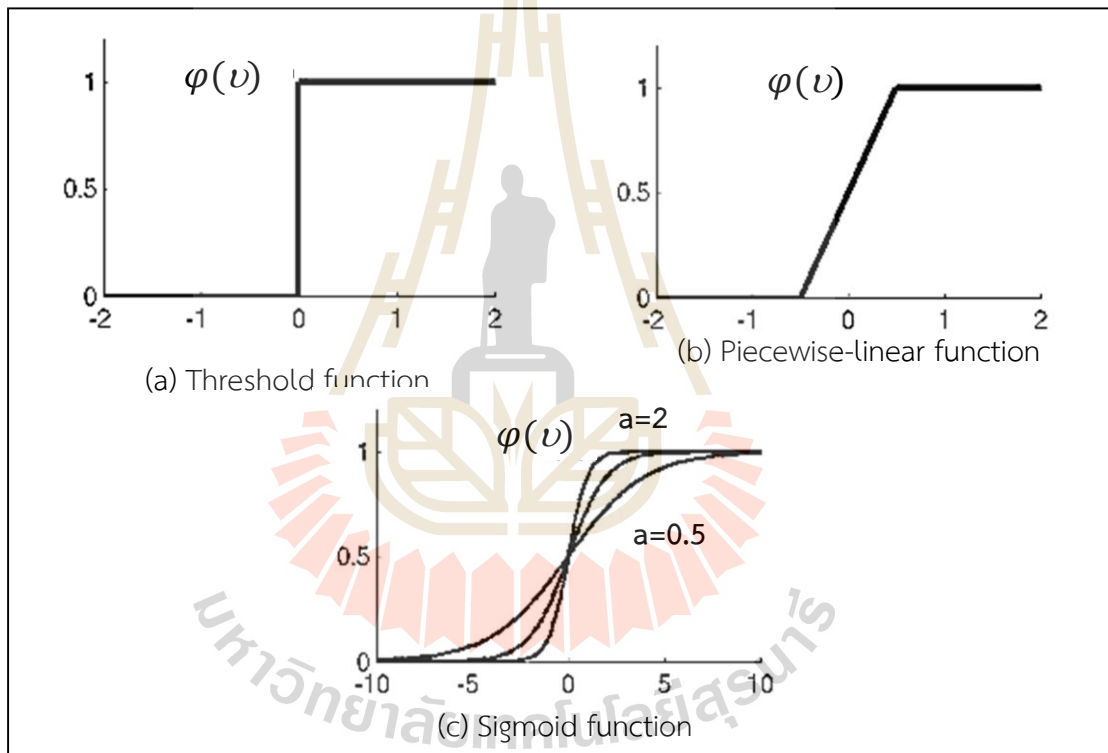
2. พีชไวส์ลิเนียร์ฟังก์ชัน (Piecewise-linear function) เป็นแอกติเวชันฟังก์ชัน ที่มีลักษณะลาดเอียง แสดงได้ดังรูปที่ 2.10b และนิยามฟังก์ชันได้ดังสมการที่ 2.50 (Haykin, 1994, p.12)

$$\varphi(v) = \begin{cases} 1, & v \geq \frac{1}{2} \\ v, & -\frac{1}{2} < v < \frac{1}{2} \\ 0, & v \leq -\frac{1}{2} \end{cases} \quad \text{----- (2.50)}$$

3. ซิกมอยด์ฟังก์ชัน (Sigmoid function) เป็นแอคติเวชันฟังก์ชัน ที่ส่วนลาดเอียง เป็นเส้นโค้ง แสดงได้ดังรูปที่ 2.10c และนิยามฟังก์ชันได้ดังสมการที่ 2.51 (Haykin, 1994, p.12)

$$\varphi(v) = \frac{1}{1+\exp(-av)} \quad \text{----- (2.51)}$$

เมื่อ  $a$  คือความชัน (Slope parameter) ของซิกมอยด์ฟังก์ชัน จากรูปที่ 2.10c ถ้า  $a$  มีค่ามาก จะทำให้กราฟมีความชันมาก และซิกมอยด์ฟังก์ชันจะให้ค่าเป็นค่าต่อเนื่องที่อยู่ระหว่าง 0 ถึง 1



รูปที่ 2.10 ประเภทของแอคติเวชันฟังก์ชัน

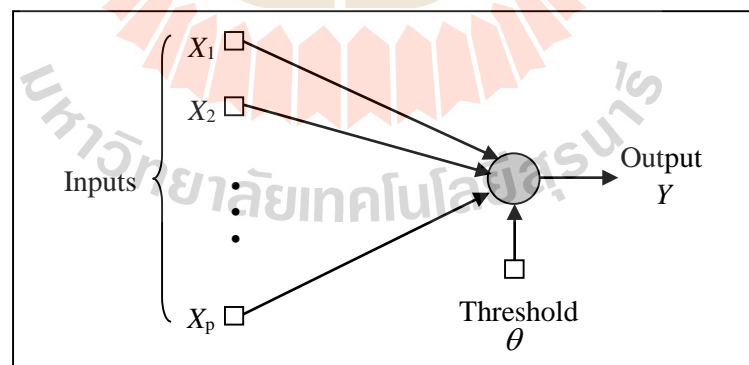
### 2.3.3 การเรียนรู้ของเครือข่ายเพอร์เซปตรอน

เพอร์เซปตรอน (Perceptron) เป็นรูปแบบที่ง่ายที่สุดของเครือข่ายประสาทเทียม ที่ใช้สำหรับจำแนกคลาสของข้อมูล โดยเป็นการจำแนกเชิงเส้น ซึ่งเครือข่ายเพอร์เซปตรอน คือ เครือข่ายแบบไปข้างหน้า (Feed forward network) ที่มีจำนวนชั้นของนิวรอน (Layer of neuron) 1 ชั้น แต่ละ

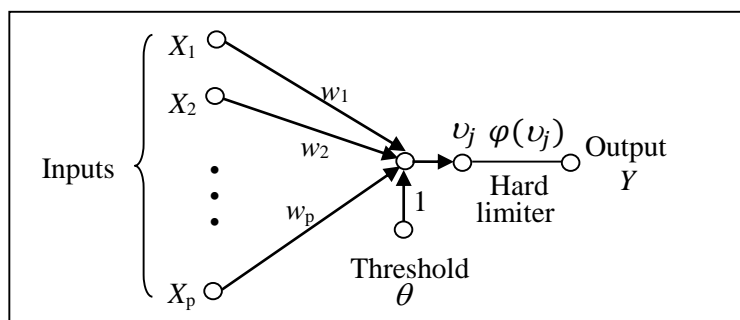
นิวรอน มีแอกติเวชันฟังก์ชันเป็นแบบเทรสโฮลด์ฟังก์ชัน (เอาต์พุตมีค่าเป็น 0 หรือ 1) ถ้าเครือข่ายมีจำนวนนิวรอนเท่ากับ  $N$  และมีจำนวนอินพุตของแต่ละนิวรอนเท่ากับ  $P$  จะกำหนดน้ำหนักซินแนปส์ในรูปของเมตริกซ์น้ำหนักซินแนปส์ ได้ดังนี้ (อาทิตย์ ศรีแก้ว, 2552, หน้า 222)

$$W = \begin{pmatrix} W_{11} & W_{12} & \dots & W_{1P} \\ W_{21} & W_{22} & \dots & W_{2P} \\ \vdots & \vdots & & \vdots \\ W_{N1} & W_{N2} & \dots & W_{NP} \end{pmatrix}$$

แถวที่  $j$  ใด ๆ ของเมตริกซ์น้ำหนักซินแนปส์ คือค่าน้ำหนักซินแนปส์ของนิวรอนตัวที่  $j$  ถ้าในกรณีที่เครือข่ายเพอร์เซปตรอนมี 1 นิวรอน สามารถเขียนเมตริกซ์น้ำหนักซินแนปส์ได้ว่า  $W = [w_1 \ w_2 \ \dots \ w_p]$  โดยเพอร์เซปตรอนที่มี 1 นิวรอน เป็นดังรูปที่ 2.11 ซึ่งใช้จำแนกคลาสได้ 2 คลาส ถ้าต้องการจำแนกคลาสมากกว่า 2 คลาส สามารถทำได้ด้วยการเพิ่มจำนวนนิวรอนในชั้นเอาต์พุต และจากตัวแบบของนิวรอน ที่ประกอบด้วยตัวรวมเชิงเส้น (Linear combiner) และตัวฮาร์ดลิมิเตอร์ (Hard limiter) ทำให้วาดกราฟสัญญาณการไหลของเพอร์เซปตรอน ได้ดังรูปที่ 2.12



รูปที่ 2.11 เพอร์เซปตรอนแบบชั้นเดียว (Haykin, 1994, p.107).



รูปที่ 2.12 กราฟสัญญาณการไหลของเพอร์เซปตรอน (Haykin, 1994, p.107).

สำหรับเพอร์เซปตรอนดังรูปที่ 2.12 แอคติเวชันฟังก์ชัน ( $\varphi(u_j)$ ) ของนิวรอนเป็นเทรสโอสต์ฟังก์ชัน และนิวรอนจะให้เอาต์พุตเป็น +1 ถ้าตัวฮาร์ดลิมิตเตอร์เป็นค่าบวก และให้เอาต์พุตเป็น -1 ถ้าตัวฮาร์ดลิมิตเตอร์เป็นค่าลบ โดยมี  $[w_1 \ w_2 \ \dots \ w_p]$  เป็นเมตริกซ์น้ำหนักซินแนปส์ของเพอร์เซปตรอน และมี  $[x_1 \ x_2 \ \dots \ x_p]^T$  เป็นอินพุตเวกเตอร์ของเพอร์เซปตรอน ส่วนค่าเทรสโอสต์ภายนอกคือ  $\theta$  ทำให้ได้เอาต์พุตของตัวรวมเชิงเส้น หรืออินพุตของตัวฮาร์ดลิมิตเตอร์เป็นดังสมการที่ 2.52

$$u = WX + \theta \quad \text{----- (2.52)}$$

เมื่อ  $W = [w_1 \ w_2 \ \dots \ w_p]$  คือ เมตริกซ์น้ำหนักซินแนปส์ของเพอร์เซปตรอน และ  $X = [x_1 \ x_2 \ \dots \ x_p]^T$  คือ อินพุตเวกเตอร์ที่กำหนดในรูปแบบของทรานส์โพสเมตริกซ์ (Transpose matrix) ดังนั้น เทอม  $WX$  ในสมการที่ 2.52 คือ การคูณของเมตริกซ์ ทำให้เขียนสมการที่ 2.52 ในรูปของผลรวมเชิงเส้น ได้ดังสมการที่ 2.53 (Haykin, 1994, p.107)

$$u = \sum_{i=1}^p w_i x_i + \theta \quad \text{----- (2.53)}$$

เนื่องจากแอคติเวชันฟังก์ชันของเพอร์เซปตรอนเป็นเทรสโอสต์ฟังก์ชัน หรือฮาร์ดลิมิตฟังก์ชัน ดังนั้น เอาต์พุตของเครือข่ายเพอร์เซปตรอน 1 นิวรอน สามารถแสดงได้ดัง

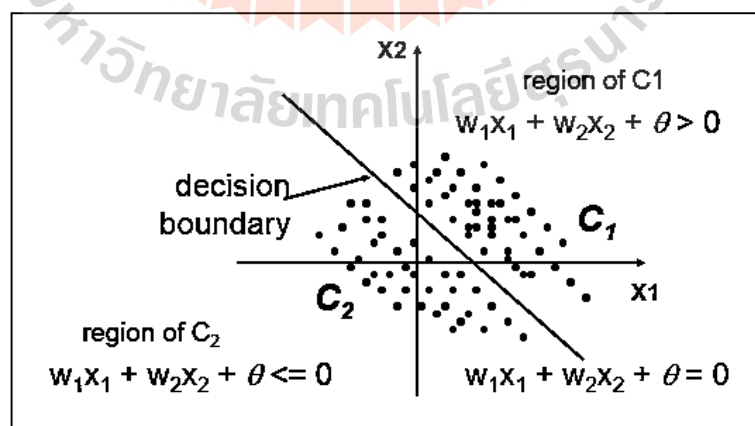
สมการที่ 2.54 (อาทิตย์ ศรีแก้ว, 2552, หน้า 223) ซึ่งฟังก์ชัน  $\text{hardlim}()$  ในสมการที่ 2.54 มีการนิยามเช่นเดียวกับสมการที่ 2.47

$$y = \varphi(v) = \text{hardlim}(WX + \theta) = \text{hardlim}\left(\sum_{i=1}^p w_i x_i + \theta\right) \quad \text{---- (2.54)}$$

การนำเพอร์เซปตรอนไปจำแนกคลาส 2 คลาส โดยมีตัวแปรอินพุตจำนวน  $p$  ตัว คือ  $x_1, x_2, \dots, x_p$  เป็นการสร้างเส้นแบ่งเขตการตัดสินใจออกเป็น 2 เขต และเรียกเส้นแบ่งเขตการตัดสินใจว่า ไฮเปอร์เพลน (Hyperplane) ซึ่งเส้นไฮเปอร์เพลน กำหนดได้ดังสมการที่ 2.55 (Haykin, 1994, p.107)

$$\sum_{i=1}^p w_i x_i + \theta = 0 \quad \text{----- (2.55)}$$

กรณีที่มีอินพุต 2 ตัว คือ  $x_1$  และ  $x_2$  โดยมีน้ำหนักซินแนปส์เป็น  $w_1$  และ  $w_2$  จะสามารถสร้างเขตการตัดสินใจ (Decision boundary) เพื่อแบ่งเป็น 2 คลาส ด้วยเส้นแบ่งเชิงเส้น ได้ดังรูปที่ 2.13 ซึ่งจุด  $(x_1, x_2)$  ที่อยู่เหนือเส้นแบ่ง จะถูกจำแนกให้อยู่ในคลาส  $C_1$  ส่วนจุด  $(x_1, x_2)$  ที่อยู่ใต้เส้นแบ่ง จะถูกจำแนกให้อยู่ในคลาส  $C_2$  โดยที่สมการของเส้นแบ่งเขตการตัดสินใจ ได้จากสมการที่ 2.53 เมื่อ  $p = 2$



รูปที่ 2.13 เส้นแบ่งเขตการตัดสินใจเชิงเส้น 2 มิติ



การออกแบบเครือข่ายเพอร์เซปตรอน สิ่งที่ต้องคำนวณหา คือ ค่าของเมตริกซ์น้ำหนักซินแนปส์ โดยใช้กฎการเรียนรู้แบบเพอร์เซปตรอน โดยมีอินพุตเป็นคู่อันดับของอินพุตเวกเตอร์  $x_1, x_2, \dots, x_i$  และตัวแปรทาร์เก็ต (Target)  $t_1, t_2, \dots, t_i$  หรืออินพุตของเครือข่าย คือ คู่อันดับ  $\{x_1, t_1\}, \{x_2, t_2\}, \dots, \{x_i, t_i\}$  ซึ่งตัวแปรทาร์เก็ตคือเอาต์พุตของชุดข้อมูลฝึกสอน (Training set) โดยมี  $y_i$  เป็นเอาต์พุตที่ได้จากเครือข่าย จากนั้น นำ  $y_i$  ไปเปรียบเทียบกับ  $t_i$  ซึ่งเครือข่ายจะทำการปรับค่าน้ำหนักซินแนปส์ เพื่อให้เอาต์พุตถูกปรับให้เข้าใกล้ทาร์เก็ตมากที่สุด ด้วยกฎการเรียนรู้แบบเพอร์เซปตรอน (อาทิตย์ ศรีแก้ว, 2552, หน้า 225) ดังนี้

$$\left. \begin{array}{l} \text{ถ้า } t = 1 \text{ และ } y = 0 \\ \text{ถ้า } t = 0 \text{ และ } y = 1 \\ \text{ถ้า } t = y \end{array} \right\} \begin{array}{l} \text{แล้ว } W^{\text{new}} = W^{\text{old}} + X \\ \text{แล้ว } W^{\text{new}} = W^{\text{old}} - X \\ \text{แล้ว } W^{\text{new}} = W^{\text{old}} \end{array} \quad \text{----- (2.56)}$$

ถ้ากำหนดค่าความผิดพลาด (Error) ของเครือข่ายเพอร์เซปตรอนด้วยสัญลักษณ์  $e$  จะนิยาม  $e$  ได้ว่า  $e = t - y$  และแปลงกฎการเรียนรู้แบบเพอร์เซปตรอน ด้วยเงื่อนไขของค่าความผิดพลาดได้ ดังนี้

$$\left. \begin{array}{l} \text{ถ้า } e = 1 \\ \text{ถ้า } e = -1 \\ \text{ถ้า } e = 0 \end{array} \right\} \begin{array}{l} \text{แล้ว } W^{\text{new}} = W^{\text{old}} + X \\ \text{แล้ว } W^{\text{new}} = W^{\text{old}} - X \\ \text{แล้ว } W^{\text{new}} = W^{\text{old}} \end{array} \quad \text{----- (2.57)}$$

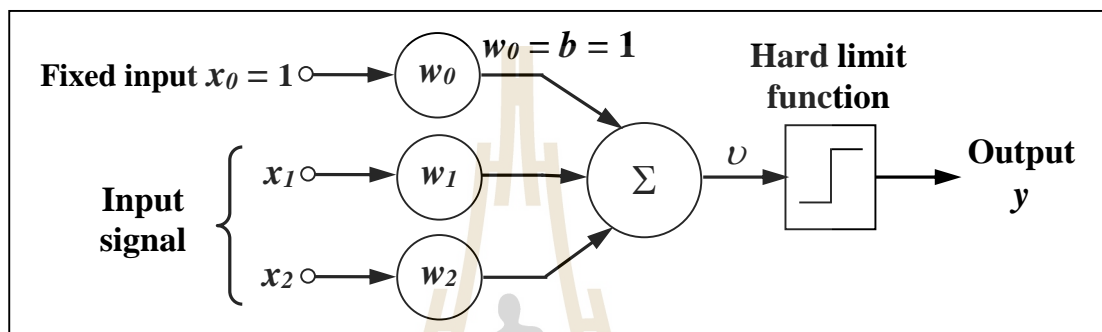
จากสมการที่ 2.57 สามารถยุบรวมเป็นสมการเดียวได้ ดังสมการที่ 2.58

$$W^{\text{new}} = W^{\text{old}} + e.X \quad \text{----- (2.58)}$$

สำหรับค่าเทรสโฮลด์ ซึ่งเป็นพารามิเตอร์ภายนอก สามารถเปลี่ยนให้เป็นน้ำหนักได้ ด้วยการกำหนดอินพุต  $x_0 = 1$  และเรียกว่า ค่าไบแอส (b) ดังที่อธิบายไว้ในสมการที่ 2.45 ค่าไบแอสจึงถูกปรับด้วยกฎการเรียนรู้เดียวกัน ได้ว่า

$$b^{\text{new}} = b^{\text{old}} + e \quad \text{----- (2.59)}$$

เพื่อแสดงการทำงานในกระบวนการเรียนรู้ของเครือข่ายเพอร์เซปตรอน จะยกตัวอย่างการเรียนรู้ของเพอร์เซปตรอน เพื่อหาฟังก์ชันแทนตัวปฏิบัติการตรรกศาสตร์ OR โดยมีโครงสร้างของเครือข่ายเป็นดังรูปที่ 2.14



รูปที่ 2.14 โครงสร้างของเครือข่ายเพอร์เซปตรอนเพื่อหาฟังก์ชันแทนตัวปฏิบัติการ OR

จากรูปที่ 2.14 กำหนดอินพุตเวกเตอร์เป็น  $X = \langle X_1, X_2, X_3, X_4 \rangle$  และ  $t = \langle t_1, t_2, t_3, t_4 \rangle$  เป็นทาร์เก็ต แต่ละอินพุตเวกเตอร์  $= [x_1, x_2]^T$  เมื่อ  $x_1, x_2 \in \{0,1\}$  ซึ่ง  $X$  และ  $t$  เป็นดังนี้

$$\{ X_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, t_1 = [0] \} \quad \{ X_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, t_2 = [1] \}$$

$$\{ X_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, t_3 = [1] \} \quad \{ X_4 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, t_4 = [1] \}$$

เมื่อกำหนดค่าเริ่มต้นของน้ำหนักซินแนปส์เป็น  $[-1.2 \quad -0.5]$  และค่าเริ่มต้นของไบแอส  $b = 1$  จากนั้น คำนวณค่าเอาต์พุต  $y$  ตามสมการที่ 2.54 และจากสมการที่ 2.58 และ 2.59 ทำให้การปรับค่าน้ำหนักซินแนปส์และไบแอส แต่ละรอบ เป็นดังนี้

รอบที่ 1. สำหรับ  $X_1 = [0 \quad 0]^T$   $y = \text{hardlim}(w_1 x_1 + w_2 x_2 + b) = +1$

$$e = t_1 - y = -1$$

$$W^{\text{new}} = W^{\text{old}} + e.X = [-1.2+(-1*0) \quad -0.5+(-1*0)] = [-1.2 \quad -0.5]$$

$$b^{\text{new}} = b^{\text{old}} + e = 1+(-1)=0$$

รอบที่ 2. สำหรับ  $X_2 = [0 \ 1]^T$   $y = \text{hardlim}(w_1 x_1 + w_2 x_2 + b) = 0$

$$e = t_2 - y = +1$$

$$W^{\text{new}} = W^{\text{old}} + e.X = [-1.2+(1*0) \quad -0.5+(1*1)] = [-1.2 \quad +0.5]$$

$$b^{\text{new}} = b^{\text{old}} + e = 0+1=+1$$

รอบที่ 3. สำหรับ  $X_3 = [1 \ 0]^T$   $y = \text{hardlim}(w_1 x_1 + w_2 x_2 + b) = 0$

$$e = t_3 - y = +1$$

$$W^{\text{new}} = W^{\text{old}} + e.X = [-1.2+(1*1) \quad 0.5+(1*0)] = [-0.2 \quad +0.5]$$

$$b^{\text{new}} = b^{\text{old}} + e = 1+1=+2$$

รอบที่ 4. สำหรับ  $X_4 = [1 \ 1]^T$   $y = \text{hardlim}(w_1 x_1 + w_2 x_2 + b) = +1$

$$e = t_4 - y = 0$$

$$W^{\text{new}} = W^{\text{old}} + e.X = [-0.2+(0*1) \quad +0.5+(0*1)] = [-0.2 \quad +0.5]$$

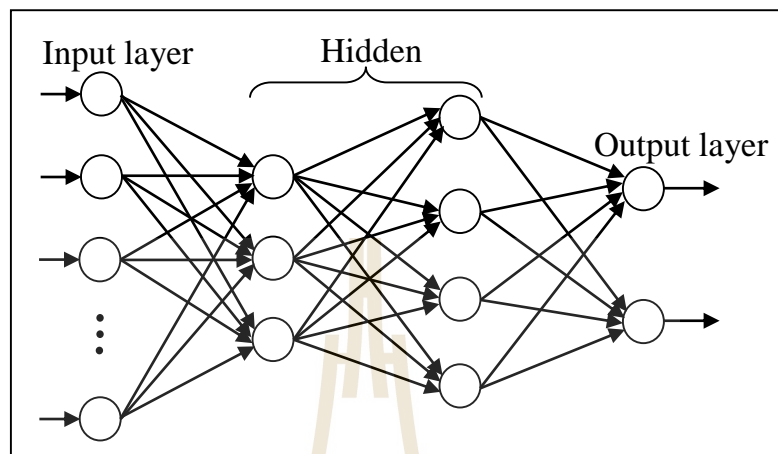
$$b^{\text{new}} = b^{\text{old}} + e = 2+(0)=+2$$

รอบที่ 5. จะวนกลับมาที่  $X_1$  อีกครั้ง ซึ่งการเรียนรู้ของเครือข่ายจะหยุดลง เมื่อคำนวณหาเอาต์พุตจากอินพุตทุกตัว แล้วค่าเอาต์พุตที่ได้เป็นไปตามทาร์เก็ตทุก ๆ อินพุต (นั่นคือค่าความผิดพลาด  $e=0$ ) โดยจะเกิดการซ้ำถึงรอบที่ 29 ได้ค่าน้ำหนักซินแนปส์สุดท้ายเท่ากับ  $[+1.8 \quad +1.5]$  และค่าไบแอสสุดท้ายเท่ากับ  $-1.0$  และจากตัวอย่างนี้ แสดงให้เห็นถึงการทำงานและกระบวนการเรียนรู้ของเครือข่ายเพอร์เซปตรอน ซึ่งพิจารณาได้ว่าการฝึกสอนเครือข่ายประสาทเทียม เป็นการค้นหาค่าน้ำหนักซินแนปส์และไบแอสที่เหมาะสมที่สุดของเครือข่าย

### 2.3.4 การเรียนรู้แบบแพร่กลับ

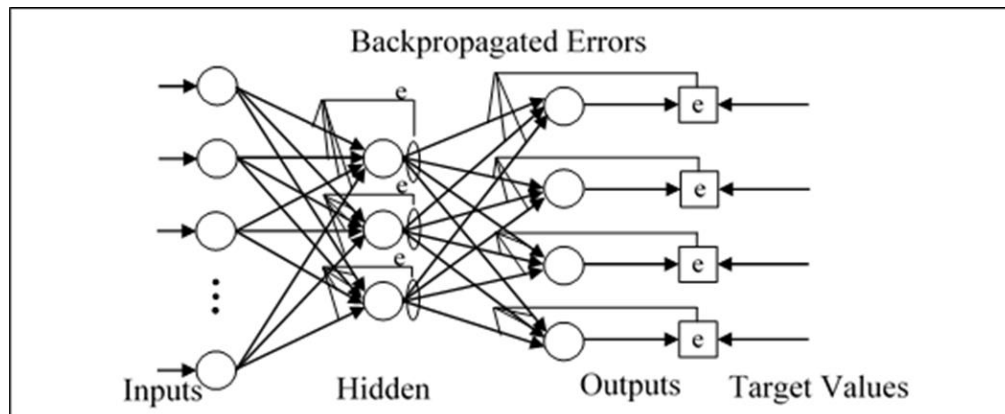
การเรียนรู้แบบแพร่กลับ (Backpropagation learning) เป็นเครือข่ายประสาทเทียมที่มีโครงสร้างไปข้างหน้าแบบหลายชั้น (Multi-layer feed forward networks) ประกอบด้วยชั้นอินพุต (Input layer) ชั้นซ่อนเร้น (Hidden layer) และชั้นเอาต์พุต (Output layer) โดยมีโครงสร้างแสดงได้ดังรูปที่ 2.15 สำหรับชั้นซ่อนเร้น อาจมีได้มากกว่า 1 ชั้น โดยชั้นซ่อนเร้นที่ 1 จะรับอินพุตเวกเตอร์ร่วมกับน้ำหนักซินแนปส์ระหว่างชั้นอินพุตและชั้นซ่อนเร้น ( $w_{in}$ ) เพื่อแปลงให้เป็นสัญญาณเอาต์พุตภายในที่จะส่งต่อไปยังชั้นซ่อนเร้นถัดไป ซึ่งเอาต์พุตที่ได้จากแต่ละชั้นจะถูกส่งต่อแบบไปข้างหน้าเพื่อเป็นอินพุตให้ชั้นถัดไป และจะกระทำการส่งต่อแบบไปข้างหน้าถึงชั้นเอาต์พุต เป็นการสิ้นสุดรอบ

การทำงานที่ 1 จากนั้น กระบวนการเดิมจะถูกกระทำซ้ำเป็นรอบต่อไป จนกระทั่งเข้าเงื่อนไขให้เกิดการคำนวณรอบสุดท้าย ได้เป็นเอาต์พุตของเครือข่ายออกมา



รูปที่ 2.15 เครือข่ายประสาทเทียมที่มีโครงสร้างไปข้างหน้าแบบหลายชั้น (Patterson, 1996, p.143).

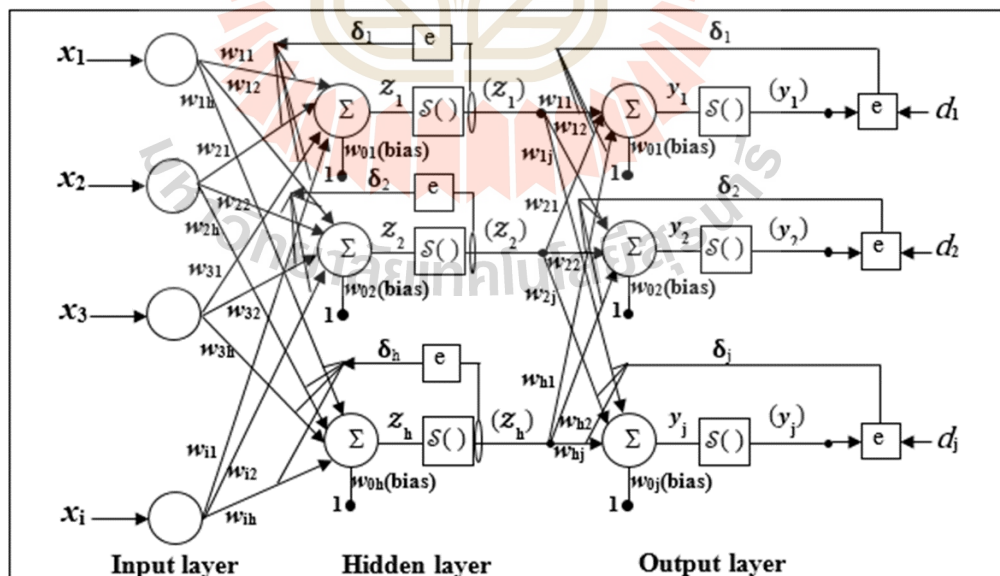
เครือข่ายประสาทเทียมที่มีโครงสร้างไปข้างหน้าแบบหลายชั้น ได้ถูกนำมาใช้ร่วมกับอัลกอริทึมแบบแพร่กลับ (Backpropagation) ที่นำเสนอโดย Rumelhart และ Zipser (Rumelhart and Zipser, 1985) ในปีค.ศ. 1985 เพื่อให้สามารถแก้ปัญหาแบ่งแยกเชิงเส้น (Linearly separable) โดยการเรียนรู้แบบแพร่กลับ คือการส่งอินพุตเข้าสู่เครือข่าย ผ่านแต่ละชั้นภายในเครือข่ายจนได้เอาต์พุต จากนั้น นำเอาต์พุตที่ได้ไปเปรียบเทียบกับทาร์เก็ต วัตถุประสงค์เป็นค่าคลาดเคลื่อน (Error) แล้วนำค่าคลาดเคลื่อนมาป้อนกลับเป็นอินพุต เพื่อปรับน้ำหนักซินแนปส์จากชั้นหนึ่งสู่อีกชั้นหนึ่ง ในทิศทางย้อนกลับ ซึ่งเครือข่ายประสาทไปข้างหน้าแบบหลายชั้น ที่ใช้การฝึกสอนแบบแพร่กลับ แสดงได้ดังรูปที่ 2.16



รูปที่ 2.16 เครือข่ายประสาทเทียมที่มีโครงสร้างไปข้างหน้าแบบหลายชั้นและแพร่กลับ

(Patterson, 1996, p.145).

เพื่อให้เกิดความเข้าใจได้ง่ายขึ้น สำหรับการอธิบายอัลกอริทึมแบบแพร่กลับ จะใช้เครือข่ายที่มีชั้นซ่อนเร้น 1 ชั้น หรือเครือข่ายไปข้างหน้าแบบ 3 ชั้น (Three layers feed forward networks) ซึ่งมีโครงสร้างของเครือข่ายเป็นดังรูปที่ 2.17



รูปที่ 2.17 เครือข่ายไปข้างหน้าแบบ 3 ชั้น ด้วยการเรียนรู้แบบแพร่กลับ

สำหรับอัลกอริทึมแบบแพร่กลับ จะใช้ชุดข้อมูลฝึกสอนที่เป็นคู่อันดับของอินพุตเวกเตอร์และทาร์เก็ตเวกเตอร์ ( $\mathbf{X}_k, \mathbf{D}_k$ ) จากนั้น นำมาผ่านการทำข้อมูลให้เป็นบรรทัดฐาน (Data normalization) โดยมีอัลกอริทึมที่แสดงด้วยรหัสเทียม (Pseudo code) ดังรูปที่ 2.18

1	Repeat {
2	$(x_i^k) = x_i^k$ , $i=1, 2, \dots, n$
3	$(x_0^k) = 1$
4	$z_h^k = \sum_{i=0}^n w_{ih}^k x_i^k$ , $h=1, 2, \dots, q$
5	$(z_h^k) = \frac{1}{1+\exp(-z_h^k)}$ ,
6	$(z_0^k) = 1$ ,
7	$y_j^k = \sum_{h=0}^q w_{hj}^k \mathcal{S}(z_h^k)$ , $j=1, 2, \dots, p$
8	$(y_j^k) = \frac{1}{1+\exp(-y_j^k)}$
9	$\delta_j^k = (d_j^k - \mathcal{S}(y_j^k))(\mathcal{S}(y_j^k))(1 - \mathcal{S}(y_j^k))$ , $j=1, 2, \dots, p$
10	$\Delta w_{hj}^k = \eta \delta_j^k \mathcal{S}(z_h^k)$ , $h=0, 1, \dots, q; j=1, 2, \dots, p$
11	$\delta_h^k = (\sum_{j=1}^p \delta_j^k w_{hj}^k)(\mathcal{S}(z_h^k))(1 - \mathcal{S}(z_h^k))$ , $h=1, 2, \dots, q$
12	$\Delta w_{ih}^k = \eta \delta_h^k x_i^k$ , $i=0, 1, \dots, n; h=1, 2, \dots, q$
13	$w_{hj}^{k+1} = w_{hj}^k + \Delta w_{hj}^k + \alpha \Delta w_{hj}^{k-1}$ , $h=0, 1, \dots, q; j=1, 2, \dots, q$
14	$w_{ih}^{k+1} = w_{ih}^k + \Delta w_{ih}^k + \alpha \Delta w_{ih}^{k-1}$ , $i=0, 1, \dots, n; h=1, 2, \dots, q$
15	} until $(\varepsilon_{av} = \frac{1}{Q} \sum_{k=1}^Q \varepsilon_k < \tau)$

รูปที่ 2.18 อัลกอริทึมแบบแพร่กลับ (Kumar, 2004).

อัลกอริทึมแบบแพร่กลับ มีสิ่งที่จะต้องกำหนดให้และการกำหนดค่าเริ่มต้น ก่อนที่จะดำเนินการตามขั้นตอนของอัลกอริทึม ดังนี้

สิ่งที่ต้องกำหนดให้

1. อินพุตเวกเตอร์  $\mathbf{X}_k (x_1, x_2, \dots, x_i)$

2. ทาร์เก็ตเวกเตอร์  $\mathbf{D}_k (d_1, d_2, \dots, d_j)$
3. จำนวนโหนดของชั้นอินพุต ( $n$ ) จำนวนนิวรอนของชั้นซ่อนเร้น ( $q$ ) และจำนวนนิวรอนของชั้นเอาต์พุต ( $p$ )

#### กำหนดค่าเริ่มต้น

1. น้ำหนักซินแนปส์และน้ำหนักไบแอสเริ่มต้น ระหว่างชั้นอินพุตและชั้นซ่อนเร้น ( $w_{ih}$  เมื่อ  $i = 0, 1, 2, \dots, n$  และ  $h = 0, 1, 2, \dots, q$ ) โดยกำหนดให้มีค่าน้อย ๆ ระหว่าง -1 ถึง 1
2. น้ำหนักซินแนปส์และน้ำหนักไบแอสเริ่มต้น ระหว่างชั้นซ่อนเร้นและชั้นเอาต์พุต ( $w_{hj}$  เมื่อ  $h = 0, 1, 2, \dots, q$  และ  $j = 0, 1, 2, \dots, p$ ) โดยกำหนดให้มีค่าน้อย ๆ ระหว่าง -1 ถึง 1
3.  $k$  คือ รอบการคำนวณซ้ำ (Epoch) โดยเริ่มจาก  $k=1$
4.  $\eta$  คือ ค่าอัตราการเรียนรู้ (Learning rate) เป็นค่าคงที่
5.  $\alpha$  คือ ค่าโมเมนตัม (momentum) เป็นค่าคงที่
6.  $\tau$  คือ ค่าความผิดพลาดที่ยอมรับได้ (Error tolerance) เป็นค่าคงที่

สำหรับการอธิบายอัลกอริทึมแบบแพร่กลับ ดังรูปที่ 2.18 สามารถอธิบายเป็นขั้นตอนตามหมายเลขบรรทัดคำสั่ง ได้ดังนี้

1. เริ่มต้นรอบการทำซ้ำ จนกว่าจะเข้าเงื่อนไขสิ้นสุดการทำงาน โดยเริ่มจากรอบที่  $k = 1$
2. กำหนดค่าสัญญาณแบบไปข้างหน้า โดยชั้นอินพุตมีแอกติเวชันฟังก์ชันเป็นแบบเพียวลิเนียร์ (Pure linear)
3. กำหนดค่าสัญญาณของไบแอสในชั้นอินพุต ( $x_0^k$ ) เท่ากับ 1 เพื่อนำไปคูณกับไบแอส ( $w_0^k$ ) ในชั้นซ่อนเร้น แล้วได้เป็นค่าไบแอสดั้งเดิม
4. อินพุตของแอกติเวชันฟังก์ชันในชั้นซ่อนเร้น คือ ผลรวมของผลคูณระหว่างน้ำหนักซินแนปส์จากชั้นอินพุตไปยังชั้นซ่อนเร้น ( $w_{ih}^k$ ) และค่าสัญญาณของชั้นอินพุต ( $x_i^k$ )
5. กำหนดให้แอกติเวชันฟังก์ชันของชั้นซ่อนเร้นเป็นซิกมอยด์ฟังก์ชัน
6. กำหนดค่าสัญญาณของไบแอสในชั้นซ่อนเร้น ( $z_0^k$ ) เท่ากับ 1 เพื่อนำไปคูณกับไบแอส ( $w_0^k$ ) ในชั้นเอาต์พุต แล้วได้เป็นค่าไบแอสดั้งเดิม
7. อินพุตของแอกติเวชันฟังก์ชันในชั้นเอาต์พุต คือ ผลรวมของผลคูณระหว่างน้ำหนักซินแนปส์ จากชั้นซ่อนเร้นไปยังชั้นเอาต์พุต ( $w_{hj}^k$ ) และค่าสัญญาณของชั้นซ่อนเร้น ( $S(z_h^k)$ )

8. กำหนดให้แอกติเวชันฟังก์ชันของชั้นเอาต์พุตเป็นซิกมอยด์ฟังก์ชัน
9. คำนวณค่าเดลตาที่ชั้นเอาต์พุต (Deltas:  $\delta^k$ )
10. คำนวณค่าผลต่างน้ำหนักซินแนปส์ที่เปลี่ยนแปลงไปที่ชั้นเอาต์พุต ( $\Delta w_{hj}^k$ )

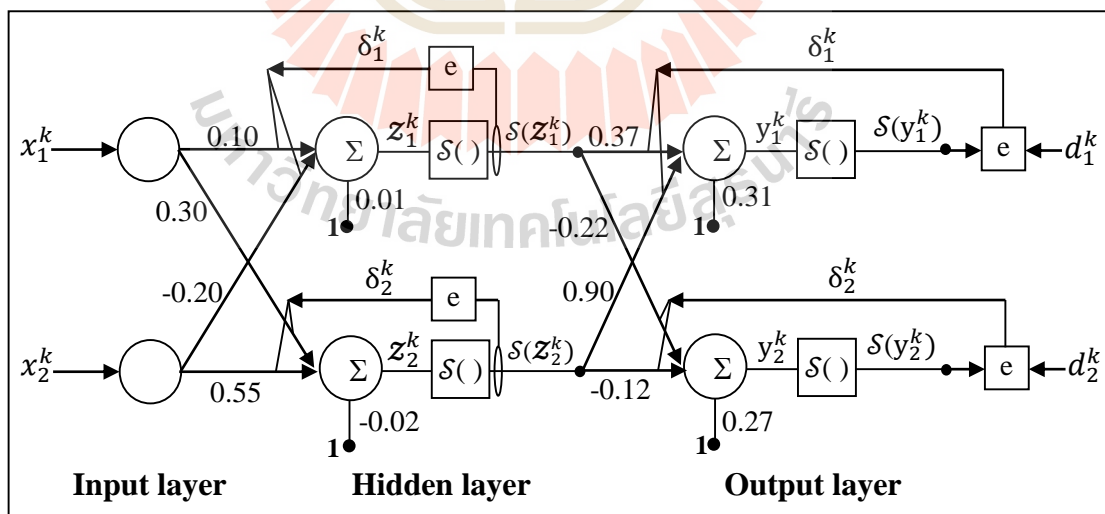
จากค่าเดลตาที่ชั้นเอาต์พุต

11. คำนวณค่าเดลตาที่ชั้นซ่อนเร้น (Deltas:  $\delta_h^k$ )
12. คำนวณค่าผลต่างน้ำหนักซินแนปส์ที่เปลี่ยนแปลงไปที่ชั้นซ่อนเร้น ( $\Delta w_{hj}^k$ )

จากค่าเดลตาที่ชั้นซ่อนเร้น

13. ปรับปรุงน้ำหนักซินแนปส์จากชั้นซ่อนเร้นไปยังชั้นเอาต์พุต เป็นค่าใหม่สำหรับรอบการคำนวณซ้ำ (Epoch) รอบถัดไป
14. ปรับปรุงน้ำหนักซินแนปส์จากชั้นอินพุตไปยังชั้นซ่อนเร้น เป็นค่าใหม่สำหรับรอบการคำนวณซ้ำ (Epoch) รอบถัดไป
15. เก็บค่าผิดพลาดรอบที่  $k$  ( $\epsilon_k$ ) เพื่อนำไปเป็นเงื่อนไขสิ้นสุดรอบการคำนวณซ้ำ และวนรอบคำนวณซ้ำจนกระทั่ง ค่าผิดพลาดเฉลี่ย ( $\epsilon_{av}$ ) น้อยกว่าค่าผิดพลาดที่ยอมรับได้ ( $\tau$ )

เพื่อแสดงกระบวนการปรับน้ำหนักซินแนปส์ของอัลกอริทึมแบบแพร่กลับ จะยกตัวอย่างเครือข่ายไปข้างหน้า 3 ชั้น แบบแพร่กลับ ที่มีโครงสร้างดังรูปที่ 2.19 มาอธิบายการทำงาน



รูปที่ 2.19 ตัวอย่างเครือข่ายไปข้างหน้า 3 ชั้น แบบแพร่กลับ



จากรูปที่ 2.19 กำหนดให้ อินพุตเวกเตอร์และทาร์เก็ตเวกเตอร์ สำหรับรอบการคำนวณที่ 1 และ 2 เป็นดังตารางที่ 2.4

ตารางที่ 2.4 ค่าของอินพุตเวกเตอร์และทาร์เก็ตเวกเตอร์ สำหรับรอบการคำนวณที่ 1 และ 2

รอบการคำนวณที่ (k)	$x_1^k$	$x_2^k$	$d_1^k$	$d_2^k$
1	0.5	-0.5	0.9	0.1
2	-0.5	0.5	0.1	0.9

เริ่มต้นรอบการทำงานที่ 1 ( $k=1$ ) และกำหนดค่า  $\eta = 1.2$  จากนั้นคำนวณค่าสัญญาณแบบไปข้างหน้า ด้วยอัลกอริทึมแบบแพร่กลับ โดยสามารถสรุปค่าสัญญาณต่าง ๆ ดังตารางที่ 2.5

ตารางที่ 2.5 ค่าสัญญาณของเครือข่าย สำหรับรอบการทำงานที่ 1

k	$x_1^k$	$x_2^k$	$s(x_1^k)$	$s(x_2^k)$	$z_1^k$	$z_2^k$	$s(z_1^k)$	$s(z_2^k)$	$y_1^k$	$y_2^k$	$s(y_1^k)$	$s(y_2^k)$
1	0.5	-0.5	0.5	-0.5	0.16	-0.145	0.5399	0.4638	0.9271	0.0955	0.7164	0.5238

ขั้นตอนต่อไป คือ การคำนวณค่าคลาดเคลื่อน แล้วป้อนกลับไปในทิศทางย้อนกลับ เพื่อนำไปปรับค่าน้ำหนักซินแนปส์ จากชั้นเอาต์พุตไปยังชั้นซ่อนเร้น ดังนี้

$$e_1^1 = d_1^1 - s(y_1^1) = 0.9 - 0.7164 = 0.1836$$

$$e_2^1 = d_2^1 - s(y_2^1) = 0.1 - 0.5238 = -0.4238$$

$$\delta j_1^1 = e_1^1 (s(y_1^1)) (1 - s(y_1^1)) = 0.1836 * 0.7164 * (1 - 0.7164) = 0.0373$$

$$\delta j_2^1 = e_2^1 (s(y_2^1)) (1 - s(y_2^1)) = -0.4238 * 0.5238 * (1 - 0.5238) = -0.1057$$

$$\delta h_1^1 = (\sum_{j=1}^2 \delta_j^1 w_{1j}^1) (s(z_1^1)) (1 - s(z_1^1)) = 0.0092$$

$$\delta h_2^1 = (0.0373 * 0.90 + (-0.1057 * -0.12)) * 0.4638 (1 - 0.4638) = 0.0115$$

คำนวณค่าผลต่างน้ำหนักซินแนปส์ที่เปลี่ยนแปลงไประหว่างชั้นซ่อนเร้นและชั้นเอาต์พุต ตามสมการ  $\Delta w_{hj}^k = \eta \delta_j^k S(\mathbf{z}_h^k)$

$$\begin{aligned}\Delta w_{01}^1 &= 1.2 * 0.0373 * 1.0 = 0.0447 \\ \Delta w_{11}^1 &= 1.2 * 0.0373 * 0.5399 = 0.0241 \\ \Delta w_{21}^1 &= 1.2 * 0.0373 * 0.4638 = 0.0207 \\ \Delta w_{02}^1 &= 1.2 * -0.1057 * 1.0 = -0.1268 \\ \Delta w_{12}^1 &= 1.2 * -0.1057 * 0.5399 = -0.0684 \\ \Delta w_{22}^1 &= 1.2 * -0.1057 * 0.4638 = -0.0588\end{aligned}$$

คำนวณค่าผลต่างน้ำหนักซินแนปส์ที่เปลี่ยนแปลงไประหว่างชั้นอินพุตและชั้นซ่อนเร้น ตามสมการ  $\Delta w_{ih}^k = \eta \delta_h^k x_i^k$

$$\begin{aligned}\Delta w_{01}^1 &= 1.2 * 0.0092 * 1.0 = 0.0110 \\ \Delta w_{11}^1 &= 1.2 * 0.0092 * 0.5 = 0.0055 \\ \Delta w_{21}^1 &= 1.2 * 0.0092 * -0.5 = -0.0055 \\ \Delta w_{02}^1 &= 1.2 * 0.0115 * 1.0 = 0.0138 \\ \Delta w_{12}^1 &= 1.2 * 0.0115 * 0.5 = 0.0069 \\ \Delta w_{22}^1 &= 1.2 * 0.0115 * -0.5 = -0.0069\end{aligned}$$

คำนวณค่าน้ำหนักซินแนปส์ระหว่างชั้นซ่อนเร้นและชั้นเอาต์พุต สำหรับรอบถัดไป (k=2) ตามสมการ  $w_{hj}^{k+1} = w_{hj}^k + \Delta w_{hj}^k + \alpha \Delta w_{hj}^{k-1}$  (หมายเหตุ  $\Delta w_{hj}^0 = 0$ )

$$\begin{aligned}w_{01}^2 &= 0.31 + 0.0447 + 0 = 0.3547 \\ w_{11}^2 &= 0.37 + 0.0241 + 0 = 0.3941 \\ w_{21}^2 &= 0.90 + 0.0207 + 0 = 0.9207 \\ w_{02}^2 &= 0.27 - 0.1268 + 0 = 0.1432 \\ w_{12}^2 &= -0.22 - 0.0684 + 0 = -0.2884 \\ w_{22}^2 &= -0.12 - 0.0588 + 0 = -0.1788\end{aligned}$$

คำนวณค่าน้ำหนักซินแนปส์ระหว่างชั้นอินพุตและชั้นซ่อนเร้น สำหรับรอบถัดไป (k=2) ตามสมการ  $w_{ih}^{k+1} = w_{ih}^k + \Delta w_{ih}^k + \alpha \Delta w_{ih}^{k-1}$  (หมายเหตุ  $\Delta w_{ih}^0 = 0$ )

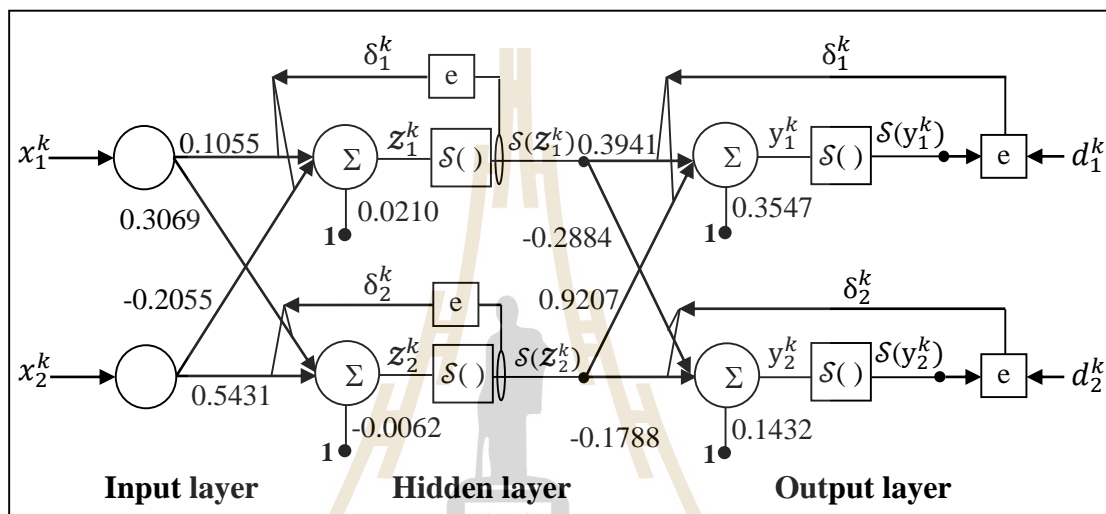
$$\begin{aligned}w_{01}^2 &= 0.01 + 0.0110 + 0 = 0.021 \\ w_{11}^2 &= 0.10 + 0.0055 + 0 = 0.1055 \\ w_{21}^2 &= -0.20 - 0.0055 + 0 = -0.2055\end{aligned}$$

$$Wih_{02}^2 = -0.02 + 0.0138 + 0 = -0.0062$$

$$Wih_{12}^2 = 0.30 + 0.0069 + 0 = 0.3069$$

$$Wih_{22}^2 = 0.55 - 0.0069 + 0 = 0.5431$$

โครงสร้างของเครือข่ายหลังจากผ่านการปรับน้ำหนักซินแนปส์ในรอบแรก (k=1) เป็นดังรูปที่ 2.20



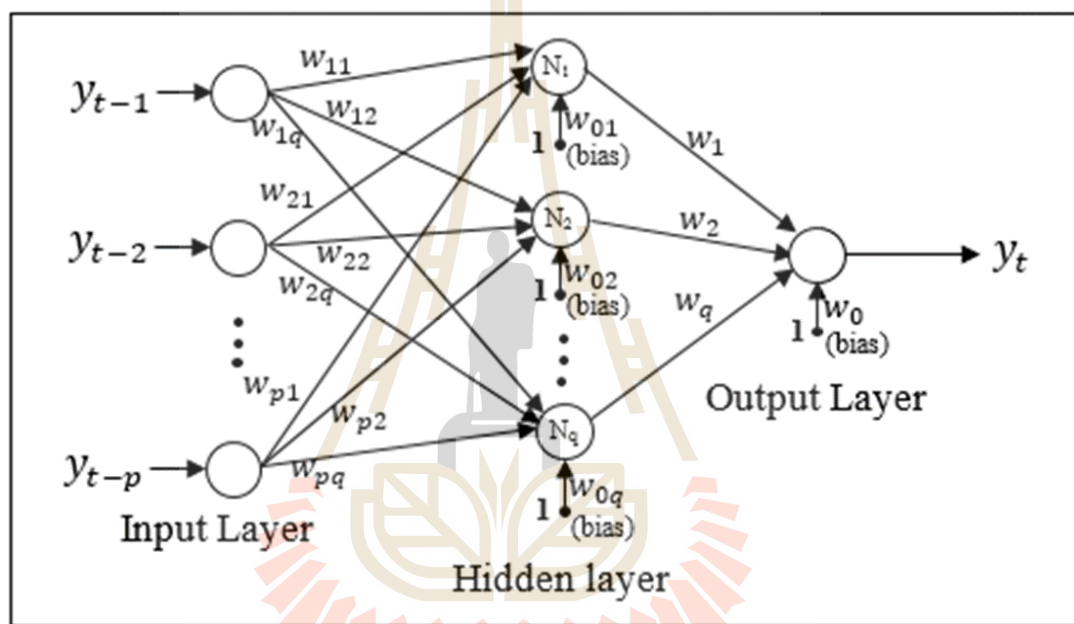
รูปที่ 2.20 โครงสร้างของตัวอย่างเครือข่ายหลังจากผ่านการปรับน้ำหนักซินแนปส์ในรอบแรก

จากนั้น กระบวนการปรับน้ำหนักซินแนปส์จะถูกกระทำซ้ำด้วยกระบวนการเดิม โดยใช้โครงสร้างของเครือข่าย ดังรูปที่ 2.20 มาคำนวณการปรับน้ำหนักซินแนปส์ในรอบการคำนวณที่ 2 (k=2) ซึ่งจะเกิดรอบการทำซ้ำด้วยกระบวนการเดิม จนกระทั่งเข้าสู่เงื่อนไขสิ้นสุดรอบการทำซ้ำ และได้น้ำหนักซินแนปส์สุดท้ายของเครือข่าย ที่ได้จากการปรับน้ำหนักด้วยการเรียนรู้แบบแพร่กลับ

### 2.3.5 การพยากรณ์อนุกรมเวลาด้วยเครือข่ายประสาทเทียม

การนำเครือข่ายประสาทเทียมมาใช้กับการพยากรณ์อนุกรมเวลาเชิงเดี่ยว (Univariate time series) เป็นการนำเครือข่ายประสาทเทียมที่มีโครงสร้างไปข้างหน้าแบบหลายชั้น ด้วยการเรียนรู้แบบแพร่กลับ มาใช้ในการพยากรณ์ค่าสังเกตล่วงหน้า โดยมีเอาต์พุตของเครือข่ายเป็นค่าสังเกต ณ เวลา  $t$  ( $y_t$ ) และอินพุตเวกเตอร์เป็นค่าสังเกต ณ เวลาไล่หลังไป 1 ถึง  $p$  คาบเวลา ( $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ ) ดังนั้น ชุดข้อมูลฝึกสอน (Training data set) ที่ใช้สร้างเครือข่ายจะประกอบด้วย

ข้อมูล 2 ชุด ได้แก่ ชุดที่ 1 คือ อินพุตเวกเตอร์ เป็นค่าสังเกต ณ เวลา \$t\$ หลังไป 1 ถึง \$p\$ คาบเวลา และชุดที่ 2 คือ ทาร์เก็ตหรือเอาต์พุต เป็นค่าสังเกต ณ เวลา \$t\$ ซึ่งเป็นค่าตัวเลข 1 ค่า ทำให้มีนิวรอนในชั้นเอาต์พุต 1 นิวรอน และจากงานวิจัยของ Wang และ Meng (2012) กล่าวว่าเมื่อนำเครือข่ายประสาทเทียมมาใช้ในการพยากรณ์อนุกรมเวลา ควรจะมีชั้นซ่อนเร้นเพียงชั้นเดียว เพื่อไม่ให้เกิดปัญหาโอเวอร์ฟิตติง (Over-fitting problem) ดังนั้น โครงสร้างของเครือข่ายประสาทเทียมสำหรับการพยากรณ์อนุกรมเวลาเชิงเดี่ยว สามารถแสดงได้ ดังรูปที่ 2.21



รูปที่ 2.21 โครงสร้างของเครือข่ายประสาทเทียมสำหรับการพยากรณ์อนุกรมเวลาเชิงเดี่ยว

โครงสร้างของเครือข่าย ดังรูปที่ 2.21 เป็นการแสดงความสัมพันธ์ระหว่างเอาต์พุต ( $y_t$ ) และอินพุต ( $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ ) ซึ่งสามารถแสดงเป็นสมการคณิตศาสตร์ได้ดังสมการที่ 2.60 (Wang and Meng, 2012)

$$y_t = w_0 + \sum_{j=1}^q w_j g(w_{0j} + \sum_{i=1}^p w_{ij} y_{t-i}) + e_t \quad \text{----- (2.60)}$$

เมื่อ  $w_j$  ( $j = 1, 2, \dots, q$ ) เป็นน้ำหนักชั้นแนบระหว่างชั้นซ่อนเร้นและชั้นเอาต์พุต และ  $w_{i,j}$  ( $i = 0, 1, 2, \dots, p; j = 1, 2, \dots, q$ ) เป็นน้ำหนักชั้นแนบระหว่างชั้นอินพุตและชั้นซ่อนเร้น โดยที่  $p$  เป็นจำนวนโหนดของชั้นอินพุต และ  $q$  เป็นจำนวนโหนดของชั้นซ่อนเร้น ส่วนฟังก์ชัน  $g()$  เป็นแอคติเวชันฟังก์ชันของชั้นซ่อนเร้น ซึ่งเป็นฟังก์ชันแบบซิกมอยด์ และเนื่องจากจำนวนโหนดของชั้นเอาต์พุต มี 1 โหนด ดังนั้น การนำเครือข่ายประสาทเทียมไปใช้กับการพยากรณ์อนุกรมเวลาเชิงเดี่ยว จึงเป็นการพยากรณ์ล่วงหน้า 1 คาบเวลา เท่านั้น

จากงานวิจัยของ Wang และ Meng (2012) กล่าวว่า การนำเครือข่ายประสาทเทียมไปใช้สำหรับการพยากรณ์อนุกรมเวลา ควรกำหนดให้โครงสร้างของเครือข่ายอยู่ในรูปแบบที่ง่ายไม่ซับซ้อน นั่นคือ ควรกำหนดให้จำนวนโหนดของชั้นซ่อนเร้น มีจำนวนน้อย ( $q$  มีค่าน้อย) เพราะการกำหนดจำนวนโหนดของชั้นซ่อนเร้นมากเกินไป อาจทำให้เกิดปัญหาโอเวอร์ฟิตติง คือ ในขั้นตอนการสร้างตัวแบบด้วยชุดข้อมูลฝึกสอน จะมีความคลาดเคลื่อนต่ำ แต่เมื่อได้ตัวแบบ แล้วนำไปทดสอบกับชุดข้อมูลทดสอบ จะให้ค่าความคลาดเคลื่อนสูง ซึ่งปัญหานี้ เป็นปัญหาที่พบบ่อยในกระบวนการสร้างตัวแบบเครือข่ายประสาทเทียม

## 2.4 การพยากรณ์อนุกรมเวลาด้วยซัพพอร์ตเวกเตอร์แมชชีนและจินตคณิตอัลกอริทึม

ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine: SVM) เป็นวิธีการด้านแมชชีนเลิร์นนิง (Machine learning) ที่ถูกนำเสนอ โดย Vapnik (1995) เพื่อใช้ในการจำแนกข้อมูลที่มี 2 คลาส (Class) โดยมีเป้าหมาย คือ สร้างเส้นไฮเปอร์เพลนที่ดีที่สุด (Optimal hyperplane) เพื่อจำแนกข้อมูลเป็น 2 คลาส ซึ่งเส้นไฮเปอร์เพลนที่ดีที่สุด คือ เส้นที่มีมาร์จิน (Margin) ทั้ง 2 ด้าน กว้างมากที่สุด ขณะเดียวกัน มาร์จินแต่ละด้านจะต้องตัดผ่านหรือครอบคลุมข้อมูลนำเข้าให้น้อยที่สุด เปรียบเสมือนการสร้างถนนตัดผ่าน 2 หมู่บ้าน ถนนที่ใช้แบ่งแยกหมู่บ้านได้ดีที่สุด คือ ถนนที่มีความกว้างมากที่สุด และทำลายบ้านเรือนทั้ง 2 ฝั่งของถนนให้น้อยที่สุดด้วย ซึ่งซัพพอร์ตเวกเตอร์แมชชีน สามารถนำมาใช้กับการพยากรณ์อนุกรมเวลาได้ เรียกว่า ซัพพอร์ตเวกเตอร์รีเกรสชัน (Support Vector Regression: SVR) โดยสร้างสมการไฮเปอร์เพลน ที่แสดงความสัมพันธ์ระหว่างอินพุตเวกเตอร์และเอาต์พุต และใช้ชุดข้อมูลฝึกสอนที่ประกอบด้วยอินพุตเวกเตอร์และทาร์เก็ตในการสร้างสมการไฮเปอร์เพลน เมื่อได้สมการไฮเปอร์เพลน จะสามารถพยากรณ์ค่าเอาต์พุตได้จากอินพุตเวกเตอร์ที่ใส่เข้าไป ซึ่งความแม่นยำในการพยากรณ์ของซัพพอร์ตเวกเตอร์รีเกรสชัน ขึ้นอยู่กับการกำหนดค่าพารามิเตอร์

ของซัพพอร์ตเวกเตอร์รีเกรสชันให้เหมาะสม โดยพารามิเตอร์ที่สำคัญคือ ค่า  $C$  (Cost) ค่า  $\mathcal{E}$  (Epsilon) และค่าพารามิเตอร์ของเคอร์เนล (Kernel parameter) ซึ่งเนื้อหาส่วนท้ายของหัวข้อนี้ จะกล่าวถึงการนำจีเนติกอัลกอริทึม (Genetic algorithm) มาใช้ในการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดสำหรับซัพพอร์ตเวกเตอร์รีเกรสชัน ดังนั้นเนื้อหาในหัวข้อนี้ ประกอบด้วย ไฮเปอร์เพลนและมาร์จิน ซัพพอร์ตเวกเตอร์แมชชีนกับกรณีที่เป็นการแบ่งแยกเชิงเส้น ซัพพอร์ตเวกเตอร์แมชชีนแบบซอฟต์แวร์ มาร์จิน เคอร์เนลฟังก์ชันกับซัพพอร์ตเวกเตอร์แมชชีน ตัวอย่างการจำแนกข้อมูลด้วยซัพพอร์ตเวกเตอร์แมชชีน ซัพพอร์ตเวกเตอร์รีเกรสชันกับการพยากรณ์อนุกรมเวลา และจีเนติกอัลกอริทึมกับการสร้างซัพพอร์ตเวกเตอร์รีเกรสชันที่เหมาะสมที่สุด

#### 2.4.1 ไฮเปอร์เพลนและมาร์จิน

ซัพพอร์ตเวกเตอร์แมชชีน เป็นเทคนิคที่ใช้ในการจำแนกชุดข้อมูลที่อยู่ในรูปแบบ  $D = \{x_i, y_i\}_{i=1}^n$  โดย  $x_i$  คือข้อมูลอินพุตจำนวน  $n$  ตัว ซึ่งเป็นจุดจำนวน  $n$  จุด ในปริภูมิมิติ  $d$  ( $D$ -dimension space) และต้องการจำแนกข้อมูล  $x_i$  เป็นคลาส ( $y_i$ ) 2 คลาส คือคลาสบวกและคลาสลบ ( $y_i \in \{+1, -1\}$ ) ดังนั้น จุดประสงค์หลักของซัพพอร์ตเวกเตอร์แมชชีน คือ การหาเส้นไฮเปอร์เพลนที่ใช้สำหรับแบ่งข้อมูลออกเป็น 2 คลาส ได้ดีที่สุด ซึ่งเนื้อหาในหัวข้อนี้ จะอธิบายความหมายของไฮเปอร์เพลน ระยะทางของจุดข้อมูลไปยังไฮเปอร์เพลน มาร์จินและซัพพอร์ตเวกเตอร์ของไฮเปอร์เพลน ดังนี้

1. ไฮเปอร์เพลน (Hyperplane) คือ ระนาบที่ใช้ในการแบ่งแยกจุด  $x_i$  ออกเป็น 2 คลาส ( $y_i$ ) ซึ่งไฮเปอร์เพลน ( $h(x)$ ) เป็นฟังก์ชันเชิงเส้นใน  $d$  มิติ ที่กำหนดได้ด้วยสมการที่ 2.61

$$\begin{aligned} h(x) &= w^T x + b && \text{----- (2.61)} \\ &= w_1x_1 + w_2x_2 + \dots + w_ax_a + b \end{aligned}$$

เมื่อ  $w$  คือ เวกเตอร์น้ำหนัก (Weight vector) ใน  $d$  มิติ และ  $b$  คือ ค่าไบแอส (Bias) มีค่าเป็นจำนวนจริง โดยจุดทุกจุดที่อยู่บนไฮเปอร์เพลน เป็นจุดที่ทำให้สมการไฮเปอร์เพลนตามสมการที่ 2.61 มีค่าเท่ากับ 0 หรือแสดงได้ดังสมการที่ 2.62

$$h(x) = w^T x + b = 0 \quad \text{----- (2.62)}$$

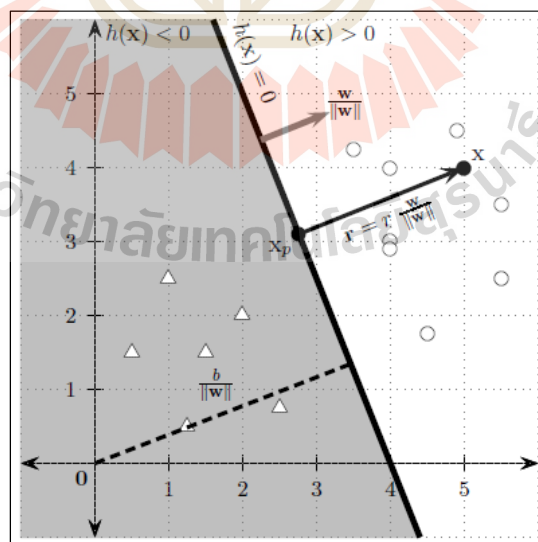
เนื่องจาก ไฮเปอร์เพลน เป็นระนาบที่ใช้แบ่งแยกข้อมูล  $x_i$  หรือเรียกว่า อินพุต  
เวกเตอร์ ออกเป็นคลาส  $y_i$  ที่มี 2 คลาส คือ +1 และ -1 ถ้าข้อมูลที่นำเข้ามาจำแนก สามารถจำแนก  
แบบลิเนียร์ได้ จะสามารถหาไฮเปอร์เพลนที่เป็นเส้นตรง สำหรับแบ่งแยกข้อมูลได้ โดยจุดใด ๆ ที่มีค่า  
 $y_i = -1$  จะได้ค่า  $h(x_i) < 0$  และจุดใด ๆ ที่มีค่า  $y_i = +1$  จะได้ค่า  $h(x_i) > 0$  หรือนิยามได้ดังสมการที่ 2.63

$$y = \begin{cases} +1 & \text{if } h(x) > 0 \\ -1 & \text{if } h(x) < 0 \end{cases} \quad \text{----- (2.63)}$$

2. ระยะทางจากจุดไปยังไฮเปอร์เพลน (Distance from point to hyperplane) ถ้า  
กำหนดให้จุด  $x \in \mathbb{R}^d$  เป็นจุดใด ๆ ที่ไม่ได้อยู่บนไฮเปอร์เพลน และจุด  $x_p$  เป็นจุดใด ๆ บนไฮเปอร์เพลน  
ที่ตั้งฉากกับจุด  $x$  ดังนั้น ระยะจากจุด  $x$  ไปตั้งฉากกับไฮเปอร์เพลนที่จุด  $x_p$  คือ  $r = x - x_p$  มีลักษณะดัง  
รูปที่ 2.22 และนิยามเป็นสมการได้ดังสมการที่ 2.64

$$x = x_p + r \quad \text{----- (2.64)}$$

$$x = x_p + r \frac{w}{\|w\|}$$



รูปที่ 2.22 ไฮเปอร์เพลนเชิงเส้น 2 มิติ (Zaki and Meira, 2014, p. 568).



เมื่อ  $r$  เป็นระยะทางจากจุด  $x$  ไปยัง  $x_p$  ซึ่งจะอยู่ในเทอมของยูนิทเวกเตอร์  $\frac{w}{\|w\|}$  ถ้า  $r$  อยู่ในทิศทางเดียวกันกับ  $w$  ค่า  $r$  จะมีค่าเป็นบวก แต่ถ้า  $r$  อยู่ในทิศทางตรงกันข้ามกับ  $w$  ค่า  $r$  จะมีค่าเป็นลบและถ้าแทนค่า  $x$  จากสมการที่ 2.64 ลงในสมการที่ 2.62 จะได้ว่า

$$\begin{aligned} h(x) &= h\left(x_p + r \frac{w}{\|w\|}\right) \\ &= w^T \left(x_p + r \frac{w}{\|w\|}\right) + b \\ &= w^T x_p + b + r \frac{w^T w}{\|w\|} \\ &= h(x_p) + r \|w\| \\ &= r \|w\| \end{aligned}$$

ดังนั้น ระยะทางจากจุด  $x$  ไปตั้งฉากกับไฮเปอร์เพลน  $r = \frac{h(x)}{\|w\|}$  ซึ่งระยะทางจะต้องมีค่าเป็นบวก แต่จากรูปที่ 2.22 จุด  $x$  ใด ๆ อาจอยู่ในคลาส +1 โดยมี  $h(x) > 0$  หรืออยู่ในคลาส -1 ที่มี  $h(x) < 0$  ดังนั้น เพื่อให้ระยะทางมีค่าเป็นบวกเสมอ จึงคูณคลาส  $y_i$  เข้าไป ได้เป็นสมการที่ 2.65

$$\delta = y_i r = \frac{y_i h(x)}{\|w\|} \quad \text{----- (2.65)}$$

กรณีที่เป็นจุดกำเนิด (Origin) หรือ  $x=0$  ระยะทางจากจุดกำเนิดไปยังไฮเปอร์เพลน นิยามได้ว่า  $r = \frac{h(0)}{\|w\|} = \frac{w^T 0 + b}{\|w\|} = \frac{b}{\|w\|}$

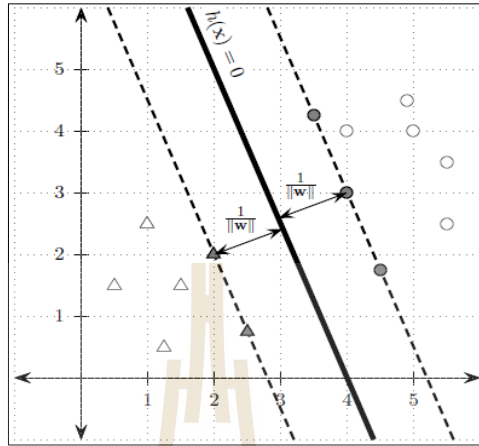
3. มارجินและซัพพอร์ตเวกเตอร์ของไฮเปอร์เพลน (Margin and Support Vector of hyperplane) ถ้ากำหนดให้ชุดข้อมูลฝึกสอนอยู่ในรูป  $D = \{x_i, y_i\}_{i=1}^n$  เมื่อ  $y_i \in \{+1, -1\}$  และไฮเปอร์เพลน  $h(x) = 0$  ดังนั้น จากสมการที่ 2.65 จะได้ระยะจากจุด  $x_i$  ใด ๆ ไปยังไฮเปอร์เพลน ดังนี้

$$\delta_i = \frac{y_i h(x_i)}{\|w\|} = \frac{y_i (w^T x_i + b)}{\|w\|}$$

สำหรับจุดทั้งหมด  $n$  จุด สามารถนิยามมارجินของการจำแนกเชิงเส้นเป็นแนวของจุด  $x_i$  ที่อยู่ใกล้กับไฮเปอร์เพลนมากที่สุด ดังรูปที่ 2.23 และนิยามสมการ ได้ดังสมการที่ 2.66



$$\delta^* = \min_{x_i} \left\{ \frac{y_i(w^T x_i + b)}{\|w\|} \right\} \quad \text{----- (2.66)}$$



รูปที่ 2.23 มارجินของไฮเปอร์เพลน (Zaki and Meira, 2014, p. 570).

จากรูปที่ 2.23 สามารถนิยามซัพพอร์ตเวกเตอร์ได้ว่า ซัพพอร์ตเวกเตอร์คือ จุดทุกจุดที่อยู่บนแนวมาร์จิน โดยมีระยะตั้งฉากกับไฮเปอร์เพลนเท่ากับ  $\frac{1}{\|w\|}$  หรือกล่าวอีกนัยได้ว่า ถ้าให้  $x^*$  เป็นซัพพอร์ตเวกเตอร์ จะต้องสอดคล้องกับเงื่อนไข ดังนี้

$$\delta^* = \frac{y^*(w^T x^* + b)}{\|w\|}$$

เมื่อ  $y^*$  คือคลาสของ  $x^*$  และเทอม  $y^*(w^T x^* + b)$  เป็นระยะทางสัมบูรณ์ของซัพพอร์ตเวกเตอร์ไปยังไฮเปอร์เพลน โดยมี  $\|w\|$  เป็นส่วนหาร เพื่อให้เป็นระยะทางสัมพัทธ์ในเทอมของน้ำหนัก  $w$

#### 2.4.2 ซัพพอร์ตเวกเตอร์แมชชีนกับกรณีที่เป็นการแบ่งแยกเชิงเส้น

แนวคิดพื้นฐานของการจำแนกข้อมูลด้วย SVM คือ การเลือกไฮเปอร์เพลนที่เหมาะสมที่สุด โดยเป็นไฮเปอร์เพลนที่มีความกว้างของมาร์จินมากที่สุด (Maximum margin hyperplane) ซึ่งถูกกำหนดโดยค่าเวกเตอร์ถ่วงน้ำหนัก  $w$  และ ไบแอส  $b$  โดยสมการของไฮเปอร์เพลนที่เหมาะสมที่สุด นิยามได้ดังสมการที่ 2.62 ถ้า  $\delta_h^*$  คือ ค่ามาร์จินสำหรับไฮเปอร์เพลน  $h(x) = 0$

ดังนั้นจุดมุ่งหมายของการหาไฮเปอร์เพลนที่ดีที่สุดและเหมาะสมที่สุดคือการหาค่า  $h^*$  ที่มี  $\delta_h^*$  มากที่สุด ซึ่งแสดงได้ดังสมการที่ 2.67

$$h^* = \arg \max_h \{\delta_h^*\} = \arg \max_{w,b} \left\{ \frac{1}{\|w\|} \right\} \quad \text{----- (2.67)}$$

เป้าหมายของ SVM คือการหาไฮเปอร์เพลนที่มีมาร์จินกว้างมากที่สุด ซึ่งระยะห่างของไฮเปอร์เพลนกับมาร์จินเท่ากับ  $\frac{1}{\|w\|}$  ภายใต้เงื่อนไข  $y_i(w^T x_i + b) \geq 1$  สำหรับทุกจุด  $X_i \in D$  และแทนที่จะทำการหาค่ามาร์จินที่กว้างที่สุดด้วยค่า  $\frac{1}{\|w\|}$  จะทำการหาค่า  $\|w\|$  ที่น้อยที่สุด ด้วยสูตรที่นิยามดังสมการที่ 2.68

$$\text{Objective Function: } \min_{w,b} \left\{ \frac{\|w\|^2}{2} \right\} \quad \text{----- (2.68)}$$

$$\text{Linear Constraints: } y_i(w^T x_i + b) \geq 1, \quad \forall x_i \in D$$

สำหรับการค้นหาไฮเปอร์เพลนที่เหมาะสมที่สุด จะใช้วิธีการแก้ปัญหาแบบดual (Dual problem) ด้วยการใช้ตัวคูณลากรองจ์ (Lagrange multiplier:  $\alpha_i$ ) คุณเข้าไปในสมการที่ 2.68 ภายใต้เงื่อนไขของ Karush-Kuhn-Tucker (KKT) ดังนั้น ผลจากการคูณด้วยตัวคูณลากรองจ์ ได้เป็นสมการที่ 2.69

$$\alpha_i(y_i(w^T x_i + b) - 1) = 0 \quad \text{และ } \alpha_i \geq 0 \quad \text{----- (2.69)}$$

จากการใช้ตัวคูณลากรองจ์ ทำให้สามารถปรับฟังก์ชันวัตถุประสงค์ (Objective function) ของสมการที่ 2.68 ได้ใหม่ โดยเรียกว่า ลากรองเจียน (Lagrangian) ดังสมการที่ 2.70

$$\min L = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i (y_i(w^T x_i + b) - 1) \quad \text{----- (2.70)}$$

จากสมการที่ 2.70 เมื่อต้องการหาค่าต่ำสุดของ L จะหาค่าอนุพันธ์ของ L เทียบกับค่าเวกเตอร์ถ่วงน้ำหนัก  $w$  และค่าไบแอส  $b$  ให้เท่ากับศูนย์ ทำให้ได้เป็นสมการที่ 2.71 และ 2.72 ดังนี้

$$\frac{\partial}{\partial w} L = w - \sum_{i=1}^n \alpha_i y_i x_i = 0 \quad \text{หรือ} \quad w = \sum_{i=1}^n \alpha_i y_i x_i \quad \text{----- (2.71)}$$

$$\frac{\partial}{\partial b} L = \sum_{i=1}^n \alpha_i y_i = 0 \quad \text{----- (2.72)}$$

สมการที่ 2.71 และ 2.72 เป็นสมการที่สำคัญสำหรับการหาค่าเวกเตอร์ถ่วงน้ำหนัก  $w$  ที่เหมาะสมที่สุด โดยสมการที่ 2.71 หมายความว่า ค่า  $w$  คือผลรวมเชิงเส้นของจุดข้อมูล  $x_i$  ที่คูณอยู่กับตัวคูณลากรองจ์  $\alpha_i$  ส่วนสมการที่ 2.72 หมายความว่า ผลรวมของตัวคูณลากรองจ์เครื่องหมาย  $(\alpha_i y_i)$  มีค่าเท่ากับศูนย์ และเมื่อนำสมการที่ 2.71 และ 2.72 แทนที่ลงไปในสมการที่ 2.70 จะได้ฟังก์ชันวัตถุประสงค์ เรียกว่า คู่อัลลากรองจ์ (Dual Lagrangian objective function) ที่อยู่ในเทอมของตัวคูณลากรองจ์ ดังนี้

$$\begin{aligned} L_{\text{dual}} &= \frac{1}{2} w^T w - w^T \left( \sum_{i=1}^n \alpha_i y_i x_i \right) - b \sum_{i=1}^n \alpha_i y_i + \sum_{i=1}^n \alpha_i \\ &= -\frac{1}{2} w^T w + \sum_{i=1}^n \alpha_i \\ &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j \end{aligned}$$

$$\text{Objective Function: } \max_{\alpha} L_{\text{dual}} = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$\text{Linear Constraints: } \alpha_i \geq 1, \quad \forall i \in D, \text{ and } \sum_{i=1}^n \alpha_i y_i = 0 \quad \text{----- (2.73)}$$

การหาค่าเวกเตอร์ถ่วงน้ำหนัก  $w$  และ ค่าไปแอต  $b$  จะต้องหาค่า  $\alpha_i$  เมื่อ  $i = 1, \dots, n$  ก่อน จากนั้น จึงนำไปหาค่า  $w$  และค่า  $b$  ได้ และจากสมการที่ 2.69 จะได้ผลลัพธ์เป็น 2 กรณี คือ

$$\text{i) } \alpha_i = 0, \text{ or}$$

$$\text{ii) } y_i(w^T x_i + b) - 1 = 0, \text{ which implies } y_i(w^T x_i + b) = 1$$

จากผลลัพธ์ทั้ง 2 กรณี สรุปได้ว่า ถ้า  $\alpha_i > 0$  ดังนั้น  $y_i(w^T x_i + b) = 1$  โดยจุด  $x_i$  คือ ซัพพอร์ตเวกเตอร์ ในทางกลับกัน ถ้า  $y_i(w^T x_i + b) > 1$  ดังนั้น  $\alpha_i = 0$  หมายความว่า

จุดใด ๆ ที่ไม่ใช่ซัพพอร์ตเวกเตอร์ จะมีค่า  $\alpha_i = 0$  และเนื่องจากค่า  $\alpha_i$  จะต้องถูกนำไปคูณกับทุกจุดข้อมูล ดังนั้น สามารถคำนวณค่าเวกเตอร์ถ่วงน้ำหนัก  $w$  ได้ดังสมการที่ 2.74

$$w = \sum_{i, \alpha_i > 0} \alpha_i y_i x_i \quad \text{----- (2.74)}$$

จากสมการที่ 2.74 ค่า  $w$  ที่ได้จะเป็นผลรวมเชิงเส้นของซัพพอร์ตเวกเตอร์ทั้งหมด ซึ่ง  $\alpha_i$  แต่ละตัวจะบ่งบอกถึงน้ำหนักของซัพพอร์ตเวกเตอร์ ดังนั้นหากค่า  $\alpha_i = 0$  นั้นหมายถึงจุดนั้น ๆ ไม่ใช่ซัพพอร์ตเวกเตอร์ จึงไม่จำเป็นต้องคำนวณหาค่า  $w$  และการคำนวณค่าไบแอส  $b$  จะต้องคำนวณค่า  $b$  ของซัพพอร์ตเวกเตอร์ แต่ละตัวก่อน ดังนี้

$$\alpha_i (y_i (w^T x_i + b) - 1) = 0 \quad \text{----- (2.75)}$$

$$y_i (w^T x_i + b) = 1 \quad \text{----- (2.76)}$$

$$b_i = \frac{1}{y_i} - w^T x_i = y_i - w^T x_i \quad \text{----- (2.77)}$$

และค่าเฉลี่ยของไบแอสจากซัพพอร์ตเวกเตอร์ทั้งหมด เป็นดังสมการที่ 2.78

$$b = \text{avg}_{\alpha_i > 0} \{b_i\} \quad \text{----- (2.78)}$$

การจำแนกข้อมูลด้วยซัพพอร์ตเวกเตอร์แมชชีน มีจุดประสงค์สำคัญ คือ การหาค่าไฮเปอร์เพลนที่เหมาะสมที่สุด ซึ่งสมการไฮเปอร์เพลน คือ  $h(x) = w^T x + b$  และจุดข้อมูลใหม่ที่น่าเข้ามาจำแนกนั้น สามารถทำนายได้ดังสมการที่ 2.79

$$\hat{y} = \text{sign}(h(z)) = \text{sign}(w^T z + b) \quad \text{----- (2.79)}$$

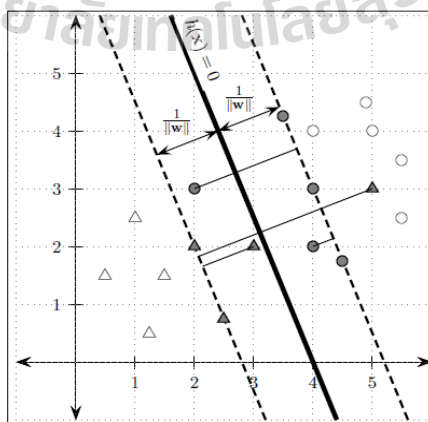
โดยค่า  $\text{sign}(\cdot)$  จะให้ค่าเป็น +1 ถ้าผลลัพธ์เป็นบวก และให้ค่าเป็น -1 เมื่อผลลัพธ์เป็นลบ

### 2.4.3 ซัพพอร์ตเวกเตอร์แมชชีนแบบซอฟต์มาร์จิน

จากหัวข้อที่กล่าวมาก่อนหน้า ได้สมมติว่า การจำแนกชุดข้อมูลฝึกสอน กระทำได้อย่างสมบูรณ์ นั่นคือ ซัพพอร์ตเวกเตอร์ทุกตัว ต้องอยู่บนเส้นมาร์จิน และซัพพอร์ตเวกเตอร์ต้องอยู่ในคลาสที่ถูกต้อง เท่านั้น แต่ในหัวข้อนี้ จะกล่าวถึงการจำแนกที่อนุญาตให้ซัพพอร์ตเวกเตอร์ไม่ต้องอยู่บนเส้นมาร์จิน และอนุญาตให้ซัพพอร์ตเวกเตอร์บางตัวอยู่ผิดคลาสได้ ซึ่งทำให้ซัพพอร์ตเวกเตอร์แมชชีนมีความยืดหยุ่นในการจำแนกมากขึ้น โดยเรียกว่า ซัพพอร์ตเวกเตอร์แมชชีนแบบซอฟต์มาร์จิน (Soft margin SVM) ดังแสดงในรูปที่ 4.3 ซึ่งวิธีการนี้จะทำการเพิ่มตัวแปรที่ใช้กำหนดความยืดหยุ่นของมาร์จิน เรียกว่า ตัวแปรสแลค (Slack variable:  $\xi_i$ ) และทำให้เงื่อนไข  $y_i(w^T x_i + b) \geq 1$  จากสมการที่ 2.68 เปลี่ยนเป็นสมการที่ 2.80

$$y_i(w^T x_i + b) \geq 1 - \xi_i \quad \text{----- (2.80)}$$

จากสมการที่ 2.80 ค่าของตัวแปรสแลค จะมีค่ามากกว่าหรือเท่ากับศูนย์ ( $\xi_i \geq 0$ ) ซึ่งค่า  $\xi_i$  แสดงถึงขนาดการล่งล้ำเขตของจุดที่เป็นซัพพอร์ตเวกเตอร์ ภายในบริเวณของมาร์จิน โดยค่า  $\xi_i$  แบ่งเป็น 3 ประเภท ถ้า  $\xi_i = 0$  แสดงว่า จุด  $x_i$  อยู่ห่างจากไฮเปอร์เพลนอย่างน้อย  $\frac{1}{\|w\|}$  ถ้า  $0 < \xi_i < 1$  แสดงว่าจุดที่เป็นซัพพอร์ตเวกเตอร์อยู่ภายในมาร์จิน และจำแนกได้ถูกคลาส หรืออยู่ในฝั่งของไฮเปอร์เพลนที่ถูกต้อง แต่ถ้า  $\xi_i \geq 1$  แสดงว่ายอมให้จุดที่เป็นซัพพอร์ตเวกเตอร์ถูกจำแนกผิดคลาส หรืออยู่ผิดฝั่งของไฮเปอร์เพลน



รูปที่ 2.24 ซอฟต์มาร์จินไฮเปอร์เพลน (Zaki and Meira, 2014, p. 577).

จากรูปที่ 2.24 จุดที่ระบายทึบ เป็นจุดที่เป็นซัพพอร์ตเวกเตอร์ โดยมีระยะห่างของแนวมาร์จินกับไฮเปอร์เพลนเท่ากับ  $\frac{1}{\|w\|}$  และจุดที่เป็นซัพพอร์ตเวกเตอร์ ซึ่งมีค่า  $\xi_i$  มากกว่าศูนย์ จะมีเส้นที่ลากไปตั้งฉากกับแนวมาร์จินเป็นเส้นทึบ โดยเรียกแนวมาร์จินที่มีค่า  $\xi_i > 0$  ในลักษณะนี้ว่า ซอฟต์มาร์จิน (Soft margin) ซึ่งเป้าหมายหลักของซัพพอร์ตเวกเตอร์แมชชีน ยังเป็นการหาไฮเปอร์เพลนที่เหมาะสมที่สุด และมีมาร์จินกว้างที่สุด โดยอนุญาตให้มีตัวแปรสแลคได้ แต่ต้องเป็นค่าที่เหมาะสม ไม่มากหรือน้อยเกินไป ดังนั้น ฟังก์ชันวัตถุประสงค์ เมื่อมีตัวแปรสแลค จึงเปลี่ยนใหม่เป็นดังสมการที่ 2.81

$$\text{Objective Function: } \min_{w, b, \xi_i} \left\{ \frac{\|w\|^2}{2} + C \sum_{i=1}^n (\xi_i)^k \right\} \quad \text{----- (2.81)}$$

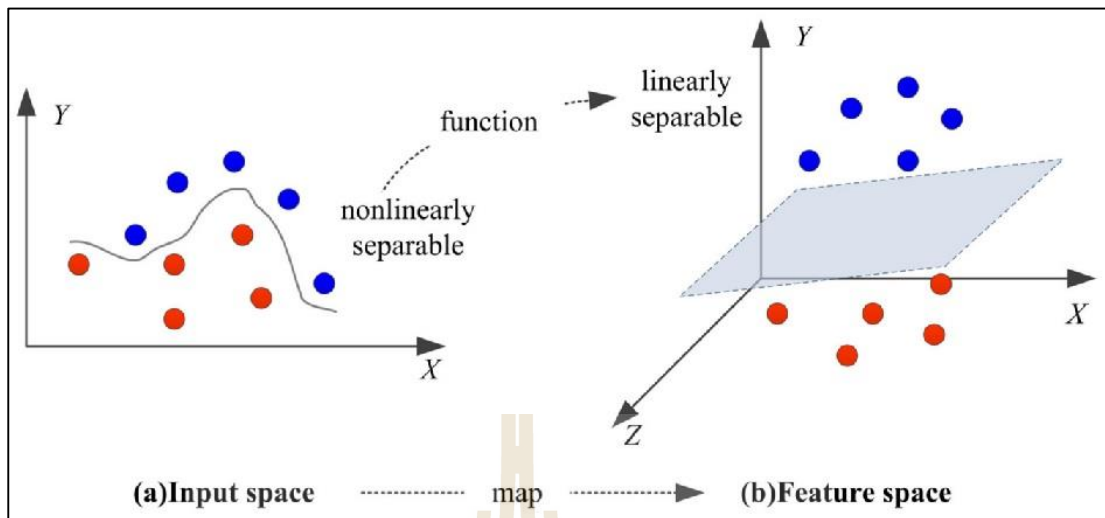
$$\text{Linear Constraints: } y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \forall x_i \in D$$

$$\xi_i \geq 0, \quad \forall x_i \in D$$

เมื่อ  $C$  และ  $k$  เป็นค่าคงที่ ซึ่งค่า  $C$  เป็นตัวบ่งบอกถึงค่าใช้จ่าย (Cost) ของการจำแนกที่ผิดพลาด (Misclassification) เช่น ถ้าค่า  $C$  มีค่าเป็นศูนย์ หรือเข้าใกล้ศูนย์ หมายถึง ไม่อนุญาตให้จุดที่เป็นซัพพอร์ตเวกเตอร์ ถูกจำแนกผิดพลาด แต่ถ้า  $C$  มีค่ามากขึ้น เช่น  $C$  เข้าใกล้ค่าอนันต์ (Infinity) ก็จะมีผลกระทบต่อค่ามาร์จินที่เหมาะสมอย่างมาก ส่วนค่า  $k$  จะมีการตั้งค่าเป็น 2 ค่าด้วยกัน คือ 1 และ 2 ถ้า  $k=1$  จะเรียกว่า ฮินจ์ลอสส์ (Hinge loss) จุดประสงค์คือ ต้องการหาจำนวนผลรวมของค่าตัวแปรสแลคที่น้อยที่สุด แต่ถ้า  $k=2$  เรียกว่า ควอดราติกลอสส์ (Quadratic loss) จุดประสงค์คือ ต้องการหาค่าผลรวมยกกำลังสองของค่าตัวแปรสแลคที่น้อยที่สุด

#### 2.4.4 เคอร์เนลฟังก์ชันกับซัพพอร์ตเวกเตอร์แมชชีน

ในกรณีที่ไม่สามารถจำแนกข้อมูลโดยใช้ซัพพอร์ตเวกเตอร์แมชชีนแบบเชิงเส้น จะต้องทำการแปลงข้อมูลให้อยู่ในมิติข้อมูลที่สูงขึ้น (High-dimensional feature space) ด้วยการใช้เคอร์เนลฟังก์ชัน (Kernel function) หลักการคือ ใช้เคอร์เนลฟังก์ชัน ทำการแปลง (Map) จุดข้อมูล  $x_i$  ในอินพุตสเปซขนาด  $D$  มิติ (D-dimensional input space) ไปเป็น จุด  $\phi(x_i)$  ในมิติข้อมูลที่สูงขึ้น ดังรูปที่ 2.25

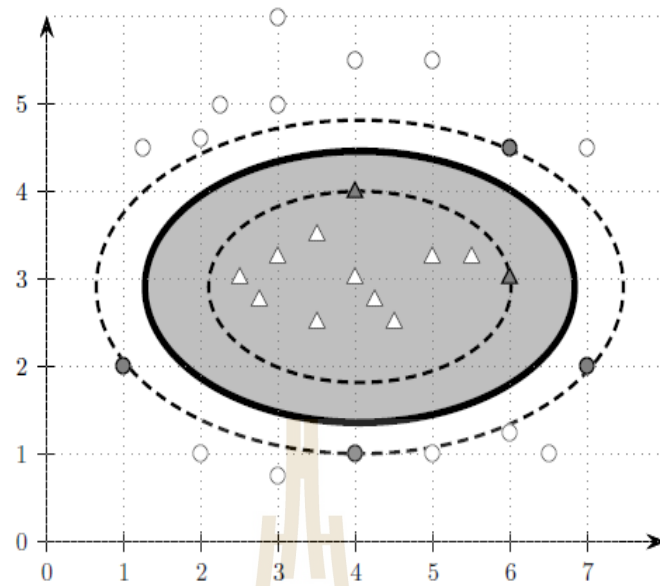


รูปที่ 2.25 การแปลงจุดข้อมูลจากอินพุตสเปซไปยังฟีเจอร์สเปซ (Cheng et. al., 2015, p. 4483).

เพื่อให้เกิดความเข้าใจ สำหรับการจำแนกข้อมูลที่ไม่เป็นเชิงเส้น ด้วยการใช้เคอร์เนลฟังก์ชัน จะยกตัวอย่างการจำแนกข้อมูลดังรูปที่ 2.26 ซึ่งเป็นการจำแนกข้อมูลออกเป็น 2 คลาส โดยที่ไม่สามารถจำแนกข้อมูลแบบเชิงเส้นได้ จึงต้องใช้เคอร์เนลฟังก์ชันในการแปลงข้อมูลจากอินพุตสเปซไปยังฟีเจอร์สเปซ โดยกำหนดให้อินพุตสเปซขนาด 2 มิติ  $X = (x_1, x_2)^T$  แปลงไปเป็นฟีเจอร์สเปซ (Feature space) ที่มีมิติ  $(x_1, x_2, x_1^2, x_2^2, x_1x_2)$  โดยใช้สูตรการแปลง  $\phi(X) = (\sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2)^T$  และจากรูปที่ 2.26 จะเห็นว่า รูปวงรีที่เป็นเส้นทึบสีดำสามารถจำแนกคลาसออกเป็น 2 คลาสได้ (แสดงด้วยรูปสามเหลี่ยมและวงกลม) ส่วนจุดที่เป็นซัพพอร์ตเวกเตอร์ เป็นจุดที่ระบายสีทึบ ซึ่งอยู่บนเส้นมาร์จิน (แสดงด้วยเส้นประ) ดังนั้น การใช้เคอร์เนลฟังก์ชัน ในการจำแนกข้อมูลที่ไม่เป็นเชิงเส้น จะต้องใช้เคอร์เนลฟังก์ชัน  $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$  เพื่อแปลงชุดข้อมูลที่อยู่ในรูปแบบ  $D = \{x_i, y_i\}_{i=1}^n$  ไปเป็นชุดข้อมูลใหม่ในฟีเจอร์สเปซ ที่อยู่ในรูปแบบ  $D_\phi = \{\phi(x_i), y_i\}_{i=1}^n$  และฟังก์ชันวัตถุประสงค์ ตามสมการที่ 2.73 จะถูกเปลี่ยนไปเป็นสมการที่ 2.82

$$\text{Objective Function: } \max_{\alpha} L_{\text{dual}} = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

$$\text{Linear Constraints: } 0 \leq \alpha_i \leq C, \forall i \in D, \text{ and } \sum_{i=1}^n \alpha_i y_i = 0 \quad \text{----- (2.82)}$$



รูปที่ 2.26 การจำแนกข้อมูลไม่เป็นเชิงเส้นของ SVM (Zaki and Meira, 2014, p. 584).

สำหรับเคอร์เนลฟังก์ชันมีหลายแบบให้เลือกใช้งาน ในกรณีที่ไม่สามารถจำแนกข้อมูลแบบเชิงเส้น จะสามารถเลือกใช้เคอร์เนลฟังก์ชัน เพื่อจำแนกข้อมูลได้ แต่ชุดข้อมูลแต่ละชุด มีการกระจายและมีข้อมูลที่แตกต่างกัน โดยส่วนใหญ่แล้ว นักวิจัยจะเลือกใช้เคอร์เนลฟังก์ชันหลาย ๆ แบบ จากนั้น นำผลความแม่นยำมาเปรียบเทียบกัน ซึ่งอาจบอกไม่ได้ว่า เคอร์เนลฟังก์ชันตัวใดเหมาะกับข้อมูลประเภทใด ดังนั้น จึงต้องมีการทดลองกับเคอร์เนลฟังก์ชันมากกว่าหนึ่งแบบ และเคอร์เนลฟังก์ชันที่นิยมใช้ในปัจจุบัน สามารถสรุปได้ดังตารางที่ 2.6

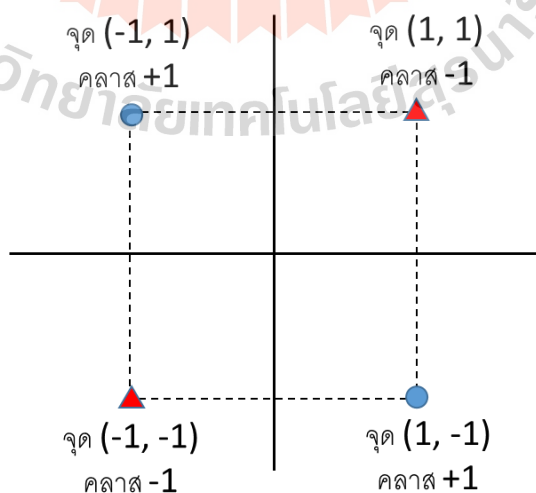


ตารางที่ 2.6 เคอร์เนลฟังก์ชันที่นิยมใช้ร่วมกับซัพพอร์ตเวกเตอร์แมชชีน (อาทิตย์ ศรีแก้ว, 2552. หน้า 386).

Kernel Function	$K(x_i, x_j)$
Radial Basis Function (RBF)	$\exp\left(\frac{-\ x_i - x_j\ ^2}{2\sigma^2}\right)$
Polynomial Function	$(1 + x_i^T \cdot x_j)^d$
Sigmoidal Function	$\tanh(\rho(x_i^T \cdot x_j) + \theta)$
Linear Function	$x_i^T \cdot x_j$
Fourier Series Function	$\frac{\sin(N + \frac{1}{2})(x_i - x_j)}{\sin(\frac{1}{2}(x_i - x_j))}$

#### 2.4.5 ตัวอย่างการจำแนกข้อมูลด้วยซัพพอร์ตเวกเตอร์แมชชีน

เพื่อให้เกิดความเข้าใจในการนำเทคนิค SVM มาใช้ในการจำแนกข้อมูล จะยกตัวอย่างการจำแนกข้อมูลไบนารี (Binary) แบบเอ็กซ์คลูซีฟอออร์ (Exclusive OR: XOR) ซึ่งเป็นจุดข้อมูลจำนวน 4 จุด โดยจำแนกเป็น 2 คลาส คือ +1 และ -1 ดังรูปที่ 2.27



รูปที่ 2.27 ปัญหาการจำแนกข้อมูล XOR

จากรูปที่ 2.27 ทำให้ทราบว่า ไม่สามารถจำแนกจุดข้อมูล 4 จุด ได้ด้วยการจำแนกเชิงเส้น ดังนั้นจึงต้องใช้เคอร์เนลฟังก์ชัน เพื่อทำการแปลงจุดข้อมูลไปยังมิติที่สูงขึ้น โดยเลือกใช้โพลิโนเมียลฟังก์ชัน (Polynomial function) ที่มีดีกรี (Degree)  $d = 2$  เป็นเคอร์เนลฟังก์ชัน ดังนั้น  $K(x_i, x_j) = (1 + x_i^T \cdot x_j)^2$  และชุดข้อมูลในรูปของอินพุตเวกเตอร์และทาร์เก็ต (เอาต์พุต) สำหรับสร้างสมการไฮเปอร์เพลนที่เหมาะสมที่สุด (Optimal hyperplane) เป็นดังตารางที่ 2.7

ตารางที่ 2.7 อินพุตเวกเตอร์และทาร์เก็ตของปัญหา XOR

Input Vector	Target
(1, 1)	-1
(1, -1)	+1
(-1, 1)	+1
(-1, -1)	-1

สิ่งที่ต้องการค้นหา คือ สมการไฮเปอร์เพลนที่เหมาะสมที่สุดในการจำแนกข้อมูลของปัญหา XOR โดยใช้เคอร์เนลฟังก์ชัน  $K(x_i, x_j) = (1 + x_i^T \cdot x_j)^2 = (1 + x_{i1}^2 x_{j1}^2 + 2x_{i1}x_{i2}x_{j1}x_{j2} + x_{i2}^2 x_{j2}^2 + 2x_{i1}x_{j1} + 2x_{i2}x_{j2})$  ซึ่งนำมาเขียนในรูปเมทริกซ์  $K$  ได้ดังนี้

$$K = \begin{bmatrix} X_1X_1 & X_1X_2 & X_1X_3 & X_1X_4 \\ X_2X_1 & X_2X_2 & X_2X_3 & X_2X_4 \\ X_3X_1 & X_3X_2 & X_3X_3 & X_3X_4 \\ X_4X_1 & X_4X_2 & X_4X_3 & X_4X_4 \end{bmatrix} = \begin{bmatrix} 9 & 1 & 1 & 1 \\ 1 & 9 & 1 & 1 \\ 1 & 1 & 9 & 1 \\ 1 & 1 & 1 & 9 \end{bmatrix}$$

และจากฟังก์ชันวัตถุประสงค์ ตามสมการที่ 2.79

$$\max_{\alpha} L_{\text{dual}} = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

จะได้ว่า

$$y_i y_j K = \begin{bmatrix} y_1 y_1^9 & y_1 y_2^1 & y_1 y_3^1 & y_1 y_4^1 \\ y_2 y_1^1 & y_2 y_2^9 & y_2 y_3^1 & y_2 y_4^1 \\ y_3 y_1^1 & y_3 y_2^1 & y_3 y_3^9 & y_3 y_4^1 \\ y_4 y_1^1 & y_4 y_2^1 & y_4 y_3^1 & y_4 y_4^9 \end{bmatrix}$$

$$= \begin{bmatrix} +9 & -1 & -1 & +1 \\ -1 & +9 & +1 & -1 \\ -1 & +1 & +9 & -1 \\ +1 & -1 & -1 & +9 \end{bmatrix}$$

และได้สมการของ  $L_{\text{dual}}$  ในเทอมของตัวคูณลากรองจ์ ( $\alpha$ ) ดังนี้

$$L_{\text{dual}}(\alpha) = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 - \frac{1}{2}(9\alpha_1^2 - 2\alpha_1\alpha_2 - 2\alpha_1\alpha_3 + 2\alpha_1\alpha_4 + 9\alpha_2^2 + 2\alpha_2\alpha_3 - 2\alpha_2\alpha_4 + 9\alpha_3^2 - 2\alpha_3\alpha_4 + 9\alpha_4^2)$$

จากนั้น ทำการหาอนุพันธ์ (Derivative) เทียบกับ  $\alpha_1$  ถึง  $\alpha_4$  แล้วนำมาเท่ากับศูนย์ (ต้องการหาค่าสูงสุด ซึ่งค่าสูงสุดจะมีความชันเท่ากับศูนย์) ทำให้ได้สมการในเทอมของ  $\alpha_i$  เมื่อ  $i = 1$  ถึง 4 ดังนี้

$$\begin{aligned} 9\alpha_1 - \alpha_2 - \alpha_3 + \alpha_4 &= 1 \\ -\alpha_1 + 9\alpha_2 + \alpha_3 - \alpha_4 &= 1 \\ -\alpha_1 + \alpha_2 + 9\alpha_3 - \alpha_4 &= 1 \\ \alpha_1 - \alpha_2 - \alpha_3 + 9\alpha_4 &= 1 \end{aligned}$$

แก้สมการ เพื่อหาค่า  $\alpha_1$  ถึง  $\alpha_4$  ได้ว่า  $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.125$  แล้วคำนวณเวกเตอร์น้ำหนัก  $w = \sum_{i=1}^4 \alpha_i y_i \phi(x_i)$  ซึ่ง  $\phi(x_i) = (1, x_1^2, \sqrt{2}x_1x_2, x_2^2, \sqrt{2}x_1, \sqrt{2}x_2)^T$  ดังนี้

$$\begin{aligned} w &= -0.125*(1, 1, 1.414, 1, -1.414, -1.414)^T \\ &\quad + 0.125*(1, 1, -1.414, 1, -1.414, 1.414)^T \\ &\quad + 0.125*(1, 1, -1.414, 1, 1.414, -1.414)^T \\ &\quad - 0.125*(1, 1, 1.414, 1, 1.414, 1.414)^T \end{aligned}$$

$$w = (0, 0, -0.707, 0, 0, 0)^T$$

$$\text{คำนวณค่าไบแอสจากสมการ } b = \frac{1}{n_{sv}} (\sum_{\alpha_i > 0} y_i - \sum_{\alpha_i > 0} w^T \phi(x_i)) = 0$$

ดังนั้น สมการไฮเปอร์เพลนที่เหมาะสมที่สุด (Optimal hyperplane) คือ

$$h(X) = w^T \phi(X) + b$$

$$h(X) = (0, 0, -0.707, 0, 0, 0) \phi(X)$$

#### 2.4.6 ซัพพอร์ตเวกเตอร์รีเกรสชันกับการพยากรณ์อนุกรมเวลา

ซัพพอร์ตเวกเตอร์รีเกรสชัน (SVR) เป็นการนำวิธีการของซัพพอร์ตเวกเตอร์แมชชีน (SVM) มาใช้วิเคราะห์ความถดถอยระหว่างอินพุตเวกเตอร์และตัวแปรเอาต์พุต ซึ่งนำมาใช้กับการพยากรณ์อนุกรมเวลาได้ โดยเปลี่ยนการจำแนกคลาสด้วย SVM เป็นการทำนายค่าด้วย SVR ที่มีเป้าหมาย คือ ต้องการค้นหาความสัมพันธ์เชิงเส้นระหว่างอินพุตเวกเตอร์ใน  $n$  มิติ ( $X \in \mathbb{R}^n$ ) และตัวแปรเอาต์พุต ( $y \in \mathbb{R}$ ) และเนื่องจาก SVR ดัดแปลงมาจาก SVM ดังนั้น สมการความถดถอยของ SVR จึงคล้ายคลึงกับสมการไฮเปอร์เพลน ของ SVM โดยสมการความถดถอยของ SVR แสดงได้ดังสมการที่ 2.83 (Bagheripour et al., 2015)

$$f(x) = w^T x + b \quad \text{----- (2.83)}$$

เมื่อ  $w$  และ  $b$  เป็นเป็นความชันและออฟเซตของเส้นความถดถอย และการกำหนดค่าของ  $w$  และ  $b$  ทำได้ด้วยการหาค่าต่ำสุดของสมการที่ 2.84 (Bagheripour et al., 2015)

$$R = \frac{1}{2} \|w\|^2 + \frac{C}{1} \sum_{i=1}^l |y_i - f(x_i)|_\epsilon \quad \text{----- (2.84)}$$

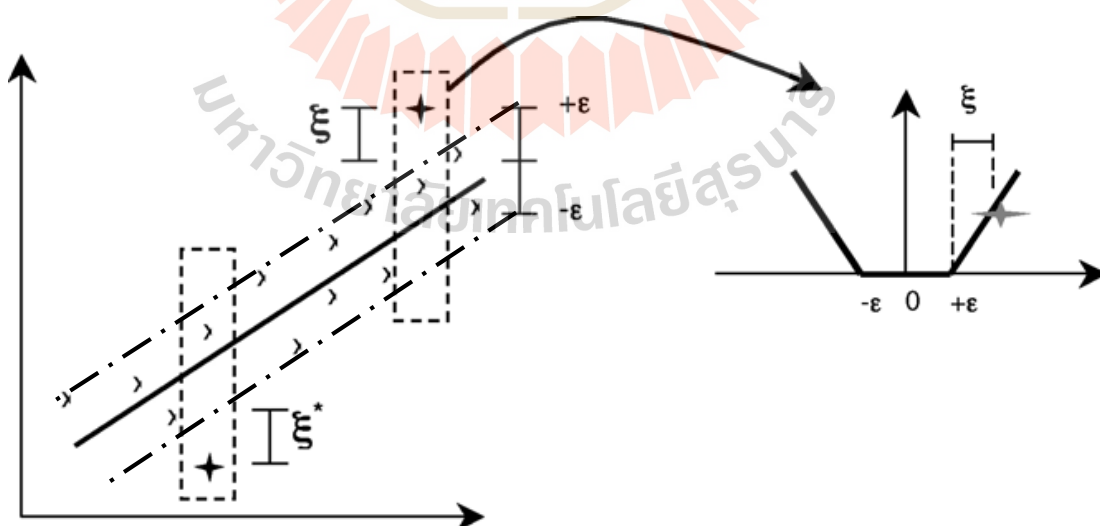
การใช้เทคนิค SVR ทำนายค่าเอาต์พุตจากอินพุตเวกเตอร์ จะมีการสร้างท่อเอปซิลอน (Epsilon tube) โดยใช้ฟังก์ชันสูญเสีย (Loss function) ที่มีหลากหลายรูปแบบ ซึ่งฟังก์ชันสูญเสียที่นิยมใช้ คือ  $\mathcal{E}$ -insensitive loss function ที่นำเสนอโดย Vapnik (1995) ดังสมการที่ 2.85

$$|y_i - f(x_i)|_\varepsilon = \begin{cases} 0 & \text{if } |y_i - f(x_i)| \leq \varepsilon \\ |y_i - f(x_i)|_\varepsilon - \varepsilon & \text{Otherwise} \end{cases} \quad \text{----- (2.85)}$$

ในความเป็นจริง เพื่อให้เกิดความยืดหยุ่นในการสร้างท่อเปิดซิลอน ซึ่งเป็นท่อที่มีรัศมีเท่ากับ  $\varepsilon$  รอบเส้นความถดถอยของ SVR จะนำวิธีการแบบซอฟต์แวร์มาร์จิน ด้วยการเพิ่มตัวแปรสแลค ( $\xi_i$ ) ทำให้การหาค่าต่ำสุดของสมการที่ 2.84 มีเงื่อนไขเปลี่ยนไป โดยอยู่ภายใต้เงื่อนไขตามสมการที่ 2.86 (Ustun et al., 2005)

$$\text{Subject to } \begin{cases} y_i - f(x_i) \leq \varepsilon + \xi_i \\ f(x_i) - y_i \leq \varepsilon + \xi_i^* \\ \varepsilon, \xi_i, \xi_i^* \geq 0 \end{cases} \quad \text{----- (2.86)}$$

รูปที่ 2.28 แสดงการสร้างท่อเปิดซิลอนแบบมีตัวแปรสแลค รอบสมการความถดถอยแบบเชิงเส้นของ SVR ซึ่งจุดที่อยู่นอกท่อเปิดซิลอน จะมีค่าของตัวแปรสแลคเท่ากับ 0 และเป็นจุดที่เรียกว่า ซัพพอร์ตเวกเตอร์ (Support Vector: SV) โดยจุดที่เป็นซัพพอร์ตเวกเตอร์จะเป็นตัวแทนของชุดข้อมูลทั้งหมด สำหรับการกำหนดสมการความถดถอยของ SVR



รูปที่ 2.28 การสร้างท่อเปิดซิลอนแบบมีตัวแปรสแลค รอบสมการความถดถอยเชิงเส้นของ SVR

(Ustun et al., 2005, p. 293).

การแก้ปัญหาของสมการที่ 2.84 ที่มีเงื่อนไขตามสมการที่ 2.85 สามารถปรับให้อยู่ในรูปแบบการแก้ปัญหาแบบคู่อัล (Dual problem) ด้วยการใช้อยู่ตัวคูณลากรองจ์ (Lagrange multipliers) ดังสมการที่ 2.87-2.88 (Bagheripour et al., 2015)

$$\begin{aligned} \text{Maximize } L_p(\alpha_i, \alpha_i^*) = & -\frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) x_i^T x_j \\ & -\varepsilon \sum_{i=1}^l (\alpha_i + \alpha_i^*) + \sum_{i=1}^l (\alpha_i - \alpha_i^*) y_i \end{aligned} \quad \text{----- (2.87)}$$

$$\text{Subject to } \begin{cases} \sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0 \\ 0 \leq \alpha_i \leq C, & i = 1, \dots, l \\ 0 \leq \alpha_i^* \leq C, & i = 1, \dots, l \end{cases} \quad \text{----- (2.88)}$$

เมื่อ  $\alpha_i, \alpha_i^*$  เป็นตัวคูณลากรองจ์ ส่วน  $C$  เป็นค่าคงที่ ซึ่งเป็นค่าใช้จ่าย (Cost) เมื่อมีความคลาดเคลื่อนเกิดขึ้น เอปซิลอน ( $\varepsilon$ ) เป็นความกว้างของเอปซิลอนทูป หรือเป็นความคลาดเคลื่อนของชุดข้อมูลฝึกสอน และแอล ( $l$ ) คือจำนวนของซัพพอร์ตเวกเตอร์ ซึ่งอินพุตเวกเตอร์ที่เป็นซัพพอร์ตเวกเตอร์ จะมี  $\alpha_i, \alpha_i^* > 0$  ส่วนอินพุตเวกเตอร์ที่ไม่ใช่ซัพพอร์ตเวกเตอร์ จะมี  $\alpha_i, \alpha_i^* = 0$  และหลังจากที่คำนวณค่า  $\alpha_i$  และ  $\alpha_i^*$  จากชุดข้อมูลฝึกสอน จะสามารถสร้างสมการ SVR เพื่อใช้ทำนายค่าเอาต์พุตจากอินพุตเวกเตอร์ ได้ดังสมการที่ 2.89 (Bagheripour et al., 2015)

$$f(x) = w_0^T x + b = \sum_{i=1}^l (\alpha_i - \alpha_i^*) x_i^T x + b \quad \text{----- (2.89)}$$

โดยที่เวกเตอร์ถ่วงน้ำหนักของไฮเปอร์เพลนถดถอย (Regression hyperplane:  $w_0$ ) เป็นดังสมการที่ 2.90

$$w_0 = \sum_{i=1}^l (\alpha_i - \alpha_i^*) x_i \quad \text{----- (2.90)}$$

สมการที่ 2.89 อยู่ในรูปแบบการถดถอยเชิงเส้น แต่ในกรณีที่เป็นกรณิที่เป็นการถดถอยแบบไม่เป็นเชิงเส้น สามารถส่งอินพุตเวกเตอร์ไปยังมิติที่สูงขึ้น (High dimensional feature space) โดยใช้เคอร์เนลฟังก์ชันเช่นเดียวกับ SVM ซึ่งเคอร์เนลฟังก์ชันที่นิยมใช้ใน SVR มีดังนี้ (Sajan et al., 2015)

- (1) ลีเนียร์เคอร์เนล (Linear kernel):  $k(x_i, x) = x_i^T x$
- (2) โพลีโนเมียลเคอร์เนล (Polynomial kernel):  $k(x_i, x) = (1 + x_i \cdot x)^d$
- (3) เกาส์เซียนเคอร์เนล (Gaussian (RBF) kernel):  $k(x_i, x) = \exp\left(-\frac{\|x_i - x\|^2}{2\sigma^2}\right)$

ดังนั้น สมการที่ 2.89 สามารถเขียนใหม่ในรูปแบบการถดถอยไม่เป็นเชิงเส้น โดยใช้เคอร์เนลฟังก์ชันได้ดังสมการที่ 2.91 (Bagheripour et al., 2015)

$$f(x) = \sum_{i=1}^l (\alpha_i - \alpha_i^*) k(x_i, x) + b \quad \text{----- (2.91)}$$

สมการที่ 2.91 เป็นสมการพยากรณ์ของ SVR โดยพยากรณ์ค่าเอาต์พุตจากอินพุตเวกเตอร์ และการกำหนดสมการพยากรณ์ของ SVR มีขั้นตอนคล้ายคลึงกับการกำหนดสมการไฮเปอร์เพลนที่เหมาะสมที่สุดของ SVM

#### 2.4.7 จีเนติกอัลกอริทึมกับการสร้างซัพพอร์ตเวกเตอร์รีเกรสชันที่เหมาะสมที่สุด

จีเนติกอัลกอริทึม (Genetic Algorithm: GA) เป็นขั้นตอนวิธีการค้นหาค่าตอบที่มีแนวคิดจากทฤษฎีวิวัฒนาการของสิ่งมีชีวิต โดยการนำเสนอของ Holland ในปี ค.ศ. 1975 ซึ่งเทคนิค GA เป็นวิธีการค้นหาค่าตอบที่เหมาะสมที่สุด โดยเริ่มต้นจากการสุ่มกลุ่มของคำตอบ (Solution) มาเป็นสมาชิกของประชากรรุ่นแรก (Initial population) แต่ละคำตอบจะถูกเข้ารหัสและเรียกว่าโครโมโซม (Chromosome) แต่ละโครโมโซมประกอบไปด้วยยีน (Gene) ซึ่งเป็นพารามิเตอร์ที่ต้องการหาค่าที่เหมาะสมที่สุด จากนั้นคำตอบจากประชากรรุ่นก่อนหน้าหรือเรียกว่า รุ่นพ่อแม่ (Parent) จะถูกนำไปสร้างเป็นประชากรรุ่นถัดไปหรือเรียกว่า รุ่นลูกหลาน (Offspring) ด้วยการดำเนินการแบบจีเนติก (Genetic operator) ที่แบ่งเป็น 2 ประเภท คือ การข้ามสายพันธุ์ (Crossover) และการกลายพันธุ์ (Mutation) จากนั้น จะเป็นขั้นตอนการแทนที่ (Replacement) ประชากรรุ่นพ่อแม่ด้วยประชากรรุ่นลูกหลาน และกระบวนการจะถูกกระทำซ้ำจนกว่าจะเข้าสู่เงื่อนไขสิ้นสุดการทำงาน

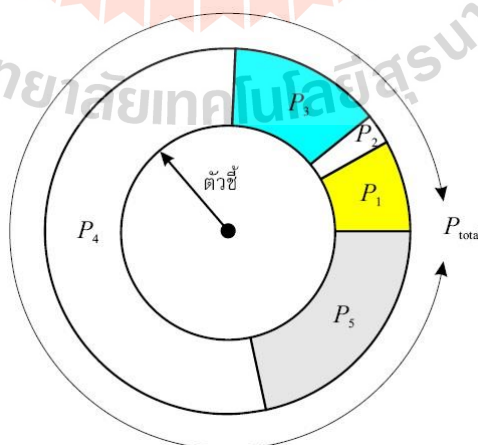
สำหรับการสร้างสมการพยากรณ์ของ SVR มีพารามิเตอร์ที่สำคัญ 3 ตัว ซึ่งมีผลต่อความแม่นยำของการพยากรณ์ ได้แก่ พารามิเตอร์ C พารามิเตอร์  $\epsilon$  และพารามิเตอร์ของเคอร์เนลฟังก์ชัน ดังนั้น สามารถนำเทคนิค GA มาใช้ในการค้นหาค่าของพารามิเตอร์ที่เหมาะสมที่สุด ซึ่งทำให้สมการ

พยากรณ์ของ SVR มีความคลาดเคลื่อนต่ำที่สุด และกระบวนการของ GA สำหรับการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดของ SVR เป็นดังนี้

1. การกำหนดโครโมโซม (Chromosome representation) เป็นขั้นตอนของการระบุรูปแบบของยีน (Gene) ภายในโครโมโซม เมื่อนำมาใช้กับการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดของ SVR ดังนั้น แต่ละโครโมโซม ประกอบด้วยยีน ที่เป็นตัวแทนของพารามิเตอร์ โดยมีการเข้ารหัสโครโมโซมเป็นค่าจำนวนจริง (Floats chromosome) จากนั้น สร้างประชากรรุ่นแรก (Initial population) แบบสุ่ม เพื่อนำไปประเมินค่าความเหมาะสมในขั้นตอนต่อไป

2. การประเมินค่าความเหมาะสม (Fitness evaluation) เป็นการประเมินค่าความเหมาะสมของแต่ละโครโมโซม เพื่อใช้ในการคัดเลือกโครโมโซมลูกหลานที่จะไปเป็นประชากรในรุ่นถัดไป ซึ่งจะประเมินค่าความเหมาะสมด้วยการวัดค่าคลาดเคลื่อน เช่น ค่า MAE (Mean Absolute Error) ค่า RMSE (Root Mean Squared Error) หรือค่า MAPE (Mean Absolute Percentage Error) เป็นต้น

3. การดำเนินการคัดเลือก (Selection operation) เป็นขั้นตอนในการคัดเลือกโครโมโซมลูกหลาน เพื่อไปเป็นประชากรในรุ่นถัดไป ซึ่งวิธีการคัดเลือกที่นิยมใช้ได้แก่ วิธีการแบบวงล้อรูเล็ต (Roulette wheel) โดยกำหนดค่าโอกาสในการถูกคัดเลือกของแต่ละโครโมโซม ( $P$ ) จากนั้น สุ่มค่าตัวชี้ ซึ่งเป็นตัวเลข ระหว่าง 0 ถึง  $P_{total}$  ซึ่งค่าของ  $P_{total}$  คือ ผลรวมของค่าโอกาสในการถูกคัดเลือกทั้งหมด โดยมีค่าเท่ากับเส้นรอบวงของวงล้อรูเล็ต ถ้าค่าตัวชี้ที่สุ่มมา ไปตรงกับโครโมโซมใดบนวงล้อรูเล็ต โครโมโซมนั้นจะถูกคัดเลือกให้ไปอยู่ในประชากรรุ่นถัดไป ดังรูปที่ 2.29



รูปที่ 2.29 วิธีการคัดเลือกโครโมโซมแบบวงล้อรูเล็ต (อาทิตย์ ศรีแก้ว, 2552, หน้า 81).



4. การข้ามสายพันธุ์ (Crossover) เป็นการสร้างโครโมโซมลูกหลาน ด้วยการนำโครโมโซมพ่อแม่จำนวน 2 โครโมโซม มาดำเนินการข้ามสายพันธุ์ ด้วยอัตราการข้ามสายพันธุ์ที่กำหนดโดยผู้ใช้ (User) ซึ่งงานวิจัยของ Wehrens และ Buydens (Wehrens and Buydens, 1998) กล่าวว่า อัตราการข้ามสายพันธุ์ ควรมีค่าระหว่าง 0.5-0.9 และวิธีการข้ามสายพันธุ์มี 3 วิธี ได้แก่

4.1 การข้ามสายพันธุ์แบบจุดเดียว (Single-point crossover) เป็นการสร้างโครโมโซมลูกหลานด้วยการสลับค่าของยีนในโครโมโซมพ่อแม่ โดยมีจุดที่ใช้สลับค่ายีนเพียงจุดเดียว ซึ่งการข้ามสายพันธุ์แบบจุดเดียว จะใช้ในกรณีที่ 1 โครโมโซม มี 2 ยีน (2 พารามิเตอร์)

4.2 การข้ามสายพันธุ์แบบหลายจุด (Multiple-point crossover) เป็นการสร้างโครโมโซมลูกหลานด้วยการสลับค่าของยีนในโครโมโซมพ่อแม่ โดยมีจุดที่ใช้สลับค่ายีนหลายจุด ซึ่งการข้ามสายพันธุ์แบบหลายจุด จะใช้ในกรณีที่ 1 โครโมโซม มียีนมากกว่า 2 ยีน ขึ้นไป โดยจุดสลับค่ายีน จะอยู่ระหว่างยีนคู่หนึ่ง ๆ

4.3 การข้ามสายพันธุ์แบบสม่ำเสมอ (Uniform crossover) เป็นการข้ามสายพันธุ์ด้วยการสร้างตัวพราง (Mask) เพื่อกำหนดจุดในการสลับค่ายีน โดยค่าของตัวพรางเป็นตัวเลขไบนารี และจุดสลับค่ายีน จะมีค่าของตัวพรางเป็น 1 ซึ่งวิธีการนี้ เหมาะที่จะนำไปใช้กับโครโมโซมที่มีการเข้ารหัสแบบไบนารี

5. การกลายพันธุ์ (Mutation) เป็นการสร้างโครโมโซมลูกหลาน ด้วยการนำโครโมโซมพ่อแม่จำนวน 1 โครโมโซม มาดำเนินการกลายพันธุ์ เพื่อให้หลุดพ้นออกจากคำตอบที่เหมาะสมที่สุดแบบวงแคบเฉพาะถิ่น (Local optimum) โดยต้องกำหนดอัตราการกลายพันธุ์ให้มีค่าต่ำ ซึ่งงานวิจัยของ Wehrens และ Buydens (1998) กล่าวว่า อัตราการกลายพันธุ์ ควรมีค่าระหว่าง 0.001-0.05

6. การแทนที่ประชากรเดิม (Replacement) เป็นการแทนที่ประชากรเดิมด้วยประชากรใหม่ที่มีค่าความเหมาะสมดีขึ้น จากนั้น ทำซ้ำขั้นตอนที่ 2 ถึง 6 จนกว่าจะเข้าสู่เงื่อนไขสิ้นสุดการทำงานของ GA

ทั้งหมดที่กล่าวมาในหัวข้อนี้ คือ ขั้นตอนการนำจีเนติกอัลกอริทึม มาใช้ในการค้นหาค่าพารามิเตอร์ที่เหมาะสมของ SVR ซึ่งค่าพารามิเตอร์ที่เหมาะสม จะทำให้การพยากรณ์ด้วยตัวแบบ SVR มีความคลาดเคลื่อนต่ำ และเหมาะสมต่อการนำไปใช้พยากรณ์อนุกรมเวลา

## 2.5 งานวิจัยที่เกี่ยวข้อง

งานวิจัยในอดีต ที่เกี่ยวข้องกับการพยากรณ์อนุกรมเวลา จะดำเนินการสร้างตัวแบบพยากรณ์จากเทคนิคหลากหลาย ที่แตกต่างกัน โดยเทคนิคที่นำมาใช้ในการสร้างตัวแบบพยากรณ์อนุกรมเวลา แบ่งเป็น 2 ประเภท ได้แก่ ประเภทที่ 1 คือ เทคนิคการวิเคราะห์อนุกรมเวลาด้วยหลักสถิติ ซึ่งเป็นวิธีการแบบดั้งเดิม โดยสร้างตัวแบบวิเคราะห์อนุกรมเวลาเชิงเดี่ยว เช่น ตัวแบบวิเคราะห์ความถดถอยอย่างง่าย ตัวแบบปรับให้เรียบแบบเอกซ์โพเนนเชียล หรือตัวแบบ ARIMA ส่วนประเภทที่ 2 คือ เทคนิคอัจฉริยะ ซึ่งเป็นวิธีการด้านแมชชีนเลิร์นนิง ได้แก่ เครือข่ายประสาทเทียม หรือซอฟต์แวร์เรกเรชัน มาใช้สร้างตัวแบบพยากรณ์อนุกรมเวลา เพื่อพยากรณ์ข้อมูลอนุกรมเวลา ซึ่งงานวิจัยที่นำเทคนิคทั้ง 2 ประเภท มาใช้สร้างตัวแบบพยากรณ์อนุกรมเวลาที่สามารถนำมาใช้ได้ มีดังนี้

Cadenas et al. (2010) ศึกษาการพยากรณ์ความเร็วลมในเขตจังหวัด คิวินตานารู (Quintana Roo) ประเทศเม็กซิโก ซึ่งวัดความเร็วลมที่ระดับ 10 เมตร เหนือพื้นดิน ทุก 10 นาที และเริ่มต้นการวิจัยด้วยการวิเคราะห์อนุกรมเวลาโดยใช้หลักสถิติที่มีการวัดค่าสถิติอย่างเป็นแบบแผน จากนั้นนำข้อมูลความเร็วลมของวันสุดท้ายที่ทำการวัด มาสร้างตัวแบบการพยากรณ์ด้วยเทคนิค SES (Single Exponential Smoothing) ซึ่งกำหนดค่าอัลฟา ( $\alpha$ ) ที่เหมาะสมที่สุดของตัวแบบ SES โดยใช้ในการวัดค่าคลาดเคลื่อน ด้วยมาตรวัดค่าคลาดเคลื่อน 5 ตัว ได้แก่ ME (Mean Error), MSE (Mean Squared error), MAE (Mean Absolute Error), MPE (Mean Percentage Error) และ MAPE (Mean Absolute Percentage Error) สุดท้าย สร้างตัวแบบการพยากรณ์ด้วยเทคนิค ANN (Artificial Neural Network) แล้วนำค่าพยากรณ์ของตัวแบบ SES ตัวแบบ ANN และค่าจริงมาเปรียบเทียบกัน ผลการทดลอง พบว่า ค่าอัลฟาที่เหมาะสมที่สุดของตัวแบบ SES เท่ากับ 0.9 และการพยากรณ์ด้วยตัวแบบ SES และตัวแบบ ANN เมื่อเปรียบเทียบกับค่าจริงแล้ว ให้ผลลัพธ์ที่ดีทั้งคู่ โดยตัวแบบ SES ให้ผลลัพธ์ดีกว่าตัวแบบ ANN เนื่องจากให้ค่าความคลาดเคลื่อนต่ำกว่า ผู้วิจัยจึงสรุปว่า เทคนิค SES เป็นทางเลือกที่ดีสำหรับการนำมาพยากรณ์ความเร็วลม

Jirong et al. (2011) ศึกษาการพยากรณ์ราคาบ้านพักอาศัยในประเทศจีน ซึ่งเป็นอนุกรมเวลาที่มีคาบเวลาเป็นรายปี ตั้งแต่ปีค.ศ. 1993 ถึง 2002 โดยใช้เทคนิค SVR (Support Vector Regression) ร่วมกับการกำหนดค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินเนติกอัลกอริทึม (Genetic algorithm) และเรียกว่าเทคนิค G-SVR จากนั้น เปรียบเทียบความแม่นยำของการพยากรณ์กับตัวแบบ GM (Grey

model) ด้วยการเปรียบเทียบค่าพยากรณ์กับค่าจริง และการวัดค่าคลาดเคลื่อนสัมพัทธ์แบบสัมบูรณ์ (Absolute relative error) ผลการศึกษา พบว่า ตัวแบบ G-SVR มีความแม่นยำในการพยากรณ์มากกว่าตัวแบบ GM

Ogcu et al. (2012) ศึกษาการพยากรณ์ปริมาณการบริโภคพลังงานไฟฟ้าในประเทศตุรกี ซึ่งเป็นอนุกรมเวลาเชิงเดี่ยวที่มีคาบเวลาเป็นรายเดือน โดยใช้เทคนิค ANN และเทคนิค SVR ในการสร้างตัวแบบการพยากรณ์ และข้อมูลที่ใช้ศึกษา เป็นข้อมูลตั้งแต่เดือนมกราคม ปีค.ศ. 1970 ถึงเดือนธันวาคม ปีค.ศ. 2011 ซึ่งการสร้างตัวแบบพยากรณ์ด้วยเทคนิค ANN และ SVR ได้กำหนดอินพุต 5 ตัว ได้แก่ ปริมาณการใช้ไฟฟ้าล่าช้าไป 12 คาบเวลา และ 24 คาบเวลา ดัชนีฤดูกาล ดัชนีเวลา และ ดัชนีเดือน ส่วนเอาต์พุตมี 1 ตัว คือ ปริมาณการใช้ไฟฟ้ารายเดือน โดยเปรียบเทียบความแม่นยำของตัวแบบ ANN และ SVR ด้วยการวัดค่าคลาดเคลื่อน MAPE ผลการศึกษา พบว่า ตัวแบบ SVR มีความแม่นยำในการพยากรณ์มากกว่าตัวแบบ ANN

Wang et., al. (2012) ศึกษาการพยากรณ์โหลดไฟฟ้าประจำปีของเมืองปักกิ่ง ประเทศจีน ซึ่งเป็นอนุกรมเวลาเชิงเดี่ยวที่มีคาบเวลารายปี โดยใช้เทคนิค SVR ร่วมกับการคัดเลือกพารามิเตอร์ที่เหมาะสมด้วยขั้นตอนวิธีแบบ DE (Differential Evolution algorithm) เรียกว่า เทคนิค DESVR จากนั้น เปรียบเทียบความแม่นยำในการพยากรณ์กับตัวแบบที่สร้างจากเทคนิค BPNN (Back Propagation Neural Network) และเทคนิคการถดถอย (Regression) ข้อมูลที่ใช้ศึกษา เป็นข้อมูลตั้งแต่ปีค.ศ. 1978 ถึงปีค.ศ. 2008 โดยการสร้างตัวแบบพยากรณ์ด้วยเทคนิค DESVR ได้เลือกอินพุต 6 ตัว เป็นค่าโหลดไฟฟ้า ณ เวลาล่าช้าไป  $t-1$  ถึง  $t-6$  คาบเวลา เพื่อพยากรณ์ ค่าโหลดไฟฟ้า ณ เวลา  $t$  จากนั้น เปรียบเทียบความแม่นยำของตัวแบบทั้งหมด ด้วยการวัดค่าคลาดเคลื่อน MAPE ผลการศึกษา พบว่า ตัวแบบ DESVR มีค่าคลาดเคลื่อน MAPE ต่ำที่สุด เมื่อวัดกับชุดข้อมูลทดสอบ

Wang and Meng (2012) ศึกษาการพยากรณ์การบริโภคพลังงานถ่านหินในเขตจังหวัดเหอเป่ย์ (Hebei) ประเทศจีน ซึ่งข้อมูลเป็นอนุกรมเวลาเชิงเดี่ยว ที่มีคาบเวลาเป็นรายปี ตั้งแต่ปี ค.ศ. 1980 ถึงปี ค.ศ. 2008 ซึ่งดำเนินการทดลอง โดยใช้เทคนิคการสร้างตัวแบบ ARIMA (Autoregressive Integrated Moving Average) ตัวแบบ ANN (Artificial Neural Network) และตัวแบบผสมผสาน ARIMA-ANN จากนั้น เปรียบเทียบความแม่นยำของแต่ละตัวแบบ ด้วยมาตรวัดความคลาดเคลื่อน RMSE (Root Mean Squared Error) และ MAE (Mean Absolute Error) ผลการทดลอง พบว่า ตัวแบบผสมผสาน ARIMA-ANN ให้ค่า RMSE และ MAE ต่ำกว่าตัวแบบ ARIMA และตัวแบบ ANN ดังนั้น ผู้วิจัย จึงสรุปว่า

เทคนิคการสร้างตัวแบบผสมผสาน ARIMA-ANN ให้ความแม่นยำในการพยากรณ์อนุกรมเวลาได้ดีกว่าการใช้เทคนิค ARIMA หรือเทคนิค ANN อย่างใดอย่างหนึ่งเพียงอย่างเดียว

Wang et al. (2012) ได้ศึกษาการปรับปรุงความแม่นยำของค่าพยากรณ์ความต้องการไฟฟ้าในเขตไฟฟ้าภาคตะวันออกเฉียงเหนือของประเทศไทย ที่ได้จากตัวแบบ Seasonal ARIMA ซึ่งข้อมูลเป็นอนุกรมเวลาเชิงเดี่ยว ที่มีคาบเวลาเป็นรายปี โดยทำการปรับปรุงความแม่นยำของค่าพยากรณ์ ด้วยวิธีการปรับปรุงค่าคลาดเคลื่อน (Residual) ของตัวแบบ Seasonal ARIMA ซึ่งปรับปรุงค่าคลาดเคลื่อนด้วยวิธีการ 3 วิธี ได้แก่ วิธีที่ 1 คือ วิธีการปรับปรุงค่าคลาดเคลื่อนด้วย Particle Swarm Optimization (PSO) optimal Fourier วิธีที่ 2 คือ วิธีการปรับปรุงค่าคลาดเคลื่อนด้วย Seasonal ARIMA และวิธีที่ 3 คือ วิธีแบบผสมผสานระหว่างวิธี PSO optimal Fourier และวิธี Seasonal ARIMA จากนั้น เปรียบผลการปรับปรุงความแม่นยำในการพยากรณ์ของทั้ง 3 วิธี ด้วยมาตรวัดความคลาดเคลื่อน MAPE, RMSE, RE (Relative Error) และ SEP (Standard Error of Prediction) ผลการศึกษา สรุปได้ว่า วิธีการปรับปรุงค่าคลาดเคลื่อนทั้ง 3 วิธี ให้ความแม่นยำสูงขึ้น และวิธีแบบผสมผสานให้ความแม่นยำสูงที่สุด

Chujai, Kerdprasop and Kerdprasop (2013) ศึกษาการวิเคราะห์ข้อมูลอนุกรมเวลาของปริมาณการใช้กระแสไฟฟ้าภายในบ้าน ซึ่งชุดข้อมูลที่ใช้ เป็นชุดข้อมูลมาตรฐาน UCI โดยนำมาสร้างเป็นข้อมูลอนุกรมเวลา ที่แบ่งคาบเวลาเป็น 4 ประเภท คือ รายวัน รายสัปดาห์ รายเดือน และรายไตรมาส แล้วนำข้อมูลมาสร้างตัวแบบการพยากรณ์ 2 ตัวแบบ คือ ตัวแบบ ARIMA และตัวแบบ ARMA (Autoregressive Moving Average) ด้วยภาษาอาร์ (R language) และ โปรแกรม Rstudio จากนั้น เปรียบเทียบความถูกต้องของค่าพยากรณ์จากตัวแบบทั้งสอง โดยวัดค่าคลาดเคลื่อนด้วยค่า AIC (Akaike Information Criterion) และค่า RMSE ผลการทดลอง พบว่า ตัวแบบ ARIMA เป็นตัวแบบที่เหมาะสมสำหรับการพยากรณ์คาบเวลารายเดือนและรายไตรมาส ส่วนตัวแบบ ARMA เป็นตัวแบบที่เหมาะสมสำหรับการพยากรณ์คาบเวลารายวันและรายสัปดาห์

Keka and Hamiti (2013) ศึกษาการหาตัวแบบทางคณิตศาสตร์ ที่แสดงความสัมพันธ์ระหว่างพลังงานไฟฟ้าและเวลา ด้วยหลักการวิเคราะห์ความถดถอยแบบเชิงเส้น (Linear regression) ซึ่งข้อมูลที่ใช้หาตัวแบบทางคณิตศาสตร์ ได้ใช้ข้อมูลการจ่ายกระแสไฟฟ้าของสถานีจ่ายไฟฟ้าย่อย ที่มีการวัดปริมาณการจ่ายกระแสไฟฟ้าทุก 15 นาที แล้วนำมาสร้างเป็นข้อมูลอนุกรมเวลาด้วยภาษาอาร์ โดยแบ่งคาบเวลาเป็นรายวัน รายสัปดาห์ และรายเดือน จากนั้นคำนวณค่าพารามิเตอร์ทางสถิติ และนำเสนอตัวแบบเชิงเส้นทางคณิตศาสตร์ ที่เป็นสมการแสดงความสัมพันธ์ระหว่างพลังงานไฟฟ้าและเวลา

Kaytez et al. (2015) ศึกษาการพยากรณ์ปริมาณการบริโภคพลังงานไฟฟ้าในตุรกี ด้วยการเปรียบเทียบเทคนิค ANN เทคนิค MLR (Multiple Linear Regression) และเทคนิค LSSVR (Least Squares Support Vector Machines) โดยเป็นการพยากรณ์อนุกรมเวลาเชิงพหุ ที่มีคาบเวลาเป็นรายปี ซึ่งมีตัวแปรอิสระ 4 ตัว เป็นอินพุต ได้แก่ กำลังการผลิตพลังงานไฟฟ้าสุทธิ (Gross electricity generation) กำลังการผลิตพลังงานไฟฟ้าที่ติดตั้งแล้ว (Installed capacity) จำนวนโรงผลิตพลังงานไฟฟ้าย่อยทั้งหมด (Total subscribership) และจำนวนประชากร (Population) ส่วนเอาต์พุตเป็นปริมาณการบริโภคไฟฟ้า โดยวัดความแม่นยำของการพยากรณ์ด้วยค่า MAPE, ค่า MSE, ค่า RMSE และค่า SSE (Sum Squared Error) ผลการศึกษา พบว่า ตัวแบบที่สร้างจากเทคนิค LSSVM มีความแม่นยำในการพยากรณ์มากที่สุด

Massana et al. (2015) ศึกษาการพยากรณ์ระยะสั้นของปริมาณการใช้ไฟฟ้าในอาคารที่มีใช้บ้านพักอาศัยด้วยตัวแบบการพยากรณ์และแอททริบิวต์ (Attributes) ที่แตกต่างกัน ซึ่งปริมาณการใช้ไฟฟ้าที่นำมาศึกษา เป็นข้อมูลการใช้ไฟฟ้าภายในอาคารสำนักงานของมหาวิทยาลัยยิโรนา (University of Girona) และข้อมูลเป็นอนุกรมเวลาที่มีคาบเวลาเป็นรายชั่วโมง โดยเก็บข้อมูลตั้งแต่เดือนพฤษภาคม ปีค.ศ. 2013 ถึงเดือนมีนาคม ปีค.ศ. 2014 ทำให้มีจำนวนข้อมูลทั้งสิ้น 7,616 อินสแตนซ์ (Instances) ในส่วนของตัวแบบการพยากรณ์ ได้สร้างตัวแบบจากเทคนิคที่ต่างกัน 3 ตัวแบบ คือ ตัวแบบ MLR (Multiple Linear Regression) ตัวแบบ MLP (Multilayer Perceptron) ซึ่งเป็นตัวแบบที่ใช้เทคนิค ANN และตัวแบบ SVR ที่กำหนดค่าพารามิเตอร์ให้เหมาะสมที่สุดด้วยเทคนิค GS (Grid search) ส่วนแอททริบิวต์ที่สัมพันธ์กับปริมาณการใช้ไฟฟ้า ซึ่งนำมาศึกษาในงานวิจัยนี้ แบ่งเป็น 3 กลุ่ม ได้แก่ ข้อมูลสภาพอากาศ (Weather data) ข้อมูลภายในอาคาร (Indoor data) และข้อมูลเวลาตามปฏิทิน (Calendar data) สุดท้าย ประเมินผลการทดลองด้วยมาตรวัดความคลาดเคลื่อน MAPE จากผลการทดลองพบว่า ตัวแบบ SVR ให้ความสมมูลระหว่างความแม่นยำในการพยากรณ์ และค่าใช้จ่ายในการคำนวณได้ดีที่สุด

Sajan et al. (2015) ศึกษาการตรวจจับความเสถียรของแรงดันไฟฟ้าแบบออนไลน์ ด้วยการสร้างตัวแบบ SVR (Support vector regression) และกำหนดค่าพารามิเตอร์ของ SVR ให้เหมาะสมที่สุดด้วยเทคนิค GA (Genetic algorithm) เรียกว่า ตัวแบบ GA-SVR โดยข้อมูลที่ใช้ศึกษาเป็นข้อมูลจากเครือข่ายไฟฟ้าในเขตพื้นที่ New England และ Indian Northern ของประเทศอินเดีย ซึ่งตัวแบบ GA-SVR มีอินพุตเวกเตอร์ (Input vector) เป็นขนาดแรงดันไฟฟ้า (Voltage magnitude) และมุมองศาของเฟส (Phase angle) ส่วนเอาต์พุตเวกเตอร์ (Output vector) เป็นดัชนีชายขอบความเสถียรของแรงดันไฟฟ้า (Voltage stability margin index) จากนั้น วัดประสิทธิภาพของตัวแบบ GA-SVR โดยเปรียบเทียบกับตัวแบบ GS-SVM



(Grid search support vector machine) และตัวแบบ ANN ด้วยมาตรวัดความคลาดเคลื่อน MAPE , NMSE (Normalized Mean Square Error) และ WIA (Willmott's index of agreement) ผลการทดลอง พบว่า ตัวแบบ GA-SVR ให้ความคลาดเคลื่อนต่ำที่สุด จึงสรุปได้ว่า ตัวแบบ GA-SVR มีประสิทธิภาพมากที่สุด สำหรับการนำมาตรวจจับความเสถียรของแรงดันไฟฟ้าแบบออนไลน์

Szoplik (2015) ศึกษาการพยากรณ์ปริมาณการบริโภคก๊าซธรรมชาติในครัวเรือนของเมือง สเกซชิน ประเทศโปแลนด์ โดยใช้เทคนิคเครือข่ายประสาทเทียม (Artificial neural network: ANN) ซึ่งดำเนินการทดลองด้วยการสร้างตัวแบบ ANN ที่มีโครงสร้างแบบหลายชั้น (Multilayer) โดยมีชั้นซ่อนเร้น (Hidden layer) เพียง 1 ชั้น เท่านั้น จากนั้น ดำเนินการสร้างตัวแบบ ANN ที่มีจำนวนนิวรอน (Neuron) ในชั้นซ่อนเร้นแตกต่างกัน เพื่อค้นหาตัวแบบ ANN ที่มีความเหมาะสมที่สุดสำหรับนำมาพยากรณ์ข้อมูล โดยวัดความเหมาะสมด้วยค่าสัมประสิทธิ์สหสัมพันธ์ (Correlation coefficients), SSE, RMSE, MAPE และค่า nRMSE (Normalized root mean square error) ใน ส่วนของ ข้อมูลที่นำมาใช้ ดำเนินการทดลอง เป็นปริมาณการบริโภคก๊าซธรรมชาติ โดยมีคาบเวลาเป็นรายชั่วโมง และมีอินพุตเป็นอุณหภูมิ เลขชั่วโมงในหนึ่งวัน เลขวันในหนึ่งสัปดาห์ เลขวันในหนึ่งเดือน และเลขเดือน ทำให้มีจำนวนอินพุตเท่ากับ 22 ตัว และมีจำนวนเอาต์พุต 1 ตัว คือ ปริมาณการบริโภคก๊าซธรรมชาติใน 1 ชั่วโมง ผลการทดลอง พบว่า ตัวแบบ ANN ที่เหมาะสมที่สุด คือตัวแบบที่มีโครงสร้างเป็น 22-36-1 นั่นคือ มีจำนวนโหนดในชั้นอินพุต 22 โหนด มีจำนวนนิวรอนในชั้นซ่อนเร้น 36 นิวรอน และมีจำนวนนิวรอนในชั้นเอาต์พุต 1 นิวรอน โดยสามารถนำตัวแบบ ANN ที่มีโครงสร้างดังกล่าว ไปใช้พยากรณ์ปริมาณการบริโภคก๊าซรายชั่วโมง ในแต่ละวันได้

Hussain et al. (2016) ศึกษาการพยากรณ์อนุกรมเวลา ซึ่งเป็นข้อมูลการบริโภคพลังงานไฟฟ้าแต่ละภาคส่วน และการใช้พลังงานไฟฟ้าโดยรวมของประเทศปากีสถาน โดยเป็นข้อมูลตั้งแต่ปี ค.ศ. 1980 ถึงปี ค.ศ. 2011 และเทคนิคที่นำมาใช้สร้างตัวแบบพยากรณ์อนุกรมเวลา ได้แก่ เทคนิคการปรับให้เรียบแบบเอกซ์โพเนนเชียลของโฮลท์และวินเตอร์ (Holt-Winters model) และเทคนิคการสร้างตัวแบบ ARIMA จากนั้น เปรียบเทียบความแม่นยำในการพยากรณ์ของตัวแบบทั้ง 2 ด้วยมาตรวัดความคลาดเคลื่อน RMSE และ MAPE ผลการทดลอง พบว่า ตัวแบบของโฮลท์และวินเตอร์ ให้ค่า RMSE และ MAPE ต่ำกว่าตัวแบบ ARIMA ในทุกภาคส่วนและภาครวม ดังนั้น ผู้วิจัยจึงสรุปว่า ตัวแบบของโฮลท์และวินเตอร์ เหมาะสมที่สุดสำหรับการนำมาใช้พยากรณ์อนุกรมเวลาชุดดังกล่าวนี้

Yuan et al. (2016) ศึกษาการพยากรณ์ปริมาณการบริโภคพลังงานหลักของประเทศจีน (China's primary energy consumption) ซึ่งข้อมูลที่ใช้ศึกษาเป็นอนุกรมเวลาเชิงเดี่ยว (Univariate time series) ที่มี

คาบเวลาเป็นรายปี ตั้งแต่ปี ค.ศ. 1965 ถึงปี ค.ศ. 2014 จากนั้น ดำเนินการทดลองด้วยการสร้างตัวแบบ ARIMA ตัวแบบ GM(1, 1) และตัวแบบผสมผสานระหว่าง 2 ตัวแบบ (Hybrid model) โดยเปรียบเทียบผลการพยากรณ์ของทั้ง 3 ตัวแบบ ด้วยมาตรวัดความคลาดเคลื่อน MAPE ผลการทดลอง พบว่า ทั้ง 3 ตัวแบบ ให้ความแม่นยำในการพยากรณ์ในเกณฑ์ที่สูง โดยตัวแบบผสมผสานให้ความแม่นยำได้ดีที่สุด เนื่องจากมีค่า MAPE ต่ำที่สุด นอกจากนี้ เมื่อพยากรณ์ปริมาณการบริโภคพลังงานล่วงหน้า ตั้งแต่ปีค.ศ. 2015 ถึงปีค.ศ. 2020 พบว่า การบริโภคพลังงานเพิ่มมากขึ้น แต่เพิ่มขึ้นในอัตราที่ลดลง เมื่อเปรียบเทียบกับทศวรรษแรกของศตวรรษที่ 21

จากการศึกษางานวิจัยที่เกี่ยวข้อง พบว่า มีการใช้เทคนิคการสร้างตัวแบบการพยากรณ์ที่แตกต่างกัน ได้แก่ เทคนิคอัจฉริยะ ซึ่งเป็นเทคนิคทางด้านแมชชีนเลิร์นนิง และเทคนิคการวิเคราะห์อนุกรมเวลา ที่เป็นเทคนิคการสร้างตัวแบบพยากรณ์ โดยใช้การวิเคราะห์ด้วยหลักสถิติแบบดั้งเดิม ซึ่งเทคนิคการพยากรณ์ที่เหมาะสมกับอนุกรมเวลา ในแต่ละงานวิจัย มีความแตกต่างกัน ขึ้นกับลักษณะของอนุกรมเวลาที่ใช้ศึกษา นอกจากนี้ พบว่า บางงานวิจัย (Wang and Meng, 2012; Wang et al., 2012; Yuan et al., 2016) ได้นำเทคนิคการพยากรณ์ทั้ง 2 เทคนิค มาผสมผสานกัน เพื่อให้สามารถตรวจจับรูปแบบเชิงเส้นและรูปแบบไม่เป็นเชิงเส้นของอนุกรมเวลา ให้ความแม่นยำของค่าพยากรณ์เพิ่มมากขึ้น ดังนั้น วิทยานิพนธ์ฉบับนี้ จึงนำเสนอวิธีการสร้างตัวแบบผสมผสาน (Hybrid model) ระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ซึ่งเป็นวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ร่วมกับวิธีการสร้างตัวแบบพยากรณ์ด้วยเทคนิคอัจฉริยะ โดยใช้รูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ในการเลือกค่าล่าหลัง (Lag time value) ของค่าสังเกต เพื่อใช้เป็นอินพุตเวกเตอร์ของชุดข้อมูลฝึกสอน สำหรับฝึกสอนตัวแบบที่สร้างจากเทคนิคอัจฉริยะ ภายใต้สมมติฐานว่า ถ้าใช้อินพุตที่เป็นค่าสังเกตล่าหลัง ซึ่งมีความสัมพันธ์กับเอาต์พุตที่เป็นค่าสังเกต ณ เวลา  $t$  ใด ๆ ตามรูปแบบออโตรีเกรสซีฟ มาใช้ฝึกสอนตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ จะทำให้ค่าพยากรณ์ของตัวแบบมีความแม่นยำเพิ่มมากขึ้น ซึ่งวิธีการนี้ เป็นวิธีการใหม่ในการสร้างตัวแบบผสมผสาน ไม่พบว่ามีการใช้วิธีการดังกล่าว ในงานวิจัยที่นำมาทบทวน ผลการทดลอง จะสามารถสรุปได้ว่า เมื่อเปรียบเทียบตัวแบบผสมผสานกับตัวแบบพยากรณ์ที่สร้างจากเทคนิคการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ซึ่งเป็นวิธีการมาตรฐาน จะมีความแม่นยำเพิ่มขึ้น และเหมาะสมมากขึ้นเพียงใด สำหรับนำมาพยากรณ์หน่วยจำหน่ายไฟฟ้าในประเทศไทย สุดท้ายนี้ สามารถแสดงตารางเปรียบเทียบงานวิจัยที่เกี่ยวข้องกับการพยากรณ์อนุกรมเวลา และงานวิจัยของวิทยานิพนธ์ฉบับนี้ ได้ดังตารางที่ 2.8-2.9





ตารางที่ 2.9 การเปรียบเทียบงานวิจัยที่เกี่ยวข้องกับการพยากรณ์อนุกรมเวลาและงานวิจัยของวิทยานิพนธ์ฉบับนี้ สำหรับประเด็นเปรียบเทียบด้านมาตรวัดความคลาดเคลื่อนและตัวแบบที่เหมาะสมที่สุด

ประเด็น เปรียบเทียบ	งานวิจัยที่เกี่ยวข้อง*														
	ก	ข	ค	ง	จ	ฉ	ช	ซ	ฅ	ญ	ฎ	ฏ	ท	ฒ	ณ*
	ปี	ปี	ปี	ปี	ปี	ปี	ปี	ปี	ปี	ปี	ปี	ปี	ปี	ปี	ปี
	2010	2011	2012	2012	2012	2012	2013	2013	2015	2015	2015	2015	2016	2016	-
มาตรวัดความคลาดเคลื่อน															
1. ME	✓														
2. MSE	✓								✓						
3. MAE	✓				✓										✓
4. MPE	✓														
5. MAPE	✓		✓	✓		✓			✓	✓	✓	✓	✓		✓
6. RMSE					✓	✓	✓		✓			✓	✓		✓
7. SSE									✓			✓			✓
8. RE						✓									
9. SEP						✓									
10. AIC							✓								
11. nRMSE												✓			
12. NMSE											✓				
13. WIA											✓				
14. Absolute relative error		✓													
15. อื่น ๆ หรือไม่ระบุ								✓				✓			
Best Model	SLR	SVR	SVR	SVR	ARIMA +ANN	Hybrid ARIMA +PSO	ARIMA	LR	SVR	SVR	SVR	ANN	ES	Hybrid ARIMA +GM	?

\* งานวิจัยที่เกี่ยวข้องประกอบด้วย

งานวิจัย ก	แทนงานวิจัยของ Cadenas, Jaramillo and Rivera (2010)
งานวิจัย ข	แทนงานวิจัยของ Jirong, Mingcang and Liuguangyan (2011)
งานวิจัย ค	แทนงานวิจัยของ Ogcun, Demirel and Zaim (2012)
งานวิจัย ง	แทนงานวิจัยของ Wang et al. (2012)
งานวิจัย จ	แทนงานวิจัยของ Wang and Meng (2012)
งานวิจัย ฉ	แทนงานวิจัยของ Wang et al. (2012)
งานวิจัย ช	แทนงานวิจัยของ Chujai, Kerdprasop and Kerdprasop (2013)
งานวิจัย ซ	แทนงานวิจัยของ Keka and Hamiti (2013)
งานวิจัย ฌ	แทนงานวิจัยของ Kaytez et al. (2015)
งานวิจัย ญ	แทนงานวิจัยของ Massana et al. (2015)
งานวิจัย ฎ	แทนงานวิจัยของ Sajan, Kumar and Tyagi (2015)
งานวิจัย ฏ	แทนงานวิจัยของ Szoplik (2015)
งานวิจัย ท	แทนงานวิจัยของ Hussain, Rahman and Memon (2016)
งานวิจัย ฒ	แทนงานวิจัยของ Yuan, Liu and Fang (2016)
งานวิจัย ณ*	แทนงานวิจัยของวิทยานิพนธ์ฉบับนี้

## บทที่ 3

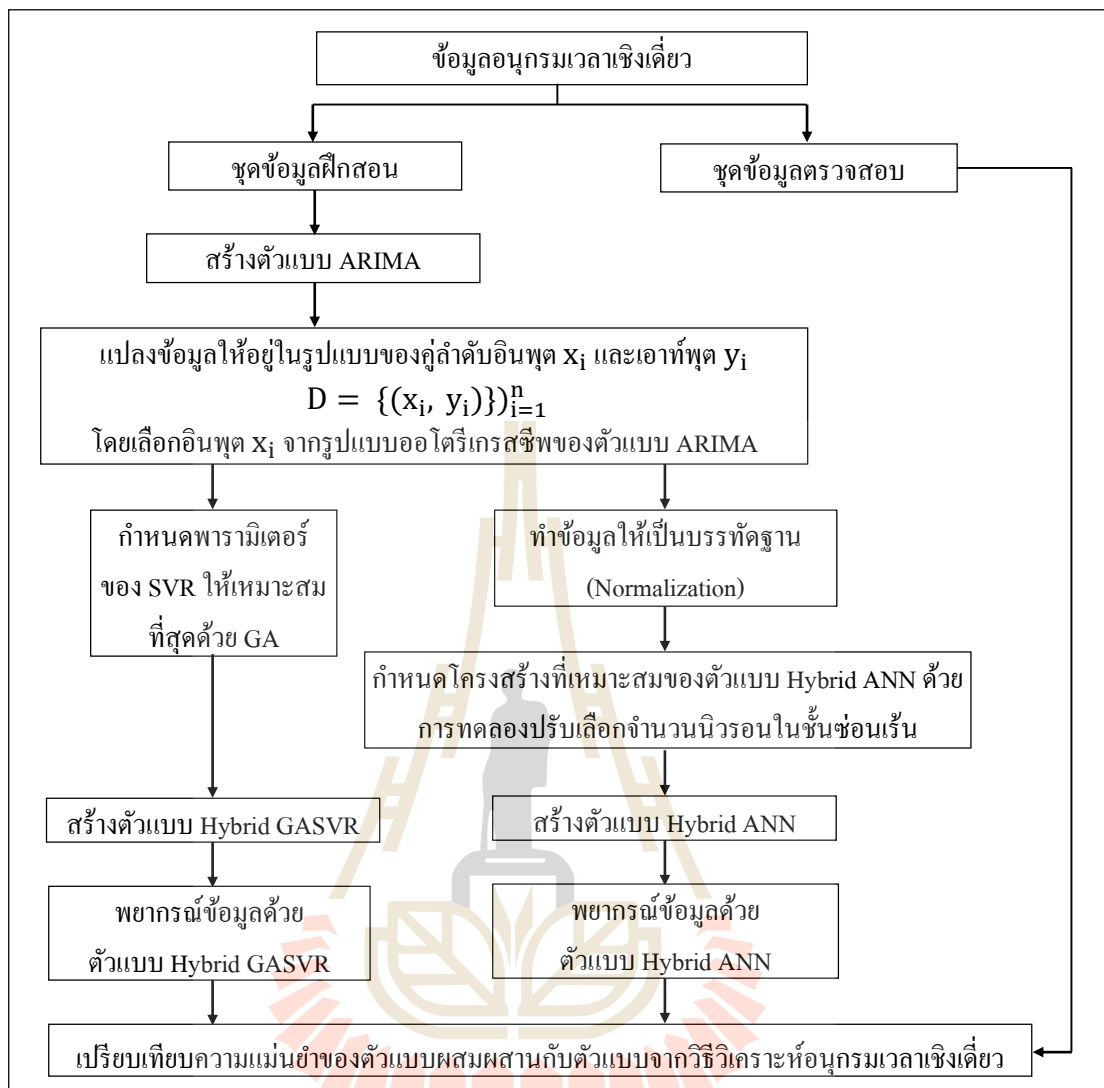
### วิธีดำเนินการวิจัย

งานวิจัยนี้มีวัตถุประสงค์ เพื่อศึกษาและพัฒนาตัวแบบผสมผสาน (Hybrid model) ระหว่างรูปแบบออโตรีเกรสซีฟ (Autoregressive) ของตัวแบบ ARIMA ร่วมกับเทคนิคอัจฉริยะ เพื่อพยากรณ์หน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง และการไฟฟ้าส่วนภูมิภาค เขต 3 แห่งประเทศไทย ซึ่งเทคนิคอัจฉริยะที่นำมาใช้ในงานวิจัยนี้ เป็นวิธีการด้านแมชชีนเลิร์นนิง ได้แก่ เทคนิคเครือข่ายประสาทเทียม (Artificial neural network : ANN) และเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชันร่วมกับอัลกอริทึม (Genetic algorithm with support vector regression : GASVR) จากนั้น เปรียบเทียบความแม่นยำของตัวแบบผสมผสานกับตัวแบบพยากรณ์ที่สร้างด้วยวิธีการวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ได้แก่ การสร้างตัวแบบ SLR (Simple linear regression) ด้วยวิธีการวิเคราะห์การถดถอยอย่างง่าย การสร้างตัวแบบ Holt-Winters ด้วยวิธีการปรับให้เรียบของโสมท์และวินเตอร์ และการสร้างตัวแบบ ARIMA ด้วยวิธีการของบอกซ์และเจนกินส์ โดยตัวแบบพยากรณ์ที่แม่นยำที่สุด คือ ตัวแบบพยากรณ์ที่ให้ค่าคลาดเคลื่อนจากมาตรวัดความคลาดเคลื่อนต่ำที่สุด สำหรับตัวแบบผสมผสานที่งานวิจัยนี้ได้นำเสนอ เป็นการสร้างตัวแบบพยากรณ์ด้วยเทคนิคอัจฉริยะ และมีการเลือกอินพุตเวกเตอร์ เพื่อสร้างชุดข้อมูลฝึกสอนให้กับตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ โดยใช้รูปแบบออโตรีเกรสซีฟ (Autoregressive) ของตัวแบบ ARIMA เป็นตัวกำหนดค่าสังเกตล่าหลังที่จะนำมาเป็นอินพุตให้กับตัวแบบ ANN และตัวแบบ GASVR ด้วยสมมุติฐานว่า ถ้าใช้อินพุตที่เป็นค่าสังเกตล่าหลัง ซึ่งมีความสัมพันธ์กับเอาต์พุตที่เป็นค่าสังเกต ณ เวลา  $t$  ใด ๆ มาใช้ในการฝึกสอนตัวแบบ จะทำให้ค่าพยากรณ์ของตัวแบบมีความแม่นยำขึ้น โดยบทนี้จะกล่าวถึง วิธีการวิจัย เครื่องมือที่ใช้ในการวิจัย และกระบวนการต่าง ๆ ของการวิจัย ซึ่งมีรายละเอียดดังนี้

#### 3.1 กรอบแนวคิดของการวิจัย

แนวคิดหลักของงานวิจัยนี้ คือ ต้องการนำเสนอตัวแบบพยากรณ์ผสมผสาน ระหว่างเทคนิคอัจฉริยะ ซึ่งเป็นวิธีการด้านแมชชีนเลิร์นนิง ร่วมกับวิธีการวิเคราะห์หอนุกรมเวลาเชิงเดี่ยว สำหรับนำมาใช้สร้างตัวแบบการพยากรณ์ข้อมูลที่อยู่ในรูปแบบของหอนุกรมเวลาเชิงเดี่ยว

นอกเหนือจากวิธีการสร้างตัวแบบการพยากรณ์ด้วยวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ซึ่งตัวแบบที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ได้แก่ ตัวแบบ SLR ตัวแบบ Holt-Winters และตัวแบบ ARIMA จะใช้ข้อมูลสำหรับสร้างตัวแบบการพยากรณ์ เป็นค่าสังเกตที่สนใจ โดยนำมาเรียงตามลำดับเวลา ให้เป็นข้อมูลที่อยู่ในรูปแบบของอนุกรมเวลา จากนั้นแบ่งข้อมูลเป็น 2 ชุด ได้แก่ ชุดข้อมูลฝึกสอน (Training data set) และชุดข้อมูลตรวจสอบ (Validation data set) โดยชุดข้อมูลฝึกสอน เป็นชุดข้อมูลสำหรับนำมาใช้สร้างตัวแบบการพยากรณ์ ส่วนชุดข้อมูลตรวจสอบ เป็นชุดข้อมูลที่ใช้วัดความแม่นยำของตัวแบบการพยากรณ์ด้วยมาตรวัดความคลาดเคลื่อน แต่การสร้างตัวแบบการพยากรณ์ด้วยเทคนิคอัจฉริยะ ได้แก่ ตัวแบบ ANN และตัวแบบ GASVR มีชุดข้อมูลฝึกสอนที่อยู่ในรูปแบบ  $(D = \{(x_i, y_i)\}_{i=1}^n)$  ซึ่งประกอบด้วยเซตของอินพุต ( $x_i$ ) ที่เป็นค่าสังเกตล่าหลัง ตั้งแต่ 1 ถึง  $p$  คาบเวลา และเซตของเอาต์พุต ( $y_i$ ) ที่เป็นค่าสังเกต ณ เวลา  $t$  ใด ๆ ดังนั้น การนำข้อมูลที่อยู่ในรูปแบบของอนุกรมเวลาเชิงเดี่ยว มาใช้เป็นชุดข้อมูลฝึกสอนเพื่อสร้างตัวแบบพยากรณ์ด้วยเทคนิคอัจฉริยะ จะต้องดำเนินการเลือกค่าสังเกตล่าหลังที่คาบเวลาใดบ้าง สำหรับนำมาใช้เป็นเซตของอินพุตในชุดข้อมูลฝึกสอน โดยงานวิจัยนี้ นำเสนอวิธีการเลือกค่าสังเกตล่าหลัง ด้วยรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA เพื่อใช้เป็นเซตของอินพุตในชุดข้อมูลฝึกสอน สำหรับสร้างตัวแบบพยากรณ์ด้วยเทคนิคอัจฉริยะ ซึ่งตัวแบบพยากรณ์ดังกล่าว มีลักษณะเป็นตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ที่เป็นวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ร่วมกับเทคนิคอัจฉริยะ จากนั้น ประเมินผลการพยากรณ์ของตัวแบบผสมผสาน เปรียบเทียบกับตัวแบบที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ด้วยการวัดความคลาดเคลื่อน ด้วยมาตรวัดความคลาดเคลื่อน SSE (Sum square error), MAE (Mean absolute error), RMSE (Root mean square error) และ MAPE (Mean absolute percentage error) เพื่อแสดงให้เห็นถึงประสิทธิภาพในด้านความแม่นยำของค่าพยากรณ์จากตัวแบบผสมผสาน โดยมีกรอบแนวคิดของการวิจัย ดังรูปที่ 3.1



รูปที่ 3.1 กรอบแนวคิดของการวิจัย

### 3.1.1 ชุดข้อมูล

ชุดข้อมูล (Data set) ที่นำมาใช้ดำเนินการวิจัย ประกอบด้วยข้อมูล 2 ชุด โดยข้อมูลทั้ง 2 ชุด คือ หน่วยจำหน่ายไฟฟ้ารายเดือน ซึ่งเป็นข้อมูลเชิงปริมาณ ที่อยู่ในรูปของตัวเลข (Numeric) และข้อมูลทั้ง 2 ชุด จะถูกนำมาสร้างเป็นอนุกรมเวลา 2 ชุด จากนั้น ดำเนินการวิจัยด้วยกรอบแนวคิดของการวิจัย ดังรูปที่ 3.1 เช่นเดียวกันทั้ง 2 ชุด โดยรายละเอียดของข้อมูลทั้ง 2 ชุด เป็นดังนี้

3.1.1.1 ข้อมูลชุดที่ 1 เป็นข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวงแห่งประเทศไทย ที่จำหน่ายให้กับผู้บริโภคประเภทบ้านอยู่อาศัย ที่อยู่ในเขตพื้นที่จังหวัด กรุงเทพมหานคร

สมุทรปราการ และนนทบุรี ซึ่งเป็นข้อมูลทุติยภูมิจากรายงานสถานการณ์การจำหน่ายไฟฟ้าประจำเดือน ตั้งแต่เดือนมกราคม 2553 ถึงเดือนพฤษภาคม 2558 ที่เผยแพร่ในเว็บไซต์ของการไฟฟ้านครหลวง (2558) โดยมีตัวอย่างของข้อมูลหน่วยจำหน่ายไฟฟ้าในรายงานสถานการณ์การจำหน่ายไฟฟ้าประจำเดือนธันวาคม 2554 ที่นำมาสร้างเป็นอนุกรมเวลา ดังรูปที่ 3.2

รายการ	ก	ข	ค	สะสมสำหรับ ปี 2554	% เปลี่ยนแปลง	
	เดือน พ.ย. 2554	เดือน ส.ค. 2553	เดือน ส.ค. 2554		ต่อปีที่ แล้ว*	ต่อเดือนที่ แล้ว**
<b>1. ข้อมูลการใช้ไฟฟ้า</b>						
1.1 ข้อมูลหน่วยซื้อ(หน่วย : ล้านหน่วย)	3,544.74	3,588.98	3,428.88	45,753.98	-1.90	-3.33
1.1.1 ข้อมูลหน่วยซื้อจากฟผ.(หน่วย : ล้านหน่วย)	3,543.80	3,588.95	3,428.20	45,748.88	-1.90	-3.32
1.1.1.1 แยกตามแรงดัน						
1.1.1.1.1 69 เควี	1,708.83	1,715.35	1,818.35	21,857.47	-0.74	-5.17
1.1.1.1.2 115 เควี	1,418.88	1,501.87	1,431.93	19,278.87	-0.83	1.08
1.1.1.1.3 230 เควี	420.81	369.93	375.92	4,810.32	-10.52	-10.83
1.1.1.2 จำนวนสถานีต้นทาง แยกตามแรงดัน						
1.1.2.1 69 เควี	11	11	11	11	0.00	0.00
1.1.2.2 115 เควี	12	12	12	12	0.00	0.00
1.1.2.3 230 เควี	3	3	3	3	0.00	0.00
1.1.2 ข้อมูลหน่วยซื้อจาก VSPP(หน่วย : ล้านหน่วย)	0.84	0.01	0.48	7.30	30.70	-42.88
1.2 หน่วยจำหน่าย(หน่วย : ล้านหน่วย)	3,416.40	3,472.17	3,277.95	43,847.75	-1.94	-4.05
1.2.1 แยกตามแรงดัน						
1.2.1.1 220 โวลท์	1,598.05	1,604.93	1,531.37	20,485.59	-1.84	-4.05
1.2.1.2 12 - 24 เควี	1,527.73	1,587.07	1,485.82	19,807.49	-1.88	-4.05
1.2.1.3 69 - 115 เควี	292.82	300.17	280.76	3,755.67	-2.75	-4.05
1.2.2 ตามประเภทผู้ใช้ไฟฟ้า						
1.2.2.1 บ้านอยู่อาศัย	819.79	773.99	703.08	10,124.81	-5.49	-14.24
1.2.2.1.1 พลังงานไฟฟ้าไม่เต็ม 150 หน่วย	32.24	30.26	28.16	422.48	1.47	-12.88

รูปที่ 3.2 ตัวอย่างหน่วยจำหน่ายไฟฟ้าประจำเดือนธันวาคม 2554 ของข้อมูลชุดที่ 1

3.1.1.2 ข้อมูลชุดที่ 2 เป็นข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้าส่วนภูมิภาคแห่งประเทศไทย เขต 3 ซึ่งเป็นหน่วยจำหน่ายไฟฟ้ารวมทุกประเภท ไม่รวมไฟฟ้าสาธารณะ ที่จำหน่ายให้ผู้บริโภคในเขตพื้นที่จังหวัด นครราชสีมา ชัยภูมิ บุรีรัมย์ และสุรินทร์ โดยได้รับความอนุเคราะห์ข้อมูลจากแผนกข้อมูลสถิติและการวิเคราะห์ การไฟฟ้าส่วนภูมิภาคเขต 3 ภาค 2 จังหวัดนครราชสีมา (2559) ซึ่งข้อมูลชุดที่ 2 แสดงได้ดังรูปที่ 3.3 โดยเป็นปริมาณการใช้ไฟฟ้ารายเดือน หน่วยเป็น

กิโลวัตต์ต่อชั่วโมง (kWh) ซึ่ง 1 กิโลวัตต์ต่อชั่วโมงเท่ากับ 1 หน่วยจำหน่ายไฟฟ้า ดังนั้น ข้อมูลชุดที่ 2 เป็นหน่วยจำหน่ายไฟฟ้าประจำเดือน ตั้งแต่เดือนมกราคม 2553 ถึงเดือนพฤษภาคม 2558 เช่นเดียวกับข้อมูลชุดที่ 1 แต่เป็นหน่วยจำหน่ายไฟฟ้าจากแหล่งข้อมูลต่างกัน

หน่วย : kWh						
	หน่วยจำหน่ายของ การไฟฟ้าส่วนภูมิภาคเขต 3 ภาค 2 นครราชสีมา (นครราชสีมา,ชัยภูมิ,บุรีรัมย์,สุรินทร์)					
	2553	2554	2555	2556	2557	2558
มกราคม	468,062,972	469,157,349	521,979,337	564,154,766	536,580,506	577,229,384
กุมภาพันธ์	468,898,512	462,186,363	545,191,569	545,133,959	538,054,440	554,355,585
มีนาคม	535,744,297	537,535,332	614,103,715	643,263,476	661,596,387	690,865,069
เมษายน	518,356,810	499,419,690	562,769,468	621,981,893	614,493,780	648,949,884
พฤษภาคม	545,534,513	537,553,805	618,524,641	644,196,549	665,002,672	691,640,099
มิถุนายน	512,131,218	529,568,345	578,384,158	605,815,254	642,815,098	
กรกฎาคม	529,979,351	550,872,927	602,212,643	616,542,081	652,499,449	
สิงหาคม	516,412,482	553,122,223	589,982,675	623,199,820	653,130,098	
กันยายน	499,848,063	539,200,364	567,805,850	603,366,962	636,184,546	
ตุลาคม	510,020,024	539,923,042	592,874,725	605,633,376	649,222,900	
พฤศจิกายน	477,126,337	520,393,726	596,460,130	595,859,522	625,487,417	
ธันวาคม	485,088,053	503,933,835	589,796,657	554,817,334	594,811,933	
รวม	6,067,202,632	6,242,867,001	6,980,085,569	7,223,964,992	7,469,879,226	3,163,040,021
ที่มา : แผนกข้อมูลสถิติและวิเคราะห์ การไฟฟ้าส่วนภูมิภาคเขต 3 ภาค 2 นครราชสีมา						

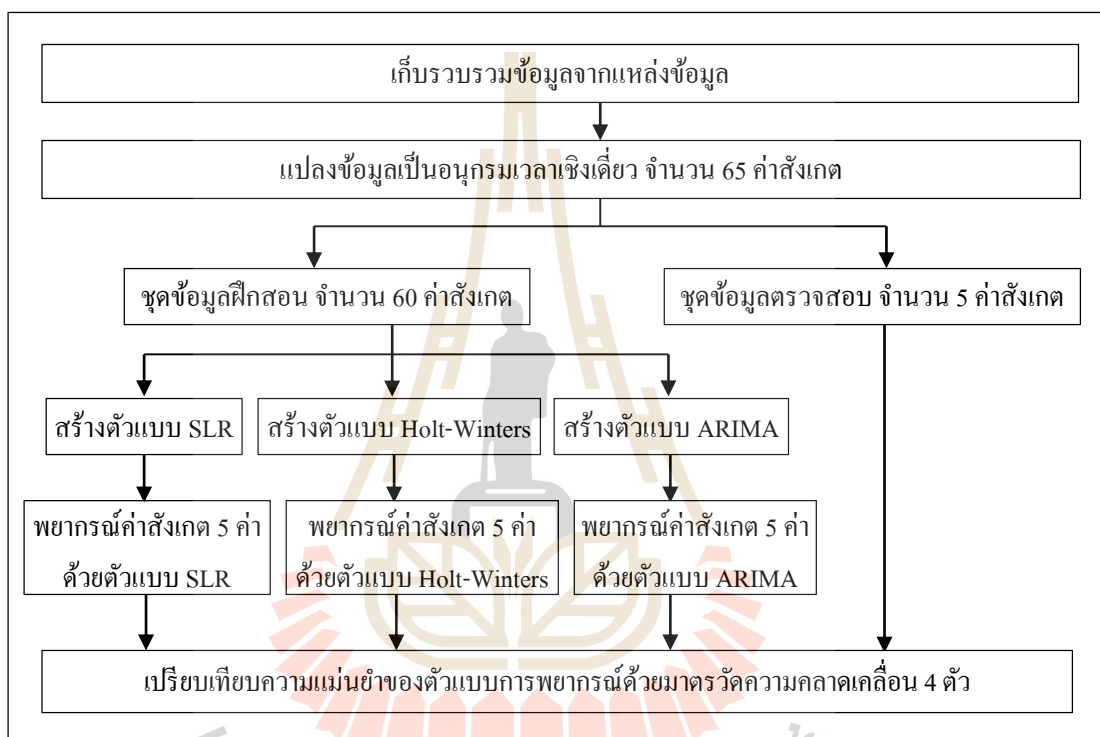
รูปที่ 3.3 หน่วยจำหน่ายไฟฟ้า ตั้งแต่เดือนมกราคม 2553 ถึงเดือนพฤษภาคม 2558 ของข้อมูลชุดที่ 2

### 3.1.2 การสร้างตัวแบบการพยากรณ์ด้วยวิธีการวิเคราะห์ห่ออนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม

เนื่องจากวิทยานิพนธ์ฉบับนี้ เป็นการศึกษาและพัฒนาตัวแบบการพยากรณ์ด้วยเทคนิคการวิเคราะห์ห่ออนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม และเทคนิคอัจฉริยะ ซึ่งทั้ง 2 เทคนิค มีขั้นตอนการสร้างตัวแบบการพยากรณ์ที่แตกต่างกัน ดังนั้น จึงดำเนินการวิจัยโดยแบ่งเป็น 2 ส่วน และจากกรอบแนวคิดของการวิจัย จะดำเนินการสร้างตัวแบบผสมผสานที่มีการเลือกอินพุตของชุดข้อมูลฝึกสอนสำหรับเทคนิคอัจฉริยะ ด้วยการวิเคราะห์รูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ซึ่งเป็นตัวแบบการพยากรณ์ที่สร้างจากเทคนิคการวิเคราะห์ห่ออนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ดังนั้น จึงเริ่มต้นดำเนินการวิจัยด้วยการสร้างตัวแบบการพยากรณ์จากวิธีวิเคราะห์ห่ออนุกรมเวลาเชิงเดี่ยวแบบ



ดั้งเดิม โดยดำเนินการทดลองสร้างตัวแบบพยากรณ์ 2 ครั้ง กับข้อมูล 2 ชุด ด้วยขั้นตอนเดียวกัน ซึ่งข้อมูลทั้ง 2 ชุด เป็นหน่วยจำหน่ายไฟฟ้าเหมือนกัน แต่มาจากแหล่งข้อมูลต่างกัน จากนั้นวัดความแม่นยำของแต่ละตัวแบบพยากรณ์ ด้วยมาตรวัดความคลาดเคลื่อน SSE, MAE, RMSE และ MAPE เพื่อนำไปเปรียบเทียบความแม่นยำกับตัวแบบผสมผสานในภายหลัง โดยขั้นตอนการสร้างตัวแบบการพยากรณ์ แสดงได้ดังรูปที่ 3.4



รูปที่ 3.4 ขั้นตอนการสร้างตัวแบบการพยากรณ์ด้วยวิธีการวิเคราะห์อนุกรมเวลาแบบดั้งเดิม

จากรูปที่ 3.4 อธิบายแต่ละขั้นตอนได้ดังนี้

1. เก็บรวบรวมข้อมูลจากแหล่งข้อมูล โดยเป็นข้อมูลที่มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้าประจำเดือน ตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 จำนวน 2 ชุด จากแหล่งข้อมูล 2 แหล่ง ได้แก่ หน่วยจำหน่ายไฟฟ้าประเภทบ้านอยู่อาศัย จากรายงานสถานการณ์การจำหน่ายไฟฟ้าประจำเดือน ของการไฟฟ้านครหลวง และหน่วยจำหน่ายไฟฟ้ารวม ยกเว้นไฟฟ้าสาธารณะของการไฟฟ้าส่วนภูมิภาค เขต 3 โดยข้อมูลทั้ง 2 ชุด มีจำนวนเท่ากัน คือ 65 ค่า



2. แปลงข้อมูลที่เก็บรวบรวมมา ให้อยู่ในรูปแบบของอนุกรมเวลา ด้วยการนำค่าสังเกตเรียงตามลำดับเวลา โดยมีคาบเวลาเป็นรายเดือน ตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 ทั้งหมด 65 ค่าสังเกต ดังรูปที่ 3.5

	A	B	C	D
1	Years	Months	<b>Electric Units</b>	TimeID
2	2010	1	776.32	1
3	2010	2	836.16	2
4	2010	3	925.55	3
5	2010	4	1026.5	4
6	2010	5	1082.18	5
7	2010	6	1003.41	6
8	2010	7	930.91	7
9	2010	8	876.83	8
10	2010	9	873.47	9
11	2010	10	820.6	10
12	2010	11	787.33	11
13	2010	12	773.99	12

รูปที่ 3.5 ข้อมูลอนุกรมเวลา

3. แบ่งข้อมูลอนุกรมเวลาออกเป็น 2 ชุด ได้แก่ ชุดข้อมูลฝึกสอน ซึ่งใช้ข้อมูลตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนธันวาคม ปีพ.ศ. 2557 จำนวน 60 ค่าสังเกต สำหรับสร้างตัวแบบพยากรณ์ และชุดข้อมูลตรวจสอบ ซึ่งใช้ข้อมูลตั้งแต่เดือนมกราคมถึงเดือนพฤษภาคม ปีพ.ศ. 2558 จำนวน 5 ค่าสังเกต สำหรับวัดความแม่นยำของตัวแบบพยากรณ์ด้วยมาตรวัดความคลาดเคลื่อน

4. สร้างตัวแบบ SLR ตัวแบบ Holt-Winters และตัวแบบ ARIMA จากชุดข้อมูลฝึกสอนด้วยภาษาอาร์ (R language) ซึ่งภาษาอาร์มีฟังก์ชันที่ใช้สร้างตัวแบบทั้ง 3 อยู่ในแพ็คเกจ (Package) ชื่อ “forecast” ที่เผยแพร่โดย Hyndman (2015) ในส่วนเก็บฟังก์ชันของภาษาอาร์ (CRAN repository) โดยการสร้างตัวแบบ SLR ใช้ฟังก์ชันชื่อ “glm()” การสร้างตัวแบบ Holt-Winters ใช้ฟังก์ชันชื่อ “HoltWinters()” และการสร้างตัวแบบ ARIMA ใช้ฟังก์ชันชื่อ “arima()” ซึ่งคำสั่งสร้างตัวแบบทั้ง 3 ด้วยภาษาอาร์ สรุปได้ดังรูปที่ 3.6

5. นำแต่ละตัวแบบมาพยากรณ์ค่าสังเกต จำนวน 5 ค่า โดยเป็นค่าสังเกตตั้งแต่เดือนมกราคมถึงเดือนพฤษภาคม ปีพ.ศ. 2558 เพื่อนำไปวัดความคลาดเคลื่อน ด้วยการเปรียบเทียบกับค่าสังเกตจริงของชุดข้อมูลตรวจสอบ ซึ่งเป็นค่าสังเกตในช่วงเวลาเดียวกัน
6. วัดความแม่นยำของตัวแบบการพยากรณ์ ด้วยมาตรวัดความคลาดเคลื่อน SSE, MAE (Hamzacebi, 2008), RMSE และ MAPE (Bergmeir and Benítez, 2012) ซึ่งมีสมการดังนี้

$$SSE = \sum_{t=1}^n (Y_t - \hat{Y}_t)^2 \quad \text{----- (3.1)}$$

$$MAE = \frac{1}{n} \sum_{t=1}^n |Y_t - \hat{Y}_t| \quad \text{----- (3.2)}$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \quad \text{----- (3.3)}$$

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| 100 \frac{y_t - \hat{y}_t}{y_t} \right| \quad \text{----- (3.4)}$$

เมื่อ  $y_t$  = ค่าสังเกตที่เวลา  $t$ ,  $\hat{y}_t$  = ค่าพยากรณ์ที่เวลา  $t$  และ  $n$  = จำนวนคาบเวลา

ที่พยากรณ์

```
SLRDF1 <- read.csv(file="d:/CodeR/Data/EuseMEA_SLRmodel.csv", header=TRUE, sep=",")
SLRfit <- glm(SLRDF1[, 4] ~ SLRDF1[, 3])
```

(ก) ชุดคำสั่งสร้างตัวแบบ SLR

```
library("forecast", lib.loc=~R/win-library/3.2")
HoltWintersDF1 <- read.csv(file="d:/CodeR/Data/EuseMEA2010_2014.csv",header=TRUE, sep=",")
HoltWintersTS1 <- ts(HoltWintersDF1[, 3], start=c(2010, 1), end=c(2014, 12), frequency=12)
HWfit <- HoltWinters(HoltWintersTS1)
```

(ข) ชุดคำสั่งสร้างตัวแบบ Holt-Winters

```
library("forecast", lib.loc=~R/win-library/3.2")
arimaDF1 <- read.csv(file="d:/CodeR/Data/EuseMEA2010_2014.csv", header=TRUE, sep=",")
arimaTS1 <- ts(arimaDF1[, 3], start=c(2010, 1), end=c(2014, 12), frequency=12)
p<-arimaorder(auto.arima(arimaTS1))[1]
d<-arimaorder(auto.arima(arimaTS1))[2]
q<-arimaorder(auto.arima(arimaTS1))[3]
P<-arimaorder(auto.arima(arimaTS1))[4]
D<-arimaorder(auto.arima(arimaTS1))[5]
Q<-arimaorder(auto.arima(arimaTS1))[6]
arimaM1 <- arima(arimaTS1, order=c(p, d, q),list(order=c(P, D, Q), period=12))
```

(ค) ชุดคำสั่งสร้างตัวแบบ ARIMA

รูปที่ 3.6 ชุดคำสั่งของภาษาอาร์ สำหรับสร้างตัวแบบ SLR ตัวแบบ Holt-Winter และตัวแบบ ARIMA

### 3.1.3 การสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA

ร่วมกับเทคนิคอัจฉริยะ

ตัวแบบ ANN และตัวแบบ GASVR เป็นตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ ซึ่งเป็นวิธีการด้านแมชชีนเลิร์นนิง ที่มีการเรียนรู้แบบมีผู้ฝึกสอน (Supervised learning) โดยใช้ชุดข้อมูล  $D$  ที่ประกอบด้วยอินพุต ( $x_i$ ) และเอาต์พุต ( $y_i$ ) ซึ่ง  $D = \{(x_i, y_i)\}_{i=1}^n$  โดยอินพุตของตัวแบบคือค่าสังเกตที่ล่าช้าไปตั้งแต่ 1 ถึง  $p$  คาบเวลา และเอาต์พุต คือ ค่าสังเกต ณ เวลา  $t$  ใด ๆ ซึ่งวิทยานิพนธ์ฉบับนี้ นำเสนอวิธีการเลือกอินพุตของตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ โดยใช้การวิเคราะห์รูปแบบออโตรีเกรสซีฟ ของตัวแบบ ARIMA ที่มีวิธีการวิเคราะห์ ดังนี้

จากรูปแบบเต็มของตัวแบบ ARIMA ที่กล่าวไว้ในบทที่ 2 คือ รูปแบบ ARIMA (p, d, q) x (P, D, Q)<sub>S</sub> โดยมีรูปแบบทั่วไปที่อยู่ในรูปของตัวดำเนินการย้อนกลับ (B) ดังสมการที่ 3.5

$$\emptyset(B)\phi_p(B^S)(1 - B^d)(1 - B^{DS})Y_t = \theta_0 + \theta(B)\Theta_Q(B^S)\varepsilon_t \quad \text{----- (3.5)}$$

เมื่ออันดับ p และ P หมายถึง อันดับของรูปแบบออโตรีเกรสซีฟ อันดับ d และ D หมายถึง อันดับของผลต่างคาบเวลาสำหรับการปรับให้อนุกรมเวลามีสถานะนิ่ง (Stationary) และ อันดับ q และ Q หมายถึง อันดับของรูปแบบมูฟวิงเอเวอร์เรจ (Moving average) ซึ่งสิ่งที่ต้องการวิเคราะห์ คือ รูปแบบออโตรีเกรสซีฟ ดังนั้น จึงกำหนดให้อันดับ q และ Q ของสมการที่ 3.3 เท่ากับ 0 และได้เป็นสมการที่ 3.6 ดังนี้

$$\emptyset(B)\phi_p(B^S)(1 - B^d)(1 - B^{DS})Y_t = \theta_0 + \varepsilon_t \quad \text{----- (3.6)}$$

$$\begin{aligned} \text{เมื่อ } \emptyset(B) &= 1 - \emptyset_1 B - \emptyset_2 B^2 - \dots - \emptyset_p B^p \\ \phi_p(B^S) &= 1 - \phi_S B^S - \phi_{2S} B^{2S} - \dots - \phi_{pS} B^{pS} \end{aligned}$$

เพื่อแสดงการวิเคราะห์รูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ในการเลือกค่าสังเกตล่าหลัง สำหรับนำไปเป็นอินพุตให้กับตัวแบบ ANN และตัวแบบ GASVR จะใช้ตัวแบบ ARIMA(1, 0, 0)x(1, 0, 0)<sub>12</sub> เป็นตัวอย่างในการอธิบาย ดังนี้

จากตัวแบบ ARIMA(1, 0, 0)x(1, 0, 0)<sub>12</sub> หมายความว่า อันดับ p=P=1 และจำนวนคาบเวลาใน 1 ฤดูกาล (S) เท่ากับ 12 ดังนั้น เมื่อแทนค่าอันดับลงในสมการที่ 3.6 จะได้ดังนี้

$$\begin{aligned} (1 - \emptyset_1 B)(1 - \phi_{12} B^{12})Y_t &= \theta_0 + \varepsilon_t \\ (1 - \emptyset_1 B - \phi_{12} B^{12} + \emptyset_1 \phi_{12} B^{13})Y_t &= \theta_0 + \varepsilon_t \\ \theta_0 + \varepsilon_t + \emptyset_1 Y_{t-1} + \phi_{12} Y_{t-12} - \emptyset_1 \phi_{12} Y_{t-13} &= Y_t \quad \text{----- (3.7)} \end{aligned}$$

จากสมการที่ 3.7 จะเลือกค่าสังเกตล่าหลัง 1, 12 และ 13 คาบเวลา ( $Y_{t-1}$ ,  $Y_{t-12}$ ,  $Y_{t-13}$ ) เป็นอินพุตของตัวแบบ และมีเอาท์พุตเป็นค่าสังเกต ณ เวลา t ( $Y_t$ ) โดยเป็นชุดข้อมูล ดังรูปที่ 3.7

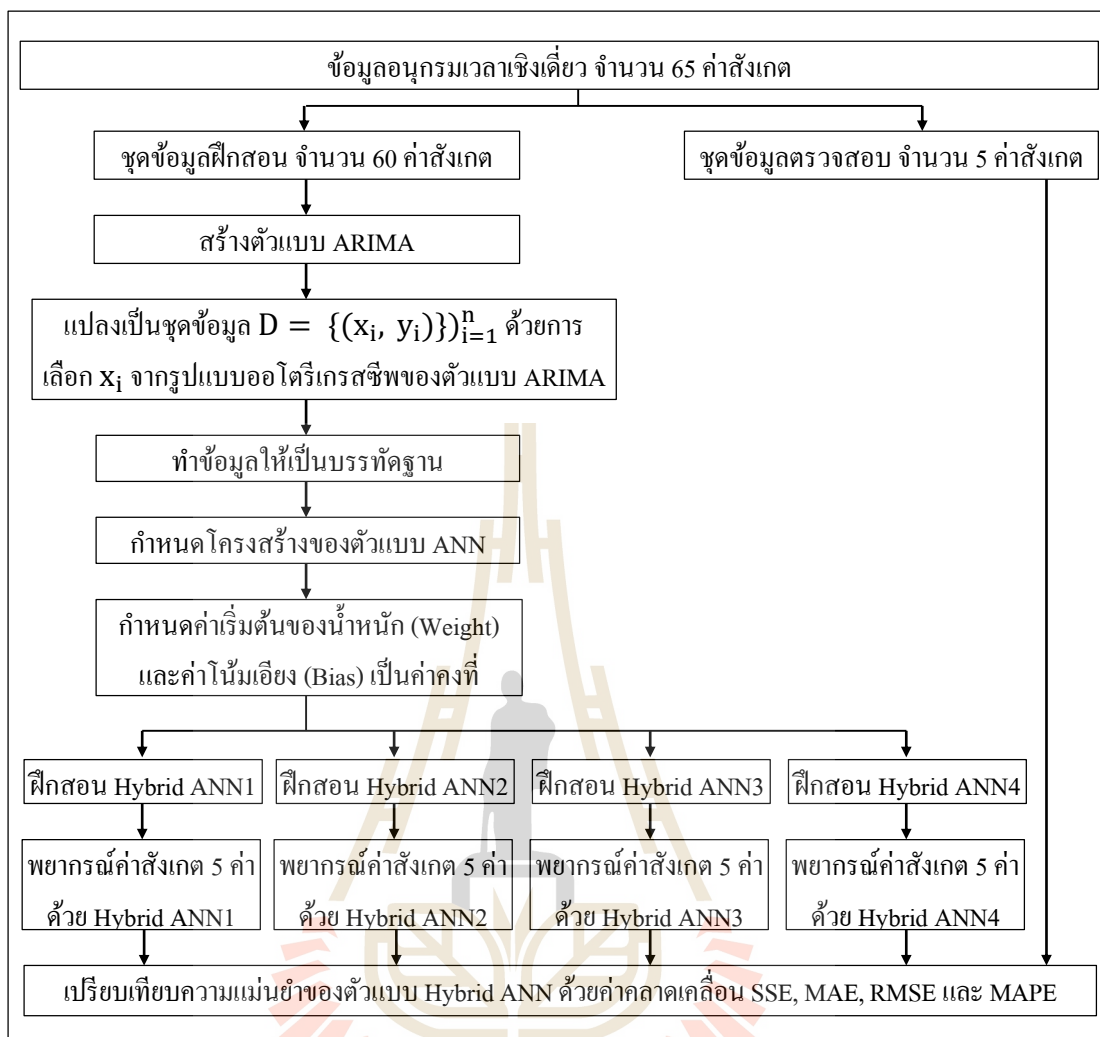
Time_ID	Output( $y_t$ )	Input1( $y_{t-1}$ )	Input2( $y_{t-2}$ )	Input3( $y_{t-3}$ )
14	787.29	t=13 707.16	t=2 836.16	t=1 776.32
15	806.64	t=14 787.29	t=3 925.55	t=2 836.16
16	864.58	t=15 806.64	t=4 1026.5	t=3 925.55
17	968.86	• 864.58	• 1082.18	• 1026.5
18	933.59	• 968.86	• 1003.41	• 1082.18
19	912.8	• 933.59	• 930.91	• 1003.41
20	902.59	• 912.8	• 876.83	• 930.91
21	886.08	• 902.59	• 873.47	• 876.83
22	830.9	• 886.08	• 820.6	• 873.47
23	886.93	t=22 830.9	t=9 787.33	t=10 820.6
24	703.06	t=23 886.93	t=10 773.99	t=11 787.33

รูปที่ 3.7 ตัวอย่างชุดข้อมูลฝึกสอนที่ได้จากการวิเคราะห์รูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA

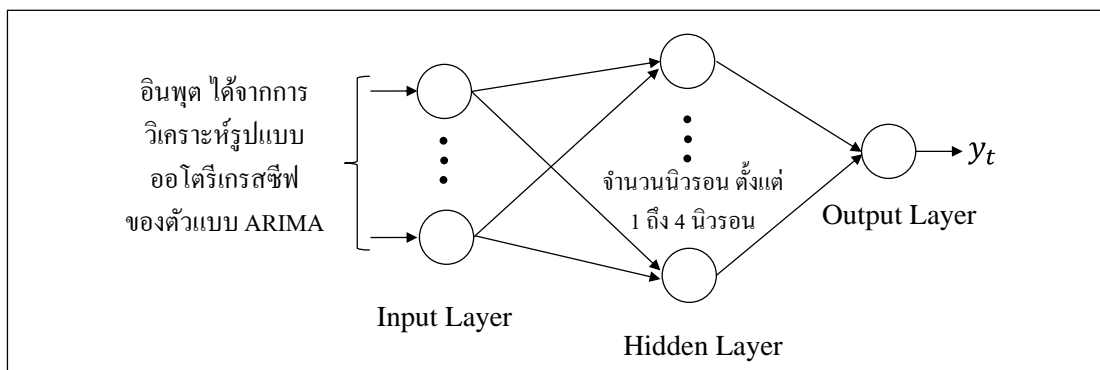
เมื่อทราบค่าสังเกตล่าสุดที่เป็นอินพุตให้กับตัวแบบการพยากรณ์ จะนำชุดข้อมูลฝึกสอนที่อยู่ในรูปแบบของอนุกรมเวลา มาแปลงเป็นชุดข้อมูล  $D = \{(x_i, y_i)\}_{i=1}^n$  เพื่อนำไปดำเนินการสร้างตัวแบบ Hybrid ANN และตัวแบบ Hybrid GASVR ซึ่งมีขั้นตอนดำเนินการวิจัยที่แตกต่างกัน ดังนี้

### 3.1.3.1 ขั้นตอนการสร้างตัวแบบ Hybrid ANN

สำหรับตัวแบบ Hybrid ANN เป็นตัวแบบที่สร้างจากเทคนิคเครือข่ายประสาทเทียม ซึ่งเป็นเทคนิคที่ไม่มีพารามิเตอร์ (Nonparametric) จึงไม่ต้องใช้จินตคณิตที่ซับซ้อนในการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดเหมือนกับเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชัน แต่เครือข่ายประสาทเทียม มีข้อเสีย คือ ปัญหาโอเวอร์ฟิตติง (Over-fitting) ดังนั้น เพื่อลดปัญหาโอเวอร์ฟิตติงจึงดำเนินการทดลองด้วยการกำหนดโครงสร้างของเครือข่ายประสาทเทียมไม่ให้ซับซ้อน โดยมีชั้นซ่อนเร้น (Hidden layer) 1 ชั้น และมีจำนวนนิวรอน (Neuron) ในชั้นซ่อนเร้นตั้งแต่ 1 ถึง 4 นิวรอน เพื่อสร้างเป็นตัวแบบ Hybrid ANN จำนวน 4 ตัวแบบ และเลือกตัวแบบที่มีโครงสร้างเหมาะสมที่สุดสำหรับนำมาพยากรณ์อนุกรมเวลาที่ศึกษา โดยการวัดค่าคลาดเคลื่อนด้วยมาตรวัดความคลาดเคลื่อน SSE, MAE, RMSE และ MAPE ซึ่งขั้นตอนการสร้างตัวแบบ Hybrid ANN ที่มีโครงสร้างเหมาะสมที่สุดเป็นดังรูปที่ 3.8 และโครงสร้างของตัวแบบ Hybrid ANN ที่ใช้ดำเนินการทดลอง แสดงได้ดังรูปที่ 3.9



รูปที่ 3.8 ขั้นตอนการสร้างตัวแบบ Hybrid ANN ที่มีโครงสร้างเหมาะสมที่สุด



รูปที่ 3.9 โครงสร้างของตัวแบบ Hybrid ANN ที่ใช้ดำเนินการทดลอง

สำหรับการสร้างตัวแบบ Hybrid ANN จะการดำเนินการทดลอง โดยใช้โปรแกรม MatLab เป็นเครื่องมือ และจากขั้นตอนการสร้างตัวแบบ Hybrid ANN ดังรูปที่ 3.8 อธิบายแต่ละขั้นตอนได้ดังนี้

1. จากข้อมูลอนุกรมเวลา แบ่งข้อมูลเป็นชุดข้อมูลฝึกสอน และชุดข้อมูลตรวจสอบเช่นเดียวกับการสร้างตัวแบบพยากรณ์ด้วยวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม
2. สร้างตัวแบบ ARIMA จากชุดข้อมูลฝึกสอน
3. แปลงชุดข้อมูลฝึกสอนให้อยู่ในรูปแบบของชุดข้อมูล  $D = \{(x_i, y_i)\}_{i=1}^n$  ด้วยการวิเคราะห์รูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA
4. ทำข้อมูลให้เป็นบรรทัดฐาน (Data normalization) มีค่าระหว่าง 0 ถึง 1 เพื่อป้องกันการเกิดดาตาโอเวอร์โฟลว์ (Data overflow) เนื่องจากกระบวนการคำนวณซ้ำของเครือข่ายประสาทเทียม
5. กำหนดโครงสร้างของตัวแบบ Hybrid ANN จำนวน 4 ตัวแบบ แต่ละตัวแบบ แตกต่างที่จำนวนนิวรอนในชั้นซ่อนเร้น โดยใช้ฟังก์ชันชื่อ “newff()” ซึ่งเป็นฟังก์ชันในการสร้างตัวแบบ Hybrid ANN ที่มีโครงสร้างเป็นเครือข่ายประสาทเทียมไปข้างหน้าแบบแพร่กลับ (Feed forward neural network with backpropagation)
6. กำหนดค่าเริ่มต้นของน้ำหนัก (Weight) และค่าโน้มเอียง (Bias) เป็นค่าคงที่ แทนการสุ่มค่า เพื่อให้เอาท์พุทของตัวแบบ Hybrid ANN คู่เข้าสู่คำตอบได้ดีขึ้น โดยค่าที่กำหนด ได้จากการทดลองสุ่ม ที่ทำให้ค่าคลาดเคลื่อนมีค่าน้อยที่สุด
7. ฝึกสอนตัวแบบ Hybrid ANN จำนวน 4 ตัวแบบ ตามโครงสร้างที่กำหนดไว้ในข้อ 5 โดยใช้ข้อมูลที่ผ่านมาการทำให้เป็นบรรทัดฐานของข้อ 4 สำหรับฝึกสอนตัวแบบ Hybrid ANN ทั้ง 4 ตัวแบบ
8. นำตัวแบบ Hybrid ANN ทั้ง 4 ตัวแบบ พยากรณ์ค่าสังเกตล่วงหน้า 5 คาบเวลา จากนั้น ทำการคืนอร์มอลไลซ์ (Denormalize) แปลงข้อมูลกลับเป็นค่าปกติ ซึ่งเป็นค่าสังเกตของเดือนมกราคมถึงเดือนพฤษภาคม ปีพ.ศ. 2558 เพื่อนำไปวัดค่าคลาดเคลื่อน SSE, MAE, RMSE และ MAPE
9. เลือกตัวแบบ Hybrid ANN ที่เหมาะสมที่สุด โดยพิจารณาจากค่าคลาดเคลื่อนทั้ง 4 ตัว สำหรับนำไปพยากรณ์ค่าสังเกตล่วงหน้า เพื่อเปรียบเทียบความแม่นยำในการ

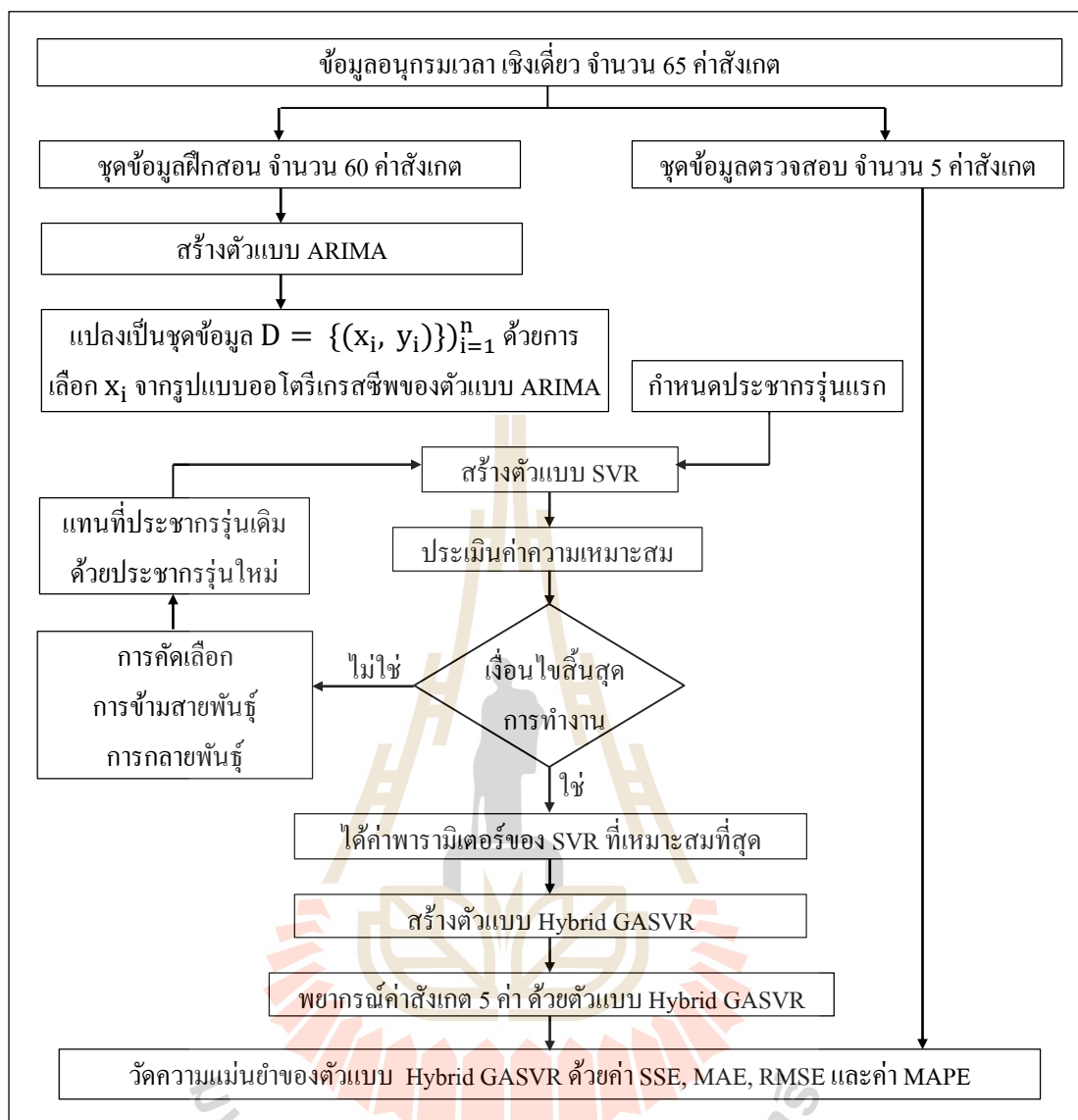
พยากรณ์กับตัวแบบ Hybrid GASVR และตัวแบบพยากรณ์ที่สร้างจากวิธีการวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม

### 3.1.3.2 ขั้นตอนการสร้างตัวแบบ Hybrid GASVR

การวิจัยนี้ ดำเนินการสร้างตัวแบบ Hybrid GASVR ด้วยการใช้ภาษาอาร์ (R language) ซึ่งมีฟังก์ชัน svm() เป็นฟังก์ชันในการสร้างตัวแบบ SVR โดยอยู่ในแพ็คเกจชื่อ “e1071” ที่เผยแพร่โดย Meyer et al. (2015) ในส่วนเก็บฟังก์ชันของภาษาอาร์ (CRAN repository) ซึ่งการสร้างตัวแบบ SVR จะต้องกำหนดค่าพารามิเตอร์ที่สำคัญ 3 ตัว ได้แก่ Cost (C), Epsilon ( $\mathcal{E}$ ) และพารามิเตอร์ของเคอร์เนลฟังก์ชัน (Parameter of kernel function) ซึ่งวิทยานิพนธ์ฉบับนี้ นำเสนอวิธีการกำหนดค่าพารามิเตอร์ของ SVR ที่เหมาะสมที่สุด โดยใช้เจเนติกอัลกอริทึม ซึ่งขั้นตอนการสร้างตัวแบบ Hybrid GASVR แสดงได้ดังรูปที่ 3.10







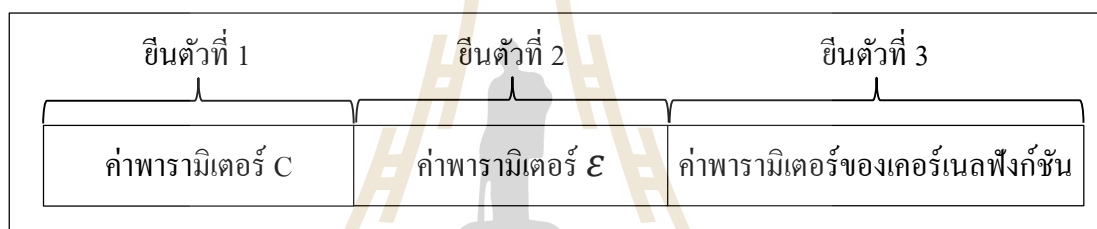
รูปที่ 3.10 ขั้นตอนการสร้างตัวแบบ Hybrid GASVR

จากขั้นตอนการสร้างตัวแบบ Hybrid GASVR ดังรูปที่ 3.10 อธิบายแต่ละขั้นตอนได้ดังนี้

1. แบ่งข้อมูลอนุกรมเวลา เป็นชุดข้อมูลฝึกสอน และชุดข้อมูลตรวจสอบ จากนั้น แปลงชุดข้อมูลฝึกสอนให้อยู่ในรูปแบบของข้อมูล  $D = \{(x_i, y_i)\}_{i=1}^n$  ด้วยการวิเคราะห์รูปแบบออโตรีเกรสซีฟ ของตัวแบบ ARIMA เช่นเดียวกับขั้นตอนการสร้างตัวแบบ Hybrid ANN
2. สร้างตัวแบบ ARIMA จากชุดข้อมูลฝึกสอน

3. สร้างตัวแบบ SVR โดยใช้จีเนติกอัลกอริทึม ค้นหาค่าพารามิเตอร์ของ SVR ที่เหมาะสมที่สุด ซึ่งภาษาอาร์มีฟังก์ชัน “rgba()” ในแพคเกจชื่อ “genalg” เผยแพร่โดย Willighagen and Ballings (2015) ในส่วนเก็บฟังก์ชันของภาษาอาร์ (CRAN repository) สำหรับค้นหาคำตอบตามกระบวนการของจีเนติกอัลกอริทึม ดังนี้

3.1 กำหนดประชากรรุ่นแรก (Initial population) ด้วยการเลือกโครโมโซม (Chromosome) เข้ามาเป็นสมาชิกของประชากรรุ่นแรกแบบสุ่ม จำนวน 200 โครโมโซม แต่ละโครโมโซมประกอบด้วยยีน (Gene) จำนวน 3 ตัว ยีนแต่ละตัว หมายถึง ค่าพารามิเตอร์ของ SVR ( $C$ ,  $\epsilon$  และพารามิเตอร์ของเคอร์เนลฟังก์ชัน) มีชนิดเป็นตัวเลขจำนวนจริง โดยเรียกโครโมโซมลักษณะแบบนี้ว่า ฟลอยต์โครโมโซม (Floats chromosome) ดังรูปที่ 3.11



รูปที่ 3.11 โครงสร้างของโครโมโซม

3.2 การประเมินค่าความเหมาะสม (Fitness evaluation) เป็นการประเมินค่าความเหมาะสมของแต่ละโครโมโซม เพื่อใช้ในการคัดเลือกโครโมโซมลูกหลาน (Offspring) ที่จะไปเป็นประชากรในรุ่นถัดไป โดยงานวิจัยนี้ กำหนดฟังก์ชันการประเมิน (Evaluation function) ด้วยมาตรวัดค่าคลาดเคลื่อน RMSE

3.3 ตรวจสอบเงื่อนไขสิ้นสุดการทำงานของจีเนติกอัลกอริทึม ซึ่งงานวิจัยนี้ กำหนดเงื่อนไขสิ้นสุดการทำงานของจีเนติกอัลกอริทึม เมื่อสร้างประชากรได้ 100 รุ่น

3.4 ถ้ายังไม่เข้าเงื่อนไขสิ้นสุดการทำงานของจีเนติกอัลกอริทึม ให้ไปดำเนินการคัดเลือก (Selection) ซึ่งเป็นขั้นตอนในการคัดเลือกโครโมโซมลูกหลาน เพื่อไปเป็นประชากรในรุ่นถัดไป โดยงานวิจัยนี้ ใช้วิธีการคัดเลือกแบบวงล้อรูเล็ต (Roulette wheel) ซึ่งเป็นวิธีการคัดเลือกที่เป็นค่าเริ่มต้นของฟังก์ชัน rgba() ที่ใช้ในการค้นหาคำตอบตามกระบวนการของจีเนติกอัลกอริทึม

3.5 ดำเนินการข้ามสายพันธุ (Crossover) ซึ่งเป็นการสร้างโครโมโซมลูกหลาน ด้วยการนำโครโมโซมพ่อแม่จำนวน 2 โครโมโซม มาดำเนินการข้ามสายพันธุ โดยกำหนดอัตราการข้ามสายพันธุเท่ากับ 0.8

3.6 ดำเนินการกลายพันธุ (Mutation) ซึ่งเป็นการสร้างโครโมโซมลูกหลาน ด้วยการนำโครโมโซมพ่อแม่จำนวน 1 โครโมโซม มาดำเนินการกลายพันธุ โดยกำหนดอัตราการกลายพันธุเท่ากับ 0.01

3.7 ดำเนินการแทนที่ประชากรเดิม (Replacement) ซึ่งเป็นการแทนที่ประชากรเดิมด้วยประชากรใหม่ที่มีค่าความเหมาะสมดีขึ้น โดยดำเนินการแทนที่โครโมโซมพ่อแม่ด้วยโครโมโซมลูกหลาน ตามอัตราการข้ามสายพันธุเท่ากับ 0.8 หรือ 20% (โครโมโซมพ่อแม่ 40 และโครโมโซมลูกหลาน 160)

3.8 ดำเนินการสร้างตัวแบบ SVR ด้วยประชากรรุ่นถัดไปของจีเนติกอัลกอริทึม และทำซ้ำขั้นตอนที่ 3.2 ถึง 3.7 จนกระทั่งเข้าสู่เงื่อนไขสิ้นสุดการทำงานของจีเนติกอัลกอริทึม

4. เมื่อได้ค่าพารามิเตอร์ของ SVR ที่เหมาะสมที่สุด จากจีเนติกอัลกอริทึม จะนำไปสร้างตัวแบบ SVR โดยใช้ชุดข้อมูลฝึกสอน ได้เป็นตัวแบบ Hybrid GASVR

5. พยากรณ์ค่าสังเกตจำนวน 5 ค่า ด้วยตัวแบบ Hybrid GASVR ซึ่งเป็นค่าสังเกตของเดือนมกราคมถึงเดือนพฤษภาคม ปี พ.ศ. 2558 เพื่อนำไปวัดความแม่นยำในการพยากรณ์ร่วมกับค่าสังเกตจริงในช่วงเวลาเดียวกันจากชุดข้อมูลทดสอบ

6. วัดความแม่นยำของตัวแบบ Hybrid GASVR ด้วยมาตรวัดความคลาดเคลื่อน SSE, MAE, RMSE และ MAPE เพื่อนำไปเปรียบเทียบกับตัวแบบ Hybrid ANN และตัวแบบที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม

จากที่กล่าวมาทั้งหมด เป็นวิธีดำเนินการวิจัยภายใต้กรอบแนวคิดการวิจัย ซึ่งดำเนินการวิจัย เพื่อแสดงวิธีการสร้างตัวแบบผสมผสาน (Hybrid model) ระหว่างรูปแบบออโตรีเกรสซีฟ (Autoregressive) ของตัวแบบ ARIMA ร่วมกับเทคนิคอัจฉริยะ จากข้อมูลที่อยู่ในรูปแบบของอนุกรมเวลาเชิงเดี่ยว โดยดำเนินการศึกษากับอนุกรมเวลาจำนวน 2 ชุด ได้แก่ ชุดที่ 1 เป็นอนุกรมเวลาที่มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวงแห่งประเทศไทย และชุดที่ 2 เป็นอนุกรมเวลาที่มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้าของการไฟฟ้าส่วนภูมิภาคแห่งประเทศไทย เขต 3 ซึ่งข้อมูลทั้ง 2 ชุด เป็นหน่วยจำหน่ายไฟฟ้าตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558

จำนวน 65 ค่าสังเกต และการสร้างตัวแบบผสมผสาน จะมีการแปลงอนุกรมเวลาเชิงเดี่ยวให้เป็นชุดข้อมูลฝึกสอนที่จะนำไปสร้างตัวแบบพยากรณ์ด้วยเทคนิคอัจฉริยะ ซึ่งชุดข้อมูลฝึกสอนอยู่ในรูปแบบคู่ลำดับของอินพุต  $x_t$  และเอาต์พุต  $y_t$  โดยการกำหนดอินพุต  $x_t$  ที่เป็นค่าสังเกตล่าช้า จะใช้วิธีการเลือกจากรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ทำให้ตัวแบบที่ได้ เป็นตัวแบบผสมผสานระหว่างตัวแบบ ARIMA และตัวแบบจากเทคนิคอัจฉริยะ ที่เป็นวิธีการด้านแมชชีนเลิร์นนิง ได้เป็นตัวแบบ Hybrid ANN และตัวแบบ Hybrid GASVR เมื่อได้ตัวแบบผสมผสาน จะนำไปเปรียบเทียบความแม่นยำในการพยากรณ์ค่าสังเกตล่วงหน้ากับตัวแบบพยากรณ์ที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ได้แก่ ตัวแบบ SLR ตัวแบบ Holt-Winters และตัวแบบ ARIMA ซึ่งผลการวิจัยที่ได้รับ จะสามารถสรุปได้ว่า ตัวแบบผสมผสานที่งานวิจัยนี้ได้นำเสนอ มีความแม่นยำมากน้อยเพียงใดสำหรับนำไปพยากรณ์อนุกรมเวลา ที่มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้าในประเทศไทย

### 3.2 เครื่องมือที่ใช้ในการวิจัย

เครื่องมือที่ใช้ในการพัฒนางานวิจัยนี้ ประกอบด้วย

1. เครื่องคอมพิวเตอร์สำหรับการพัฒนา มีรายละเอียดดังนี้
  - 1.1 หน่วยประมวลผลกลาง: Intel Core i5
  - 1.2 หน่วยความจำสำรอง: SSD 256 GB
  - 1.3 หน่วยความจำหลัก: 8 GB
2. ระบบปฏิบัติการและโปรแกรมประยุกต์สำหรับการพัฒนา ประกอบไปด้วย
  - 2.1 ระบบปฏิบัติการ: Windows 8 ระบบ 32 bits
  - 2.2 เครื่องมือที่ใช้ในการพัฒนา: Matlab2009b และ RStudio

## บทที่ 4

### การทดสอบและอภิปรายผล

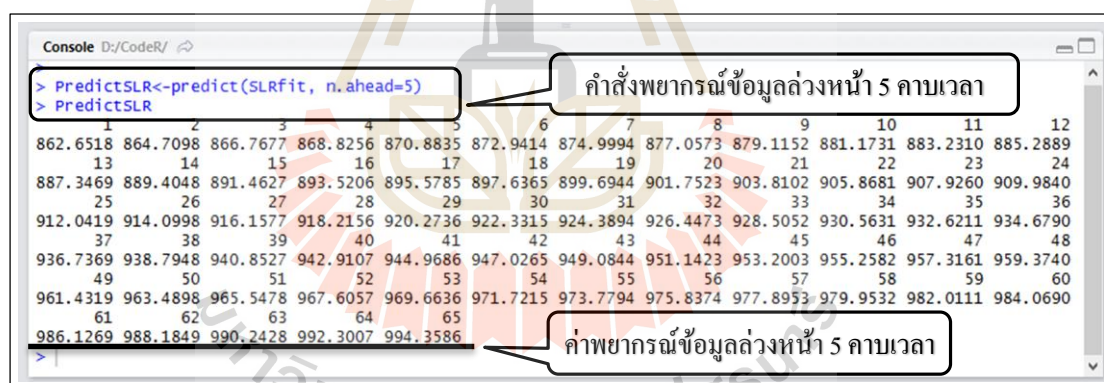
งานวิจัยนี้ได้ศึกษาและพัฒนาตัวแบบผสมผสาน เพื่อพยากรณ์หน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง และการไฟฟ้าส่วนภูมิภาค เขต 3 แห่งประเทศไทย โดยสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟ (Autoregressive) ของตัวแบบ ARIMA ซึ่งเป็นวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ร่วมกับวิธีการสร้างตัวแบบพยากรณ์ด้วยเทคนิคอัจฉริยะ ที่เป็นวิธีการด้านแมชชีนเลิร์นนิง ได้แก่ การสร้างตัวแบบ ANN (Artificial neural network) ด้วยเทคนิคเครือข่ายประสาทเทียม และการสร้างตัวแบบ GASVR (Genetic algorithm with support vector regression) ด้วยเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชันร่วมกับจินตคณิตอัลกอริทึม โดยใช้รูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA มาเป็นตัวกำหนดค่าสังเกตล่าช้าที่จะนำมาเป็นอินพุตให้กับตัวแบบ ANN และตัวแบบ GASVR จากนั้น เปรียบเทียบความแม่นยำของค่าพยากรณ์จากตัวแบบผสมผสานกับตัวแบบพยากรณ์ที่สร้างด้วยวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ที่เป็นวิธีการมาตรฐานในการสร้างตัวแบบพยากรณ์อนุกรมเวลาเชิงเดี่ยว ได้แก่ การสร้างตัวแบบ SLR (Simple linear regression) ด้วยวิธีการวิเคราะห์การถดถอยอย่างง่าย การสร้างตัวแบบ Holt-Winters ด้วยวิธีการปรับให้เรียบของโฮลท์และวินเตอร์ และการสร้างตัวแบบ ARIMA ด้วยวิธีการของบ็อกซ์และเจนกินส์ ซึ่งการเปรียบเทียบความแม่นยำของค่าพยากรณ์ จะวัดด้วยมาตรวัดความคลาดเคลื่อน SSE (Sum square error), MAE (Mean absolute Error), RMSE (Root mean square error) และ MAPE (Mean absolute percentage error) โดยได้ผลการทดสอบ ดังต่อไปนี้

#### 4.1 ผลการทดสอบของตัวแบบพยากรณ์จากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม

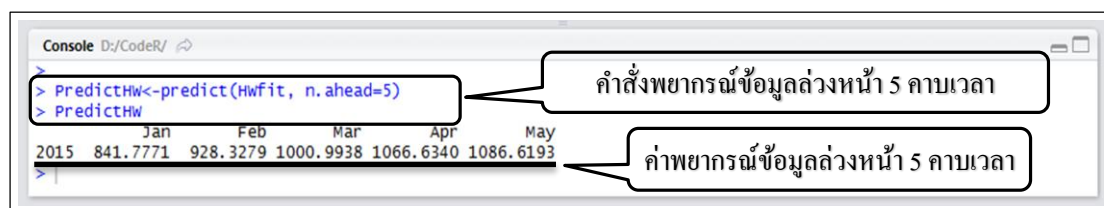
การสร้างตัวแบบพยากรณ์ด้วยวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ประกอบด้วย ตัวแบบ SLR ตัวแบบ Holt-Winters และตัวแบบ ARIMA ได้ดำเนินการวิจัย โดยเขียนโปรแกรมด้วยภาษาอาร์ และมีชุดคำสั่งสำหรับสร้างตัวแบบพยากรณ์ดังรูปที่ 3.6 ในบทที่ 3 ซึ่งข้อมูลที่ใช้ดำเนินการวิจัยมี 2 ชุด ได้แก่ ชุดที่ 1 คือ หน่วยจำหน่ายไฟฟ้ารายเดือนประเภท

บ้านพักอาศัย ของการไฟฟ้านครหลวงแห่งประเทศไทย และชุดที่ 2 คือ หน่วยจำหน่ายไฟฟ้ารายเดือนไม่รวมไฟฟ้าสาธารณะ ของการไฟฟ้าส่วนภูมิภาคแห่งประเทศไทย เขต 3 ภาค 2 โดยข้อมูลทั้ง 2 ชุด เป็นหน่วยจำหน่ายไฟฟ้ารายเดือน ตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 ซึ่งแบ่งข้อมูลตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนธันวาคม ปีพ.ศ. 2557 จำนวน 60 ค่าสังเกต สำหรับสร้างตัวแบบพยากรณ์ ส่วนข้อมูลตั้งแต่เดือนมกราคมถึงเดือนพฤษภาคม ปีพ.ศ. 2558 จำนวน 5 ค่าสังเกต แบ่งไว้เป็นข้อมูลตรวจสอบ สำหรับวัดความแม่นยำของตัวแบบพยากรณ์ โดยคำสั่งและค่าพยากรณ์หน่วยจำหน่ายไฟฟ้า เดือนมกราคมถึงเดือนพฤษภาคม ปีพ.ศ. 2558 จำนวน 5 ค่าสังเกต ของแต่ละตัวแบบ เป็นดังรูปที่ 4.1-4.3

เมื่อได้ค่าพยากรณ์ของแต่ละตัวแบบ จึงนำมาวัดความแม่นยำด้วยการวัดค่าคลาดเคลื่อน SSE, MAE, RMSE และ MAPE ซึ่งมีตัวอย่างชุดคำสั่งคำนวณค่าคลาดเคลื่อน สำหรับชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง ดังรูปที่ 4.4 โดยสามารถสรุปคำสั่งเท็จจริง ค่าพยากรณ์ ค่าคลาดเคลื่อนของแต่ละตัวแบบ สำหรับอนุกรมเวลาที่ใช้ศึกษาในงานวิจัยทั้ง 2 ชุด ได้ดังตารางที่ 4.1 และตารางที่ 4.2 ตามลำดับ



รูปที่ 4.1 คำสั่งและค่าพยากรณ์ข้อมูลของตัวแบบ SLR



รูปที่ 4.2 คำสั่งและค่าพยากรณ์ข้อมูลของตัวแบบ Holt-Winters



```

Console D:/CodeR/
> PredictARIMA1 <- predict(arimaM1, n.ahead=5)
> PredictARIMA1
$pred
      Jan      Feb      Mar      Apr      May
2015 799.2394 859.3280 969.4038 1065.1187 1117.6332
$se
      Jan      Feb      Mar      Apr      May
2015 57.44790 71.17448 77.52626 80.71932 82.37672
    
```

รูปที่ 4.3 คำสั่งและค่าพยากรณ์ข้อมูลของตัวแบบ ARIMA

```

Console D:/CodeR/
> TestSet1<-read.csv(file="d:/CodeR/Data/EtestMEA1_5_2015.csv")
> Actual1<-c(TestSet1$Electric_Units)
> ForecastSLR1<-c(PredictSLR1[61:65])
> err1SLR1 <- Actual1-ForecastSLR1
> err2SLR1 <- (Actual1-ForecastSLR1)/Actual1
> sseSLR1<-sum(err1SLR1^2)
> maeSLR1<-mean(abs(err1SLR1))
> rmseSLR1 <- sqrt(mean(err1SLR1^2))
> mapeSLR1<-mean(abs(100*err2SLR1))
> ForecastHW1<-c(PredictHW1[1:5])
> err1HW1 <- Actual1-ForecastHW1
> err2HW1 <- (Actual1-ForecastHW1)/Actual1
> sseHW1<-sum(err1HW1^2)
> maeHW1<-mean(abs(err1HW1))
> rmseHW1 <- sqrt(mean(err1HW1^2))
> mapeHW1<-mean(abs(100*err2HW1))
> ForecastARIMA1<-c(PredictARIMA1$pred[1:5])
> err1ARIMA1 <- Actual1-ForecastARIMA1
> err2ARIMA1 <- (Actual1-ForecastARIMA1)/Actual1
> sseARIMA1<-sum(err1ARIMA1^2)
> maeARIMA1<-mean(abs(err1ARIMA1))
> rmseARIMA1 <- sqrt(mean(err1ARIMA1^2))
> mapeARIMA1<-mean(abs(100*err2ARIMA1))
> sseSLR1
[1] 137892
> maeSLR1
[1] 146.7159
> rmseSLR1
[1] 166.0675
> mapeSLR1
[1] 15.76985
> sseHW1
[1] 33192.94
> maeHW1
[1] 70.40561
> rmseHW1
[1] 81.47753
> mapeHW1
[1] 7.418392
    
```

รูปที่ 4.4 ชุดคำสั่งคำนวณค่าคลาดเคลื่อนของตัวแบบ SLR, Holt-Winters และ ARIMA สำหรับชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง

ตารางที่ 4.1 ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบพยากรณ์จากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม สำหรับชุดข้อมูลที่ 1

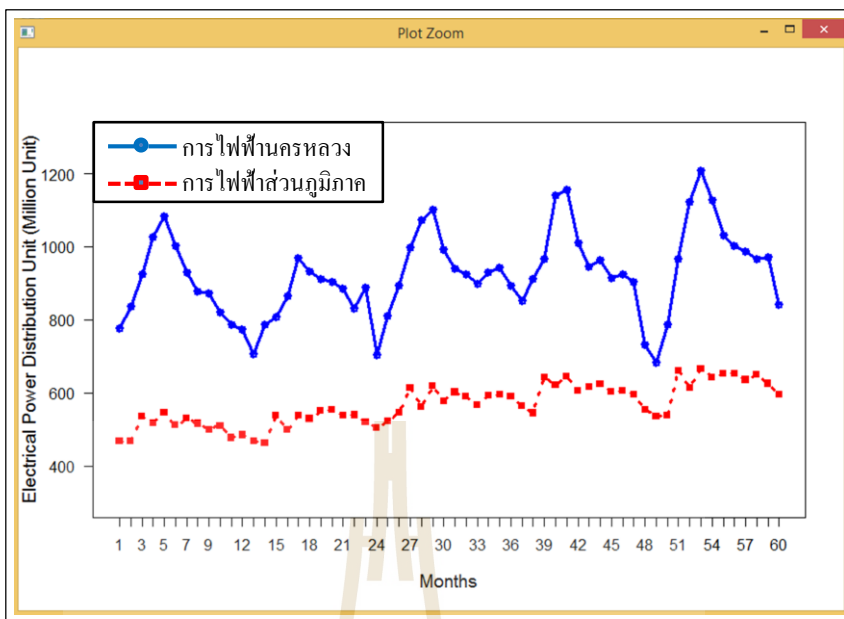
หน่วยจำหน่ายไฟฟ้า	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	SSE	MAE	RMSE	MAPE
ค่าสังเกตจริง	2558	2558	2558	2558	2558	-	-	-	-
ค่าพยากรณ์ของ ตัวแบบ SLR	986.13	988.18	990.24	992.30	994.36	137892	146.72	166.07	15.77
ค่าพยากรณ์ของ ตัวแบบ Holt-Winters	841.80	928.46	1001.88	1067.98	1087.20	33193	70.41	81.48	7.42
ค่าพยากรณ์ของ ตัวแบบ ARIMA	799.24	859.33	969.05	1064.46	1116.82	16348	48.52	57.18	4.80

ตารางที่ 4.2 ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบพยากรณ์จากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม สำหรับชุดข้อมูลที่ 2

หน่วยจำหน่ายไฟฟ้า	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	SSE	MAE	RMSE	MAPE
ค่าสังเกตจริง	2558	2558	2558	2558	2558	-	-	-	-
ค่าพยากรณ์ของ ตัวแบบ SLR	641.79	644.30	646.81	649.32	651.83	15785	47.75	56.19	7.92
ค่าพยากรณ์ของ ตัวแบบ Holt-Winters	593.08	593.48	682.67	629.45	664.80	2950	21.90	24.29	3.58
ค่าพยากรณ์ของ ตัวแบบ ARIMA	571.34	561.08	670.49	635.23	670.32	1138	13.61	15.09	2.08

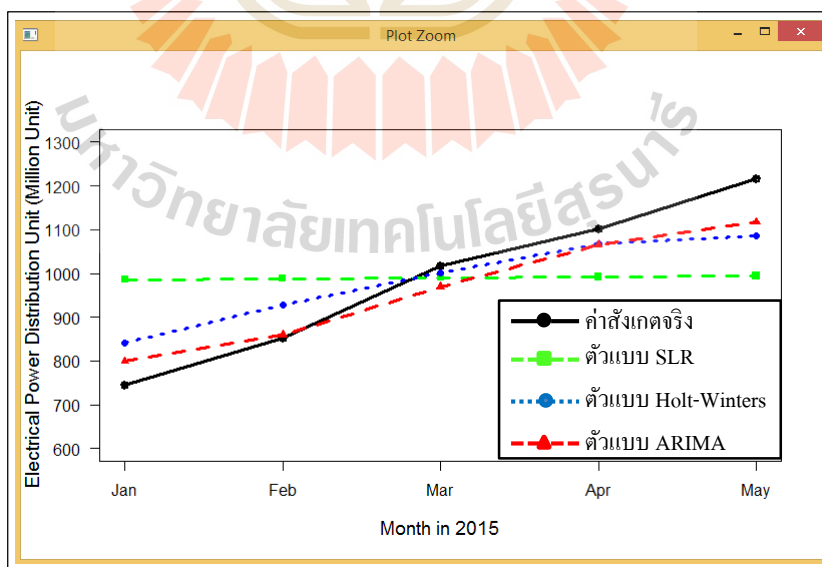
เมื่อนำข้อมูลที่ใช้สร้างตัวแบบพยากรณ์ทั้ง 2 ชุด มาสร้างกราฟเส้น เพื่อแสดงการเคลื่อนไหวของค่าสังเกตตามลำดับเวลา สามารถแสดงได้ดังรูปที่ 4.5



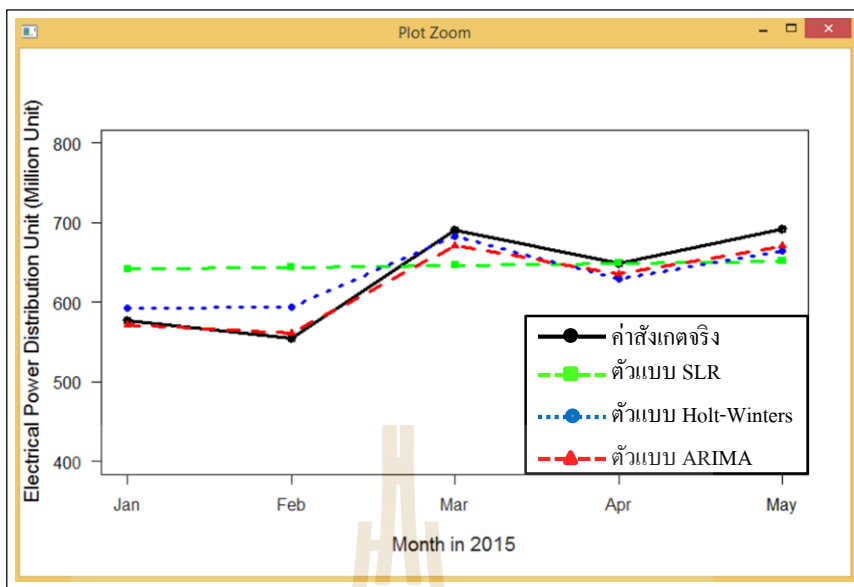


รูปที่ 4.5 กราฟค่าสังเกตที่ใช้สร้างตัวแบบพยากรณ์ของวิธีวิเคราะห์อนุกรมเวลาแบบดั้งเดิม

เมื่อนำค่าพยากรณ์ของแต่ละตัวแบบ มาสร้างกราฟเส้นเปรียบเทียบกับค่าสังเกตจริง สำหรับอนุกรมเวลาทั้ง 2 ชุด ได้ผลดังรูปที่ 4.6 และรูปที่ 4.7

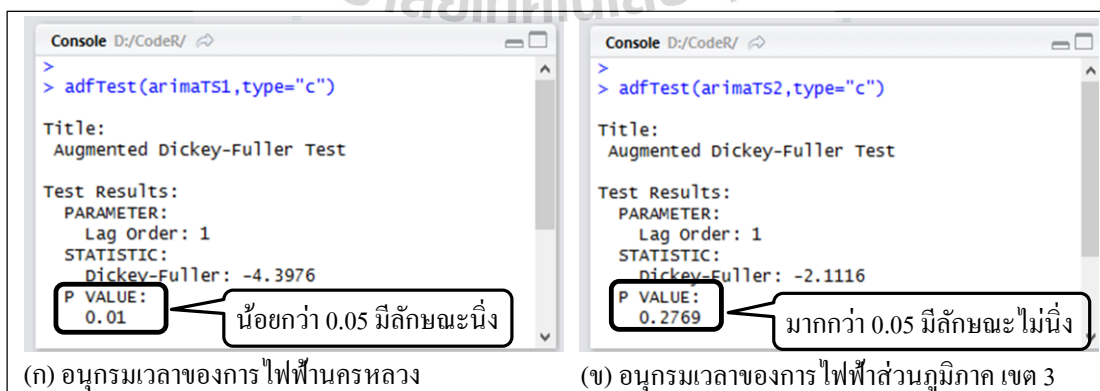


รูปที่ 4.6 กราฟเปรียบเทียบค่าพยากรณ์ของตัวแบบที่สร้างจากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม และค่าสังเกตจริงของชุดข้อมูลที่ 1



รูปที่ 4.7 กราฟเปรียบเทียบค่าพยากรณ์ของตัวแบบที่สร้างจากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม และค่าสังเกตจริงของชุดข้อมูลที่ 2

สำหรับตัวแบบ ARIMA มีผลการวิจัยเพิ่มเติมในส่วนของการทดสอบลักษณะนิ่ง (Stationary) ของอนุกรมเวลา และรูปแบบที่เหมาะสมที่สุดของตัวแบบ ARIMA โดยการทดสอบลักษณะนิ่งของอนุกรมเวลา จะทดสอบด้วยวิธี Augmented Dickey-Fuller โดยใช้ฟังก์ชันชื่อ “adfTest” ในแพ็คเกจชื่อ “fUnitRoots” ที่เผยแพร่โดย Wuertz (2013) ในส่วนเก็บข้อมูลของภาษาอาร์ (CRAN repository) ซึ่งผลการทดสอบลักษณะนิ่งของอนุกรมเวลาทั้ง 2 ชุด เป็นดังรูปที่ 4.8



รูปที่ 4.8 ผลการทดสอบลักษณะนิ่งของอนุกรมเวลาทั้ง 2 ชุด

การวิเคราะห์รูปแบบที่เหมาะสมที่สุดของตัวแบบ ARIMA คือ การค้นหาอันดับ  $p, d, q$  และอันดับฤดูกาล  $P, D, Q$  ที่ทำให้ค่า AIC (Akaike information criterion) มีค่าต่ำที่สุด ซึ่งภาษาอาร์ มีฟังก์ชันชื่อ “auto.arima” ที่ใช้สำหรับค้นหาอันดับ  $p, d, q$  และอันดับฤดูกาล  $P, D, Q$  ที่เหมาะสมที่สุด โดยผลการวิจัย พบว่า อนุกรมเวลาของการไฟฟ้านครหลวง มีอันดับ  $p, d, q = (1, 0, 0)$  และอันดับฤดูกาล  $P, D, Q = (1, 0, 0)$  ส่วนอนุกรมเวลาของการไฟฟ้าส่วนภูมิภาค มีอันดับ  $p, d, q = (1, 0, 0)$  และอันดับฤดูกาล  $P, D, Q = (1, 1, 0)$  ดังรูปที่ 4.9

```

Console D:/CodeR/
> auto.arima(arimaTS1)
Series: arimaTS1
ARIMA(1,0,0)(1,0,0)[12] with non-zero mean

Coefficients:
      ar1      sar1  intercept
      0.7314  0.6425  929.9975
s.e.    0.0833  0.0980   57.5709

sigma^2 estimated as 3300: log likelihood=-33
1.78
AIC=671.56  AICC=672.29  BIC=679.94
>

```

(ก) อนุกรมเวลาของการไฟฟ้านครหลวง

```

Console D:/CodeR/
> auto.arima(arimaTS2)
Series: arimaTS2
ARIMA(1,0,0)(1,1,0)[12] with drift

Coefficients:
      ar1      sar1  drift
      0.7427  -0.5025  2.4735
s.e.    0.0966  0.1188  0.5157

sigma^2 estimated as 256.7: log likelihood=-2
03.39
AIC=414.79  AICC=415.72  BIC=422.27
>

```

(ข) อนุกรมเวลาของการไฟฟ้าส่วนภูมิภาค

รูปที่ 4.9 ผลการวิเคราะห์รูปแบบที่เหมาะสมที่สุดของตัวแบบ ARIMA ของอนุกรมเวลาทั้ง 2 ชุด

## 4.2 ผลการทดสอบของตัวแบบผสมผสาน

ตัวแบบผสมผสาน เป็นตัวแบบพยากรณ์ที่สร้างด้วยเทคนิคอัจฉริยะ ซึ่งใช้ชุดข้อมูลฝึกสอนที่อยู่ในรูปแบบ  $D = \{(x_i, y_i)\}_{i=1}^n$  สำหรับฝึกสอนตัวแบบพยากรณ์ เมื่อ  $x_i$  คือ อินพุตเวกเตอร์ ซึ่งเป็นค่าสังเกตล่าช้า ส่วน  $y_i$  คือ เอาท์พุต ซึ่งเป็นค่าสังเกต ณ เวลา  $t$  โดยงานวิจัยนี้ ได้นำเสนอวิธีการเลือกอินพุตเวกเตอร์ จากรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA เพื่อแปลงอนุกรมเวลาเชิงเดี่ยวให้เป็นชุดข้อมูลฝึกสอน สำหรับฝึกสอนตัวแบบพยากรณ์ที่สร้างด้วยเทคนิคอัจฉริยะ ดังนั้นจึงต้องวิเคราะห์รูปแบบออโตรีเกรสซีฟ จากตัวแบบ ARIMA ของชุดข้อมูลทั้ง 2 ชุด ดังนี้

### 4.2.1 ผลการวิเคราะห์รูปแบบออโตรีเกรสซีฟจากตัวแบบ ARIMA

จากรูปที่ 4.9 แสดงผลการวิเคราะห์รูปแบบที่เหมาะสมที่สุดของตัวแบบ ARIMA สำหรับชุดข้อมูลที่อยู่ในรูปแบบของอนุกรมเวลาเชิงเดี่ยวทั้ง 2 ชุด โดยอนุกรมเวลาชุดที่ 1 มีรูปแบบ

ARIMA(1, 0, 0)x(1, 0, 0)<sub>12</sub> ส่วนอนุกรมเวลาชุดที่ 2 มีรูปแบบ ARIMA(1, 0, 0)x(1, 1, 0)<sub>12</sub> และจากตัวอย่างการวิเคราะห์รูปแบบออโตรีเกรสซีฟที่กล่าวไว้ในบทที่ 3 ตามสมการที่ 3.4 ทำให้สามารถวิเคราะห์รูปแบบออโตรีเกรสซีฟ จากตัวแบบ ARIMA ของอนุกรมเวลาทั้ง 2 ชุด ได้ดังนี้

1. การวิเคราะห์รูปแบบออโตรีเกรสซีฟของอนุกรมเวลาชุดที่ 1

สำหรับอนุกรมเวลาชุดที่ 1 ซึ่งเป็นชุดข้อมูลของการไฟฟ้านครหลวง มีรูปแบบ ARIMA(1, 0, 0)x(1, 0, 0)<sub>12</sub> หมายความว่า อันดับ  $p=1, d=0, q=0$  และอันดับฤดูกาล  $P=1, D=0, Q=0$  ดังนั้นเมื่อแทนค่าอันดับ  $p=1, P=1, d=0$  และ  $D=0$  ลงในสมการที่ 3.4 ทำให้สามารถพิสูจน์สมการ ได้ดังนี้

$$\begin{aligned} \phi(B)\phi_p(B^S)(1-B^d)(1-B^{DS})Y_t &= \theta_0 + \varepsilon_t \\ (1-\phi_1B)(1-\phi_{12}B^{12})Y_t &= \theta_0 + \varepsilon_t \\ (1-\phi_1B-\phi_{12}B^{12}+\phi_1\phi_{12}B^{13})Y_t &= \theta_0 + \varepsilon_t \\ \theta_0 + \varepsilon_t + \phi_1Y_{t-1} + \phi_{12}Y_{t-12} - \phi_1\phi_{12}Y_{t-13} &= Y_t \end{aligned} \quad \text{----- (4.1)}$$

จากการพิสูจน์สมการที่ 4.1 ทำให้ทราบว่า รูปแบบออโตรีเกรสซีฟของอนุกรมเวลาชุดที่ 1 ประกอบด้วยค่าล่าช้าที่เวลา 1, 12 และ 13 คาบเวลา ( $Y_{t-1}, Y_{t-12}, Y_{t-13}$ ) ดังนั้น เลือก  $Y_{t-1}, Y_{t-12}$  และ  $Y_{t-13}$  เป็นอินพุตเวกเตอร์ โดยมีเอาต์พุต คือ  $Y_t$  สำหรับสร้างเป็นชุดข้อมูลฝึกสอนให้กับตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ และตัวอย่างชุดข้อมูลฝึกสอนของอนุกรมเวลาชุดที่ 1 แสดงได้ดังรูปที่ 4.10

TimeID	Output ( $Y_t$ )	$Y_{t-1}$	$Y_{t-12}$	$Y_{t-13}$
14	787.29	707.16	836.16	776.32
15	806.64	787.29	925.55	836.16
16	864.58	806.64	1026.5	925.55
17	968.86	864.58	1082.18	1026.5
18	933.59	968.86	1003.41	1082.18
19	912.8	933.59	930.91	1003.41
20	902.59	912.8	876.83	930.91
21	886.08	902.59	873.47	876.83
22	830.9	886.08	820.6	873.47
23	886.93	830.9	787.33	820.6
24	703.06	886.93	773.99	787.33
25	811.41	703.06	707.16	773.99

รูปที่ 4.10 ตัวอย่างชุดข้อมูลฝึกสอนของอนุกรมเวลาชุดที่ 1

2. การวิเคราะห์รูปแบบออโตรีเกรสซีฟของอนุกรมเวลาชุดที่ 2

สำหรับอนุกรมเวลาชุดที่ 2 ซึ่งเป็นชุดข้อมูลของการไฟฟ้าส่วนภูมิภาค เขต 3 ภาค 2 มีรูปแบบ ARIMA(1, 0, 0)x(1, 1, 0)<sub>12</sub> หมายความว่า อันดับ p=1, d=0, q=0 และอันดับฤดูกาล P=1, D=1, Q=0 ดังนั้น เมื่อแทนค่าอันดับ p=1, P=1, d=0 และ D=1 ลงในสมการที่ 3.4 ทำให้สามารถพิสูจน์สมการ ได้ดังนี้

$$\begin{aligned} \phi(B)\phi_p(B^S)(1 - B^d)(1 - B^{DS})Y_t &= \theta_0 + \varepsilon_t \\ (1 - \phi_1 B)(1 - \phi_{12} B^{12})(1 - B^{12})Y_t &= \theta_0 + \varepsilon_t \\ (1 - \phi_1 B - \phi_{12} B^{12} + \phi_1 \phi_{12} B^{13})(1 - B^{12})Y_t &= \theta_0 + \varepsilon_t \\ \theta_0 + \varepsilon_t + \phi_1 Y_{t-1} + (1 + \phi_{12})Y_{t-12} - (1 + \phi_{12})\phi_1 Y_{t-13} \\ - \phi_{12} Y_{t-24} + \phi_1 \phi_{12} Y_{t-25} &= Y_t \end{aligned} \quad \text{----- (4.2)}$$

จากการพิสูจน์สมการที่ 4.2 ทำให้ทราบว่า รูปแบบออโตรีเกรสซีฟของอนุกรมเวลาชุดที่ 2 ประกอบด้วยค่าล่าหลังที่เวลา 1, 12, 13, 24 และ 25 คาบเวลา (Y<sub>t-1</sub>, Y<sub>t-12</sub>, Y<sub>t-13</sub>, Y<sub>t-24</sub>, Y<sub>t-25</sub>) ดังนั้น เลือกร Y<sub>t-1</sub>, Y<sub>t-12</sub>, Y<sub>t-13</sub>, Y<sub>t-24</sub> และ Y<sub>t-25</sub> เป็นอินพุตเวกเตอร์ โดยมีเอาต์พุต คือ Y<sub>t</sub> สำหรับสร้างเป็นชุดข้อมูลฝึกสอนให้กับตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ และตัวอย่างชุดข้อมูลฝึกสอนของอนุกรมเวลาชุดที่ 2 แสดงได้ดังรูปที่ 4.11

TimeID	Output (Y <sub>t</sub> )	Y <sub>t-1</sub>	Y <sub>t-12</sub>	Y <sub>t-13</sub>	Y <sub>t-24</sub>	Y <sub>t-25</sub>
26	545.19	521.98	462.19	469.16	468.9	468.06
27	614.1	545.19	537.54	462.19	535.74	468.9
28	562.77	614.1	499.42	537.54	518.36	535.74
29	618.52	562.77	537.55	499.42	545.53	518.36
30	578.38	618.52	529.57	537.55	512.13	545.53
31	602.21	578.38	550.87	529.57	529.98	512.13
32	589.98	602.21	553.12	550.87	516.41	529.98
33	567.81	589.98	539.2	553.12	499.85	516.41
34	592.87	567.81	539.92	539.2	510.02	499.85
35	596.46	592.87	520.39	539.92	477.13	510.02
36	589.8	596.46	503.93	520.39	485.09	477.13
37	564.15	589.8	521.98	503.93	469.16	485.09

รูปที่ 4.11 ตัวอย่างชุดข้อมูลฝึกสอนของอนุกรมเวลาชุดที่ 2

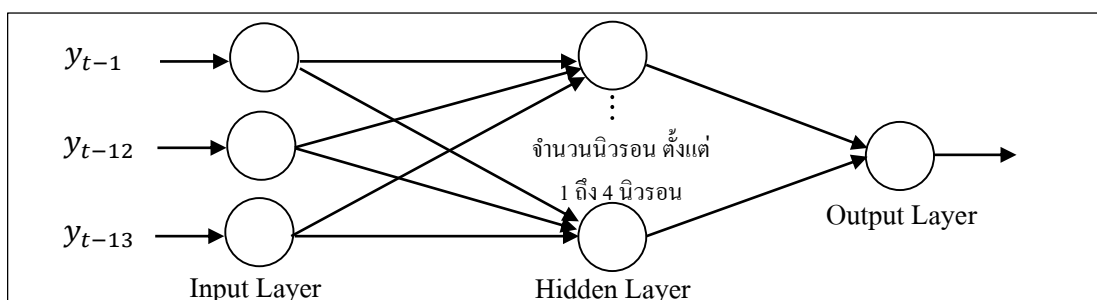
ผลการวิเคราะห์รูปแบบออโตรีเกรสซีฟของอนุกรมเวลาทั้ง 2 ชุด เพื่อแปลงอนุกรมเวลาเชิงเดี่ยวให้อยู่ในรูปแบบของชุดข้อมูลฝึกสอนสำหรับสร้างตัวแบบพยากรณ์ด้วยเทคนิคอัจฉริยะ สรุปได้ว่า ชุดข้อมูลฝึกสอนของอนุกรมเวลาชุดที่ 1 มีอินพุต 3 ตัว คือ  $Y_{t-1}$ ,  $Y_{t-12}$  และ  $Y_{t-13}$  โดยมีเอาต์พุต คือ  $Y_t$  ส่วนชุดข้อมูลฝึกสอนของอนุกรมเวลาชุดที่ 2 มีอินพุต 5 ตัว คือ  $Y_{t-1}$ ,  $Y_{t-12}$ ,  $Y_{t-13}$ ,  $Y_{t-24}$  และ  $Y_{t-25}$  โดยมีเอาต์พุต คือ  $Y_t$  จากนั้น นำชุดข้อมูลฝึกสอนทั้ง 2 ชุด ไปสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟ ร่วมกับเทคนิคอัจฉริยะ จำนวน 2 เทคนิค ได้แก่ เทคนิคเครือข่ายประสาทเทียม(ANN) และเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชันร่วมกับจีเนติกอัลกอริทึม (GASVR) ซึ่งมีผลการทดสอบ ดังต่อไปนี้

#### 4.2.2 ผลการทดสอบสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟกับเทคนิคเครือข่ายประสาทเทียม

งานวิจัยนี้ ดำเนินการทดสอบสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟ ร่วมกับเทคนิคเครือข่ายประสาทเทียม กับอนุกรมเวลาที่แปลงเป็นชุดข้อมูลฝึกสอน จำนวน 2 ชุด แต่ละชุดข้อมูล ดำเนินการสร้างตัวแบบ ANN ขึ้นมาทดสอบ จำนวน 4 ตัวแบบ โดยมีโครงสร้างของเครือข่ายประสาทเทียม ที่ใช้ดำเนินการทดสอบ ดังรูปที่ 3.9 ซึ่งแสดงไว้ในบทที่ 3 และเนื่องจากชุดข้อมูลฝึกสอนทั้ง 2 ชุด มีอินพุตแตกต่างกัน ดังนั้น จึงแบ่งการแสดงผลการทดสอบเป็น 2 ส่วน ดังนี้

1. ผลการทดสอบตัวแบบผสมผสาน ANN สำหรับชุดข้อมูลที่ 1

ชุดข้อมูลที่ 1 เป็นข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวงประเภทบ้านอยู่อาศัย เมื่อวิเคราะห์รูปแบบออโตรีเกรสซีฟ พบว่า ค่าสังเกต  $Y_t$  ขึ้นกับค่าสังเกต  $Y_{t-1}$ ,  $Y_{t-12}$  และ  $Y_{t-13}$  ดังนั้น จึงกำหนดให้  $Y_{t-1}$ ,  $Y_{t-12}$  และ  $Y_{t-13}$  เป็นอินพุตของเครือข่ายประสาทเทียม ทำให้โครงสร้างของเครือข่ายประสาทเทียมที่ใช้ดำเนินการทดสอบสำหรับชุดข้อมูลที่ 1 เป็นดังรูปที่ 4.12



รูปที่ 4.12 โครงสร้างของตัวแบบผสมผสาน ANN ที่ใช้ดำเนินการทดสอบ ของชุดข้อมูลที่ 1



จากรูปที่ 4.12 ดำเนินการสร้างตัวแบบผสมผสาน ANN จำนวน 4 ตัวแบบ แต่ละตัวแบบ มีชั้นอินพุต (Input layer) และชั้นเอาต์พุต (Output layer) ดังรูปที่ 4.12 เหมือนกัน แต่มีความแตกต่างที่จำนวนนิวรอนในชั้นซ่อนเร้น (Hidden layer) ตั้งแต่ 1 ถึง 4 นิวรอน ได้เป็นตัวแบบผสมผสาน ANN1 ถึง ANN4 และดำเนินการสร้างตัวแบบผสมผสาน ANN ด้วยการเขียนโปรแกรม Matlab ตามขั้นตอนที่ระบุไว้ในบทที่ 3 โดยแสดงตัวอย่างชุดคำสั่งที่ใช้สร้างตัวแบบผสมผสาน ANN2 ดังรูปที่ 4.13 ซึ่งชุดคำสั่ง มีการใช้ไฟล์ข้อมูล 3 ตัว ได้แก่ AllEuse.xlsx, InputTraining.xlsx และ TargetTraining.xlsx โดยตัวอย่างข้อมูลของทั้ง 3 ไฟล์ มีลักษณะดังรูปที่ 4.14

```

1 - AllData = xlsread('AllEuse.xlsx');
2 - minD = min(min(AllData));
3 - maxD = max(max(AllData));
4 - DataNor = (AllData - minD)/(maxD-minD)+0.1;
5 - data = xlsread('InputTraining.xlsx');
6 - P= (data - minD)/(maxD-minD)+0.1;
7 - P=P';
8 - Target = xlsread('TargetTraining.xlsx');
9 - T=(Target - minD)/(maxD-minD)+0.1;
10 - T = T';
11 - net=newff(minmax(P),[2 1],{'logsig','purelin'},'trainlm'); % ANN2
12 - net.trainParam.epochs=3000;
13 - net.trainParam.goal=0.0001;
14 - net.trainParam.min_grad = 1.00e-25;
15 - net.trainParam.mu_Max = 1.00e+10;
16 - net.iw{1,1}=[0.5 0.5 0.5;0.5 0.5 0.5];
17 - net.lw{2,1}=[0.5 0.5];
18 - net.b{1}=[0.5 0.5]';
19 - net.b{2}=[0.5]';
20 - net = train(net,P,T);
21 - DataTest = [DataNor(60) DataNor(49) DataNor(48);
22 -             DataNor(61) DataNor(50) DataNor(49);
23 -             DataNor(62) DataNor(51) DataNor(50);
24 -             DataNor(63) DataNor(52) DataNor(51);
25 -             DataNor(64) DataNor(53) DataNor(52)]';
26 - TarTest = [DataNor(61) DataNor(62) DataNor(63) DataNor(64) DataNor(65)];
27 - aTest = sim(net,DataTest);
28 - % Show predict value and calculate SSE, MAE, RMSE and MAPE
29 - Actual_Value = (TarTest-0.1).*(maxD-minD)+minD;
30 - Predict_Value = (aTest-0.1).*(maxD-minD)+minD;
31 - Error = Actual_Value-Predict_Value;
32 - E1 = 100*Error./Actual_Value;
33 - SSE_NN = sum(Error.*Error);
34 - MAE_NN = mean(abs(Error));
35 - RMSE_NN = sqrt(sum(Error.*Error)/5);
36 - MAPE_NN = sum(abs(E1))/5;

```

กำหนดโครงสร้างของเครือข่ายให้มี  
นิวรอนในชั้นซ่อนเร้น = 2 และมี  
นิวรอนในชั้นเอาต์พุต = 1

กำหนดค่าเริ่มต้นของ Weight  
และ Bias

อินพุตของชุดข้อมูล  
ตรวจสอบ

ค่าพยากรณ์

เอาต์พุตของชุด  
ข้อมูลตรวจสอบ

วัดค่าคลาดเคลื่อน SSE,  
MAE, RMSE, MAPE

รูปที่ 4.13 ชุดคำสั่งสร้างตัวแบบผสมผสาน ANN2 ของชุดข้อมูลที่ 1



1	Electric_Units	1	$Y_{t-1}$	$Y_{t-12}$	$Y_{t-13}$	1	$Y_t$
2	776.32	2	707.16	836.16	776.32	2	787.29
3	836.16	3	787.29	925.55	836.16	3	806.64
4	925.55	4	806.64	1026.5	925.55	4	864.58
5	1026.5	5	864.58	1082.18	1026.5	5	968.86
6	1082.18	6	968.86	1003.41	1082.18	6	933.59
7	1003.41	7	933.59	930.91	1003.41	7	912.8
8	930.91	8	912.8	876.83	930.91	8	902.59
9	876.83	9	902.59	873.47	876.83	9	886.08
10	873.47	10	886.08	820.6	873.47	10	830.9
11	820.6	11	830.9	787.33	820.6	11	886.93
12	787.33	12	886.93	773.99	787.33	12	703.06
13	773.99	13	703.06	707.16	773.99	13	811.41

(a) AllEuse                      (b) InputTraining                      (c) TargetTraining

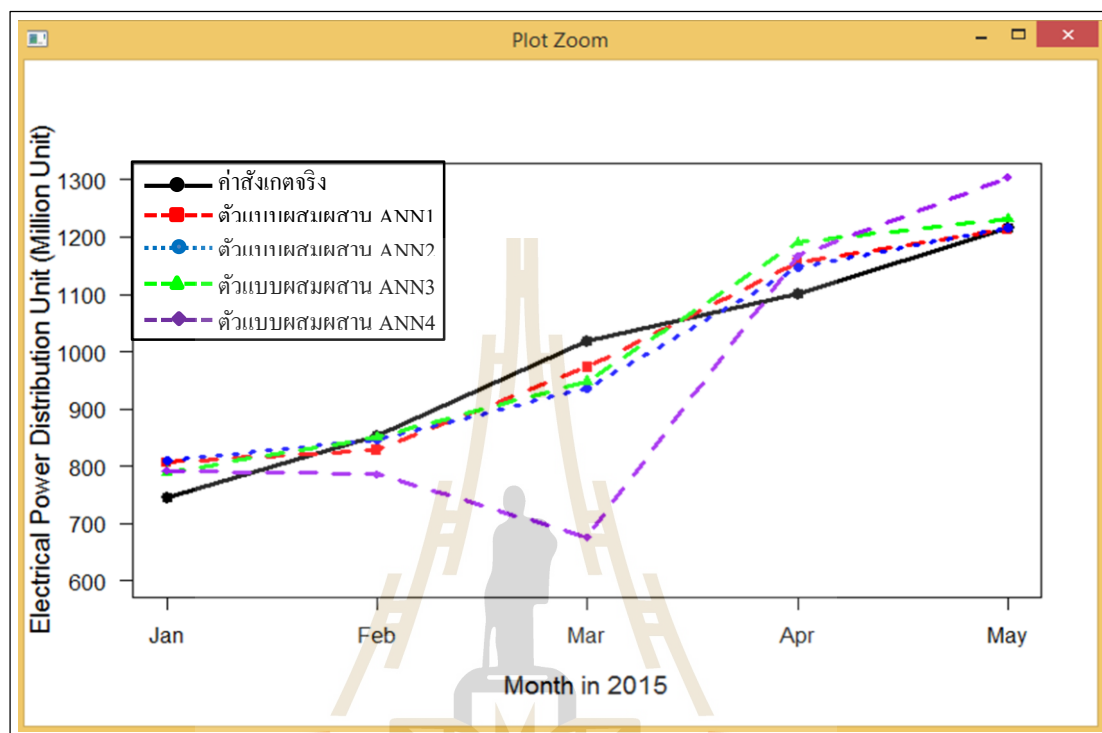
รูปที่ 4.14 ตัวอย่างข้อมูลที่ใช้ในชุดคำสั่งสร้างตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 1

เมื่อดำเนินการสร้างตัวแบบผสมผสาน ANN ทั้ง 4 ตัวแบบ ด้วยชุดคำสั่งสร้างตัวแบบผสมผสาน ANN ดังตัวอย่างที่แสดงไว้ในรูปที่ 4.13 จากนั้น ประมวลผลชุดคำสั่งของทั้ง 4 ตัวแบบ ได้ค่าพยากรณ์ 5 คาบเวลา ตั้งแต่เดือนมกราคม ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 ค่า RMSE และค่า MAPE ของแต่ละตัวแบบ เป็นดังตารางที่ 4.3

ตารางที่ 4.3 ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 1

หน่วยจำหน่ายไฟฟ้า	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	SSE	MAE	RMSE	MAPE
ค่าสังเกตจริง	2558	2558	2558	2558	2558	-	-	-	-
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN1	806.67	828.23	972.95	1155.25	1213.39	9328	37.77	43.19	4.15
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN2	809.22	845.37	936.87	1146.96	1215.41	12771	40.01	50.54	4.34
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN3	789.42	850.48	947.31	1189.77	1231.58	15004	44.19	54.78	4.49
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN4	792.05	786.36	675.80	1166.77	1303.95	135630	121.72	164.70	12.17

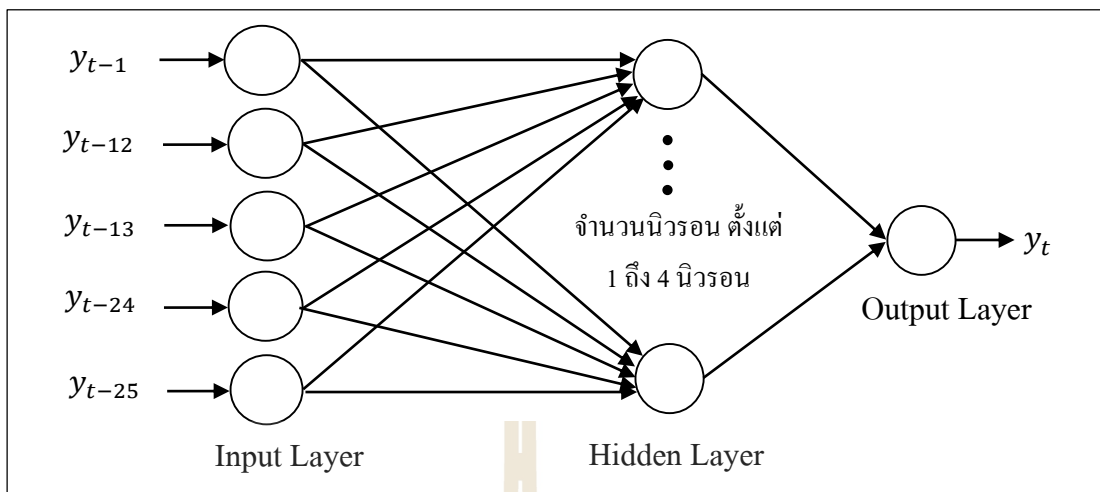
สำหรับชุดข้อมูลที่ 1 เมื่อนำค่าพยากรณ์ของตัวแบบผสมผสาน ANN ทั้ง 4 ตัวแบบ มาสร้างกราฟเส้นเปรียบเทียบกับค่าสังเกตจริง ได้ผลดังรูปที่ 4.15



รูปที่ 4.15 กราฟค่าพยากรณ์ของตัวแบบผสมผสาน ANN และค่าสังเกตจริงของชุดข้อมูลที่ 1

## 2. ผลการทดสอบตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 2

ชุดข้อมูลที่ 2 เป็นข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้าส่วนภูมิภาคเขต 3 ภาค 2 ไม่รวมไฟฟ้าสาธารณะ เมื่อวิเคราะห์รูปแบบออโตเรเกรสซีฟ พบว่า ค่าสังเกต  $Y_t$  ขึ้นกับค่าสังเกต  $Y_{t-1}$ ,  $Y_{t-12}$ ,  $Y_{t-13}$ ,  $Y_{t-24}$  และ  $Y_{t-25}$  ดังนั้น จึงกำหนดให้  $Y_{t-1}$ ,  $Y_{t-12}$ ,  $Y_{t-13}$ ,  $Y_{t-24}$  และ  $Y_{t-25}$  เป็นอินพุตของเครือข่ายประสาทเทียม ทำให้โครงสร้างของเครือข่ายประสาทเทียมที่ใช้ดำเนินการทดสอบสำหรับชุดข้อมูลที่ 2 เป็นดังรูปที่ 4.16 โดยดำเนินการสร้างตัวแบบ ANN ทั้งหมด 4 ตัวแบบ เช่นเดียวกับชุดข้อมูลที่ 1 และใช้ชุดคำสั่งสร้างตัวแบบผสมผสาน ANN คล้ายคลึงกับชุดคำสั่งในรูปที่ 4.13 แต่เปลี่ยนจำนวนนิวรอนในชั้นอินพุตเป็น 5 นิวรอน และใช้ไฟล์ข้อมูล AllEuse.xlsx, InputTraining.xlsx และ TargetTraining.xlsx ที่เป็นข้อมูลของชุดข้อมูลที่ 2 ซึ่งตัวอย่างข้อมูลของทั้ง 3 ไฟล์ มีลักษณะดังรูปที่ 4.17



รูปที่ 4.16 โครงสร้างของตัวแบบผสมผสาน ANN ที่ใช้ดำเนินการทดสอบของชุดข้อมูลที่ 2

1	Electric_Units	1	$Y_{t-1}$	$Y_{t-12}$	$Y_{t-13}$	$Y_{t-24}$	$Y_{t-25}$	1	$Y_t$
2	468.06	2	521.98	462.19	469.16	468.9	468.06	2	545.19
3	468.9	3	545.19	537.54	462.19	535.74	468.9	3	614.1
4	535.74	4	614.1	499.42	537.54	518.36	535.74	4	562.77
5	518.36	5	562.77	537.55	499.42	545.53	518.36	5	618.52
6	545.53	6	618.52	529.57	537.55	512.13	545.53	6	578.38
7	512.13	7	578.38	550.87	529.57	529.98	512.13	7	602.21
8	529.98	8	602.21	553.12	550.87	516.41	529.98	8	589.98
9	516.41	9	589.98	539.2	553.12	499.85	516.41	9	567.81
10	499.85	10	567.81	539.92	539.2	510.02	499.85	10	592.87
11	510.02	11	592.87	520.39	539.92	477.13	510.02	11	596.46
12	477.13	12	596.46	503.93	520.39	485.09	477.13	12	589.8
13	485.09	13	589.8	521.98	503.93	469.16	485.09	13	564.15

(a) AllEuse

(b) InputTraining

(c) TargetTraining

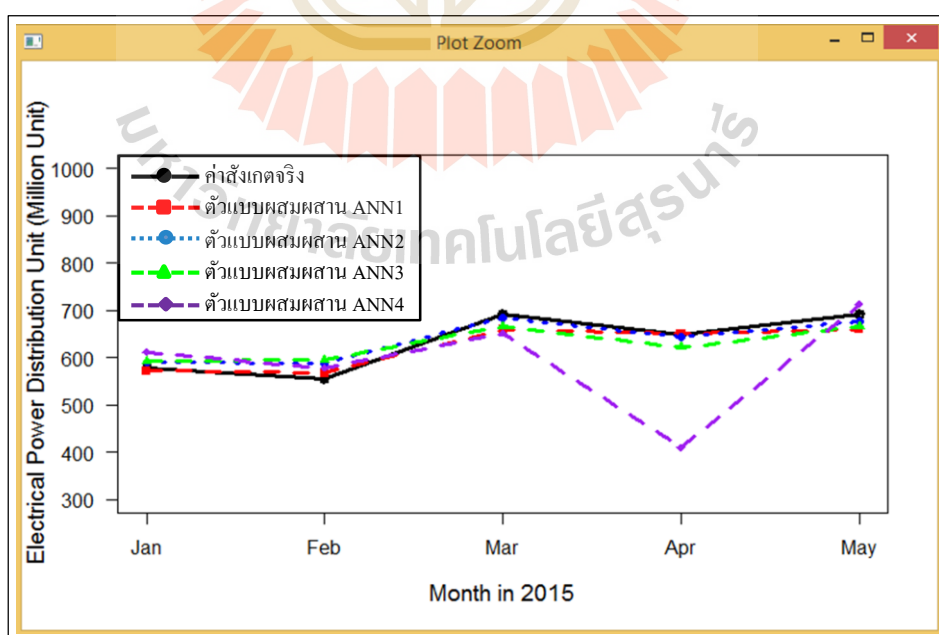
รูปที่ 4.17 ตัวอย่างข้อมูลที่ใช้ในชุดคำสั่งสร้างตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 2

เมื่อดำเนินการสร้างตัวแบบผสมผสาน ANN ทั้ง 4 ตัวแบบ โดยใช้ข้อมูลฝึกสอนชุดที่ 2 ด้วยชุดคำสั่งสร้างตัวแบบผสมผสาน ANN จากนั้น ประมวลผลชุดคำสั่งของทั้ง 4 ตัวแบบ ได้ค่าพยากรณ์ 5 คาบเวลา ตั้งแต่เดือนมกราคม ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 ค่า RMSE และค่า MAPE ของแต่ละตัวแบบ เป็นดังตารางที่ 4.4

ตารางที่ 4.4 ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบผสมผสาน ANN ของชุดข้อมูลที่ 2

หน่วยจำหน่ายไฟฟ้า	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	SSE	MAE	RMSE	MAPE
ค่าสังเกตจริง	577.23	554.35	690.87	648.95	691.64			-	-
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN1	571.62	567.08	658.42	650.11	659.33	2291	16.85	21.41	2.56
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN2	588.96	586.73	683.42	644.31	676.82	1482	14.20	17.22	2.36
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN3	591.56	594.50	665.58	621.83	665.46	3878	26.61	27.85	4.27
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN4	610.75	578.55	651.53	409.10	712.88	61235	71.63	110.67	11.18

เมื่อนำค่าพยากรณ์ของตัวแบบผสมผสาน ANN ทั้ง 4 ตัวแบบ มาสร้างกราฟเส้นเปรียบเทียบกับค่าสังเกตจริง สำหรับชุดข้อมูลที่ 2 ได้ผลดังรูปที่ 4.18



รูปที่ 4.18 กราฟค่าพยากรณ์ของตัวแบบผสมผสาน ANN และค่าสังเกตจริงของชุดข้อมูลที่ 2

### 4.2.3 ผลการทดสอบสร้างตัวแบบผสมผสานระหว่างรูปแบบบอโตรีเกรสซีฟกับเทคนิค ซัพพอร์ตเวกเตอร์รีเกรสชันร่วมกับจีเนติกอัลกอริทึม

การสร้างตัวแบบพยากรณ์ด้วยเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชัน (SVR) ใช้ชุดข้อมูลฝึกสอนเช่นเดียวกับเทคนิค ANN แต่เทคนิค SVR เป็นเทคนิคที่มีพารามิเตอร์ โดยพารามิเตอร์ของ SVR ประกอบด้วยพารามิเตอร์ C(Cost) พารามิเตอร์  $\epsilon$  (Epsilon) และพารามิเตอร์ของเคอร์เนลฟังก์ชันที่ใช้ใน SVR และจากการทดสอบสร้างตัวแบบ SVR จากชุดข้อมูลฝึกสอนทั้ง 2 ชุด โดยกำหนดค่าพารามิเตอร์แต่ละตัวเป็นค่าเริ่มต้น (Default) พบว่า เคอร์เนลฟังก์ชันที่ให้ค่า RMSE และ MAPE ต่ำที่สุด คือ ลินีเยร์เคอร์เนล (Linear kernel) ดังนั้น จึงใช้ลีนีเยร์เคอร์เนล เป็นเคอร์เนลฟังก์ชันสำหรับการสร้างตัวแบบผสมผสาน SVR โดยที่ลีนีเยร์เคอร์เนล เป็นเคอร์เนลฟังก์ชันที่ไม่มีพารามิเตอร์ ดังนั้น พารามิเตอร์ของ SVR ในงานวิจัยนี้ จึงมี 2 ตัว คือ พารามิเตอร์ C และ  $\epsilon$  โดยการกำหนดค่าพารามิเตอร์ของ SVR ที่เหมาะสมที่สุด จะดำเนินการตามขั้นตอนที่ได้กล่าวไว้ใน บทที่ 3 ด้วยจีเนติกอัลกอริทึม (Genetic algorithm) ซึ่งชุดคำสั่งของจีเนติกอัลกอริทึม ที่เขียนด้วยภาษาอาร์ สำหรับชุดข้อมูลฝึกสอนชุดที่ 1 และชุดที่ 2 เป็นดังรูปที่ 4.19 และรูปที่ 4.20 ตามลำดับ

```

TrainLoad1<-read.csv(file="d:/CodeR/Data/TrainingMEA.csv", header=TRUE, sep=",")
InV1<-data.frame(TrainLoad1$ValT1,TrainLoad1$ValT12,TrainLoad1$ValT13)
evaluate1 <- function(string=c()) {
  returnVal = NA;
  if (length(string)== 2) {
    model1 <- svm(x=InV1,y=TrainLoad1$Target, kernel="linear", cost=string[1], epsilon=string[2])
    newdata1 = data.frame(Val1=TrainLoad1$ValT1,Val2=TrainLoad1$ValT12,
                          Val3=TrainLoad1$ValT13)
    ForeSVM1 <- predict(model1,newdata=newdata1)
    error <- TrainLoad1$Target - ForeSVM1
    returnVal1=sqrt(mean(error^2))
  } else { stop("Expecting a chromosome of length 2!"); }
  returnVal1
}
rbga.results = rbga(c(1, 0.001), c(32, 0.1), evalFunc=evaluate1, verbose=TRUE, mutationChance=0.01)

```

รูปที่ 4.19 ชุดคำสั่งค้นหาพารามิเตอร์ที่เหมาะสมที่สุดด้วยจีเนติกอัลกอริทึมของข้อมูลฝึกสอนชุดที่ 1

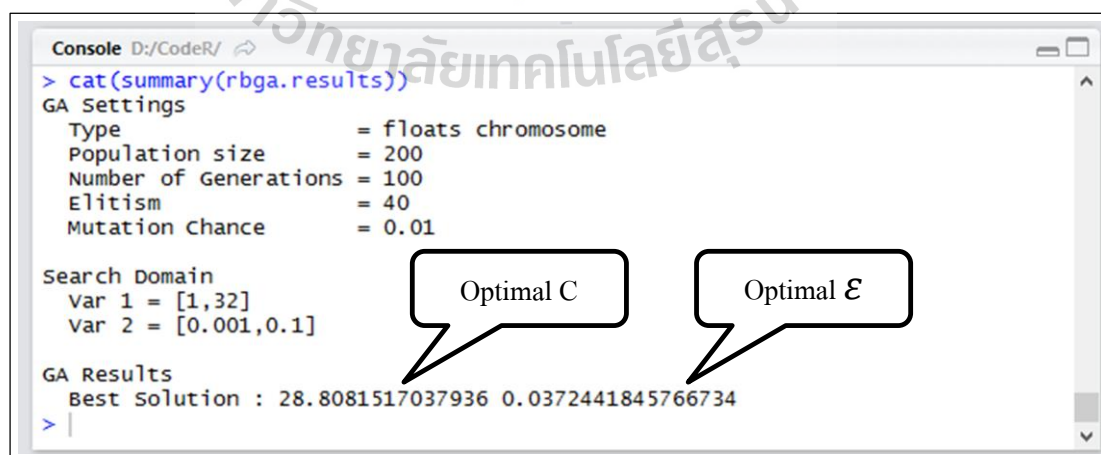
```

TrainLoad2<-read.csv(file="d:/CodeR/Data/TrainingPEA.csv", header=TRUE, sep=",")
InV2<-data.frame(TrainLoad2$ValT1,TrainLoad2$ValT12,TrainLoad2$ValT13,
                 TrainLoad2$ValT24,TrainLoad2$ValT25)
evaluate2 <- function(string=c()) {
  returnVal = NA;
  if (length(string) == 2) {
    model2 <- svm(x=InV2,y=TrainLoad2$Target, kernel="linear", cost=string[1], epsilon=string[2])
    newdata2 = data.frame(Val1= TrainLoad2$ValT1,Val2= TrainLoad2$ValT12,
                          Val3= TrainLoad2$ValT13, Val4= TrainLoad2$ValT24,Val5= TrainLoad2$ValT25)
    ForeSVM2 <- predict(model2,newdata=newdata2)
    error <- TrainLoad2$Target - ForeSVM2
    returnVal2=sqrt(mean(error^2))
  } else { stop("Expecting a chromosome of length 2!"); }
  returnVal2
}
rbga.results = rbga(c(1, 0.001), c(64, 0.1), evalFunc=evaluate2, verbose=TRUE, mutationChance=0.01)

```

รูปที่ 4.20 ชุดคำสั่งค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินเนติกอัลกอริทึมของข้อมูลฝึกสอนชุดที่ 2

เมื่อดำเนินการประมวลผลชุดคำสั่งตามรูปที่ 4.19 และ 4.20 ได้ผลการทดสอบ ดังรูปที่ 4.21 และรูปที่ 4.22



```

Console D:/CodeR/
> cat(summary(rbga.results))
GA Settings
  Type                = floats chromosome
  Population size     = 200
  Number of Generations = 100
  Elitism              = 40
  Mutation Chance     = 0.01

Search Domain
  Var 1 = [1,32]
  Var 2 = [0.001,0.1]

GA Results
  Best Solution : 28.8081517037936 0.0372441845766734
>

```

Optimal C

Optimal  $\epsilon$

รูปที่ 4.21 ผลการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินเนติกอัลกอริทึมของข้อมูลฝึกสอนชุดที่ 1



```

Console D:/CodeR/
> cat(summary(rbga.results))
GA Settings
  Type           = floats chromosome
  Population size = 200
  Number of Generations = 100
  Elitism         = 40
  Mutation Chance = 0.01

Search Domain
  Var 1 = [1,64]
  Var 2 = [0.001,0.1]

GA Results
  Best Solution : 60.4265839820728 0.0998526274086908
  >
  
```

รูปที่ 4.22 ผลการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดด้วยจินตคณิตอัลกอริทึมของข้อมูลฝึกสอนชุดที่ 2

เมื่อได้ค่าพารามิเตอร์  $C$  และ  $\epsilon$  ที่เหมาะสมที่สุดจากจินตคณิตอัลกอริทึม จึงนำไปใช้สร้างตัวแบบผสมผสาน เรียกว่า ตัวแบบผสมผสาน GASVR ด้วยชุดคำสั่งตามรูปที่ 4.23 และ 4.24

```

Trainload1<-read.csv(file="d:/CodeR/Data/TrainingMEA.csv", header=TRUE, sep=",")
InV1<-data.frame(Trainload1$ValT1,Trainload1$ValT12,Trainload1$ValT13)
model1 <- svm(x=InV1,y=Trainload1$Target,kernel="linear",
              cost=28.8081517037936,epsilon=0.0372441845766734)
TestLoad1<-read.csv(file="d:/CodeR/Data/TestMEA.csv", header=TRUE, sep=",")
newdata1 = data.frame(Val1=TestLoad1$ValT1,Val2=TestLoad1$ValT12,Val3=TestLoad1$ValT13)
ForeSVM1 <- predict(model1,newdata=newdata1)
error <- Testload1$ActualValue - ForeSVM1
err2 <- (Testload1$ActualValue - ForeSVM1)/Testload1$ActualValue
SSE_GASVR1 <- sum(error^2)
SSE_GASVR1
MAE_GASVR1 <- mean(abs(error))
MAE_GASVR1
RMSE_GASVR1 <- sqrt(mean(error^2))
RMSE_GASVR1
MAPE_GASVR1<-mean(abs(100*err2))
MAPE_GASVR1
  
```

รูปที่ 4.23 ชุดคำสั่งสร้างตัวแบบผสมผสาน GASVR ของข้อมูลฝึกสอนชุดที่ 1



```

Trainload2<-read.csv(file="d:/CodeR/Data/TrainingPEA.csv", header=TRUE, sep=",")
InV2<-data.frame(Trainload2$ValT1,Trainload2$ValT12,Trainload2$ValT13,
                 Trainload2$ValT24,Trainload2$ValT25)
model2 <- svm(x=InV2,y=Trainload2$Target,kernel="linear",
             cost=60.4265839820728,epsilon=0.0998526274086908)
Testload2<-read.csv(file="d:/CodeR/Data/TestPEA.csv", header=TRUE, sep=",")
newdata2 = data.frame(Val1=Testload2$ValT1,Val2=Testload2$ValT12,Val3=Testload2$ValT13,
                    Val4=Testload2$ValT24,Val5=Testload2$ValT25)
ForeSVM2 <- predict(model2,newdata=newdata2)
error <- Testload2$ActualValue - ForeSVM2
err2 <- (Testload2$ActualValue - ForeSVM2)/Testload2$ActualValue
SSE_GASVR2 <- sum(error^2)
SSE_GASVR2
MAE_GASVR2 <- mean(abs(error))
MAE_GASVR2
RMSE_GASVR2 <- sqrt(mean(error^2))
RMSE_GASVR2
MAPE_GASVR2<-mean(abs(100*err2))
MAPE_GASVR2

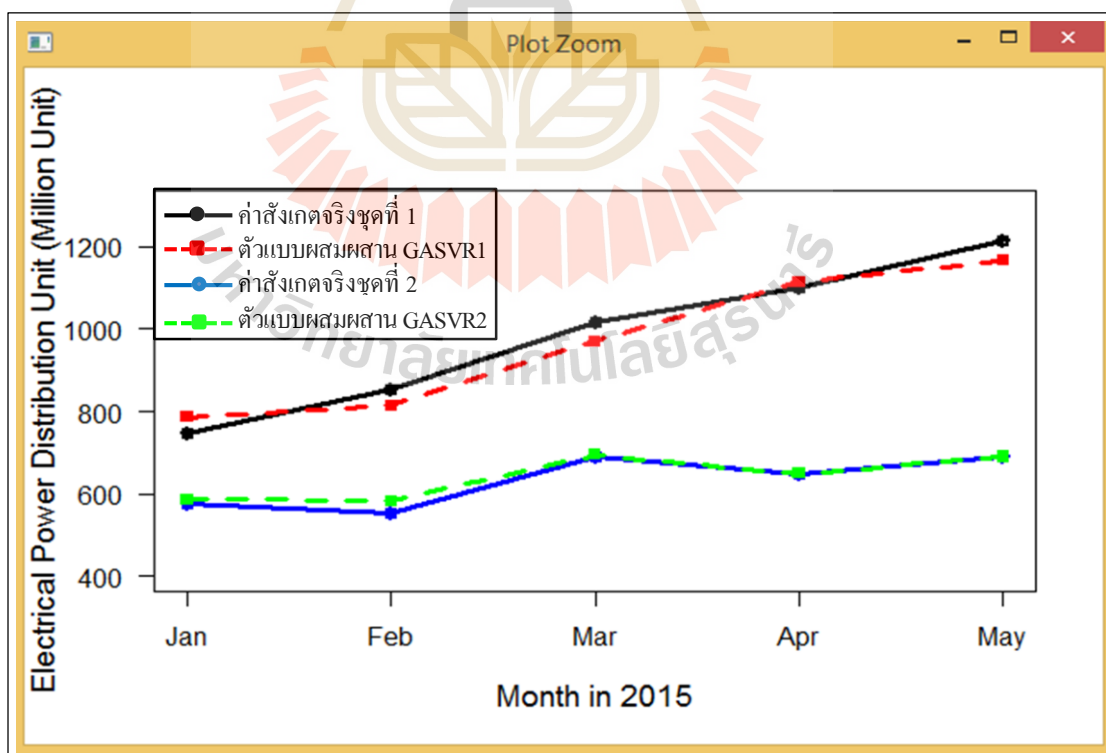
```

รูปที่ 4.24 ชุดคำสั่งสร้างตัวแบบผสมผสาน GASVR ของข้อมูลฝึกสอนชุดที่ 2

ผลการทดสอบสร้างตัวแบบผสมผสาน GASVR ได้ค่าพยากรณ์เปรียบเทียบกับค่าจริง ค่า RMSE และค่า MAPE ของชุดข้อมูลทั้ง 2 ชุด ดังตารางที่ 4.5 เมื่อนำค่าพยากรณ์ของข้อมูลทั้ง 2 ชุด ไปสร้างกราฟเส้นเปรียบเทียบกับค่าสังเกตจริง แสดงได้ดังรูปที่ 4.25

ตารางที่ 4.5 ค่าพยากรณ์และค่าคลาดเคลื่อนของตัวแบบผสมผสาน GASVR ของชุดข้อมูลทั้ง 2 ชุด

หน่วยจำหน่ายไฟฟ้า	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	SSE	MAE	RMSE	MAPE
ค่าสังเกตจริงของชุดข้อมูลการไฟฟ้านครหลวง (ข้อมูลชุดที่ 1)	745.82	853.60	1017.89	1100.93	1216.77	-	-	-	-
ค่าพยากรณ์ของตัวแบบผสมผสาน GASVR1	786.15	814.86	969.63	1114.64	1166.55	8166	38.25	40.41	4.01
ค่าสังเกตจริงของชุดข้อมูลการไฟฟ้าส่วนภูมิภาค (ข้อมูลชุดที่ 2)	577.23	554.35	690.87	648.95	691.64	-	-	-	-
ค่าพยากรณ์ของตัวแบบผสมผสาน GASVR2	586.58	581.82	694.75	649.63	689.98	860	8.61	13.12	1.50



รูปที่ 4.25 กราฟค่าพยากรณ์ของตัวแบบผสมผสาน GASVR และค่าสังเกตจริงของข้อมูลทั้ง 2 ชุด

#### 4.2.4 ผลการทดสอบวัดความทนทานของตัวแบบผสมผสาน

เพื่อทดสอบความทนทานในการพยากรณ์ของตัวแบบผสมผสาน ANN และตัวแบบผสมผสาน GASVR เปรียบเทียบกับตัวแบบ ARIMA จึงดำเนินการทดสอบด้วยการเปลี่ยนแปลงสัดส่วนของจำนวนค่าสังเกตในชุดข้อมูลฝึกสอนและจำนวนค่าสังเกตในชุดข้อมูลตรวจสอบ โดยแบ่งสัดส่วนใหม่ ด้วยการลดจำนวนค่าสังเกตของชุดข้อมูลฝึกสอน แล้วนำไปเพิ่มให้กับชุดข้อมูลตรวจสอบคราวละ 6 ค่าสังเกต จนกว่าจะไม่สามารถลดจำนวนค่าสังเกตของชุดข้อมูลฝึกสอนได้ จากนั้น เปรียบเทียบความทนทานของค่าพยากรณ์ด้วยการวัดค่าคลาดเคลื่อน RMSE และ MAPE ซึ่งดำเนินการทดสอบกับชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง และชุดข้อมูลที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 ด้วยขั้นตอนเดียวกัน ผลการทดสอบ สรุปได้ดังตารางที่ 4.6 และ 4.7



ตารางที่ 4.6 ผลการทดสอบความทนทานของค่าพยากรณ์ สำหรับชุดข้อมูลที่ 1

กลุ่ม	ชุดข้อมูลฝึกสอน		ชุดข้อมูลตรวจสอบ		รูปแบบ ARIMA	อินพุตจากรูปแบบออโตรีเกรสซีฟ	ARIMA		Hybrid ANN1		Hybrid GASVR		MAPE of Hybrid ANN1 / ARIMA	MAPE of Hybrid GASVR / ARIMA
	จำนวน	%	จำนวน	%			RMSE	MAPE	RMSE	MAPE	RMSE	MAPE		
1	60	92%	5	8%	ARIMA(1,0,0)x(1,0,0)	t-1, t-12, t-13	57.18	4.80	43.19	4.15	40.41	4.01	-13.54%	-16.41%
2	54	83%	11	17%	ARIMA(3,0,0)x(1,0,0)	t-1, t-2, t-3, t-12, t-13, t-14, t-15	72.10	6.02	48.47	3.75	35.13	3.15	-37.71%	-47.72%
3	48	74%	17	26%	ARIMA(1,0,0)x(1,1,0)	t-1, t-12, t-13, t-24, t-25	75.87	7.02	76.12	6.29	72.40	6.00	-10.40%	-14.50%
4	42	65%	23	35%	ARIMA(1,0,0)x(1,0,0)	t-1, t-12, t-13	111.46	9.83	59.62	4.99	64.12	5.48	-49.24%	-44.24%
5	36	55%	29	45%	ARIMA(1,0,0)x(1,0,0)	t-1, t-12, t-13	131.13	10.80	78.61	5.95	73.11	5.76	-44.91	-46.70%
6	30	46%	35	54%	ARIMA(0,1,0)x(0,1,0)	t-1, t-12, t-13	104.13	9.02	104.53	10.07	64.53	4.98	+11.64%	-44.80%
7	24	37%	41	63%	ARIMA(1,0,0)	t-1	176.35	14.74	89.08	7.15	89.65	7.31	-51.49%	-50.45%
8	18	28%	47	72%	ARIMA(2,0,0)	t-1, t-2	146.43	11.73	86.79	6.97	75.57	6.37	-40.58%	-45.70%
9	12	18%	53	82%	ARIMA(2,0,0)	t-1, t-2	130.78	10.41	78.00	6.67	75.43	6.60	-35.93%	-36.54%
10	6	9%	59	91%	ไม่มีรูปแบบ ARIMA	ไม่มีอินพุต	ไม่มีตัวแบบ		ไม่มีตัวแบบ		ไม่มีตัวแบบ		-	-

ตารางที่ 4.7 ผลการทดสอบความทนทานของค่าพยากรณ์ สำหรับชุดข้อมูลที่ 2

กลุ่ม	ชุดข้อมูลฝึกสอน		ชุดข้อมูลตรวจสอบ		รูปแบบ ARIMA	อินพุตจากรูปแบบ ออโตรีเกรสซีฟ	ARIMA		Hybrid ANN2		Hybrid GASVR		MAPE of Hybrid ANN2 / ARIMA	MAPE of Hybrid GASVR / ARIMA
	จำนวน	%	จำนวน	%			RMSE	MAPE	RMSE	MAPE	RMSE	MAPE		
1	60	92%	5	8%	ARIMA(1,0,0)x(1,1,0)	t-1, t-12, t-13, t-24, t-25	15.09	2.08	17.22	2.36	13.12	1.50	13.46%	-27.93%
2	54	83%	11	17%	ARIMA(1,0,0)x(1,1,0)	t-1, t-12, t-13, t-24, t-25	10.02	1.44	19.92	2.73	14.14	1.84	89.58%	28.15%
3	48	74%	17	26%	ARIMA(1,0,0)x(1,1,0)	t-1, t-12, t-13, t-24, t-25	28.24	3.99	35.49	4.97	26.20	3.30	24.56%	-17.18%
4	42	65%	23	35%	ARIMA(0,1,0)x(1,1,0)	t-1, t-12, t-13, t-24, t-25	34.01	4.53	35.58	5.15	24.79	2.93	13.69%	-35.32%
5	36	55%	29	45%	ARIMA(0,1,0)x(0,1,0)	t-1, t-12, t-13	127.22	18.90	45.54	5.75	20.32	2.67	-69.58%	-85.87%
6	30	46%	35	54%	ARIMA(0,1,0)x(0,1,0)	t-1, t-12, t-13	51.60	7.11	72.63	9.70	23.84	3.31	36.43%	-53.40%
7	24	37%	41	63%	ARIMA(1,0,0)	t-1	103.75	15.26	4,275	524.25	63.72	8.25	3,335%	-45.92%
8	18	28%	47	72%	ARIMA(0,0,0)	ไม่มีอินพุต	ค่าพยากรณ์ เป็นค่าคงที่		ไม่มีการสร้าง ตัวแบบ		ไม่มีการสร้าง ตัวแบบ		-	-

### 4.3 อภิปรายผล

จากผลการทดสอบการพัฒนาตัวแบบพยากรณ์ เพื่อพยากรณ์หน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง และการไฟฟ้าส่วนภูมิภาค เขต 3 แห่งประเทศไทย โดยสร้างตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ร่วมกับเทคนิคอัจฉริยะ ได้แก่ การสร้างตัวแบบผสมผสาน ANN (Artificial neural network) และการสร้างตัวแบบผสมผสาน GASVR (Genetic algorithm with support vector regression) เปรียบเทียบกับตัวแบบพยากรณ์ที่สร้างด้วยวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม สามารถสรุปผลการทดสอบเปรียบเทียบในด้านต่าง ๆ ได้ดังต่อไปนี้

#### 4.3.1 คุณลักษณะของข้อมูล

ข้อมูลที่น่าสนใจในงานวิจัยนี้ ประกอบด้วยข้อมูล 2 ชุด ได้แก่ ชุดที่ 1 คือ ข้อมูลหน่วยจำหน่ายไฟฟ้ารายเดือนประเภทบ้านอยู่อาศัย ของการไฟฟ้านครหลวงแห่งประเทศไทย และชุดที่ 2 คือ ข้อมูลหน่วยจำหน่ายไฟฟ้ารายเดือนรวมทุกประเภท ยกเว้นไฟฟ้าสาธารณะของการไฟฟ้าส่วนภูมิภาคแห่งประเทศไทย เขต 3 โดยข้อมูลทั้ง 2 ชุด มีลักษณะเป็นอนุกรมเวลาเชิงเดี่ยว ที่มีคาบเวลารายเดือน มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้า ซึ่ง 1 หน่วยจำหน่ายไฟฟ้า หมายถึง ปริมาณการใช้ไฟฟ้า 1 กิโลวัตต์ต่อชั่วโมง (Kw/h) จากการนำข้อมูลทั้ง 2 ชุด มาสร้างกราฟเส้นตามลำดับเวลา ดังรูปที่ 4.5 พบว่า ข้อมูลทั้ง 2 ชุด มีการเคลื่อนไหวคล้ายคลึงกัน คือ มีรูปแบบการเคลื่อนไหวซ้ำเดิมตามรอบฤดูกาล และเนื่องจากข้อมูลทั้ง 2 ชุด มีคาบเวลาเป็นรายเดือน จึงทำให้ข้อมูลทั้ง 2 ชุด มี 1 รอบฤดูกาลเท่ากับ 12 เดือน ตามรอบปีปฏิทินโดยธรรมชาติ เมื่อนำข้อมูลทั้ง 2 ชุด มาทดสอบลักษณะนิ่ง (Stationary) พบว่า ข้อมูลชุดที่ 1 ของการไฟฟ้านครหลวง เป็นอนุกรมเวลาที่มีลักษณะนิ่ง (Stationary) หมายความว่า มีค่าเฉลี่ยและความแปรปรวนคงที่ตลอดช่วงเวลาที่สนใจ สามารถนำไปใช้สร้างตัวแบบพยากรณ์ด้วยวิธีการแบบดั้งเดิมได้ โดยไม่ต้องหาผลต่างของค่าสังเกต เพื่อสร้างอนุกรมเวลาใหม่ ก่อนนำไปสร้างตัวแบบพยากรณ์ ส่วนข้อมูลชุดที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 เป็นอนุกรมเวลาที่มีลักษณะไม่นิ่ง (Non-stationary) หมายความว่า มีค่าเฉลี่ยและความแปรปรวนไม่คงที่ตลอดช่วงเวลาที่สนใจ จะต้องนำมาหาผลต่างของค่าสังเกต เพื่อสร้างอนุกรมเวลาใหม่ ก่อนนำไปสร้างตัวแบบพยากรณ์ ซึ่งผลการทดสอบลักษณะนิ่ง สอดคล้องกับผลการวิเคราะห์รูปแบบที่เหมาะสมที่สุดของตัวแบบ ARIMA ของอนุกรมเวลาทั้ง 2 ชุด ดังแสดงตามรูปที่ 4.9

#### 4.3.2 อภิปรายผลการพยากรณ์ของตัวแบบที่สร้างจากวิธีวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม

วิธีวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม เป็นวิธีค้นหารูปแบบการเคลื่อนไหวของหอนุกรมเวลาเชิงเดี่ยว ซึ่งเป็นค่าสังเกตที่อยู่ในรูปแบบของตัวเลข 1 ค่า ในแต่ละคาบเวลา ภายใต้สมมุติฐานว่า รูปแบบการเคลื่อนไหวของหอนุกรมเวลาในอดีต คล้ายคลึงกับรูปแบบการเคลื่อนไหวของหอนุกรมเวลาในปัจจุบันและอนาคต ซึ่งตัวแบบที่สร้างจากวิธีวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม คือ สมการคณิตศาสตร์ที่เป็นตัวแทนของรูปแบบการเคลื่อนไหวของหอนุกรมเวลา ที่มีค่าสังเกตเปลี่ยนแปลงไปตามลำดับเวลา และสามารถใช้ตัวแบบพยากรณ์ค่าสังเกตในอนาคตได้ โดยความแม่นยำของค่าพยากรณ์ ขึ้นกับวิธีการวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแต่ละวิธีว่าจะสามารถตรวจจับการเคลื่อนไหวของหอนุกรมเวลาได้มากน้อยเพียงใด สำหรับงานวิจัยนี้ ได้ศึกษาและพัฒนาตัวแบบพยากรณ์ที่สร้างจากวิธีวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม จำนวน 3 ตัวแบบ ได้แก่ ตัวแบบ SLR (Simple linear regression) ตัวแบบ Holt-Winters และตัวแบบ ARIMA จากนั้นประเมินความแม่นยำในการพยากรณ์ของแต่ละตัวแบบ โดยวัดความคลาดเคลื่อนด้วยมาตรวัดความคลาดเคลื่อน RMSE และ MAPE ซึ่งเป็นมาตรวัดความคลาดเคลื่อนที่นิยมใช้ในงานวิจัยในอดีต (Wang et al., 2012; Kaytez et al., 2015; Szoplik, 2015; Hussain et al., 2016)

ผลการทดสอบ พบว่า ตัวแบบ ARIMA มีความแม่นยำในการพยากรณ์มากที่สุดสำหรับหอนุกรมเวลาที่ใช้ศึกษาในงานวิจัยนี้ ทั้ง 2 ชุด โดยหอนุกรมเวลาชุดที่ 1 ของการไฟฟ้านครหลวง ตัวแบบ ARIMA มีค่า SSE, MAE, RMSE และ MAPE ต่ำที่สุด เท่ากับ 16,348, 48.52, 57.18 และ 4.80 ตามลำดับ ส่วนหอนุกรมเวลาชุดที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 ตัวแบบ ARIMA มีค่า SSE, MAE, RMSE และ MAPE ต่ำที่สุด เท่ากับ 1,138, 13.61, 15.09 และ 2.08 ตามลำดับ เมื่อพิจารณากราฟเปรียบเทียบค่าพยากรณ์ของตัวแบบ ARIMA กับค่าสังเกตจริงในคาบเวลาเดียวกันของหอนุกรมเวลาทั้ง 2 ชุด ดังรูปที่ 4.6 และรูปที่ 4.7 พบว่า กราฟค่าพยากรณ์ของตัวแบบ ARIMA มีลักษณะคล้ายคลึงกับกราฟค่าสังเกตจริงมากที่สุด หมายความว่า ตัวแบบ ARIMA สามารถตรวจจับรูปแบบความเคลื่อนไหวของหอนุกรมเวลาทั้ง 2 ชุด ได้ดีที่สุดในนั้น ตัวแบบ ARIMA เป็นตัวแบบที่เหมาะสมที่สุด สำหรับตัวแบบพยากรณ์ที่สร้างจากวิธีการวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม นอกจากนี้ เมื่อดูกราฟค่าพยากรณ์ของตัวแบบ SLR เปรียบเทียบกับกราฟของค่าสังเกตจริง ตามรูปที่ 4.6 และรูปที่ 4.7 พบว่า กราฟค่าพยากรณ์ของตัวแบบ SLR ไม่สามารถตรวจจับความเคลื่อนไหวเนื่องจากแนวโน้มและฤดูกาล ซึ่งเป็นส่วนประกอบของหอนุกรมเวลาได้ ดังนั้น สรุปได้ว่า ตัวแบบ SLR ที่มีเวลา ( $t$ ) เป็นตัวแปรอิสระ



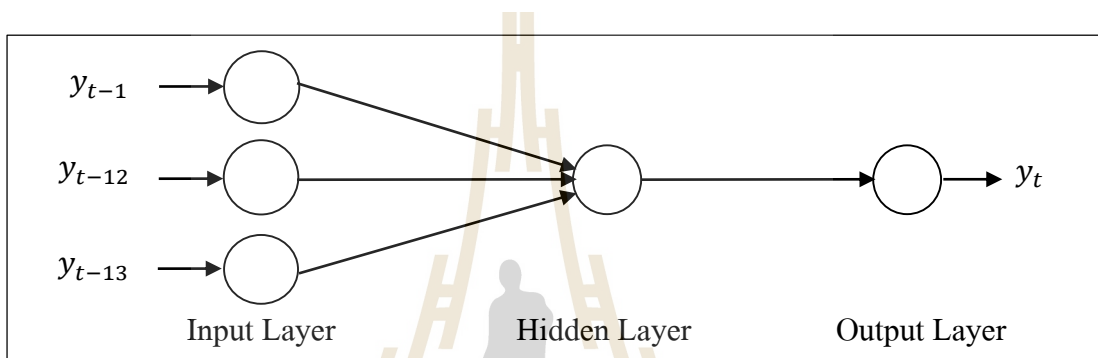
และมีตัวแปรตามเป็นค่าสังเกตของอนุกรมเวลา ไม่เหมาะสมสำหรับการนำมาพยากรณ์อนุกรมเวลา ทั้ง 2 ชุด ซึ่งผลการทดสอบในส่วนนี้ สอดคล้องกับงานวิจัยในอดีต ที่กล่าวว่า ตัวแบบ ARIMA เป็นตัวแบบที่ได้รับความนิยม เนื่องจากให้ค่าพยากรณ์ที่มีความคลาดเคลื่อนต่ำ หรือมีความแม่นยำสูง (Wang and Meng, 2012; Wang et al., 2012; Chujai et al., 2013; Yuan et al., 2016)

#### 4.3.3 อภิปรายผลการพยากรณ์ของตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟกับเทคนิคเครือข่ายประสาทเทียม

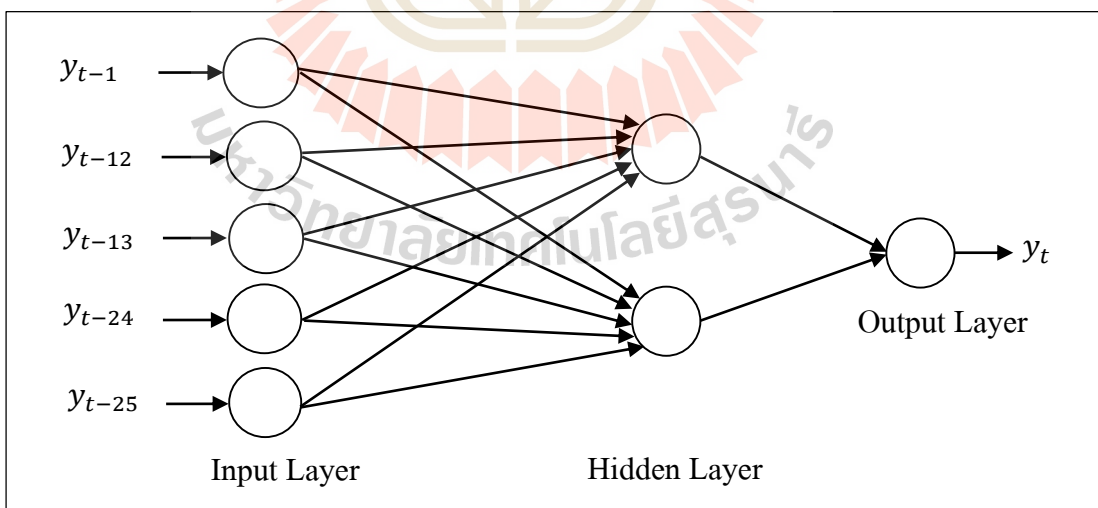
วิธีดำเนินการทดสอบสร้างตัวแบบผสมผสาน ANN กับอนุกรมเวลาทั้ง 2 ชุด ได้ใช้เครือข่ายประสาทเทียมแบบแพร่กลับ (Back propagation neural network) โดยกำหนดโครงสร้างของเครือข่ายประสาทเทียมแบบ 3 ชั้น ประกอบด้วยชั้นอินพุต ชั้นซ่อนเร้น และชั้นเอาต์พุต อย่างละ 1 ชั้น และได้สร้างตัวแบบผสมผสาน ANN ขึ้นมาทดสอบ จำนวน 4 ตัวแบบ แต่ละตัวแบบ มีความแตกต่างที่จำนวนนิวรอนในชั้นซ่อนเร้น ตั้งแต่ 1 ถึง 4 นิวรอน นั่นคือ ตัวแบบผสมผสาน ANN1 เป็นตัวแบบที่มีนิวรอนในชั้นซ่อนเร้น จำนวน 1 นิวรอน ไล่ไปตามลำดับถึงตัวแบบผสมผสาน ANN4 ที่มีนิวรอนในชั้นซ่อนเร้น จำนวน 4 นิวรอน เหตุที่เลือกกำหนดโครงสร้างของเครือข่ายประสาทเทียมดังกล่าว เพื่อทดสอบ เนื่องจาก เครือข่ายประสาทเทียมเป็นเทคนิคที่มีปัญหาโอเวอร์ฟิตติง (Over-fitting) (Hagiwara and Fukumizu, 2008) ซึ่งเป็นปัญหาที่ทำให้เกิดค่าความคลาดเคลื่อนต่ำเมื่อทดสอบกับชุดข้อมูลฝึกสอน แต่มีค่าความคลาดเคลื่อนสูงเมื่อทดสอบกับชุดข้อมูลทดสอบ เมื่อกำหนดให้โครงสร้างของเครือข่ายมีความซับซ้อนมาก จะยิ่งทำให้ปัญหาโอเวอร์ฟิตติง เพิ่มมากขึ้น ดังนั้น จึงได้กำหนดโครงสร้างของเครือข่าย ให้มีชั้นซ่อนเร้นเพียง 1 ชั้น และกำหนดนิวรอนในชั้นซ่อนเร้น ให้มีจำนวนน้อย ตั้งแต่ 1 ถึง 4 นิวรอน เพื่อทดสอบความแม่นยำของตัวแบบพยากรณ์ที่สร้างจากเทคนิคผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟกับเทคนิคเครือข่ายประสาทเทียม

ผลการทดสอบ พบว่า ชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง ตัวแบบผสมผสาน ANN1 เป็นตัวแบบที่ให้ค่า SSE, MAE, RMSE และ MAPE ต่ำที่สุด เท่ากับ 9,328, 37.77, 43.19 และ 4.15 ตามลำดับ ดังนั้น ตัวแบบผสมผสาน ANN1 เป็นตัวแบบพยากรณ์ที่เหมาะสมที่สุด สำหรับชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง โดยมีโครงสร้างของตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด ดังรูปที่ 4.26 ส่วนชุดข้อมูลที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 ตัวแบบผสมผสาน ANN2 เป็นตัวแบบที่ให้ค่า SSE, MAE, RMSE และ MAPE ต่ำที่สุด เท่ากับ 1,482, 14.20, 17.22 และ 2.36 ตามลำดับ ดังนั้น ตัวแบบผสมผสาน ANN2 เป็นตัวแบบพยากรณ์ที่เหมาะสมที่สุด สำหรับชุดข้อมูลที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 โดยมีโครงสร้างของตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด

ผังรูปที่ 4.27 นอกจากนี้ เมื่อพิจารณาค่าความคลาดเคลื่อน RMSE และ MAPE จากตารางที่ 4.3-4.4 ของข้อมูลทั้ง 2 ชุด พบว่า ตัวแบบผสมผสาน ANN4 ซึ่งเป็นตัวแบบที่มีจำนวนนิวรอนในชั้นซ่อนเร้นมากที่สุด ให้ค่าคลาดเคลื่อน RMSE และ MAPE สูงที่สุด ทั้ง 2 ชุดข้อมูล ทำให้สรุปได้ว่า ถ้าเพิ่มจำนวนนิวรอนในชั้นซ่อนเร้นให้มากขึ้น จะทำให้โครงสร้างของเครือข่ายประสาทเทียมมีความซับซ้อนมากขึ้น และทำให้ปัญหาโอเวอร์ฟิตติงเพิ่มมากขึ้น หรือกล่าวได้ว่า การเพิ่มจำนวนนิวรอนในชั้นซ่อนเร้นให้มากขึ้นจะยิ่งทำให้ค่าคลาดเคลื่อนมากขึ้น หรือมีความแม่นยำของค่าพยากรณ์ลดน้อยลง



รูปที่ 4.26 โครงสร้างของตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด สำหรับชุดข้อมูลที่ 1



รูปที่ 4.27 โครงสร้างของตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด สำหรับชุดข้อมูลที่ 2

#### 4.3.4 อภิปรายเปรียบเทียบผลการพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR

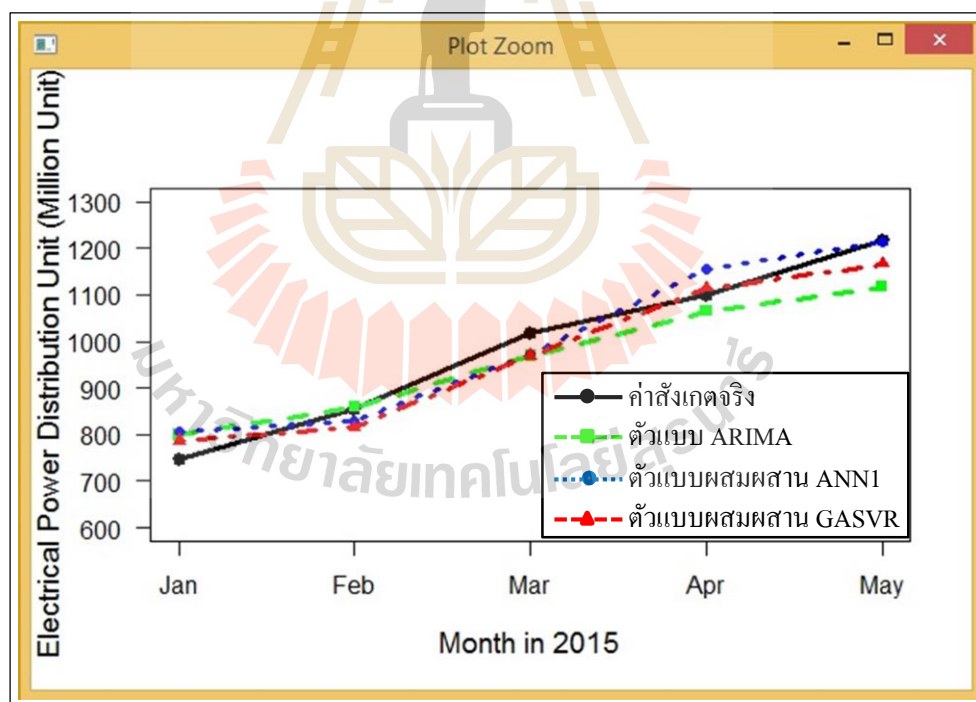
จากผลการทดสอบในส่วนของการสร้างตัวแบบพยากรณ์ด้วยวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยว พบว่า ตัวแบบ ARIMA เป็นตัวแบบพยากรณ์ที่ให้ค่าคลาดเคลื่อน RMSE และ MAPE ต่ำที่สุด ดังนั้น จึงนำผลการพยากรณ์จากตัวแบบ ARIMA มาเปรียบเทียบกับตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR เพื่อให้สอดคล้องกับวัตถุประสงค์ข้อที่ 2 ของงานวิจัยนี้ และพิสูจน์สมมุติฐานที่ตั้งไว้ว่า ถ้าใช้อินพุตที่เป็นค่าสังเกตค่าหลัง ซึ่งมีความสัมพันธ์กับเอาท์พุตที่เป็นค่าสังเกต ณ เวลา  $t$  ใด ๆ มาใช้ในการฝึกสอนตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ จะทำให้ค่าพยากรณ์มีความแม่นยำขึ้น ซึ่งผลการเปรียบเทียบ สรุปได้ดังตารางที่ 4.8 และตารางที่ 4.9 เมื่อนำค่าพยากรณ์ของทั้ง 3 ตัวแบบ มาเปรียบเทียบกับค่าสังเกตจริง สำหรับชุดข้อมูลทั้ง 2 ชุด ได้ผลดังรูปที่ 4.28 และรูปที่ 4.29

ตารางที่ 4.8 ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR สำหรับชุดข้อมูลที่ 1

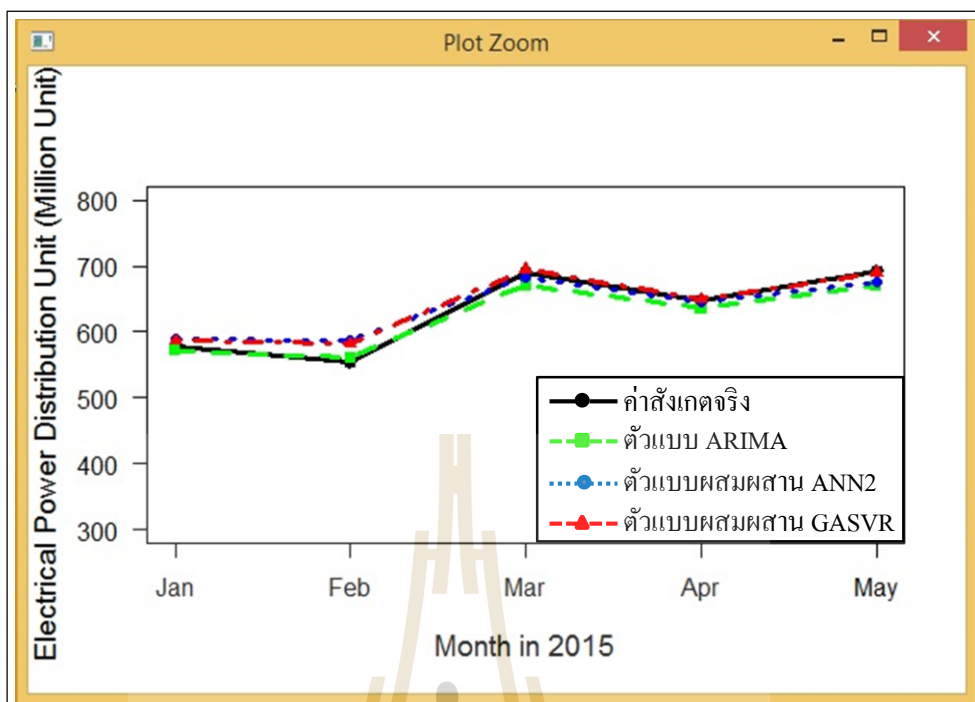
หน่วยจำหน่ายไฟฟ้า	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	SSE	MAE	RMSE	MAPE
หน่วยจำหน่ายไฟฟ้า	2558	2558	2558	2558	2558				
ค่าสังเกตจริง	745.82	853.60	1017.89	1100.93	1216.77	-	-	-	-
ค่าพยากรณ์ของตัวแบบ ARIMA	799.24	859.33	969.05	1064.46	1116.82	16348	48.52	57.18	4.80
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN1	806.67	828.23	972.95	1155.25	1213.39	9328	37.77	43.19	4.15
ค่าพยากรณ์ของตัวแบบผสมผสาน GASVR	786.15	814.86	969.63	1114.64	1166.55	8166	38.25	40.41	4.01

ตารางที่ 4.9 ผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR สำหรับชุดข้อมูลที่ 2

หน่วยจำหน่ายไฟฟ้า	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	SSE	MAE	RMSE	MAPE
ค่าสังเกตจริง	577.23	554.35	690.87	648.95	691.64	-	-	-	-
ค่าพยากรณ์ของตัวแบบ ARIMA	571.34	561.08	670.49	635.23	670.32	1138	13.61	15.09	2.08
ค่าพยากรณ์ของตัวแบบผสมผสาน ANN2	588.96	586.73	683.42	644.31	676.82	1482	14.20	17.22	2.36
ค่าพยากรณ์ของตัวแบบผสมผสาน GASVR	586.58	581.82	694.75	649.63	689.98	860	8.61	13.12	1.50



รูปที่ 4.28 กราฟเปรียบเทียบค่าสังเกตจริงและค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR สำหรับชุดข้อมูลที่ 1



รูปที่ 4.29 กราฟเปรียบเทียบค่าสังเกตจริงและค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR สำหรับชุดข้อมูลที่ 2

จากผลการเปรียบเทียบค่าพยากรณ์ของทั้ง 3 ตัวแบบ ดังตารางที่ 4.8 และตารางที่ 4.9 พบว่า ตัวแบบที่ให้ค่าคลาดเคลื่อนต่ำที่สุด ทั้ง 2 ชุดข้อมูล คือ ตัวแบบผสมผสาน GASVR ผลการเปรียบเทียบในส่วนนี้ พิสูจน์ได้ว่า สมมติฐานที่ตั้งไว้เป็นจริง โดยการเลือกอินพุตที่เป็นค่าสังเกต ล้าหลังจากรูปแบบบอโตริเกรสซีฟของตัวแบบ ARIMA ซึ่งตัวแบบ ARIMA สามารถตรวจจับรูปแบบเชิงเส้นในอนุกรมเวลาได้ดี ส่วนเทคนิคซัพพอร์ตเวกเตอร์เรกเรชัน ที่เป็นวิธีการด้านแมชชีนเลิร์นนิง สามารถตรวจจับรูปแบบไม่เป็นเชิงเส้นได้ดี (Wang and Meng, 2012; Zhang et al., 2016) เมื่อนำมาผสมผสานเป็นตัวแบบผสมผสาน GASVR จึงสามารถตรวจจับรูปแบบเชิงเส้นและไม่เป็นเชิงเส้นได้ ทั้ง 2 รูปแบบ ทำให้ความแม่นยำของค่าพยากรณ์จากตัวแบบผสมผสาน GASVR เพิ่มมากขึ้น เมื่อเปรียบเทียบกับตัวแบบ ARIMA ทั้ง 2 ชุดข้อมูล

ในส่วนของการเปรียบเทียบตัวแบบผสมผสาน ANN กับตัวแบบผสมผสาน GASVR พบว่า ตัวแบบผสมผสาน GASVR มีค่าคลาดเคลื่อนต่ำกว่า หรือมีความแม่นยำของค่าพยากรณ์สูงกว่าตัวแบบผสมผสาน ANN เนื่องจากเทคนิคซัพพอร์ตเวกเตอร์เรกเรชัน เป็นเทคนิคที่

เหมาะสำหรับชุดข้อมูลที่มีขนาดเล็ก (Zhang et al., 2016) ซึ่งชุดข้อมูลที่ใช้ศึกษาทั้ง 2 ชุด เป็นชุดข้อมูลที่มีคาบเวลารายเดือน ใช้ข้อมูลในอดีตย้อนหลังไปเป็นระยะเวลา 6 ปี มีค่าสังเกตของแต่ละชุดข้อมูล จำนวน 65 ค่าสังเกต เป็นการศึกษาการพยากรณ์ข้อมูลแบบระยะยาว (Long term period) ทำให้ชุดข้อมูลที่ใช้ศึกษา มีขนาดเล็ก นอกจากนี้ เทคนิคเครือข่ายประสาทเทียม ยังเกิดปัญหาโอเวอร์ฟิตติง (Hagiwara and Fukumizu, 2008) จากการลดค่าคลาดเคลื่อนให้ต่ำที่สุด (Fitting error) ในขั้นตอนการฝึกสอน ทำให้มีค่าคลาดเคลื่อนต่ำ เมื่อวัดกับชุดข้อมูลฝึกสอน แต่มีค่าคลาดเคลื่อนสูง เมื่อวัดกับชุดข้อมูลทดสอบ ดังนั้น เทคนิคซัพพอร์ตเวกเตอร์รีเกรสชัน จึงเหมาะสมในการนำมาใช้ในการสร้างตัวแบบพยากรณ์มากกว่าเทคนิคเครือข่ายประสาทเทียม และเป็นผลให้ตัวแบบผสมผสาน GASVR มีค่าคลาดเคลื่อน RMSE และ MAPE ต่ำกว่าตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด ทั้ง 2 ชุดข้อมูล

#### 4.3.5 อภิปรายผลการทดสอบวัดความทนทานของตัวแบบผสมผสาน

จากการทดสอบเปลี่ยนแปลงสัดส่วนของจำนวนข้อมูลในชุดข้อมูลฝึกสอนและชุดข้อมูลตรวจสอบ พบว่า เมื่อลดจำนวนข้อมูลในชุดข้อมูลฝึกสอน แล้วนำไปเพิ่มในชุดข้อมูลตรวจสอบ ตัวแบบผสมผสาน GASVR มีความทนทานต่อการเพิ่มขนาดของชุดข้อมูลตรวจสอบมากที่สุด เนื่องจาก เมื่อเพิ่มขนาดของชุดข้อมูลตรวจสอบ ตัวแบบผสมผสาน GASVR มีค่าคลาดเคลื่อน RMSE และ MAPE ต่ำกว่าตัวแบบ ARIMA และตัวแบบผสมผสาน ANN โดยให้ผลการทดสอบในลักษณะเดียวกันทั้ง 2 ชุดข้อมูล ดังที่แสดงไว้ในตารางที่ 4.6 และ 4.7 จากผลการทดสอบที่ได้รับสามารถวิเคราะห์ได้ว่า เมื่อเพิ่มขนาดของชุดข้อมูลตรวจสอบ จะทำให้รูปแบบไม่เป็นเชิงเส้นที่แฝงในอนุกรมเวลา มีแนวโน้มเพิ่มมากขึ้น แต่ตัวแบบผสมผสาน GASVR สามารถตรวจจับรูปแบบความเป็นเชิงเส้นและไม่เป็นเชิงเส้นที่แฝงอยู่ภายในอนุกรมเวลาได้ทั้ง 2 รูปแบบ ดังนั้น จึงทำให้ความคลาดเคลื่อนที่วัดด้วยค่า RMSE และ MAPE ของตัวแบบผสมผสาน GASVR มีการเปลี่ยนแปลงเพิ่มขึ้นไม่มากและรวดเร็วเท่ากับตัวแบบ ARIMA

เมื่อเปรียบเทียบตัวแบบผสมผสาน ANN กับตัวแบบ ARIMA พบว่า สำหรับชุดข้อมูลที่ 1 ตัวแบบผสมผสาน ANN มีความทนทานต่อการเพิ่มขนาดของชุดข้อมูลตรวจสอบมากกว่าตัวแบบ ARIMA เนื่องจากตัวแบบผสมผสาน ANN ให้ค่าคลาดเคลื่อน MAPE ต่ำกว่าตัวแบบ ARIMA เป็นส่วนใหญ่ ยกเว้น กลุ่มที่ 6 ซึ่งมีชุดข้อมูลตรวจสอบจำนวน 35 ค่า คิดเป็นสัดส่วน 54% เท่านั้น ที่ตัวแบบผสมผสาน ANN ให้ค่าคลาดเคลื่อน MAPE สูงกว่าตัวแบบ ARIMA ส่วนชุดข้อมูลที่ 2 ตัวแบบ ARIMA มีความทนทานต่อการเพิ่มขนาดของชุดข้อมูลตรวจสอบมากกว่าตัวแบบผสมผสาน ANN เนื่องจากตัวแบบ ARIMA ให้ค่าคลาดเคลื่อน MAPE ต่ำกว่าตัวแบบผสมผสาน ANN ทุกกลุ่ม ผลการ



ทดสอบที่ได้รับ วิเคราะห์ได้ว่า ชุดข้อมูลที่ 1 มีรูปแบบความเป็นเชิงเส้นต่ำกว่าชุดข้อมูลที่ 2 จึงทำให้ตัวแบบ ARIMA มีค่าคลาดเคลื่อนของชุดข้อมูลที่ 1 สูงกว่าชุดข้อมูลที่ 2 ดังนั้น เมื่อใช้ตัวแบบผสมผสาน ANN กับชุดข้อมูลที่ 1 ที่มีรูปแบบเชิงเส้นต่ำ หรือกล่าวอีกนัยหนึ่ง คือมีรูปแบบไม่เป็นเชิงเส้นสูง ประกอบกับการเพิ่มขนาดของชุดข้อมูลตรวจสอบ ยิ่งทำให้รูปแบบไม่เป็นเชิงเส้นของชุดข้อมูลที่ 1 เพิ่มมากขึ้น เป็นผลให้ตัวแบบผสมผสาน ANN สามารถลดค่าคลาดเคลื่อนลงได้ แต่เมื่อใช้ตัวแบบผสมผสาน ANN กับชุดข้อมูลที่ 2 ซึ่งมีรูปแบบเชิงเส้นสูง แม้จะมีการเพิ่มขนาดของชุดข้อมูลตรวจสอบ ที่ทำให้รูปแบบไม่เป็นเชิงเส้นเพิ่มมากขึ้น แต่ไม่เพียงพอที่จะทำให้ตัวแบบผสมผสาน ANN สามารถลดค่าคลาดเคลื่อนลงได้ ซึ่งแตกต่างจากตัวแบบผสมผสาน GASVR ที่สามารถลดค่าคลาดเคลื่อนให้ต่ำกว่าตัวแบบ ARIMA ได้ทั้งสองชุดข้อมูล ทั้งนี้ เนื่องจากเทคนิค SVR มีความเหมาะสมกับชุดข้อมูลฝึกสอนที่มีขนาดเล็ก มากกว่าเทคนิค ANN ประกอบกับปัญหาโอเวอร์ฟิตติงของเทคนิค ANN ที่แสดงออกได้ดีกับชุดข้อมูลฝึกสอน แต่แสดงออกได้ไม่ดีกับชุดข้อมูลตรวจสอบ ดังนั้น ยังมีจำนวนข้อมูลในชุดข้อมูลฝึกสอนน้อยลง ขณะเดียวกัน มีจำนวนข้อมูลในชุดข้อมูลตรวจสอบมากขึ้น จึงทำให้การใช้ตัวแบบผสมผสาน ANN กับชุดข้อมูลที่ 2 ที่มีรูปแบบเชิงเส้นสูง ไม่สามารถลดค่าคลาดเคลื่อนให้ต่ำกว่าตัวแบบ ARIMA ได้เช่นเดียวกับตัวแบบผสมผสาน GASVR

ผลการทดสอบและการอภิปรายผลที่ได้กล่าวมาทั้งหมด สรุปได้ว่า วิธีการสร้างตัวแบบผสมผสานระหว่างรูปแบบอตรีเกรสซีฟของตัวแบบ ARIMA และเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชัน ร่วมกับการค้นหาค่าพารามิเตอร์ที่เหมาะสมด้วยจินตคณิตอัลกอริทึม หรือตัวแบบผสมผสาน GASVR ที่งานวิจัยนี้ได้นำเสนอ สามารถเพิ่มความแม่นยำของค่าพยากรณ์ เมื่อเปรียบเทียบกับค่าพยากรณ์ที่ได้จากตัวแบบ ARIMA และตัวแบบผสมผสาน ANN นอกจากนี้ ตัวแบบผสมผสาน GASVR มีความทนทานต่อการเพิ่มขนาดของชุดข้อมูลตรวจสอบได้มากกว่าตัวแบบ ARIMA และตัวแบบผสมผสาน ANN ทั้งนี้ ข้อจำกัดของตัวแบบผสมผสาน GASVR นั้น จำเป็นต้องใช้การประมวลผลที่มีการเพิ่มขึ้นตอนในการสร้างตัวแบบพยากรณ์ที่มีความซับซ้อนมากขึ้น เพื่อให้ได้ค่าคลาดเคลื่อนที่ต่ำกว่าตัวแบบ ARIMA

จากข้อดีและข้อจำกัดข้างต้น เมื่อนำมาพิจารณาในประเด็นของความแม่นยำในการพยากรณ์ พบว่า ตัวแบบผสมผสาน GASVR ที่งานวิจัยนี้ได้นำเสนอ สามารถนำมาใช้กับชุดข้อมูลที่มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้ารายเดือนของประเทศไทย ซึ่งเป็นข้อมูลที่มีลักษณะคล้ายคลึงกับชุดข้อมูลที่ใช้ศึกษาในงานวิจัยนี้ ได้อย่างเหมาะสม



## บทที่ 5

### สรุปผลการวิจัยและข้อเสนอแนะ

พลังงานไฟฟ้าเป็นปัจจัยพื้นฐานของการดำรงชีวิตในปัจจุบัน ซึ่งระบบการผลิตพลังงานไฟฟ้าเป็นระบบขนาดใหญ่ ที่ต้องใช้ทรัพยากร บุคคลากร และค่าใช้จ่ายในการดำเนินการจำนวนมาก ถ้าสามารถพยากรณ์ปริมาณการใช้พลังงานไฟฟ้าของประชาชนได้อย่างแม่นยำ จะทำให้การวางแผนการผลิตพลังงานไฟฟ้า มีประสิทธิภาพ ช่วยลดต้นทุนในการผลิตพลังงานไฟฟ้า ไม่ก่อให้เกิดการสูญเสียค่าใช้จ่ายโดยไม่จำเป็น จากการผลิตพลังงานไฟฟ้ามากกว่าความต้องการที่แท้จริง หรือไม่ก่อให้เกิดการผลิตพลังงานไฟฟ้าไม่เพียงพอต่อความต้องการ เป็นผลให้เกิดความสูญเสียทางเศรษฐกิจ การดำเนินธุรกิจ สูญเสียเวลาและภาพลักษณ์ ดังนั้น งานวิจัยนี้ จึงนำเสนอวิธีการพัฒนาตัวแบบพยากรณ์หน่วยจำหน่ายไฟฟ้ารายเดือนของประเทศไทย โดยหน่วยจำหน่ายไฟฟ้า 1 หน่วย หมายถึงปริมาณการใช้ไฟฟ้า 1 กิโลวัตต์ต่อชั่วโมง (kW/h) นั่นคือ ข้อมูลหน่วยจำหน่ายไฟฟ้า เป็นข้อมูลที่บ่งบอกปริมาณการใช้ไฟฟ้า หรือความต้องการใช้ไฟฟ้าของผู้บริโภค ซึ่งมีลักษณะเป็นอนุกรมเวลาเชิงเดี่ยว ที่บันทึกข้อมูลเป็นรายเดือน ตามลำดับเวลา และข้อมูลหน่วยจำหน่ายไฟฟ้า ที่ใช้เป็นกลุ่มตัวอย่าง สำหรับศึกษาในงานวิจัยนี้ เป็นหน่วยจำหน่ายไฟฟ้ารายเดือนตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 ของการไฟฟ้านครหลวง และการไฟฟ้าส่วนภูมิภาค เขต 3 แห่งประเทศไทย จำนวน 2 ชุดข้อมูล

การสร้างตัวแบบพยากรณ์อนุกรมเวลาเชิงเดี่ยว สามารถแบ่งได้ 2 วิธี ได้แก่ วิธีที่ 1 คือวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยว ด้วยหลักวิเคราะห์เชิงสถิติแบบดั้งเดิม ซึ่งเป็นวิธีการศึกษารูปแบบการเคลื่อนไหว หรือเปลี่ยนแปลงของอนุกรมเวลา ด้วยการพิจารณาจากลำดับของข้อมูลที่บันทึกตามคาบเวลาในอดีต และเพื่อพยากรณ์ค่าข้อมูลในอนาคตด้วยตัวแบบพยากรณ์ หรือสมการคณิตศาสตร์ที่เหมาะสมกับข้อมูลอนุกรมเวลา โดยตัวแบบพยากรณ์ที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม ได้แก่ ตัวแบบ SLR (Simple linear regression) ที่สร้างจากวิธีการวิเคราะห์ความถดถอยอย่างง่าย ตัวแบบ Holt-Winters ที่สร้างจากวิธีการปรับให้เรียบแบบเอกซ์โพเนนเชียล และตัวแบบ ARIMA (Autoregressive integrated moving average) ที่สร้างจากวิธีการของบอซซ์และเจนกินส์ ส่วนวิธีที่ 2 คือ วิธีการสร้างตัวแบบพยากรณ์ด้วยเทคนิค

อัจฉริยะ ซึ่งเป็นวิธีการด้านแมชชีนเลิร์นนิง ได้แก่ เทคนิคเครือข่ายประสาทเทียม และเทคนิค ซัพพอร์ตเวกเตอร์รีเกรสชัน โดยดำเนินการสร้างตัวแบบพยากรณ์จากชุดข้อมูลตัวอย่างกลุ่มหนึ่ง เรียกว่า ชุดข้อมูลฝึกสอน (Training data set) ที่ประกอบด้วยข้อมูลอินพุต (Input) และข้อมูล เอาต์พุต (Output) ที่เรียกว่า ทาร์เก็ต (Target) เมื่อได้ตัวแบบการพยากรณ์ จะใช้ตัวแบบเพื่อทำนาย ข้อมูลเอาต์พุตที่ยังไม่ทราบค่า จากข้อมูลอินพุตที่ทราบค่าได้

จากที่กล่าวมา วิธีการสร้างตัวแบบพยากรณ์ มีหลากหลายวิธี ซึ่งการสร้างตัวแบบพยากรณ์ ให้สามารถพยากรณ์ข้อมูลล่วงหน้าได้อย่างแม่นยำ ควรนำเสนอการสร้างตัวแบบพยากรณ์จาก วิธีการต่างๆ หลากหลายวิธี จากนั้น เปรียบเทียบความแม่นยำของค่าพยากรณ์ด้วยการวัดค่า คลาดเคลื่อน และเลือกตัวแบบพยากรณ์ที่มีค่าคลาดเคลื่อนต่ำที่สุด สำหรับนำมาใช้พยากรณ์ข้อมูล ล่วงหน้า นอกจากนี้ วิธีการสร้างตัวแบบพยากรณ์จากวิธีวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม สามารถตรวจจับรูปแบบเชิงเส้นได้ดี แต่ตรวจจับรูปแบบไม่เป็นเชิงเส้นได้ไม่ดี ในทางกลับกัน ตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ สามารถตรวจจับรูปแบบไม่เป็นเชิงเส้นได้ดี แต่ตรวจจับ รูปแบบเชิงเส้นได้ไม่ดี ดังนั้น งานวิจัยนี้ จึงนำเสนอ วิธีการพัฒนาตัวแบบพยากรณ์ จากวิธีการ วิเคราะห์หอนุกรมเวลาเชิงเดี่ยว และเทคนิคอัจฉริยะ โดยนำเสนอวิธีการใหม่ ในการพัฒนาตัวแบบ พยากรณ์ ด้วยการผสมผสานรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA กับเทคนิคอัจฉริยะ เพื่อ สร้างเป็นตัวแบบผสมผสาน ANN และตัวแบบผสมผสาน GASVR ด้วยความมุ่งหมายว่า ตัวแบบ ผสมผสาน จะสามารถตรวจจับรูปแบบเชิงเส้นและรูปแบบไม่เป็นเชิงเส้น ที่แฝงอยู่ในอนุกรมเวลาได้ ทั้ง 2 รูปแบบ ทำให้ค่าพยากรณ์จากตัวแบบผสมผสาน มีความแม่นยำมากขึ้น สุดท้าย ประเมิน ผลการวิจัย โดยเปรียบเทียบความแม่นยำของตัวแบบผสมผสานทั้ง 2 ตัวแบบ กับตัวแบบพยากรณ์ จากวิธีวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิมที่มีความแม่นยำมากที่สุด ด้วยการวัดค่าคลาดเคลื่อน 4 ตัว ได้แก่ ค่า SSE, MAE, RMSE และ MAPE ตามวัตถุประสงค์ที่ได้กำหนดไว้ในงานวิจัยนี้

## 5.1 สรุปผลการวิจัย

ผลการทดสอบตัวแบบพยากรณ์ที่สร้างจากวิธีการวิเคราะห์หอนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม จำนวน 3 ตัวแบบ ได้แก่ ตัวแบบ SLR ตัวแบบ Holt-Winters และตัวแบบ ARIMA และผลการ ทดสอบตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA กับเทคนิคอัจฉริยะ จำนวน 2 ตัวแบบ ได้แก่ ตัวแบบผสมผสาน ANN และตัวแบบผสมผสาน GASVR สามารถสรุปผล โดยแบ่งเป็นผลสรุปย่อย ได้ดังนี้

### 5.1.1 ชุดข้อมูล

ชุดข้อมูลที่ใช้ศึกษาในงานวิจัยนี้ มีลักษณะเป็นอนุกรมเวลาเชิงเดี่ยว ที่มีค่าสังเกต 1 ค่า เป็นหน่วยจำหน่ายไฟฟ้ารายเดือน ที่บันทึกตามลำดับเวลา ตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 ของการไฟฟ้านครหลวง และการไฟฟ้าส่วนภูมิภาค เขต 3 แห่งประเทศไทย จำนวน 2 ชุดข้อมูล ซึ่งหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง เป็นหน่วยจำหน่ายไฟฟ้าของผู้บริโภคประเภทบ้านอยู่อาศัย เท่านั้น ส่วนหน่วยจำหน่ายไฟฟ้าของการไฟฟ้าส่วนภูมิภาค เขต 3 เป็นหน่วยจำหน่ายไฟฟ้ารวมทุกประเภท ยกเว้นไฟฟ้าสาธารณะ แต่ละชุดข้อมูลมีคาบเวลารายเดือน มีจำนวนคาบเวลาใน 1 ฤดูกาล เท่ากับ 12 คาบเวลา ตามรอบปีปฏิทิน และมีค่าสังเกตจำนวน 65 ค่า โดยแบ่งเป็นชุดข้อมูลที่ใช้ฝึกสอนตัวแบบพยากรณ์ทุกตัวแบบ จำนวน 60 ค่า ซึ่งเป็นค่าสังเกต ตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนธันวาคม ปีพ.ศ. 2557 และแบ่งเป็นชุดข้อมูลตรวจสอบจำนวน 5 ค่า โดยเป็นค่าสังเกต ตั้งแต่เดือนมกราคม ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 สำหรับวัดค่าคลาดเคลื่อนทั้ง 4 ตัว ของแต่ละตัวแบบพยากรณ์ เพื่อใช้เปรียบเทียบความแม่นยำของค่าพยากรณ์จากแต่ละตัวแบบ เมื่อนำข้อมูลทั้ง 2 ชุด มาทดสอบลักษณะนิ่ง (Stationary) พบว่าข้อมูลชุดที่ 1 ของการไฟฟ้านครหลวง เป็นอนุกรมเวลาที่มีลักษณะนิ่ง (Stationary) สามารถนำไปใช้สร้างตัวแบบพยากรณ์ด้วยวิธีการแบบดั้งเดิมได้ ส่วนข้อมูลชุดที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 เป็นอนุกรมเวลาที่มีลักษณะไม่นิ่ง (Non-stationary) จะต้องนำมาหาผลต่างของค่าสังเกต เพื่อสร้างอนุกรมเวลาใหม่ ก่อนนำไปสร้างตัวแบบพยากรณ์

### 5.1.2 สรุปผลการทดสอบตัวแบบพยากรณ์จากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม

งานวิจัยนี้ ดำเนินการสร้างตัวแบบพยากรณ์จากวิธีวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม จำนวน 3 ตัวแบบ ได้แก่ ตัวแบบ SLR ตัวแบบ Holt-Winters และตัวแบบ ARIMA โดยดำเนินการสร้างและประเมินผลตัวแบบพยากรณ์กับชุดข้อมูล ด้วยขั้นตอนวิธีเดียวกันทั้ง 2 ชุดข้อมูล ผลการทดสอบ พบว่า ตัวแบบ ARIMA เป็นตัวแบบที่ให้ค่าคลาดเคลื่อนต่ำที่สุดทั้ง 2 ชุดข้อมูล โดยชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง ตัวแบบ ARIMA ให้ค่า SSE, MAE, RMSE และ MAPE เท่ากับ 16,348, 48.52, 57.18 และ 4.80 ตามลำดับ ส่วนชุดข้อมูลที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 ตัวแบบ ARIMA ให้ค่า SSE, MAE, RMSE และ MAPE เท่ากับ 1,138, 13.61, 15.09 และ 2.08 ตามลำดับ ดังนั้น เมื่อเปรียบเทียบระหว่างตัวแบบพยากรณ์ที่สร้างจากวิธีการวิเคราะห์อนุกรมเวลาเชิงเดี่ยวแบบดั้งเดิม สรุปได้ว่า ตัวแบบ ARIMA เป็นตัวแบบที่เหมาะสมที่สุดของชุด

ข้อมูลทั้ง 2 ชุด สำหรับนำมาพยากรณ์ค่าสังเกตล่วงหน้า นอกจากนี้ สรุปเพิ่มเติมได้ว่า ตัวแบบ SLR ที่มีเวลา (t) เป็นตัวแปรอิสระ และมีตัวแปรตามเป็นค่าสังเกตของอนุกรมเวลา ไม่เหมาะสมสำหรับนำมาพยากรณ์อนุกรมเวลาทั้ง 2 ชุด เนื่องจากไม่สามารถตรวจจับความเคลื่อนไหวจากส่วนประกอบแนวโน้มและฤดูกาลของอนุกรมเวลาได้

### 5.1.3 สรุปผลการวิเคราะห์รูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA

ผลการค้นหารูปแบบที่เหมาะสมที่สุดของตัวแบบ ARIMA ด้วยฟังก์ชัน `auto.arima()` สำหรับชุดข้อมูลทั้ง 2 ชุด พบว่า ชุดข้อมูลที่ 1 มีรูปแบบ ARIMA(1, 0, 0)x(1, 0, 0)<sub>12</sub> ส่วนชุดข้อมูลที่ 2 มีรูปแบบ ARIMA(1, 0, 0)x(1, 1, 0)<sub>12</sub> เมื่อนำมาวิเคราะห์รูปแบบออโตรีเกรสซีฟสรุปได้ว่า รูปแบบออโตรีเกรสซีฟของชุดข้อมูลที่ 1 มีจำนวน 3 ค่าล่าหลัง ประกอบด้วย ค่าสังเกตล่าหลังที่เวลา 1, 12 และ 13 คาบเวลา ( $Y_{t-1}, Y_{t-12}, Y_{t-13}$ ) ส่วนรูปแบบออโตรีเกรสซีฟของชุดข้อมูลที่ 2 มีจำนวน 5 ค่าล่าหลัง ประกอบด้วย ค่าสังเกตล่าหลังที่เวลา 1, 12, 13, 24 และ 25 คาบเวลา ( $Y_{t-1}, Y_{t-12}, Y_{t-13}, Y_{t-24}, Y_{t-25}$ ) ดังนั้น สำหรับชุดข้อมูลที่ 1 เลือก  $Y_{t-1}, Y_{t-12}$  และ  $Y_{t-13}$  เป็นอินพุตเวกเตอร์ เพื่อสร้างเป็นชุดข้อมูลฝึกสอนให้กับตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ ส่วนชุดข้อมูลที่ 2 เลือก  $Y_{t-1}, Y_{t-12}, Y_{t-13}, Y_{t-24}$  และ  $Y_{t-25}$  เป็นอินพุตเวกเตอร์ เพื่อสร้างเป็นชุดข้อมูลฝึกสอนให้กับตัวแบบพยากรณ์ที่สร้างจากเทคนิคอัจฉริยะ

### 5.1.4 สรุปผลการทดสอบตัวแบบผสมผสาน ANN

การดำเนินการทดสอบตัวแบบผสมผสาน ANN มีจุดประสงค์เพื่อค้นหาโครงสร้างของเครือข่ายประสาทเทียมที่เหมาะสมที่สุด โดยทดสอบสร้างตัวแบบผสมผสาน ANN จำนวน 4 ตัวแบบ แต่ละตัวแบบ มีจำนวนนิวรอนในชั้นซ่อนเร้นแตกต่างกัน ตั้งแต่ 1 ถึง 4 นิวรอน เช่นเดียวกันทั้ง 2 ชุดข้อมูล จากนั้น วัดค่าคลาดเคลื่อนของแต่ละตัวแบบ เพื่อกำหนดโครงสร้างของตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด จากตัวแบบผสมผสาน ANN ที่ให้ค่าคลาดเคลื่อนต่ำที่สุด ผลการทดสอบ สรุปได้ว่า สำหรับชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง ตัวแบบผสมผสาน ANN1 เป็นตัวแบบพยากรณ์ที่เหมาะสมที่สุด เนื่องจากวัดค่าคลาดเคลื่อน SSE, MAE, RMSE และ MAPE ได้ต่ำที่สุด เท่ากับ 9,328, 37.77, 43.19 และ 4.15 ตามลำดับ โดยมีโครงสร้างของเครือข่ายประสาทเทียมเป็น 3-1-1 หมายความว่า ชั้นอินพุต มีจำนวน 3 โหนด แต่ละโหนด คือ ค่าสังเกตล่าหลัง  $Y_{t-1}, Y_{t-12}$  และ  $Y_{t-13}$  ตามลำดับ ส่วนชั้นซ่อนเร้น มีจำนวน 1 นิวรอน และชั้นเอาต์พุต มีจำนวน 1 นิวรอน คือ ค่าสังเกต ณ เวลา t ( $Y_t$ ) ส่วนชุดข้อมูลที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 ตัวแบบผสมผสาน ANN2 เป็นตัวแบบพยากรณ์ที่เหมาะสมที่สุด เนื่องจากวัดค่าคลาดเคลื่อน SSE, MAE,

RMSE และ MAPE ได้ต่ำที่สุด เท่ากับ 1,482, 14.20, 17.22 และ 2.36 ตามลำดับ โดยมีโครงสร้างของเครือข่ายประสาทเทียมเป็น 5-2-1 หมายความว่า ชั้นอินพุต มีจำนวน 5 โหนด แต่ละโหนดคือค่าสังเกตค่าหลัง  $Y_{t-1}$ ,  $Y_{t-2}$ ,  $Y_{t-3}$ ,  $Y_{t-4}$  และ  $Y_{t-5}$  ตามลำดับ ส่วนชั้นซ่อนเร้น มีจำนวน 2 นิวรอน และชั้นเอาต์พุต มีจำนวน 1 นิวรอน คือ ค่าสังเกต ณ เวลา  $t$  ( $Y_t$ ) นอกจากนี้ พบว่า ตัวแบบผสมผสาน ANN4 ซึ่งเป็นตัวแบบที่มีจำนวนนิวรอนในชั้นซ่อนเร้นมากที่สุด ให้ค่าคลาดเคลื่อนสูงที่สุดทั้ง 2 ชุดข้อมูล แสดงว่า ถ้าโครงสร้างของเครือข่ายประสาทเทียมมีความซับซ้อนมากขึ้น มีแนวโน้มทำให้ค่าคลาดเคลื่อนเพิ่มมากขึ้น เมื่อวัดกับชุดข้อมูลทดสอบ ซึ่งเป็นปัญหาโอเวอร์ฟิตติงที่พบในเทคนิคเครือข่ายประสาทเทียม

### 5.1.5 สรุปผลการเปรียบเทียบค่าพยากรณ์ของตัวแบบ ARIMA ตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR

เมื่อนำตัวแบบ ARIMA ซึ่งเป็นตัวแบบที่แม่นยำที่สุดในกลุ่มตัวแบบที่สร้างจากวิธีวิเคราะห์หอนุกรมเวลาเชิงเดี่ยว เปรียบเทียบกับตัวแบบผสมผสาน ANN ที่เหมาะสมที่สุด และตัวแบบผสมผสาน GASVR เพื่อเปรียบเทียบว่า ตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA ซึ่งสามารถตรวจจับรูปแบบเชิงเส้น ได้ดี เมื่อนำมาผสมผสานกับเทคนิคอัจฉริยะ ที่สามารถตรวจจับรูปแบบไม่เป็นเชิงเส้น ได้ดี จะสามารถทำให้ค่าพยากรณ์มีความแม่นยำเพิ่มขึ้นได้ เนื่องจาก สามารถตรวจจับรูปแบบเชิงเส้นและไม่เป็นเชิงเส้น ได้ทั้ง 2 รูปแบบ ผลการทดสอบ สรุปได้ว่า สำหรับชุดข้อมูลที่ 1 ของการไฟฟ้านครหลวง ตัวแบบผสมผสาน GASVR ให้ค่าคลาดเคลื่อน SSE, MAE ,RMSE และ MAPE ต่ำที่สุด เท่ากับ 8,166, 38.25, 40.41 และ 4.01 ตามลำดับ โดยตัวแบบผสมผสาน GASVR ให้ความแม่นยำเพิ่มขึ้น เมื่อเปรียบเทียบกับค่า MAPE กับตัวแบบ ARIMA เท่ากับ +16.46% และชุดข้อมูลที่ 2 ของการไฟฟ้าส่วนภูมิภาค เขต 3 ตัวแบบผสมผสาน GASVR ให้ค่าคลาดเคลื่อน SSE, MAE ,RMSE และ MAPE ต่ำที่สุด เท่ากับ 860, 8.61, 13.12 และ 1.50 ตามลำดับ โดยตัวแบบผสมผสาน GASVR ให้ความแม่นยำเพิ่มขึ้น เมื่อเปรียบเทียบกับค่า MAPE กับตัวแบบ ARIMA เท่ากับ +27.88% ดังนั้น สรุปได้ว่า ตัวแบบผสมผสาน GASVR ซึ่งเป็นตัวแบบผสมผสานระหว่างรูปแบบออโตรีเกรสซีฟของตัวแบบ ARIMA และเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชันร่วมกับการค้นหาค่าพารามิเตอร์ที่เหมาะสมด้วยจีเนติกอัลกอริทึม ที่งานวิจัยนี้ได้นำเสนอ สามารถเพิ่มความแม่นยำของค่าพยากรณ์ได้สูงขึ้น เมื่อเทียบกับตัวแบบ ARIMA และตัวแบบผสมผสาน GASVR เป็นตัวแบบที่ให้ค่าคลาดเคลื่อนต่ำที่สุด หรือแม่นยำมากที่สุด นอกจากนี้ เมื่อดำเนินการทดสอบวัดความทนทานของตัวแบบผสมผสาน GASVR และตัวแบบผสมผสาน

ANN เปรียบเทียบกับตัวแบบ ARIMA ด้วยการเพิ่มขนาดของชุดข้อมูลตรวจสอบ พบว่า ตัวแบบผสมผสาน GASVR ให้ค่าคลาดเคลื่อนโดยรวม ต่อการเปลี่ยนขนาดของชุดข้อมูลตรวจสอบ ต่ำที่สุด ทั้ง 2 ชุดข้อมูล ทำให้สรุปได้ว่า ตัวแบบผสมผสาน GASVR มีความทนทานต่อการเปลี่ยนขนาดของชุดข้อมูลตรวจสอบมากที่สุด ดังนั้น ตัวแบบผสมผสาน GASVR จึงเป็นตัวแบบที่เหมาะสมที่สุดสำหรับนำมาใช้พยากรณ์อนุกรมเวลาเชิงเดี่ยว ที่มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้ารายเดือนของประเทศไทย

## 5.2 ปัญหาและข้อเสนอแนะ

วิธีการพัฒนาตัวแบบผสมผสานที่งานวิจัยนี้ได้นำเสนอ สามารถนำไปประยุกต์ใช้เพื่อพยากรณ์อนุกรมเวลาเชิงเดี่ยว ที่มีค่าสังเกตเป็นหน่วยจำหน่ายไฟฟ้ารายเดือนของประเทศไทย ทำให้ได้ผลการทดสอบที่ดี เนื่องจากสามารถลดค่าคลาดเคลื่อนให้ต่ำกว่าตัวแบบ ARIMA ทั้งนี้ ตัวแบบผสมผสานดังกล่าว จำเป็นต้องใช้การประมวลผลที่มีการเพิ่มขึ้นตอนในการสร้างตัวแบบพยากรณ์ที่มีความซับซ้อนมากขึ้น และเมื่อนำไปใช้กับอนุกรมเวลาเชิงเดี่ยว ที่มีค่าสังเกตแตกต่างจากอนุกรมเวลาที่ใช้ศึกษาในงานวิจัยนี้ อาจส่งผลให้ค่าคลาดเคลื่อนสูงกว่าตัวแบบ ARIMA เนื่องจาก แต่ละอนุกรมเวลา มีส่วนประกอบของอนุกรมเวลา ได้แก่ แนวโน้ม ฤดูกาล ความผันผวนผิดปกติ และผลกระทบจากปัจจัยภายนอกที่แฝงอยู่ในอนุกรมเวลา แตกต่างกัน นอกจากนี้ งานวิจัยนี้ เป็นการศึกษาอนุกรมเวลาเชิงเดี่ยว ที่มีปัจจัยเดียว คือ หน่วยจำหน่ายไฟฟ้ารายเดือน ที่มีค่าเปลี่ยนแปลงเคลื่อนไหว ตามลำดับเวลา แต่ถ้าสามารถรวบรวมข้อมูล ซึ่งเป็นปัจจัยอื่นที่เกี่ยวข้องกับค่าหน่วยจำหน่ายไฟฟ้า ได้อย่างมีคุณภาพและน่าเชื่อถือ เช่น อุณหภูมิ หรือจำนวนประชากร เข้ามาเป็นปัจจัยร่วม สำหรับสร้างตัวแบบพยากรณ์ อาจทำให้ความแม่นยำของค่าพยากรณ์ เพิ่มสูงกว่าผลที่ได้จากงานวิจัยนี้



## รายการอ้างอิง

- การไฟฟ้านครหลวงแห่งประเทศไทย. (2558). รายงานสถานการณ์การจำหน่ายไฟฟ้า [Online].  
ได้จาก: <http://www.mea.or.th/download/index.php#>
- การไฟฟ้าส่วนภูมิภาคแห่งประเทศไทย. (2559). หน่วยจำหน่ายของ การไฟฟ้าส่วนภูมิภาคเขต 3  
ภาค 2 นครราชสีมา (นครราชสีมา, ชัยภูมิ, บุรีรัมย์, สุรินทร์). แผนกข้อมูลสถิติและ  
วิเคราะห์ การไฟฟ้าส่วนภูมิภาคเขต 3 ภาค 2 นครราชสีมา.
- ทรงศิริ แต่สมบัติ. (2549). การพยากรณ์เชิงปริมาณ. พิมพ์ครั้งที่ 1. กรุงเทพฯ:  
สำนักพิมพ์มหาวิทยาลัยเกษตรศาสตร์.
- พงศ์เทพ วิวรรณระเดช และสมชาย จาคศิริ. (2550). การวิเคราะห์อนุกรมเวลา (Time Series  
Analysis) และระบบสารสนเทศทางภูมิศาสตร์ (GIS) ในการพัฒนาองค์ความรู้ด้าน  
สิ่งแวดล้อมและสุขภาพ. พิมพ์ครั้งที่ 1. เชียงใหม่: ศูนย์ประสานข้อมูลปัญหาหมอกควัน  
ในพื้นที่ภาคเหนือ ภาควิชาวิทยาศาสตร์ชุมชน คณะแพทยศาสตร์ มหาวิทยาลัยเชียงใหม่.
- อาทิตย์ ศรีแก้ว. (2552). ปัญญาเชิงคำนวณ (Computational Intelligence). พิมพ์ครั้งที่ 1. กรุงเทพฯ:  
จรัสสนิทวงศ์การพิมพ์.
- Bagheripour, P., Gholami, A., Asoodeh, M., and Asadi, M. V. (2015). Support vector regression  
based determination of shear wave velocity. **Journal of Petroleum Science and  
Engineering**. 125: 95-99.
- Bergmeir, C., and Benítez, J. M. (2012). On the use of cross-validation for time series predictor  
evaluation. **Information Sciences**. 191: 192-213.
- Bowerman, B. L., O'Connell, R. T., and Koehler A. B. (2005). **Forecasting, time series, and  
regression: an applied approach**. 4th ed. The United States of America: Thomson Brooks.
- Box, G. E. P., Jenkins, G. L. and Reinsel G. C. (2008). **Times series analysis forecasting and  
control**. 4<sup>th</sup> ed. Hoboken, N.J. : John Wiley.
- Bunn, D. W., and Farmer, E. D. (1985). **Comparative Models for Electrical Load Forecasting**.  
New York: John Wiley and Sons.



- Cadenas, E., Jaramillo, O. A., and Rivera, W. (2010). Analysis and forecasting of wind velocity in chetumal, quintana roo, using the single exponential smoothing method. **Renewable Energy**. 35: 925-930.
- Cheng, C. T., Feng, Z. K., Niu, W. J., and Liao, S. L. (2015). Heuristic Methods for Reservoir Monthly Inflow Forecasting: A Case Study of Xinfengjiang Reservoir in Pearl River, China. **Water**. 7(8): 4477-4495.
- Chujai, P., Kerdprasop, N., and Kerdprasop, K. (2013). Time series analysis of household electric consumption with ARIMA and ARMA models. **Proceedings of the International MultiConference of Engineers and Computer Scientists**. China: Hong Kong.
- Dickey, D. A., and Fuller, W. A. (1979). Distribution of the Estimators for Autoregressive Time Series with a Unit Root. **Journal of the American Statistical Association** 74. 366: 427–431.
- Hagiwara, K., and Fukumizu, K., (2008). Relation between weight size and degree of over-fitting in neural network regression. **Neural Networks**. 21: 48–58.
- Hamzacebi, C. (2008). Improving artificial neural network's performance in seasonal time series forecasting. **Information Sciences**. 178: 4550-4559.
- Haykin, S. (1994). **Neural networks: A comprehensive foundation**. New York: Macmillan College.
- Holland, J. H. (1975) **Adaptation in Natural and Artificial Systems**. Michigan: University of Michigan Press.
- Hong, W. C. (2009). Electric load forecasting by support vector model. **Applied Mathematical Modelling**. 33: 2444–2454.
- Hussain, A., Rahman, M., and Memon, J. A. (2016). Forecasting electricity consumption in Pakistan: the way forward. **Energy Policy**. 90: 73-80
- Hyndman, R.J. (2015). **Forecasting Functions for Time Series and Linear Models**. CRAN Repository. Available from: <http://github.com/robjhyndman/forecast>.
- Ismail, Z., Yahya, A., and Mahpol, K. A. (2009). Forecasting Peak Load Electricity Demand Using Statistics and Rule Based Approach. **American Journal of Applied Sciences**. 6(8): 1618-1625.
- Jirong, G., Mingcang, Z., and Liuguangyan, J. (2011). Housing price forecasting based on genetic algorithm and support vector machine. **Expert Systems with Applications**. 38: 3383–3386.

- Kaytez, F., Taplamacioglu, M. C., Cam, E., and Hardalac, F. (2015). Forecasting electricity consumption: A comparison of regression analysis, neural networks and least squares support vector machines. **Electrical Power and Energy Systems**. 67: 431–438.
- Keka, I., and Hamiti, M. (2013). Load profile analyses using R language. **Proceedings of the ITI 2013 35th Int. Conf. on International Technology Interfaces**. Croatia: Cavtat.
- Kumar, S. (2004). **Neural Networks: A Classroom Approach**. New Delhi: Tata McGraw-Hill Education.
- Massana, J., Pous, C., Burgas, L., Melendez, J., and Colomer, J. (2015). Short-term load forecasting in a non-residential building contrasting models and attributes. **Energy and Buildings**. 92: 322–330.
- Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., Chang, C.C., and Lin, C.C. (2015). **Misc. Functions of the Department of Statistics, Probability Theory Group**. CRAN Repository. Available from: <http://cran.r-project.org/web/packages/e1071/index.html>.
- Montgomery, D. C., Jennings, C. L., and Kulahci, M. (2008). **Introduction to time series analysis and forecasting**. The United States of America: Wiley Interscience.
- Ogcu, G., Demirel, O. F., and Zaim, S. (2012). Forecasting electricity consumption with neural networks and support vector regression. **Procedia-Social and Behavioral Sciences**. 58: 1576-1585.
- Patterson, D. W. (1996). **Artificial neural networks: theory and applications**. Singapore: Prentice Hall.
- Prema, V., and Rao, K. U. (2015). Development of statistical time series models for solar power prediction. **Renewable Energy**. 83: 100-109.
- Rumerlhart, D. J., and Zipser D. (1985). Feature Discovery by Competitive Learning. **Cognitive Science**. 9: 75–112.
- Sajan, K.S., Kumar, V., and Tyagi, B. (2015). Genetic algorithm based support vector machine for on-line voltage stability monitoring. **Electrical Power and Energy Systems**. 73: 200-208
- Szoplik, J. (2015). Forecasting of natural gas consumption with artificial neural networks. **Energy**. 85: 208-220
- Tseng, F. M., and Tzeng, G. H. (2002). A fuzzy seasonal ARIMA model for forecasting. **Fuzzy Sets and Systems**. 126: 367-376.

- Ustun, B., Melssen, W. J., Oudenhuijzen, M., and Buydens, L. M. C. (2005). Determination of optimal support vector regression parameters by genetic algorithms and simplex optimization. **Analytica Chimica Acta**. 544: 292-305.
- Vapnik, V. (1995). **The Nature of Statistical Learning Theory**. Springer-Verlag.
- Wang, J., Li, L., Niu, D., and Tan, Z. (2012). An annual load forecasting model based on support vector regression with differential evolution algorithm. **Applied Energy**. 94: 65-70.
- Wang, X., and Meng, M. (2012). A Hybrid Neural Network and ARIMA Model for Energy Consumption Forecasting. **Journal of Computers**. 7: 1184-1190.
- Wang, Y., Wang, J., Zhao, G., and Dong, Y. (2012). Application of residual modification approach in seasonal ARIMA for electricity demand forecasting: A case study of China. **Energy Policy**. 48: 284-294.
- Wehrens, R., and Buydens, L. M. C. (1998). Evolutionary optimisation: a tutorial. **Trends in Analytical Chemistry**. 17(4): 193-203.
- Wei, J., and Shou, L. (2010). Model selection using modified AIC and BIC in joint modeling of paired functional data. **Statistics and Probability Letters**. 80: 1918–1924.
- Willighagen, E., and Ballings, M. (2015). **R Based Genetic Algorithm**. CRAN Repository. Available from: <http://github.com/egonw/genalg>.
- Wuertz, D. (2013). **Trends and Unit Roots**. CRAN Repository. Available from: <http://www.rmetrics.org/>.
- Yuan, C., Liu, S., and Fang, Z. (2016). Comparison of China's primary energy consumption forecasting by using ARIMA (the autoregressive integrated moving average) model and GM(1,1) model. **Energy**. 100: 384-390.
- Zaki, M. J., and Meira, W. Jr. (2014) **Data Mining and Analysis: Fundamental Concepts and Algorithms**. Cambridge University Press: 566-601.
- Zhang, F., Deb, C., Lee, S.E., , Yang, J., and Shah, K.W. (2016). Time series forecasting for building energy consumption using weighted Support Vector Regression with differential evolution optimization technique. **Energy and Buildings**. 126: 94-103.



ภาคผนวก ก

รหัสต้นฉบับของโปรแกรม

มหาวิทยาลัยเทคโนโลยีสุรนารี

```

#โปรแกรมสร้างตัวแบบ SLR ตัวแบบ Holt-Winters และตัวแบบ ARIMA ด้วยภาษาอาร์

library("forecast", lib.loc=~R/win-library/3.2")
library("fUnitRoots", lib.loc=~R/win-library/3.2")

# MEA traditional model

# Create SLR model
SLRDF1 <- read.csv(file="d:/CodeR/Data/EuseMEA_SLRmodel.csv", header=TRUE, sep=",")
SLRfit1 <- glm(SLRDF1[, 4] ~ SLRDF1[, 3])
PredictSLR1 <- predict(SLRfit1, n.ahead=5)

# Create Holt-Winters model
HoltWintersDF1 <- read.csv(file="d:/CodeR/Data/EuseMEA2010_2014.csv",header=TRUE,
sep=",")
HoltWintersTS1 <- ts(HoltWintersDF1[, 3], start=c(2010, 1), end=c(2014, 12), frequency=12)
HWfit1 <- HoltWinters(HoltWintersTS1)
PredictHW1 <- predict(HWfit1, n.ahead=5)

#Box and Jengins ARIMA model
#ARIMA of MEA
arimaDF1 <- read.csv(file="d:/CodeR/Data/EuseMEA2010_2014.csv", header=TRUE, sep=",")
arimaTS1 <- ts(arimaDF1[, 3], start=c(2010, 1), end=c(2014, 12), frequency=12)
p<-arimaorder(auto.arima(arimaTS1))[1]
d<-arimaorder(auto.arima(arimaTS1))[2]
q<-arimaorder(auto.arima(arimaTS1))[3]
P<-arimaorder(auto.arima(arimaTS1))[4]
D<-arimaorder(auto.arima(arimaTS1))[5]
Q<-arimaorder(auto.arima(arimaTS1))[6]
arimaM1 <- arima(arimaTS1, order=c(p, d, q) ,list(order=c(P, D, Q), period=12))
PredictARIMA1 <- predict(arimaM1, n.ahead=5)

#Measure error: SSE, MAE, RMSE and MAPE
TestSet1<-read.csv(file="d:/CodeR/Data/EtestMEA1_5_2015.csv")
Actual1<-c(TestSet1$Electric_Units)

```

```

ForecastSLR1<-c(PredictSLR1[61:65])
err1SLR1 <- Actual1-ForecastSLR1
err2SLR1 <- (Actual1-ForecastSLR1)/Actual1
sseSLR1<-sum(err1SLR1^2)
maeSLR1<-mean(abs(err1SLR1))
rmseSLR1 <- sqrt(mean(err1SLR1^2))
mapeSLR1<-mean(abs(100*err2SLR1))
ForecastHW1<-c(PredictHW1[1:5])
err1HW1 <- Actual1-ForecastHW1
err2HW1 <- (Actual1-ForecastHW1)/Actual1
sseHW1<-sum(err1HW1^2)
maeHW1<-mean(abs(err1HW1))
rmseHW1 <- sqrt(mean(err1HW1^2))
mapeHW1<-mean(abs(100*err2HW1))
ForecastARIMA1<-c(PredictARIMA1$pred[1:5])
err1ARIMA1 <- Actual1-ForecastARIMA1
err2ARIMA1 <- (Actual1-ForecastARIMA1)/Actual1
sseARIMA1<-sum(err1ARIMA1^2)
maeARIMA1<-mean(abs(err1ARIMA1))
rmseARIMA1 <- sqrt(mean(err1ARIMA1^2))
mapeARIMA1<-mean(abs(100*err2ARIMA1))
sseSLR1
maeSLR1
rmseSLR1
mapeSLR1
sseHW1
maeHW1
rmseHW1
mapeHW1
sseARIMA1

```

```

macARIMA1
rmseARIMA1
mapeARIMA1
# PEA traditional model
# Create SLR model
SLRDF2 <- read.csv(file="d:/CodeR/Data/EusePEA_SLRmodel.csv", header=TRUE, sep=",")
SLRfit2 <- glm(SLRDF2[, 4] ~ SLRDF2[, 3])
PredictSLR2 <- predict(SLRfit2, n.ahead=5)
# Create Holt-Winters model
HoltWintersDF2 <- read.csv(file="d:/CodeR/Data/EusePEA2010_2014.csv",header=TRUE,
sep=",")
HoltWintersTS2 <- ts(HoltWintersDF2[, 3], start=c(2010, 1), end=c(2014, 12), frequency=12)
HWfit2 <- HoltWinters(HoltWintersTS2)
PredictHW2 <- predict(HWfit2, n.ahead=5)
#ARIMA of PEA
arimaDF2 <- read.csv(file="d:/CodeR/Data/EusePEA2010_2014.csv", header=TRUE, sep=",")
arimaTS2 <- ts(arimaDF2[,3], start=c(2010,1), end=c(2014,12), frequency=12)
p<-arimaorder(auto.arima(arimaTS2))[1]
d<-arimaorder(auto.arima(arimaTS2))[2]
q<-arimaorder(auto.arima(arimaTS2))[3]
P<-arimaorder(auto.arima(arimaTS2))[4]
D<-arimaorder(auto.arima(arimaTS2))[5]
Q<-arimaorder(auto.arima(arimaTS2))[6]
arimaM2 <- arima(arimaTS2, order=c(p, d, q) ,list(order=c(P, D, Q), period=12))
PredictARIMA2 <- predict(arimaM2, n.ahead=5)
#Measure error: SSE, MAE, RMSE and MAPE
TestSet2<-read.csv(file="d:/CodeR/Data/EtestPEA1_5_2015.csv")
Actual2<-c(TestSet2$Electric_Units)
ForecastSLR2<-c(PredictSLR2[61:65])
err1SLR2 <- Actual2-ForecastSLR2

```



```
err2SLR2 <- (Actual2-ForecastSLR2)/Actual2
sseSLR2<-sum(err1SLR2^2)
maeSLR2<-mean(abs(err1SLR2))
rmseSLR2 <- sqrt(mean(err1SLR2^2))
mapeSLR2<-mean(abs(100*err2SLR2))
ForecastHW2<-c(PredictHW2[1:5])
err1HW2 <- Actual2-ForecastHW2
err2HW2 <- (Actual2-ForecastHW2)/Actual2
sseHW2<-sum(err1HW2^2)
maeHW2<-mean(abs(err1HW2))
rmseHW2 <- sqrt(mean(err1HW2^2))
mapeHW2<-mean(abs(100*err2HW2))
ForecastARIMA2<-c(PredictARIMA2$pred[1:5])
err1ARIMA2 <- Actual2-ForecastARIMA2
err2ARIMA2 <- (Actual2-ForecastARIMA2)/Actual2
sseARIMA2<-sum(err1ARIMA2^2)
maeARIMA2<-mean(abs(err1ARIMA2))
rmseARIMA2 <- sqrt(mean(err1ARIMA2^2))
mapeARIMA2<-mean(abs(100*err2ARIMA2))
sseSLR2
maeSLR2
rmseSLR2
mapeSLR2
sseHW2
maeHW2
rmseHW2
mapeHW2
sseARIMA2
maeARIMA2
rmseARIMA2
```

```

mapeARIMA2
# END PROGRAM

% โปรแกรมสร้างตัวแบบ Hybrid ANN ด้วยโปรแกรม Matlab

clear all
close all
clc
AllData = xlsread('AllEuse.xlsx');
minD = min(min(AllData))
maxD = max(max(AllData))
DataNor = (AllData - minD)/(maxD-minD)+0.1;
data = xlsread('InputTraining.xlsx');
P= (data - minD)/(maxD-minD)+0.1
P=P';
Target = xlsread('TargetTraining.xlsx');
T=(Target - minD)/(maxD-minD)+0.1
T = T';

%net=newff(minmax(P),[4 1],{'logsig','purelin'},'trainlm'); %ANN4
%net=newff(minmax(P),[3 1],{'logsig','purelin'},'trainlm'); %ANN3
%net=newff(minmax(P),[2 1],{'logsig','purelin'},'trainlm'); %ANN2
net=newff(minmax(P),[1 1],{'logsig','purelin'},'trainlm'); %ANN1
net.trainParam.epochs=3000;
net.trainParam.goal=0.0001;
net.trainParam.min_grad = 1.00e-25;
net.trainParam.mu_Max = 1.00e+10;
%ANN4 [4 1]
%net.iw{1,1}=[0.5 0.5 0.5;0.5 0.5 0.5;0.5 0.5 0.5;0.5 0.5 0.5]
%net.lw{2,1}=[0.5 0.5 0.5 0.5]
%net.b{1}=[0.5 0.5 0.5 0.5]'

```

```

%net.b{2}=[0.5]'
%ANN3 [3 1]
%net.iw{1,1}=[0.5 0.5 0.5;0.5 0.5 0.5;0.5 0.5 0.5]
%net.lw{2,1}=[0.5 0.5 0.5]
%net.b{1}=[0.5 0.5 0.5]'
%net.b{2}=[0.5]'
%ANN2 [2 1]
%net.iw{1,1}=[0.5 0.5 0.5;0.5 0.5 0.5]
%net.lw{2,1}=[0.5 0.5]
%net.b{1}=[0.5 0.5]'
%net.b{2}=[0.5]'
%ANN1 [1 1]
net.iw{1,1}=[0.5 0.5 0.5]
net.lw{2,1}=[0.5]
net.b{1}=[0.5]'
net.b{2}=[0.5]'
net = train(net,P,T);
% Calculate SSE, MAE, RMSE and MAPE
DataTest = [DataNor(60) DataNor(49) DataNor(48);
            DataNor(61) DataNor(50) DataNor(49);
            DataNor(62) DataNor(51) DataNor(50);
            DataNor(63) DataNor(52) DataNor(51);
            DataNor(64) DataNor(53) DataNor(52)];
TarTest = [DataNor(61) DataNor(62) DataNor(63) DataNor(64) DataNor(65)];
aTest = sim(net,DataTest);
Actual_Value = (TarTest-0.1).*(maxD-minD)+minD
Predict_Value = (aTest-0.1).*(maxD-minD)+minD
Error = Actual_Value-Predict_Value;
E1 = 100*Error./Actual_Value;
SSE_NN=sum(Error.*Error)

```

```

MAE_NN=mean(abs(Error))
RMSE_NN = sqrt(sum(Error.*Error)/5)
MAPE_NN = sum(abs(E1))/5
% END PROGRAM

# โปรแกรมสร้างตัวแบบ Hybrid GASVR ด้วยภาษาอาร์

# MEA
# Find optimal parameter by GA
TrainLoad1<-read.csv(file="d:/CodeR/Data/TrainingMEA.csv", header=TRUE, sep=",")
InV1<-data.frame(TrainLoad1$ValT1,TrainLoad1$ValT12,TrainLoad1$ValT13)
evaluate1 <- function(string=c()) {
  returnVal = NA;
  if (length(string) == 2) {
    model1 <-
svm(x=InV1,y=TrainLoad1$Target,kernel="linear",cost=string[1],epsilon=string[2])
    newdata1 =
data.frame(Val1=TrainLoad1$ValT1,Val2=TrainLoad1$ValT12,Val3=TrainLoad1$ValT13)
    ForeSVM1 <- predict(model1,newdata=newdata1)
    error <- TrainLoad1$Target - ForeSVM1
    returnVal1=sqrt(mean(error^2))
  } else {
    stop("Expecting a chromosome of length 2!");
  }
  returnVal1
}
rbga.results = rbga(c(1, 0.01), c(64, 1),
                    evalFunc=evaluate1, verbose=TRUE, mutationChance=0.01)
cat(summary(rbga.results))
# End of Finding optimal parameter

```

```

#create optimal GASVR model and find error
Trainload1<-read.csv(file="d:/CodeR/Data/TrainingMEA.csv", header=TRUE, sep=",")
InV1<-data.frame(Trainload1$ValT1,Trainload1$ValT12,Trainload1$ValT13)
model1 <- svm(x=InV1,y=Trainload1$Target,kernel="linear",cost=28.8081517037936,
epsilon=0.0372441845766734)
TestLoad1<-read.csv(file="d:/CodeR/Data/TestMEA.csv", header=TRUE, sep=",")
newdata1 =
data.frame(Val1=TestLoad1$ValT1,Val2=TestLoad1$ValT12,Val3=TestLoad1$ValT13)
ForeSVM1 <- predict(model1,newdata=newdata1)
error <- Testload1$ActualValue - ForeSVM1
err2 <- (Testload1$ActualValue - ForeSVM1)/Testload1$ActualValue
SSE_GASVR1 <- sum(error^2)
SSE_GASVR1
MAE_GASVR1 <- mean(abs(error))
MAE_GASVR1
RMSE_GASVR1 <- sqrt(mean(error^2))
RMSE_GASVR1
MAPE_GASVR1<-mean(abs(100*err2))
MAPE_GASVR1
# End of optimal GASVR model
# End of MEA
# PEA
# Find optimal parameter by GA
TrainLoad2<-read.csv(file="d:/CodeR/Data/TrainingPEA.csv", header=TRUE, sep=",")
InV2<-
data.frame(TrainLoad2$ValT1,TrainLoad2$ValT12,TrainLoad2$ValT13,TrainLoad2$ValT24,
TrainLoad2$ValT25)
evaluate2 <- function(string=c()) {
  returnVal = NA;
  if (length(string) == 2) {

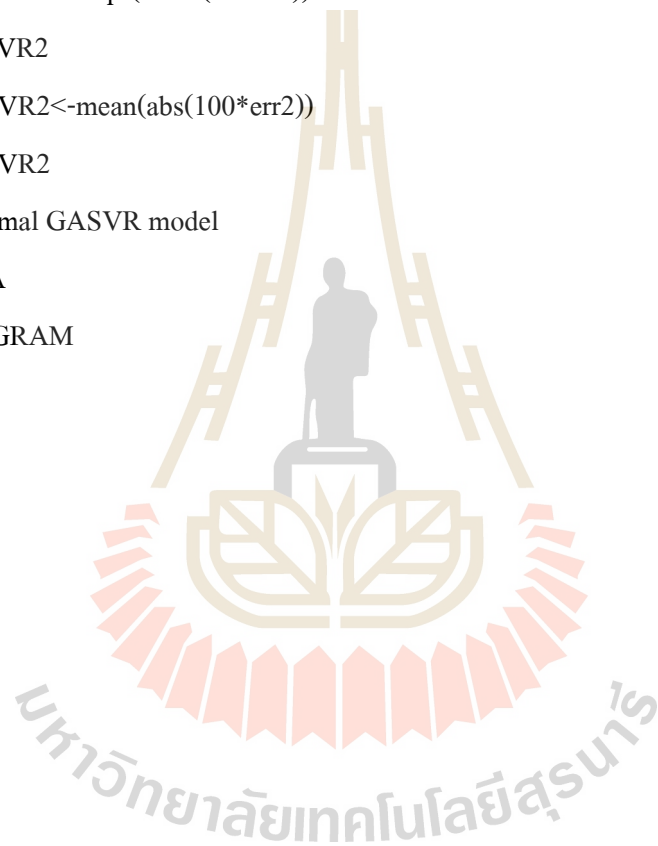
```

```

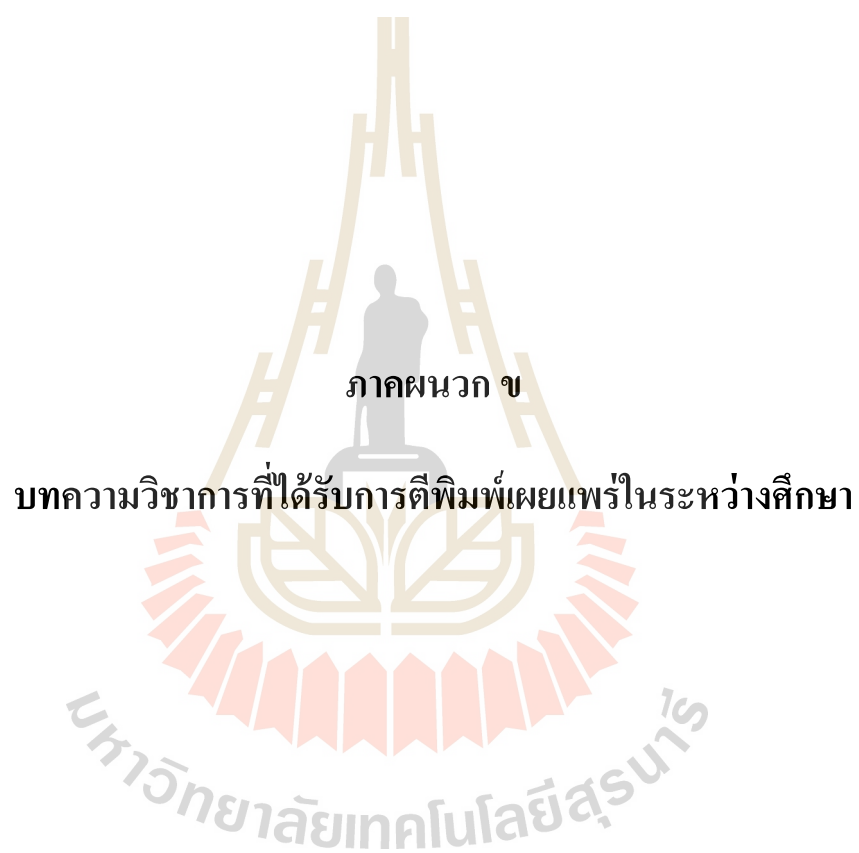
model2 <-
svm(x=InV2,y=TrainLoad2$Target,kernel="linear",cost=string[1],epsilon=string[2])
  newdata2 =
data.frame(Val1=TrainLoad2$ValT1,Val2=TrainLoad2$ValT12,Val3=TrainLoad2$ValT13,
Val4=TrainLoad2$ValT24,Val5=TrainLoad2$ValT25)
ForeSVM2 <- predict(model2,newdata=newdata2)
  error <- TrainLoad2$Target - ForeSVM2
  returnVal2=sqrt(mean(error^2))
} else {
  stop("Expecting a chromosome of length 2!");
}
returnVal2
}
rbga.results = rbga(c(1, 0.001), c(64, 0.1),
  evalFunc=evaluate2, verbose=TRUE, mutationChance=0.01)
cat(summary(rbga.results))
# End of Finding optimal parameter
#create optimal GASVR model and find error
Trainload2<-read.csv(file="d:/CodeR/Data/TrainingPEA.csv", header=TRUE, sep=",")
InV2<-
data.frame(Trainload2$ValT1,Trainload2$ValT12,Trainload2$ValT13,Trainload2$ValT24,Trainload2$ValT25)
model2 <-
svm(x=InV2,y=Trainload2$Target,kernel="linear",cost=60.4265839820728,epsilon=0.0998526274086908)
Testload2<-read.csv(file="d:/CodeR/Data/TestPEA.csv", header=TRUE, sep=",")
newdata2 =
data.frame(Val1=Testload2$ValT1,Val2=Testload2$ValT12,Val3=Testload2$ValT13,Val4=Testload2$ValT24,Val5=Testload2$ValT25)
ForeSVM2 <- predict(model2,newdata=newdata2)

```

```
error <- Testload2$ActualValue - ForeSVM2
err2 <- (Testload2$ActualValue - ForeSVM2)/Testload2$ActualValue
SSE_GASVR2 <- sum(error^2)
SSE_GASVR2
MAE_GASVR2 <- mean(abs(error))
MAE_GASVR2
RMSE_GASVR2 <- sqrt(mean(error^2))
RMSE_GASVR2
MAPE_GASVR2<-mean(abs(100*err2))
MAPE_GASVR2
# End of optimal GASVR model
# End of PEA
# END PROGRAM
```







ภาคผนวก ข

บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

## รายชื่อบทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

Ronnachai Chuentawat, Supaporn Bunrit, Chanintorn Ruangudomsakul, Nittaya Kerdprasop, and

Kittisak Kerdprasop (2016). **Artificial Neural Networks and Time Series Models for**

**Electrical Load Analysis.** The International MultiConference of Engineers and

Computer Scientists 2016, IMECS 2016, March 16 - 18, 2016, Hong Kong.

รณชัย ชื่นธวัช, ชนินทร เรืองอุดมสกุล, นิตยา เกิดประสพ, และกิตติศักดิ์ เกิดประสพ. 2559.

การวิเคราะห์ห่อหุ้มเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาตัวแบบการพยากรณ์  
ที่เหมาะสม ด้วยภาษาอาร์. วารสาร มทร.อีสาน. ปีที่ 9 ฉบับที่ 3 (กันยายน-ธันวาคม 2559):

26-43.

รณชัย ชื่นธวัช, กิตติศักดิ์ เกิดประสพ, และนิตยา เกิดประสพ. 2560.

การพยากรณ์ความต้องการใช้งานหน่วยจำหน่ายไฟฟ้าด้วยซัพพอร์ตเวกเตอร์รีเกรสชัน  
แบบตรวจสอบสลับ 3 ส่วน. วารสารวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยอุบลราชธานี.  
ปีที่ 19 ฉบับที่ 3 (กันยายน-ธันวาคม 2560).

Ronnachai Chuentawat, Kittisak Kerdprasop and Nittaya Kerdprasop (2016). **THE FORECAST**

**OF ELECTRICAL POWER DISTRIBUTION UNIT USING SUPPORT VECTOR**

**REGRESSION OPTIMIZED WITH GENETIC ALGORITHM.**

Suranaree Journal of Science and Technology.

# Artificial Neural Networks and Time Series Models for Electrical Load Analysis

Ronnachai Chuentawat\*, Supaporn Bunrit, Chanintorn Ruangudomsakul, Nittaya Kerdprasop, and Kittisak Kerdprasop

**Abstract**—We applied traditional time series analysis and artificial neural network (ANN) techniques to model and forecast the power consumption of Bangkok's metropolitan area. Time series data in terms of units of household electricity usage were obtained from the Metropolitan Electricity Authority of Thailand. The data had been collected monthly from January 2010 to May 2015. Forecasting models with different parameters are generated from both techniques using the training data, which are the series from January 2010 to December 2014. The remaining data from January 2015 to May 2015 are employed as the testing data. Forecasting performance of each model is measured by the rooted mean square error (RMSE) and the mean absolute percentage error (MAPE) metrics. The traditional time series forecasting models studied in this research are GLM, HoltWinters, and ARIMA. For ANN, we examine four models using 3 layers with different number of neurons ranging from 4 to 7: 3L-4N, 3L-5N, 3L-6N, and 3L-7N. The experimental results reveal that ARIMA is superior among the traditional time series models. For the intelligent based models, 3L-6N is the best of ANN models. Moreover, the MAPE metric of the 3L-6N model is less than the ARIMA model. As a result, we can conclude that ANN model is more powerful in forecasting power distribution units than the traditional time series models.

**Index Terms**—Artificial neural networks, forecasting, time series model, electrical load analysis

## I. INTRODUCTION

THIS research aims to perform time series analysis to forecast future data by using electricity supply information of Metropolitan Electricity Authority of Thailand. It is a monthly reported data from January 2010 to May 2015. The data are presented in unit metric in which 1 unit refers to 1000 kilowatts per hour. These data contain the time series pattern.

Manuscript December 26, 2015; revised January 18, 2016. This work has been supported in part by grants from Suranaree University of Technology through the funding of the Data Engineering Research Unit.

R. Chuentawat is a doctoral student with the School of Computer Engineering, Suranaree University of Technology, and also an assistant professor at Nakhon Ratchasima Rajabhat University, Nakhon Ratchasima 30000, Thailand (e-mail: c\_ronnachai@hotmail.com).

S. Bunrit is a doctoral student with the School of Computer Engineering, Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand (e-mail: sbunrit@sut.ac.th).

C. Ruangudomsakul is a doctoral student with the School of Computer Engineering, Suranaree University of Technology, and also a lecturer at Sisaket Rajabhat University, Thailand (e-mail: chanintorn.r@ssku.ac.th).

N. Kerdprasop is an associate professor with the School of Computer Engineering, Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand (e-mail: nittaya@sut.ac.th).

K. Kerdprasop is an associate professor with the School of Computer Engineering, Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand (e-mail: kerdpras@sut.ac.th).

An accurate forecast on the demand of electricity provides the advantageous information in resource planning, management of funding, and reducing the operation cost. Bunn and Farmer [2] found and reported that 1% of forecasting error raised by 10 million units of the operation costs. Therefore, the precision of forecast is a challenging problem, especially on the demand of electricity supply.

Traditional time series forecasting technique generates the forecasting model from the training data set by determining the unrecognized parameters. In our study, simple linear regression analysis, triple exponential smoothing and autoregressive integrated moving average (ARIMA) of Box and Jenkins method are applied. The acquired forecasting models from the three methods are then compared the accuracy using the two metrics: rooted mean square error (RMSE) and mean absolute percentage error (MAPE). The lower RMSE and MAPE, the more accurate the model is.

Artificial neural network (ANN) is a machine learning technique that mimics the concept of neuron operation in human brain. It can be used to forecast time series data by setting the output neuron to provide the observed value at time  $t$  ( $y_t$ ). Input is the observed value of lag time from 1 to  $p$  period prior to the time  $t$  ( $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ ). In this study, we used three-layer feed-forward back propagation neural network for time series analysis. We compared each model to find the most suitable one with the lowest RMSE and MAPE.

Our research is thus a comparative study of artificial neural networks and traditional time series analysis for forecasting time series data. Research objectives are:

1. Apply three traditional time series analysis methods for generating forecasting model using the R language. Such methods are generalized linear models (GLM), Holt-Winters (HoltWinters) and autoregressive integrated moving average (ARIMA).
2. Apply ANN models for time series analysis by generating forecasting models in Matlab.
3. Find the most suitable forecasting model by comparing RMSE and MAPE resulting from all traditional time series analysis methods and ANN models.

## II. LITERATURE REVIEW

There are a number of related works in literatures about time series analysis using traditional time series methods and ANNs.

Maçaira, Souza and Oliveira [3] studied about how to define forecasting model and forecast yearly electricity consumption of residential units in Brazil until 2050 using

Pegels exponential smoothing forecasting technique. They performed parameter adjustment for forecasting data along with the estimated value that is available in PDE (the ten year energy planning) and PNE (the nation energy planning). The result was that the Pegels exponential smoothing forecasting technique produced the nearest estimation of electrical consumption in both PDE and PNE.

Keka and Hamiti [4] did experiment to find mathematical model representing relationship between electrical energy and time by using linear regression techniques. They used electrical supply data collected from the electrical substation every 15 minutes and set the time interval by using day, week and month. Result from the experiment showed a linear mathematical model that can represent the relationship between electrical energy and time.

Chujai, Kerdprasop and Kerdprasop [5] studied time-series analysis of household electrical consumption using data from UCI Machine Learning Repository. They generated forecast models from ARIMA and ARMA using 4 kinds of time interval (day, week, month, and quarter) and experimented with the R language. They performed model tolerant analysis by measuring the AIC (Akaike Information Criterion) and RMSE. The result showed that the ARIMA model is suitable for forecasting data in the month and quarter interval while the ARMA model is suitable for forecasting data in day and week interval.

Wang and Ming [6] studied about forecasting energy consumption in China using ARIMA, ANN and hybrid ARIMA-ANN model. They compared the accuracy using RMSE, MAE (Mean Absolute Error), and MAPE. The result of the study showed that a hybrid model provided better accuracy in forecasting than either the ARIMA, or ANNs.

Firata, Turanb and Yurdusev [7] studied about forecasting water supply consumption in Turkey using several ANN techniques including generalized regression neural networks (GRNN), cascade correlation neural network (CCNN), and feed forward neural networks (FFNN). They defined 6 network structures and then performed statistical testing by measuring the average absolute relative error (AARE), normalized root mean square error (NRMSE), and threshold statistic (Ts). The result showed that the values of AARE and NRMSE of the M5 models are better than those of the other models and the test statistics of the M5 CCNN model is slightly better than those of GRNN and FFNN models.

### III. METHODOLOGY

#### A. Traditional Time Series Models

In traditional time series forecasting technique, we apply general linear regression, triple exponential smoothing, and Box and Jenkins methods. We then forecast data for the next 5 intervals. After that, we compare forecasted data against real data in a test set. The RMSE and MAPE metrics are used as statistical measure instruments to find the suitable forecasting model. The 3 traditional forecasting techniques (GLM, HoltWinters, and ARIMA) are described as follows:

1. *GLM* model is the forecasting model that are generated by simple linear regression analysis. GLM linear regression [8] is as shown in equation 1.

$$Y = \beta_0 + \beta_1 X + \varepsilon \quad (1)$$

where  $X$  is a time-interval variable in a monthly unit,  
 $Y$  is the amount of monthly power distribution unit,  
 $\varepsilon$  is a residual term.

2. *HoltWinters* model is deriving from triple exponential smoothing. It is a smoothing technique that can be used to generate forecast models from data with trend and seasonal [9]. HoltWinters model can be described as in equations 2-4. To forecast the future event, equation 5 which is composed of the level, trend, and seasonal parts is applied.

$$\text{Level : } L_t = \alpha(Y_t - S_{t-s}) + (1 - \alpha)(L_{t-1} + b_{t-1}) \quad (2)$$

$$\text{Trend : } b_t = \beta(L_t - L_{t-1}) + (1 - \beta)b_{t-1} \quad (3)$$

$$\text{Seasonal : } S_t = \gamma(Y_{t-1} - L_t) + (1 - \gamma)S_{t-s} \quad (4)$$

$$\text{Forecast : } F_{t+m} = L_t + b_{tm} + S_{t-s+m} \quad (5)$$

where  $\alpha$  is constant for level smoothing  
 $\beta$  is constant for trend smoothing  
 $\gamma$  is constant for seasonal smoothing  
 $L_t$  is estimated level of time series at time  $t$   
 $Y_t$  is observed value at time  $t$   
 $b_t$  is estimated slope of time series at time  $t$   
 $S_t$  is seasonal factor  
 $s$  is seasonal length, equal to 12 months ( $s = 12$ )  
 $m$  is forecast intervals  
 $F_{t+m}$  is forecasted value  $m$  intervals

3. *ARIMA* model is a model derived from Box and Jenkins method. General term of ARIMA [10] can be presented with backward shift operator ( $B$ ) as in equation 6.

$$\theta_p(B)\phi_p(B^s)(1-B)^d(1-B^s)^D Y_t = w_q(B)W_Q(B^s)\alpha_t \quad (6)$$

$$\begin{aligned} \text{where } B Y_t &= Y_{t-1} \quad \text{and} \quad B^k Y_t = Y_{t-k} \\ w_q(B) &= 1 - w_1 B - w_2 B^2 - \dots - w_q B^q \\ \theta_p(B) &= 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_p B^p \\ \phi_p(B^s) &= 1 - \phi_1 B^s - \phi_2 B^{2s} - \dots - \phi_p B^{ps} \\ W_Q(B^s) &= 1 - W_1 B^s - W_2 B^{2s} - \dots - W_Q B^{Qs} \end{aligned}$$

In order to generate an ARIMA model, we have to analyze time series data for defining suitable parameters of ARIMA( $p, d, q$ ) $\times$ ( $P, D, Q$ ) $_s$ . That means selecting the suitable  $p, d, q$  from trend and  $P, D, Q$  from seasonal. In R language, we can use function `auto.arima()` to assign suitable ARIMA( $p, d, q$ ) $\times$ ( $P, D, Q$ ) $_s$ . In this research, we use  $(p, d, q) = (1, 0, 0)$  and  $(P, D, Q)_s = (1, 0, 0)_{12}$ , where 1 season = 12 months ( $S=12$ ). Therefore, a suitable ARIMA( $p, d, q$ ) $\times$ ( $P, D, Q$ ) $_s$  is ARIMA(1, 0, 0) $\times$ (1, 0, 0) $_{12}$ .

After obtaining 3 forecasting models from the 3 methods mentioned above, statistical measuring in terms of RMSE and MAPE is compared for model accuracy. A model with the lowest RMSE and MAPE is the most suitable one. Summary of steps in applying traditional time series analysis to find a suitable forecasting model is shown in figure 1.

Step 1: Collect data from secondary source. Data are obtained from the monthly report of the Metropolitan Electricity Authority from January 2010 to May 2015. The data are the household electrical usage of residents in the Metropolitan Electricity Authority area. We then order the data to build up time series.



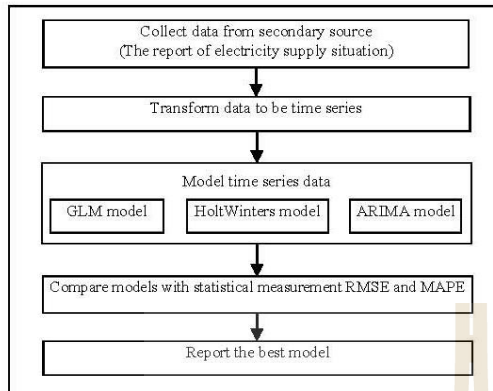


Fig. 1. Steps in selecting the most appropriated model for forecasting time series data

	A	B	C	D	E
1	Years	Months	Electric_Units	No_Consumers	Mean_Units
2	2010	1	776.32	2420767	320.69
3	2010	2	836.16	2423046	345.09
4	2010	3	925.55	2433022	380.41
5	2010	4	1026.5	2439689	420.75
6	2010	5	1082.18	2448223	442.03
7	2010	6	1003.41	2456309	408.5
8	2010	7	930.91	2463398	377.9
9	2010	8	876.83	2473742	354.45

Fig. 2. Time series data are created from secondary data

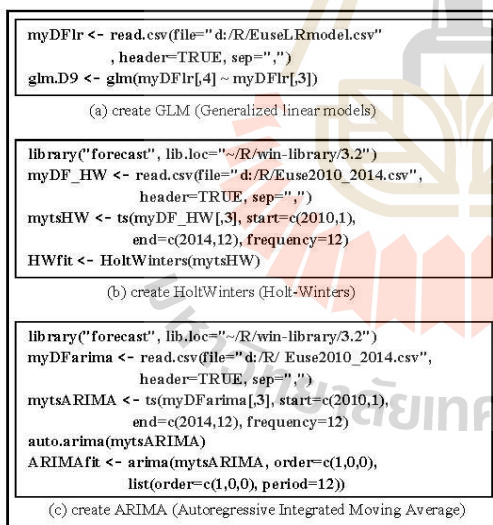


Fig. 3. R commands for generating traditional forecasting model

Step 2: Transform data to be time series. The data from the monthly report of the Metropolitan Electricity Authority is a monthly interval from January 2010 to May 2015 as shown in figure 2. We divide the data into 2 sets. The first is a training set consisting of data from January 2010 to December 2014. It is used for generating the forecasting model. The test set is data from January 2015 to May 2015. Test data are used for validating the model and evaluating by RMSE and MAPE.

Step 3: Model time series data. This step generates 3 different forecasting models from the training set. We build 3 forecasting models from the same training set with commands in R language that can be summarized in figure 3.

Step 4: Compare models with statistical measurements RMSE and MAPE. The computation of RMSE (rooted mean square error) and MAPE (mean absolute percentage error) [11] is shown in equations 7 and 8, respectively.

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \quad (7)$$

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| 100 \frac{y_t - \hat{y}_t}{y_t} \right| \quad (8)$$

where  $y_t$  is observed value at time  $t$ ,  
 $\hat{y}_t$  is forecasted value at time  $t$ , and  
 $n$  is the number of forecasting period.

Step 5: Report the best model. The best forecasting model is the one with the lowest RMSE and MAPE values.

### B. Artificial Neural Network Model

Artificial neural networks (ANNs) are the simulated networks of neurons in the human brain. We can apply ANNs to forecast time series data. ANNs perform learning from existing data by analyzing the correlation between observed values at current time with previous observed values. After getting an ANN model from the training set, we can use such model to forecast the value of new observation in the test set.

In this research, we use ANNs model calls three-layer feed-forward back propagation neural networks which has 1 hidden layer. Output from a model is a forecasting value at current time ( $y_t$ ). Input to a model is the previous observed values at 1 to  $p$  time intervals ( $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ ) and be represented as a vector. The correlation between input and output [6] can be shown as in equation 9.

$$y_t = \omega_0 + \sum_{j=1}^q \omega_j g(\omega_{0j} + \sum_{i=1}^p \omega_{i,j} y_{t-i}) + e_t \quad (9)$$

where  $\omega_j$  ( $j=1, \dots, q$ ) and  $\omega_{i,j}$  ( $i=0, \dots, p; j=1, \dots, q$ ) are model parameters which are called weights,  
 $p$  is the number of neurons in the input layer,  
 $q$  is the number of neurons in the hidden layer and have sigmoid function [6] as a transfer function presented in equation 10.

$$\text{sig}(x) = \frac{1}{(1 + \exp(-x))} \quad (10)$$

From ANNs model shown in equation 9, we can transform to nonlinear function to represent the relationship [6] between previous observed values ( $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ ) and forecasting value ( $y_t$ ) as in equation 11.

$$y_t = f(y_{t-1}, \dots, y_{t-p}, \omega) + e_t \quad (11)$$

where  $\omega$  is a vector of all parameters and  $f$  is a function used for determining the network structure and the connecting of weights. As a result, the neural network model is equivalent to and can be expressed in terms of nonlinear autoregressive model.

In the research on Wang and Meng [6], they suggested that ARIMA and ANNs have something similar. From our ARIMA model, it has  $p=1$  and  $P=1$  which means observation at time  $t$  ( $y_t$ ) relates to the previous observation with a lag of one interval ( $y_{t-1}$ ) and relates to previous observation with 1 season lag or 12

intervals ( $y_{t+2}$ ). So, we can use  $y_{t-1}$  and  $y_{t-2}$  as the input to ANNs model. Therefore, input layer of ANNs model has 2 neurons. In order to investigate the complexity of an ANN causing the over-fitting problem, we experiment with 4 ANNs models defined as follows:

- Model 1: Three layers, two input neurons, one hidden neuron and one output neuron (3L-4N).
- Model 2: Three layers, two input neurons, two hidden neurons and one output neuron (3L-5N).
- Model 3: Three layers, two input neurons, three hidden neurons and one output neuron (3L-6N).
- Model 4: Three layers, two input neurons, four hidden neurons and one output neuron (3L-7N).

Electric Units	$Y_{t-1}$	$Y_{t-2}$	$Y_t$
2	773.99	776.32	707.16
3	707.16	836.16	787.29
4	787.29	925.55	806.64
5	806.64	1026.5	861.58
6	881.58	1082.18	968.86
7	968.86	1003.41	933.59
8	913.59	910.91	912.8
9	512.8	876.83	812.54
10	902.59	873.47	886.08
11	896.08	840.6	830.9
12	830.9	787.33	886.93
13	886.93	773.99	703.06

Fig. 4. Examples of data file for generating ANNs model

```

AllData = xlsread('AllEuse.xlsx');
minD = min(min(AllData))
maxD = max(max(AllData))
DataNor = (AllData - minD)/(maxD-minD)+0.1;
data = xlsread('InputTraining.xlsx');
(a) Normalized data

% For ANN(3L-6N) #This is comment
net=newff(minmax(P),[3 1],{'logsig','purelin'},'trainlm');
net.trainParam.epochs=3000;
(b) Build model and initial configuration

% For ANN(3L-6N) #This is comment
net.iw{1,1}=[0.5 0.4;0.5 0.5;0.5 0.5]
net.lw{2,1}=[0.5 0.5 0.5]
(c) Initial weights and bias

net = train(net,P,T);
(d) Train model

DataTest = [DataNor(60) DataNor(49);
            DataNor(61) DataNor(50);
            DataNor(62) DataNor(51);
            DataNor(63) DataNor(52);
            DataNor(64) DataNor(53)];
(e) Predict new observation value
  
```

Fig. 5. Command sets in Matlab for generating ANNs model

We then implement our ANNs structures using Matlab software and the steps can be described as follows:

**Step 1.** Create 3 data files with Excel (figure 4) for generating models. The first file is *AllEuse*; it contains a power distribution unit data from January 2010 to May 2015. The second file is *InputTraining*; it contains a training set as input data consisting of  $y_{t-1}$  and  $y_{t-2}$ . The third file is *TargetTraining*; it contains an output data of the training data set  $y_t$ .

**Step 2.** Perform data normalization in *InputTraining* and *TargetTraining* to adjust the data values into the interval of 0 and 1 using a command set in figure 5(a).

**Step 3.** Create ANNs by using *newff()* function and initial configuration using a command set in figure 5(b).

**Step 4.** Define initial weights and bias as random constants using a command set in figure 5(c).

**Step 5.** Perform training model by using normalized data ( $P$  and  $T$ ) with a command set in figure 5(d).

**Step 6.** Forecast 5 intervals of power distribution units from January 2015 to May 2015. Then calculate RMSE and MAPE by comparing forecasted data against the real data using a command set in figure 5(e).

#### IV. EXPERIMENTAL EVALUATION

We aim at forecasting power distribution units of household electrical usage of residents in the Metropolitan Electricity Authority area. We apply 2 different techniques: traditional time series forecasting technique by using R language and time series forecasting with ANNs by using Matlab. We, therefore, branch the results by experiments and their comparison into 3 sections.

##### A. Results of Traditional Time Series Models

We generate forecasting models by using 3 traditional time series forecasting models, then use all 3 models to forecast power distribution unit in the next 5 intervals from January 2015 to May 2015. After that, we take the predicted values to calculate RMSE and MAPE by comparing with actual values using the R command set in figure 6. The result of experiments is shown in table 1.

```

foreGLM <- predict(glm.D9, n.ahead=5)
(a) Forecasting command set of GLM model

foreHW <- predict(HWfit,newdata=testData_ByMonth,n.ahead=5)
(b) Forecasting command set of HoltWinters model

foreARIMA <- predict(ARIMAFit,newdata=testData_ByMonth,
                    n.ahead=5)
(c) Forecasting command set of ARIMA model
  
```

Fig. 6. Forecasting command set in R language for traditional time series model

TABLE I  
PREDICTED VALUE AND ERROR FOR TRADITIONAL TIME SERIES MODELS

Value	Jan	Feb	Mar	Apr	May	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
GLM	986.13	988.18	990.24	992.3	994.36	166.07	15.78
HoltWinters	841.78	928.33	1000.99	1066.63	1086.62	81.48	7.42
ARIMA	799.24	859.33	969.4	1065.12	1117.63	57.18	4.80



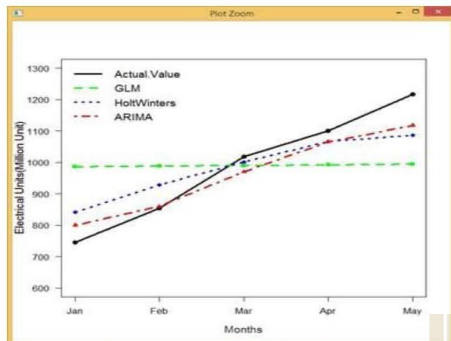


Fig. 7. Comparison graph between 5 intervals of actual value and predicted value

```

> auto.arima(myts53_57)
Series: myts53_57
ARIMA(1,0,0)(1,0,0)[12] with non-zero mean

Coefficients:
          ar1      sar1  intercept
          0.7301   0.6403   929.9639
s.e.        0.0835   0.0984   57.2129

sigma^2 estimated as 3320:  log likelihood=-331.93
AIC=671.85  AICC=672.58  BIC=680.23
    
```

Fig. 8. Function auto.arima() for defining format of ARIMA model

From table 1, the lowest RMSE and MAPE are obtained from ARIMA model. RMSE is 57.18 and MAPE is 4.80. Figure 7 presents a linear plot of actual against predicted values that are obtained from all 3 models. A graph of ARIMA shows the smallest difference between the actual and predicted values. This result is obtained from using auto.arima() for defining ARIMA model as shown in figure 8.

The forecasting ARIMA model consists of autoregressive(AR) in which  $p = 1$  and seasonal autoregressive(SAR) with  $P = 1$ . When estimating the coefficient of  $AR_1 (\hat{\theta}_1)$  and  $SAR_1 (\hat{\Theta}_1)$ , we get the values of 0.7301 and 0.6403, respectively, and the parameters  $d = D = 0$  and  $q = Q = 0$ . The forecasting equations are presented as follows:

$$\begin{aligned}
 (1 - \hat{\theta}_1 B)(1 - \hat{\Theta}_1 B^{12})Y_t &= \theta_0, \quad \text{while } \theta_0 \text{ is constant} \\
 (1 - \hat{\theta}_1 B - \hat{\Theta}_1 B^{12} + \hat{\theta}_1 \hat{\Theta}_1 B^{13})Y_t &= \theta_0 \\
 Y_t - \hat{\theta}_1 Y_{t-1} - \hat{\Theta}_1 Y_{t-12} + \hat{\theta}_1 \hat{\Theta}_1 Y_{t-13} &= \theta_0 \\
 Y_t &= \theta_0 + \hat{\theta}_1 Y_{t-1} + \hat{\Theta}_1 Y_{t-12} - \hat{\theta}_1 \hat{\Theta}_1 Y_{t-13} \\
 Y_t &= \theta_0 + 0.73Y_{t-1} + 0.64Y_{t-12} - 0.47Y_{t-13}
 \end{aligned}$$

From existing observed values of time series,  $\theta_0 = 92.42$  and the forecasting equation of ARIMA can be present as equation 12.

$$Y_t = 92.42 + 0.73Y_{t-1} + 0.64Y_{t-12} - 0.47Y_{t-13} \quad (12)$$

**B. Results of Artificial Neural Networks Model**

We define 4 different ANN models: 3L-4N (one neuron in hidden layer), 3L-5N (two neurons in hidden layer), 3L-6N (three neurons in hidden layer), and 3L-7N (four neurons in hidden layer).

From our experiment, we discover that we should not set the goal an error parameter (net.trainParam.goal) too low because it may cause over-fitting. Thus, we set goal error equal to 0.0001. In addition, when we set the initial weight and bias of each

model appropriately, RMSE and MAPE of each model are decreased. As a result, the lowest RMSE and MAPE of each model have been observed after experimentally set the most appropriate parameter values. These appropriate initial weights and bias of each model can be summarized in the table 2.

While using initial weight and bias of each model in table 2 to perform model training, then predict next 5 interval of power distribution unit from January 2015 to May 2015. After that, calculate RMSE and MAPE comparing with actual value. We summarize the result in table 3.

From table 3, we can see that 3L-6N model resulted in the lowest MAPE and table 4 shows that 3L-6N has the highest correlation coefficient. That means the 3L-6N model is highly related to the target. Figure 9 presents a regression graph between the output and the target of a 3L-6N model. When we increase 1 number of neuron, from 3L-6N to 3L-7N, the RMSE and MAPE increase. Therefore, the 3L-6N model is the most appropriate model for predicting this time series. When we use command set in Matlab to show the weight values of each layer, we can define the structure of ANNs as shown in figure 10.

TABLE II  
INITIAL WEIGHTS AND BIAS OF EACH MODEL

Model	net.iw{1,1}	net.lw{2,1}	net.b{1}	net.b{2}
3L-4N	[0.5 0.5]	[0.5]	[0.5]	[0.5]
3L-5N	[0.5 0.5; 0.1 0.1]	[0.4 0.4]	[0.5 0.5]	[0.5]
3L-6N	[0.5 0.4; 0.5 0.5; 0.5 0.5]	[0.5 0.5 0.5]	[0.5 0.5 0.5]	[0.5]
3L-7N	[0.5 0.4; 0.5 0.5; 0.5 0.5; 0.5 0.5]	[0.5 0.5 0.5 0.5]	[0.5 0.5 0.5 0.5]	[0.5]

TABLE III  
ACTUAL VERSUS PREDICTED VALUES OF ANNS MODELS

	Jan	Feb	Mar	Apr	May	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
3L-4N	786.94	789.21	911.85	1096.59	1190.53	59.67	5.21
3L-5N	785.26	809.44	905.99	1123.58	1193.65	58.47	5.08
3L-6N	753.33	816.63	890.15	1148.87	1207.41	63.46	4.58
3L-7N	759.61	760.58	913.55	1089.92	1184.74	64.66	5.33

TABLE IV  
CORRELATION COEFFICIENT AND EQUATION OF ANNS

Model	Correlation coefficient (R)	Relative equation of output and target
3L-4N	0.84323	Output=0.71*Target+0.16
3L-5N	0.87184	Output=0.76*Target+0.14
3L-6N	0.88868	Output=0.79*Target+0.12
3L-7N	0.84950	Output=0.72*Target+0.16

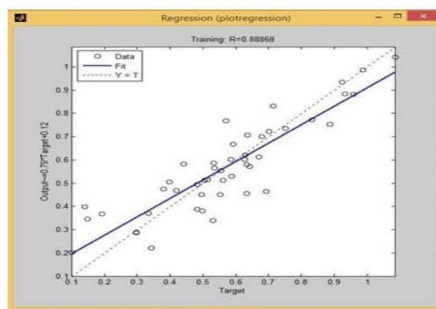


Fig. 9. Regression graph of 3L-6N model



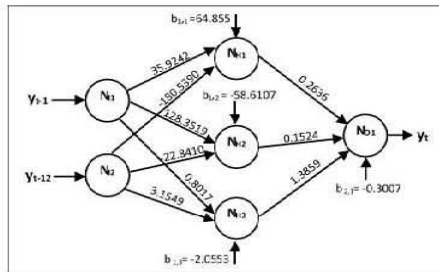


Fig. 10. Artificial neural networks for 3L-6N model

C. Comparison Between ARIMA and 3L-6N ANN Models

From the literature review [5],[6], the authors suggest that ARIMA is the accurate model to forecast data with moving by trend and season. We thus compare ARIMA with tripple exponential smoothing technique using HoltWinters model and simple linear regression using GLM model. From our experiment, we discover that the ARIMA model gives the lowest error (RMSE = 57.18 and MAPE = 4.80).

To perform time series analysis with ANNs technique, we have found that the 3L-6N model gives the lowest MAPE (=4.58) and also the highest correlation coefficient. That means the output of a 3L-6N model relates the most to the target.

When plot 5 intervals from January 2015 to May 2015 of actual observation values compared to data predicted by ARIMA and 3L-6N models (table 5 and figure 11), it can be concluded that ARIMA's graph is more similar to the actual observed values than 3L-6N's graph but a 3L-6N model can adjust irregular error in March to normal error rapidly. As a result, MAPE of 3L-6N is lower than ARIMA but RMSE is higher than ARIMA.

TABLE V  
ACTUAL VS PREDICATED VALUES OF ARIMA AND 3L-6N ANN MODELS

Value	Jan	Feb	Mar	Apr	May	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
ARIMA	799.24	859.33	969.4	1065.12	1117.63	57.18	4.80
3L-6N	753.33	816.63	890.15	1148.87	1207.41	63.46	4.58

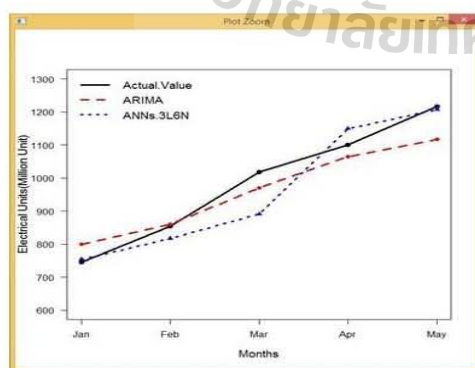


Fig. 11. Comparison graph between ARIMA and ANNs 3L-6N

V. CONCLUSIONS

We study time series analysis for forecasting power distribution unit using monthly data reported by the Metropolitan Electricity Authority from January 2010 to May 2015. We use data from January 2010 to December 2014 as a training set and data from January 2015 to May 2015 as the test set. In our experiment for model comparison, we select 3 traditional time series models (GLM, HoltWinters, ARIMA) and 4 ANNs models (3L-4N, 3L-5N, 3L-6N, 3L-7N).

In traditional time series analysis, ARIMA model is the most accurate model (RMSE = 57.18 and MAPE = 4.80). In ANNs technique, we discover that 3L-6N model (three-layer feed-forward back propagation neural network) is the most appropriate model (MAPE = 4.58, correlation coefficient = 0.88868). When compare ARIMA against the 3L-6N model, it turns out that ARIMA's graph is more similar to the actual observed values than the 3L-6N's graph, but MAPE of 3L-6N is lower than ARIMA's MAPE. As a result, predicted values from 3L-6N model are more accurate than the ARIMA model.

In conclusion, the ANNs model of the 3L-6N structure can be applied to time series forecasting at high precision. The caution is that the high number of hidden layer's neurons may cause over-fitting.

REFERENCES

- [1] Metropolitan Electricity Authority (2015). Report of electrical distribution [Online]. Available : <http://www.mea.or.th/download/index.php#>
- [2] D.W. Bunn and E.D. Farmer, *Comparative Models for Electrical Load Forecasting*, New York: John Wiley and Sons, 1985.
- [3] P. M. Maçaira, R. C. Souza and F. L. C. Oliveira, "Modelling and Forecasting the Residential Electricity Consumption in Brazil with Pegels Exponential Smoothing Techniques", *Procedia Computer Science*, vol.55, pp. 328-335, 2015.
- [4] I. Keka and M. Hamiti, "Load profile analyses using R language" in *Proceedings of the ITI 2013 35th Int. Conf. on International Technology Interfaces*, Cavtat, Croatia, 2013, pp. 245-250.
- [5] P. Chujai, N. Kerdprasop and K. Kerdprasop, "Time series analysis of household electric consumption with ARIMA and ARMA models" in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, Hong Kong, China, 2013, pp. 295-300.
- [6] X. Wang and M. Meng, "A Hybrid Neural Network and ARIMA Model for Energy Consumption Forecasting" *JOURNAL OF COMPUTERS*, vol.7, No.5, pp. 1184-1190, 2012.
- [7] M. Firata, M.E. Turanb and M.A. Yurdusev, "Comparative analysis of neural network techniques for predicting water consumption time series" *Journal of Hydrology*, vol.384, pp. 46-51, 2010.
- [8] I. Boldina and P.G. Beninger, "Strengthening statistical usage in marine ecology: Linear regression" *Journal of Experimental Marine Biology and Ecology*, vol. 474, pp. 81-91, 2016.
- [9] V. Prema, and K.U. Rao, "Development of statistical time series models for solar power prediction" *Renewable Energy*, vol. 83, pp. 100-109, 2015.
- [10] Y. Wang, J. Wang, G. Zhao and Y. Dong, "Application of residual modification approach in seasonal ARIMA for electricity demand forecasting: A case study of China" *Energy Policy*, vol. 48, pp. 284-294, 2012.
- [11] C. Bergneir and J. M. Berítez, "On the use of cross-validation for time series predictor evaluation" *Information Sciences*, vol.191, pp. 192-213, 2012.



หนังสือรับรองการตีพิมพ์บทความ  
วารสาร มทร.อีสาน

ขอรับรองว่าบทความวิจัย

เรื่อง การวิเคราะห์หอนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาตัวแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาอาร์

โดย ผู้ช่วยศาสตราจารย์รณชัย ชื่นธวัช นายชนินทร์ เรืองอุดมสกุล  
รองศาสตราจารย์ ดร.นิตยา เกิดประสพ และรองศาสตราจารย์ ดร.กิตติศักดิ์ เกิดประสพ

ได้ผ่านการประเมินจากคณะกรรมการผู้ทรงคุณวุฒิ  
และตีพิมพ์ใน วารสาร มทร.อีสาน  
ปีที่ ๙ ฉบับที่ ๓ (กันยายน - ธันวาคม พ.ศ. ๒๕๕๙)

  
(ผู้ช่วยศาสตราจารย์ ดร.พิเชฐ สุวิทยารักษ์)  
บรรณาธิการวารสาร มทร.อีสาน  
มหาวิทยาลัยเทคโนโลยีสุรนารี

มหาวิทยาลัยเทคโนโลยีสุรนารี

## การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาตัวแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาอาร์

### Time Series Analysis of Electrical Distribution Units to Find a Suitable Forecasting Model with R Language

รมชัย ชื่นรัช<sup>1\*</sup> ชนินทร เรืองอุดมสกุล<sup>1</sup> นิตยา เกิดประสพ<sup>1</sup> และกิตติศักดิ์ เกิดประสพ<sup>1</sup>

Received: November, 2015; Accepted: May, 2016

#### บทคัดย่อ

การวิจัยนี้มีวัตถุประสงค์เพื่อวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า และค้นหาตัวแบบการพยากรณ์ที่เหมาะสม โดยใช้การเขียนโปรแกรมด้วยภาษาอาร์ เป็นเครื่องมือดำเนินการวิจัย ซึ่งอนุกรมเวลาที่นำมาวิเคราะห์ เป็นข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง (กฟน.) เก็บรวบรวมข้อมูลจากรายงานสถานการณ์การจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง ตั้งแต่เดือนมกราคม ปี พ.ศ. 2553 ถึงเดือนพฤษภาคม ปี พ.ศ. 2558 ซึ่งข้อมูลที่น่าสนใจมีขอบเขตเฉพาะจำนวนหน่วยจำหน่ายไฟฟ้าให้กับผู้บริโภคประเภทบ้านอยู่อาศัย โดยนำข้อมูลตั้งแต่เดือนมกราคม ปี พ.ศ. 2553 ถึงเดือนธันวาคม ปี พ.ศ. 2557 ไปสร้างตัวแบบการพยากรณ์ที่แตกต่างกันทั้งสิ้น 3 ตัวแบบ ด้วยภาษาอาร์ ได้แก่ ตัวแบบ glm, HoltWinters และ ARIMA จากนั้นเปรียบเทียบตัวแบบทั้ง 3 ด้วยการวัดค่าคลาดเคลื่อน ได้แก่ ค่า RMSE และค่า MAPE จากข้อมูลที่แต่ละตัวแบบพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าของเดือนมกราคมถึงเดือนพฤษภาคม ปี พ.ศ. 2558 เปรียบเทียบกับข้อมูลจริงที่ได้จากรายงานสถานการณ์การจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง ในเดือนมกราคมถึงเดือนพฤษภาคม ปี พ.ศ. 2558 ซึ่งผลการทดลอง สรุปได้ว่าตัวแบบ ARIMA เป็นตัวแบบที่ให้ค่า RMSE และค่า MAPE ต่ำที่สุด ดังนั้น ตัวแบบ ARIMA เป็นตัวแบบที่เหมาะสมที่สุดสำหรับนำไปสร้างเป็นตัวแบบการพยากรณ์ในรูปแบบของสมการคณิตศาสตร์ เพื่อพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวงในส่วนของผู้บริโภคประเภทบ้านอยู่อาศัย

คำสำคัญ : ข้อมูลอนุกรมเวลา; หน่วยจำหน่ายไฟฟ้า; การพยากรณ์; ภาษาอาร์

<sup>1</sup> สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี นครราชสีมา

\* Corresponding E-mail : c\_ronnachai@hotmail.com



## Abstract

This research aimed to analyze time series of electrical distribution unit data and to find a suitable forecasting model by using R language programming as a tool to carry out this research. The analyzed time series were the electrical distribution unit data of Metropolitan Electricity Authority (MEA) which had been collected from report of electrical distribution since January 2010 to May 2015. Residential consumers were the interested specific data. The electrical distribution units as the training data set were the data from January 2010 to December 2014. This data set were modeled by 3 different forecasting models with R language programming, i.e. glm model, HoltWinters model and ARIMA model. Then the comparison among the 3 models was done by comparing the model forecasting against the actual data of MEA during January - May, 2015. From the comparative results, it was concluded that the ARIMA model is the most suitable one for creating a forecasting model in term of mathematic equation to predict the electrical distribution units of MEA for the section of residential consumers.

**Keywords:** Time Series Data; Electrical Distribution Units; Forecasting; R Language

## บทนำ

การวิเคราะห์อนุกรมเวลาเป็นวิธีการพยากรณ์เชิงปริมาณ หมายถึง การใช้ข้อมูลเชิงปริมาณ ที่มีการเก็บรวบรวมอย่างเป็นระบบ และต่อเนื่องในอดีต มาวิเคราะห์ด้วยวิธีการเชิงสถิติ เพื่อพยากรณ์ข้อมูลที่เกิดขึ้นในอนาคต ส่วนข้อมูลอนุกรมเวลา หมายถึง ข้อมูลที่เก็บบันทึกตามลำดับเวลา และการบันทึกข้อมูลอาจจะทำเป็นรายวัน รายสัปดาห์ รายเดือน รายไตรมาส หรือรายปี ขึ้นอยู่กับความเหมาะสมของการนำข้อมูลไปใช้ โดยข้อมูลอนุกรมเวลาประกอบด้วย 4 ส่วนประกอบ คือ แนวโน้ม ฤดูกาล รอบเวลา และความผิดปกติ ซึ่งการวิเคราะห์ข้อมูลอนุกรมเวลาโดยใช้เทคนิคการพยากรณ์จะทำการกำหนดตัวแบบการพยากรณ์จากข้อมูลในอดีตด้วย สมมติฐานว่าเหตุการณ์ที่เกิดขึ้นในอนาคต มีรูปแบบเหมือนกับเหตุการณ์ที่เกิดขึ้นในอดีต และงานวิจัยนี้ข้อมูลที่น่ามาพยากรณ์เป็นข้อมูลการจำหน่ายไฟฟ้าของการไฟฟ้านครหลวงที่จำหน่ายให้กับผู้บริโภคที่อยู่ในเขตพื้นที่กรุงเทพมหานคร สมุทรปราการ และนนทบุรี ประเภทบ้านอยู่อาศัย โดยอาศัยรายงานสถานการณ์การจำหน่ายไฟฟ้าประจำเดือนตั้งแต่เดือนมกราคม ปี พ.ศ. 2553 ถึงเดือนพฤษภาคม ปี พ.ศ. 2558 (Metropolitan Electricity Authority, 2015) และข้อมูลที่สังเกต คือ หน่วยจำหน่ายไฟฟ้า โดย 1 หน่วยจำหน่าย หมายถึง การใช้ไฟฟ้า 1,000 วัตต์ต่อชั่วโมง ซึ่งข้อมูลหน่วยจำหน่ายไฟฟ้า เป็นข้อมูลที่สะท้อนถึงความต้องการใช้ไฟฟ้าของผู้บริโภค มีลักษณะเป็นอนุกรมเวลาที่บันทึกข้อมูลเป็นรายเดือน ตามลำดับเวลา และการพยากรณ์ค่าความต้องการไฟฟ้าของผู้บริโภคช่วยวางแผนด้านทรัพยากร จัดสรรงบประมาณ และลดค่าใช้จ่ายในการดำเนินการลงได้

## 28 การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาแบบการพยากรณ์ที่เหมาะสม ศิวดาชาฮาร์

สอดคล้องกับงานวิจัยของ Bunn, D.W. and Farmer, E.D. (Bunn, D.W. and Farmer, E.D., 1985) ได้ศึกษาการพยากรณ์ความต้องการไฟฟ้า พบว่า ค่าใช้จ่ายการดำเนินการจะเพิ่มขึ้น 10 ล้านปอนด์ทุกปี เมื่อมีการพยากรณ์ผิดพลาดไปทุก 1% กล่าวได้ว่า การพยากรณ์ความต้องการของหน่วยจำหน่ายไฟฟ้า มีความสำคัญอย่างยิ่งต่อการวางแผนการผลิตกระแสไฟฟ้า และทำให้การดำเนินการผลิตกระแสไฟฟ้า มีประสิทธิภาพมากยิ่งขึ้น

สำหรับวิธีการวิเคราะห์อนุกรมเวลามีหลากหลายวิธี แต่ละวิธีจะดำเนินการสร้างตัวแบบการพยากรณ์ ที่ให้ค่าความแม่นยำของการพยากรณ์แตกต่างกัน โดยสามารถวัดค่าความแม่นยำด้วยการวัดค่า ความคลาดเคลื่อนทางสถิติ เช่น ค่า SSE (Sum Square Error) ค่า RMSE (Root Mean Square Error) ค่า MAE (Mean Absolute Error) หรือค่า MAPE (Mean Absolute Percentage Error) ซึ่งค่าสถิติดังกล่าว เป็นการวัดค่าความแม่นยำด้วยการเปรียบเทียบค่าพยากรณ์กับค่าสังเกตจริงในช่วงเวลาเดียวกัน ว่ามีค่าแตกต่างกันมากน้อยเพียงใด ดังนั้น ค่าสถิติดังกล่าว สามารถนำมาใช้เปรียบเทียบ ตัวแบบการพยากรณ์ที่ถูกสร้างด้วยวิธีการพยากรณ์ที่แตกต่างกันได้ และวิธีการวิเคราะห์อนุกรมเวลา เพื่อสร้างตัวแบบการพยากรณ์ที่สำคัญ มีดังนี้

1. วิธีการวิเคราะห์การถดถอย (Regression Analysis) เป็นการศึกษาความสัมพันธ์ระหว่าง ปัจจัยหรือตัวแปรตั้งแต่ 2 ตัวแปรขึ้นไป โดยเป็นความสัมพันธ์ของตัวแปรอิสระ ที่มีค่าของตัวแปรเกิดขึ้น อย่างอิสระ และตัวแปรอิสระมีตั้งแต่ 1 ตัวแปรขึ้นไป กับตัวแปรตาม 1 ตัวแปร ซึ่งค่าของตัวแปรตาม ขึ้นอยู่กับรูปแบบความสัมพันธ์กับตัวแปรอิสระ และวัตถุประสงค์ของการวิเคราะห์การถดถอย คือ เพื่อทำให้ทราบถึงลักษณะของความสัมพันธ์ ขนาดของความสัมพันธ์ และทิศทางของความสัมพันธ์ ระหว่างตัวแปรอิสระกับตัวแปรตาม

2. วิธีการปรับให้เรียบแบบเอ็กซ์โปเนนเชียล (Exponential Smoothing) เป็นวิธีการ พยากรณ์ที่มีประสิทธิภาพ เมื่อองค์ประกอบของอนุกรมเวลา ได้แก่ แนวโน้ม และฤดูกาล ที่มีการ เปลี่ยนแปลงตลอดช่วงเวลาที่น่าสนใจ สำหรับวิธีการนี้เป็นวิธีการให้น้ำหนักกับค่าสังเกตของอนุกรมเวลา ที่นำมาใช้ในการพยากรณ์ไม่เท่ากัน โดยให้น้ำหนักค่าสังเกตที่เกิดขึ้นล่าสุดมากที่สุด ส่วนน้ำหนักของ ค่าสังเกตก่อนหน้ามีค่าลดหลั่นตามลำดับเวลา หมายถึง ถ้าค่าสังเกตล่าสุดมีค่าสูงมาก ยังมีน้ำหนักน้อย และ น้ำหนักของค่าสังเกต จะลดหลั่นลงแบบเอ็กซ์โปเนนเชียลตามลำดับเวลา ซึ่งใช้ค่าคงที่ 1 หรือมากกว่า 1 ตัว กำหนดน้ำหนักให้กับค่าสังเกตแต่ละค่า และเรียกค่าคงที่ว่า ค่าคงที่ปรับให้เรียบ (Smoothing Constant)

3. วิธีของบ็อกซ์และเจนกินส์ (Box and Jenkins Method) เป็นวิธีการวิเคราะห์อนุกรมเวลา ที่อาศัยกระบวนการเชิงสุ่ม (Stochastic Process) โดยถือว่าข้อมูลที่เกิดขึ้นตามเวลาที่เปลี่ยนแปลงไป มีลักษณะการเกิดค่าของข้อมูลที่เป็นไปตามกฎความน่าจะเป็น ซึ่งวิธีของบ็อกซ์และเจนกินส์ จะพยากรณ์ ข้อมูลจากรูปแบบหลัก 2 รูปแบบ ได้แก่ รูปแบบ Autoregressive คือ รูปแบบที่กำหนดว่าค่าพยากรณ์ ที่เวลาใด ๆ ขึ้นอยู่กับค่าสังเกตก่อนหน้า และรูปแบบ Moving Average คือ รูปแบบที่กำหนดว่า ค่าพยากรณ์ที่เวลาใด ๆ ขึ้นอยู่กับค่าคลาดเคลื่อนก่อนหน้า

การศึกษาวิธีการสร้างตัวแบบการพยากรณ์ตามวิธีการวิเคราะห์อนุกรมเวลาแต่ละวิธี จะใช้โปรแกรม คอมพิวเตอร์เป็นเครื่องมือช่วยในการสร้างตัวแบบการพยากรณ์ และพยากรณ์ข้อมูลล่วงหน้าจากตัวแบบ

ที่สร้างขึ้น โดยงานวิจัยนี้นำเสนอการเขียนโปรแกรมเพื่อสร้างตัวแบบการพยากรณ์และพยากรณ์ข้อมูลล่วงหน้าด้วยภาษาอาร์ (R Language) ซึ่งเป็นภาษาคอมพิวเตอร์ที่สร้างขึ้นโดย Ross Ihaka และ Robert Gentleman แห่งมหาวิทยาลัยออกแลนด์ (The University of Auckland) โดยภาษาอาร์เป็นภาษาคอมพิวเตอร์ที่ใช้สำหรับการวิเคราะห์เชิงสถิติ ซึ่งมีฟังก์ชันทางสถิติหลากหลาย สามารถนำมาเขียนโปรแกรมและพัฒนาวิธีการทางสถิติใหม่ ๆ ที่แสดงผลในรูปของกราฟได้ ทำให้ผู้ใช้นำมาวิเคราะห์และจัดการข้อมูลได้อย่างสะดวก รวดเร็วมากขึ้น ซึ่งต่างจากโปรแกรมสำเร็จรูปทางสถิติที่ใช้โดยทั่วไป และการใช้ภาษาอาร์สำหรับการวิเคราะห์เชิงสถิติ สามารถจำแนกได้เป็น 2 ลักษณะ ได้แก่ แบบโต้ตอบ (Interactive Mode) และแบบชุดคำสั่งในลักษณะฟังก์ชันและวัตถุ (Object and Functional Programming) ดังนั้น ภาษาอาร์ ซึ่งเป็นภาษาคอมพิวเตอร์ในรูปแบบของโอเพนซอร์ส (Open Source) จึงเป็นภาษาที่มีผู้พัฒนาฟังก์ชันที่หลากหลายเพื่อใช้สำหรับการวิเคราะห์ทางสถิติ และสามารถใช้ภาษาอาร์สนับสนุนการเขียนโปรแกรมแบบ Procedural Programming หรือการเขียนโปรแกรมแบบ Object Oriented Programming โดยมีโครงสร้างข้อมูลพื้นฐาน ได้แก่ เวกเตอร์ (Vectors) เมตริกซ์ (Metrics) อาร์เรย์ (Arrays) และตาตาเฟรม (Data Frames)

จากการทบทวนวรรณกรรมและงานวิจัยที่เกี่ยวข้องกับการวิเคราะห์ข้อมูลเวลาที่เป็นข้อมูลแสดงปริมาณการใช้ไฟฟ้า เพื่อสร้างตัวแบบการพยากรณ์ข้อมูล สามารถอธิบายได้ดังนี้

1. Maçaira, P.M. et al. (Maçaira, P.M. et al., 2015) ศึกษาการกำหนดตัวแบบการพยากรณ์และพยากรณ์ปริมาณการใช้ไฟฟ้ารายปีของผู้บริโภคประเภทบ้านอยู่อาศัย ในประเทศบราซิล จนถึงปี ค.ศ. 2050 ด้วยเทคนิคการพยากรณ์แบบ Pegels Exponential Smoothing โดยปรับค่าพารามิเตอร์เพื่อพยากรณ์ข้อมูลร่วมกับค่าประมาณการของบริษัท ERC (The Energy Research Company) ที่กำหนดไว้ในรายงาน PDE (The Ten Year Energy Planning) และรายงาน PNE (The Nation Energy Planning) ผลจากการศึกษาทำให้ทราบตัวแบบการพยากรณ์ที่ให้ค่าพยากรณ์ใกล้เคียงกับค่าประมาณการปริมาณการใช้ไฟฟ้าในรายงาน PDE และ PNE โดยตัวแบบการพยากรณ์สามารถพยากรณ์ปริมาณการใช้ไฟฟ้ารายปีได้ถึงปี ค.ศ. 2050

2. Pereira, C.M. et al. (Pereira, C.M. et al., 2015) ศึกษาการเปรียบเทียบตัวแบบการพยากรณ์ข้อมูลอนุกรมเวลา 2 ตัวแบบ โดยใช้ข้อมูลการจ่ายกระแสไฟฟ้าของสถานีจ่ายไฟฟ้า Coelba ในประเทศบราซิล ซึ่งมีลักษณะเป็นข้อมูลอนุกรมเวลา มาสร้างตัวแบบการพยากรณ์ 2 ตัวแบบ คือ ตัวแบบ SARIMAX และตัวแบบ FIC (Fuzzy Inference System) โดยรวมตัวแปรภายนอก 3 ตัวแปร ได้แก่ จำนวนลูกค้า อุณหภูมิ และปริมาณน้ำฝน เข้าไปในตัวแบบทั้ง 2 จากนั้นเปรียบเทียบค่าพยากรณ์ที่ได้จากตัวแบบการพยากรณ์ด้วยค่า MAPE (Mean Average Percentage Error) และค่า SD (Standard Deviation) ผลจากการศึกษาสรุปได้ว่า ตัวแบบ FIC ให้ค่าคลาดเคลื่อนต่ำกว่าตัวแบบ SARIMAX

3. Keka, I. and Hamiti, M. (Keka, I. and Hamiti, M., 2013) ศึกษาการทำตัวแบบทางคณิตศาสตร์ที่แสดงความสัมพันธ์ระหว่างพลังงานไฟฟ้าและเวลา ด้วยหลักการวิเคราะห์ความถดถอยแบบเชิงเส้น (Linear Regression) ซึ่งข้อมูลที่เข้าหาตัวแบบทางคณิตศาสตร์ได้ใช้ข้อมูลการจ่ายกระแสไฟฟ้าของสถานีจ่ายไฟฟ้าย่อยที่มีการวัดปริมาณการจ่ายกระแสไฟฟ้าทุก 15 นาที แล้วนำมาสร้างเป็นข้อมูล



30 การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาค่าตัวแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาอาร์

อนุกรมเวลาด้วยภาษาอาร์ โดยแบ่งคาบเวลาเป็นรายวัน รายสัปดาห์ และรายเดือน จากนั้นคำนวณค่าพารามิเตอร์ทางสถิติ และนำเสนอตัวแบบเชิงเส้นทางคณิตศาสตร์ที่เป็นสมการแสดงความสัมพันธ์ระหว่างพลังงานไฟฟ้าและเวลา

4. Chujai, P. et al. (Chujai, P. et al., 2013) ได้ศึกษาการวิเคราะห์ข้อมูลอนุกรมเวลาของปริมาณการใช้กระแสไฟฟ้าภายในบ้าน ซึ่งชุดข้อมูลที่ใช้เป็นชุดข้อมูลมาตรฐาน UCI โดยนำมาสร้างเป็นข้อมูลอนุกรมเวลา ที่แบ่งคาบเวลาเป็น 4 ประเภท คือ รายวัน รายสัปดาห์ รายเดือน และรายไตรมาส แล้วนำข้อมูลมาสร้างตัวแบบการพยากรณ์ 2 ตัวแบบ คือ ตัวแบบ ARIMA และตัวแบบ ARMA ด้วยภาษาอาร์ และโปรแกรม Rstudio จากนั้นเปรียบเทียบความถูกต้องของค่าพยากรณ์จากตัวแบบทั้งสอง โดยวัดค่าคลาดเคลื่อนด้วยค่า AIC (Akaike Information Criterion) และค่า RMSE (Root Mean Square Error) ซึ่งสรุปได้ว่าตัวแบบ ARIMA เป็นตัวแบบที่เหมาะสมสำหรับการพยากรณ์คาบเวลารายเดือนและรายไตรมาส ส่วนตัวแบบ ARMA เป็นตัวแบบที่เหมาะสมสำหรับการพยากรณ์คาบเวลารายวันและรายสัปดาห์

5. Wang, Y. et al. (Wang, Y. et al., 2012) ได้ศึกษาการปรับปรุงความแม่นยำของค่าพยากรณ์ความต้องการไฟฟ้าในเขตไฟฟ้าภาคตะวันตกเฉียงเหนือของประเทศไทย ที่ได้จากตัวแบบ Seasonal ARIMA โดยทำการปรับปรุงความแม่นยำของค่าพยากรณ์ด้วยการใช้วิธีการปรับปรุงค่าคลาดเคลื่อน (Residual) ที่เกิดจากการพยากรณ์ด้วยตัวแบบ Seasonal ARIMA ซึ่งใช้วิธีการที่แตกต่างกัน 3 วิธี ได้แก่ 1) การปรับปรุงค่าคลาดเคลื่อนด้วยวิธี PSO Optimal Fourier 2) วิธี Seasonal ARIMA และ 3) วิธีแบบผสมผสานระหว่างวิธี PSO Optimal Fourier และวิธี Seasonal ARIMA ผลการศึกษารูปได้ว่า การใช้วิธีการปรับปรุงค่าคลาดเคลื่อนทั้ง 3 วิธี ร่วมกับการพยากรณ์ด้วยตัวแบบ Seasonal ARIMA ให้ความถูกต้องของค่าพยากรณ์สูงกว่าการพยากรณ์ด้วยตัวแบบ Seasonal ARIMA เพียงอย่างเดียว นอกจากนี้ วิธีการปรับปรุงค่าคลาดเคลื่อนที่ให้ผลลัพธ์ที่ดีที่สุด คือ วิธีการปรับปรุงค่าคลาดเคลื่อนแบบผสมผสาน

งานวิจัยนี้ผู้วิจัยต้องการนำเสนอวิธีการวิเคราะห์อนุกรมเวลา ด้วยการสร้างตัวแบบการพยากรณ์ จากวิธีการที่แตกต่างกัน 3 วิธี ได้แก่ วิธีการวิเคราะห์การถดถอย วิธีการปรับให้เรียบแบบเอ็กซ์โปเนนเชียลของ Holt และ Winters และวิธีของบ็อกซ์และเจนกินส์ จากนั้นประเมินผลความแม่นยำของแต่ละตัวแบบ ด้วยการวัดค่าคลาดเคลื่อน RMSE และ MAPE ซึ่งใช้การเขียนโปรแกรมด้วยภาษาอาร์ เป็นเครื่องมือในการดำเนินการวิจัย ดังนั้น วัตถุประสงค์ของงานวิจัยสามารถแบ่งเป็นประเด็นหลัก ได้ดังนี้

1. เพื่อศึกษาการวิเคราะห์อนุกรมเวลา ด้วยการสร้างตัวแบบการพยากรณ์ด้วยภาษาอาร์ และโปรแกรม Rstudio จากข้อมูลหน่วยจำหน่ายไฟฟ้า ตั้งแต่เดือนมกราคม ปี พ.ศ. 2553 ถึงเดือนธันวาคม ปี พ.ศ. 2557 ทั้งหมด 3 ตัวแบบ ได้แก่ 1) ตัวแบบ Generalized Linear Models (glm) 2) ตัวแบบ Holt-Winters (HoltWinters) และ 3) ตัวแบบ Autoregressive Integrated Moving Average (ARIMA)

2. เพื่อเปรียบเทียบตัวแบบทั้ง 3 ด้วยการทดสอบความถูกต้องของค่าพยากรณ์ โดยใช้ภาษาอาร์ คำนวณค่า RMSE และ MAPE ของแต่ละตัวแบบ ซึ่งตัวแบบที่เหมาะสมสำหรับการพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้า คือตัวแบบที่ให้ค่า RMSE และ MAPE ต่ำที่สุด



3. เพื่อสร้างตัวแบบการพยากรณ์ที่เหมาะสมที่สุด ในรูปของสมการคณิตศาสตร์ ที่สามารถนำไปพัฒนาเป็นโปรแกรมคอมพิวเตอร์ สำหรับพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าประเภทบ้านอยู่อาศัยของการไฟฟ้านครหลวง

### วิธีดำเนินการวิจัย

สำหรับงานวิจัยนี้ผู้วิจัยต้องการศึกษาข้อมูลการจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง ที่จำหน่ายให้กับผู้บริโภคประเภทบ้านอยู่อาศัย โดยแบ่งคาบเวลาเป็นรายเดือน ซึ่งแบ่งข้อมูลเป็นสองกลุ่ม กลุ่มที่ 1 เป็นกลุ่มที่ใช้สำหรับการสร้างตัวแบบการพยากรณ์ โดยใช้ข้อมูลหน่วยจำหน่ายไฟฟ้า ตั้งแต่เดือนมกราคม ปี พ.ศ. 2553 ถึงเดือนธันวาคม ปี พ.ศ. 2557 เป็นข้อมูลฝึกสอน ส่วนกลุ่มที่ 2 เป็นกลุ่มที่ใช้สำหรับทดสอบความถูกต้องของค่าพยากรณ์สำหรับแต่ละตัวแบบการพยากรณ์ที่ได้สร้างขึ้น โดยใช้ข้อมูลหน่วยจำหน่ายไฟฟ้า ตั้งแต่เดือนมกราคมถึงเดือนพฤษภาคม ปี พ.ศ. 2558 เป็นข้อมูลทดสอบ ซึ่งวิธีดำเนินการวิจัยจะดำเนินการสร้างตัวแบบการพยากรณ์ จากวิธีการพยากรณ์ที่แตกต่างกัน 3 วิธี ได้แก่ วิธีการวิเคราะห์ความถดถอย วิธีการปรับให้เรียบแบบเอ็กซ์โปเนนเชียล และวิธีของบ็อกซ์และเจนกินส์ ได้เป็นตัวแบบการพยากรณ์ 3 ตัวแบบ จากนั้น นำตัวแบบทั้ง 3 มาทำนายค่าพยากรณ์ล่วงหน้า 5 คาบเวลา แล้วนำค่าพยากรณ์ของแต่ละตัวแบบไปเปรียบเทียบกับค่าสังเกตจริง ที่แบ่งไว้เป็นชุดข้อมูลทดสอบ เพื่อวัดค่าคลาดเคลื่อนทางสถิติ ได้แก่ ค่า RMSE และค่า MAPE และใช้ค่าคลาดเคลื่อนดังกล่าว เลือกตัวแบบการพยากรณ์ที่เหมาะสมที่สุด แล้วนำไปสร้างสมการพยากรณ์ของตัวแบบที่เหมาะสมที่สุด ซึ่งตัวแบบการพยากรณ์ที่สร้างจากวิธีการพยากรณ์ที่แตกต่างกันทั้ง 3 วิธี มีดังนี้

1. ตัวแบบ glm เป็นตัวแบบที่สร้างจากวิธีการวิเคราะห์ความถดถอย โดยมีรูปแบบการถดถอยแบบเชิงเส้น ดังนี้ (Boldina, I. and Beninger, P.G., 2016)

$$Y = \beta_0 + \beta_1 X + \varepsilon \quad (1)$$

โดยที่

<b>X</b>	คือ	คาบเวลามีหน่วยเป็นเดือน ซึ่งเป็นตัวแปรอิสระ
<b>Y</b>	คือ	ค่าสังเกตที่สนใจ ในที่นี้คือหน่วยจำหน่ายไฟฟ้ารายเดือน
<b><math>\varepsilon</math></b>	คือ	ค่าคลาดเคลื่อน

2. ตัวแบบ HoltWinters เป็นตัวแบบที่สร้างจากวิธีการปรับให้เรียบแบบทริเบิลเอ็กซ์โปเนนเชียล สมูทติง (Triple Exponential Smoothing) ซึ่งเป็นวิธีการปรับให้เรียบที่นำมาใช้สร้างตัวแบบการพยากรณ์กับข้อมูลที่มีส่วนประกอบของแนวโน้ม และการเปลี่ยนแปลงตามฤดูกาล โดยมีรูปแบบ ดังนี้ (Prema, V. and Rao, K.U., 2015)

32 การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาตัวแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาฮอว์

$$\begin{aligned} \text{Level: } L_t &= \alpha(Y_t - S_{t-s}) + (1-\alpha)(L_{t-1} + b_{t-1}) \\ \text{Trend: } b_t &= \beta(L_t - L_{t-1}) + (1-\beta)b_{t-1} \\ \text{Seasonal: } S_t &= \gamma(Y_{t-1} - L_t) + (1-\gamma)S_{t-s} \end{aligned} \quad (2)$$

และสมการพยากรณ์ ที่ได้จากส่วนประกอบทั้ง 3 ส่วน แสดงได้ ดังนี้

$$\text{Forecast: } \bar{F}_{t+m} = L_t + b_m + S_{t-s+m} \quad (3)$$

โดยที่

$\alpha$	คือ	ค่าคงที่ปรับให้เรียบของส่วนประกอบระดับ มีค่าระหว่าง 0 กับ 1
$\beta$	คือ	ค่าคงที่ปรับให้เรียบของส่วนประกอบแนวโน้ม มีค่าระหว่าง 0 กับ 1
$\gamma$	คือ	ค่าคงที่ปรับให้เรียบของส่วนประกอบฤดูกาล มีค่าระหว่าง 0 กับ 1
$L_t$	คือ	ค่าประมาณระดับของอนุกรมเวลา ณ เวลา t
$Y_t$	คือ	ค่าสังเกต ณ เวลา t
$b_t$	คือ	ค่าประมาณความชันของอนุกรมเวลา ณ เวลา t
$S_t$	คือ	ส่วนประกอบฤดูกาล
$s$	คือ	ความยาวของแต่ละฤดูกาล เช่น ฤดูกาลละ 12 เดือน ( $s = 12$ )
$m$	คือ	จำนวนคาบเวลาล่วงหน้าที่ต้องการพยากรณ์
$\bar{F}_{t+m}$	คือ	ค่าพยากรณ์ล่วงหน้าที่ m คาบเวลา

3. ตัวแบบ ARIMA เป็นตัวแบบที่สร้างจากวิธีของบอกรีและเจนกินส์ โดยมีรูปแบบทั่วไปที่เขียนในรูปของตัวดำเนินการย้อนกลับ (Backward Shift Operator: B) ในรูปสมการดังนี้ (Wang, Y. et al., 2012)

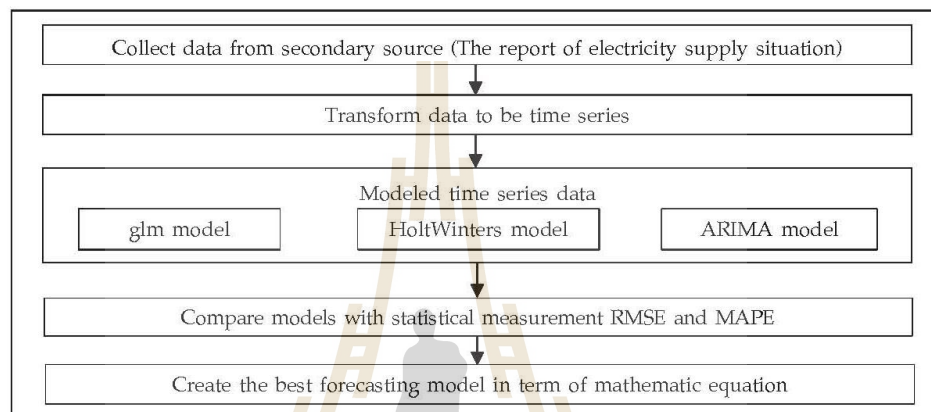
$$\theta_p(B)\Theta_p(B^S)(1-B)^d(1-B^S)^D Y_t = w_q(B)W_q(B^S)\alpha_t \quad (4)$$

เมื่อ  $BY_t = Y_{t-1}$  และ  $B^k Y_t = Y_{t-k}$

โดยที่

$$\begin{aligned} w_q(B) &= 1 - w_1 B - w_2 B^2 - \dots - w_q B^q \\ \theta_p(B) &= 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_p B^p \\ \Theta_p(B^S) &= 1 - \Theta_1 B^S - \Theta_2 B^{2S} - \dots - \Theta_p B^{pS} \\ W_q(B^S) &= 1 - W_1 B^S - W_2 B^{2S} - \dots - W_q B^{qS} \end{aligned} \quad (5)$$

เมื่อได้ 3 ตัวแบบ จึงนำตัวแบบมาเปรียบเทียบ ด้วยการวัดค่าสถิติ RMSE และ MAPE ซึ่งตัวแบบที่เหมาะสมสำหรับการพยากรณ์ข้อมูลอนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า คือ ตัวแบบที่ให้ค่า RMSE และค่า MAPE ต่ำที่สุด โดยสรุปวิธีการวิเคราะห์ข้อมูลอนุกรมเวลา เพื่อหาตัวแบบที่เหมาะสมสำหรับการพยากรณ์ข้อมูล ดังรูปที่ 1



รูปที่ 1 กรอบแนวคิดแสดงขั้นตอนการเลือกตัวแบบที่เหมาะสมที่สุดสำหรับการพยากรณ์ข้อมูลอนุกรมเวลา

จากรูปที่ 1 อธิบายแต่ละขั้นตอนได้ดังนี้

1. Collect Data from Secondary Source คือ ขั้นตอนรวบรวมข้อมูล จากข้อมูลทุติยภูมิ ที่ได้จากรายงานสถานการณ์การจำหน่ายไฟฟ้าประจำเดือน ของการไฟฟ้านครหลวง ตั้งแต่เดือนมกราคม ปี พ.ศ. 2553 ถึงเดือนพฤษภาคม ปี พ.ศ. 2558 กับผู้บริหารโคกประเภทบ้านอยู่อาศัย ซึ่งตัวอย่างของข้อมูล ในรายงาน ที่นำมาสร้างเป็นข้อมูลอนุกรมเวลา แสดงได้ดังรูปที่ 2

34 การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาตัวแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาอาร์

รายการ	ก	ข	ค
	เดือน พ.ย. 2553	เดือน ธ.ค. 2552	เดือน ธ.ค. 2553
<b>1. ข้อมูลการใช้ไฟฟ้า</b>			
1.1 ข้อมูลหน่วยซื้อ(หน่วย : ล้านหน่วย)	3,679.03	3,474.84	3,586.96
1.1.1 ข้อมูลหน่วยซื้อจากกฟผ.(หน่วย : ล้านหน่วย)	3,678.68	3,473.74	3,586.95
1.1.1.1 แยกตามแรงดัน			
1.1.1.1.1 69 เควี	1,677.30	1,626.20	1,715.35
1.1.1.1.2 115 เควี	1,546.66	1,408.64	1,501.67
1.1.1.1.3 230 เควี	454.72	438.9	369.93
1.1.1.2 จำนวนสถานีต้นทาง แยกตามแรงดัน			
1.1.1.2.1 69 เควี	11	11	11
1.1.1.2.2 115 เควี	12	12	12
1.1.1.2.3 230 เควี	3	3	3
1.1.2 ข้อมูลหน่วยซื้อจาก VSPP(หน่วย : ล้านหน่วย)	0.35	1.1	0.01
1.2 หน่วยจำหน่าย(หน่วย : ล้านหน่วย)	3,529.06	3,318.09	3,477.63
1.2.1 แยกตามแรงดัน			
1.2.1.1 220 โวลท์	1,660.78	1,650.83	1,637.19
1.2.1.2 12 - 24 เควี	1,589.66	1,407.74	1,567.07
1.2.1.3 69 - 115 เควี	304.5	285.25	300.17
1.2.2 ตามประเภทผู้ใช้ไฟฟ้า			
1.2.2.1 บ้านอยู่อาศัย	787.32	722.16	776.67

รูปที่ 2 ตัวอย่างข้อมูลหน่วยจำหน่ายไฟฟ้าประจำเดือนธันวาคม ปี พ.ศ. 2553

2. Transform Data to be Time Series คือ ขั้นตอนที่น่าข้อมูลทุติยภูมิที่รวบรวมได้ มาสร้างเป็นข้อมูลอนุกรมเวลา เรียงตามลำดับเวลา โดยมีคาบเวลาเป็นรายเดือน ตั้งแต่เดือนมกราคม ปี พ.ศ. 2553 ถึงเดือนพฤษภาคม ปี พ.ศ. 2558 แสดงได้ดังรูปที่ 3 และแบ่งข้อมูลเป็น 2 ชุด ได้แก่ 1) ชุดข้อมูลฝึกสอน (Training Set) ซึ่งใช้ข้อมูลตั้งแต่เดือนมกราคม ปี พ.ศ. 2553 ถึงเดือนธันวาคม ปี พ.ศ. 2557 สำหรับสร้างตัวแบบพยากรณ์ และ 2) ชุดข้อมูลทดสอบ (Test Set) ซึ่งใช้ข้อมูลตั้งแต่เดือนมกราคมถึงเดือนพฤษภาคม ปี พ.ศ. 2558 สำหรับทดสอบตัวแบบพยากรณ์ เพื่อวัดค่าคลาดเคลื่อน RMSE และ MAPE

	A	B	C	D	E
1	Years	Months	Electric Units	No_Consumers	Mean_Units
2	2553	1	776.32	2420767	320.69
3	2553	2	836.16	2423046	345.09
4	2553	3	925.55	2433022	380.41
5	2553	4	1026.5	2439689	420.75
6	2553	5	1082.18	2448223	442.03
7	2553	6	1003.41	2456309	408.5
8	2553	7	930.91	2463398	377.9
9	2553	8	876.83	2473742	354.45

รูปที่ 3 ข้อมูลอนุกรมเวลาที่สร้างจากข้อมูลทุติยภูมิ

3. Modeled Time Series Data คือ ขั้นตอนการสร้างตัวแบบที่แตกต่างกัน 3 ตัวแบบ จากข้อมูลอนุกรมเวลาที่แบ่งไว้เป็นชุดข้อมูลฝึกสอน โดยการสร้างตัวแบบทั้ง 3 ใช้ชุดข้อมูลฝึกสอนชุดเดียวกัน แต่ละตัวแบบมีคำสั่งสร้างตัวแบบด้วยภาษาอาร์ สรุปลงได้ดังรูปที่ 4

```
myDFlr <- read.csv(file="d:/R/EuseLRmodel.csv", header=TRUE, sep=",")
glm.D9 <- glm(myDFlr[,4] ~ myDFlr[,3])
```

(ก) คำสั่งสร้างตัวแบบ glm (Generalized Linear Models)

```
library("forecast", lib.loc=~R/win-library/3.2")
myDF53_57 <- read.csv(file="d:/R/Euse53_57.csv", header=TRUE, sep=",")
myts53_57 <- ts(myDF53_57[,3], start=c(2553,1), end=c(2557,12), frequency=12)
HWfit <- HoltWinters(myts53_57)
```

(ข) คำสั่งสร้างตัวแบบ HoltWinters (Holt-Winters)

```
library("forecast", lib.loc=~R/win-library/3.2")
myDF53_57 <- read.csv(file="d:/R/Euse53_57.csv", header=TRUE, sep=",")
myts53_57 <- ts(myDF53_57[,3], start=c(2553,1), end=c(2557,12), frequency=12)
auto.arima(myts53_57)
fitM53_57 <- arima(myts53_57, order=c(1,0,0), list(order=c(1,0,0), period=12))
```

(ค) คำสั่งสร้างตัวแบบ ARIMA (Autoregressive Integrated Moving Average)

รูปที่ 4 ชุดคำสั่งภาษาอาร์ สำหรับสร้างตัวแบบอนุกรมเวลาที่แตกต่างกัน 3 ตัวแบบ

4. Compare Models with Statistical Measurement RMSE and MAPE คือ ขั้นตอนการเปรียบเทียบตัวแบบพยากรณ์ 3 ตัวแบบ ด้วยการวัดค่าคลาดเคลื่อนทางสถิติ คือ ค่า RMSE (Root Mean Square Error) และ MAPE (Mean Absolute Percentage Error) (Bergmeir, C. and Benítez, J.M., 2012) ซึ่งมีสมการดังนี้

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \quad (6)$$

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| 100 \frac{y_t - \hat{y}_t}{y_t} \right| \quad (7)$$

เมื่อ  $y_t$  = ค่าสังเกตที่เวลา  $t$ ,  $\hat{y}_t$  = ค่าพยากรณ์ที่เวลา  $t$  และ  $n$  = จำนวนคาบเวลาที่พยากรณ์



36 การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาอาร์

5. Create the Best Forecasting Model in Term of Mathematic Equation คือ ขั้นตอนการสร้างตัวแบบการพยากรณ์ที่เหมาะสมที่สุดในรูปของสมการคณิตศาสตร์ ที่สามารถนำไปพัฒนาเป็นโปรแกรมคอมพิวเตอร์ สำหรับพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าประเภทบ้านอยู่อาศัยของการไฟฟ้านครหลวง ซึ่งตัวแบบการพยากรณ์ที่เหมาะสมที่สุด คือ ตัวแบบที่ให้ค่า RMSE และ MAPE ต่ำที่สุด

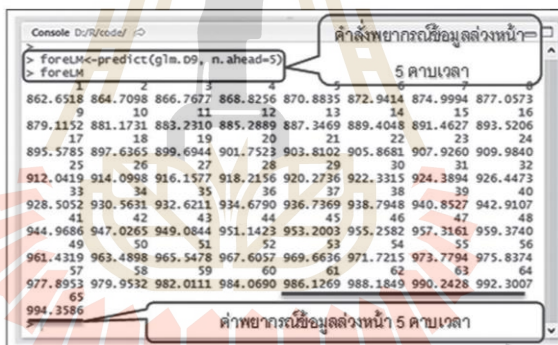
**ผลการวิจัย**

ผู้วิจัยนำตัวแบบพยากรณ์ทั้ง 3 ตัวแบบ มาพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้า 5 คาบเวลา ตั้งแต่เดือนมกราคมถึงเดือนพฤษภาคม ปี พ.ศ. 2558 เพื่อนำค่าพยากรณ์ไปวัดค่าคลาดเคลื่อน RMSE และ MAPE โดยเปรียบเทียบกับค่าสังเกตจริงที่แบ่งไว้เป็นชุดข้อมูลทดสอบ ซึ่งผลการวิจัยพบว่า

1. การพยากรณ์ข้อมูลด้วยตัวแบบพยากรณ์

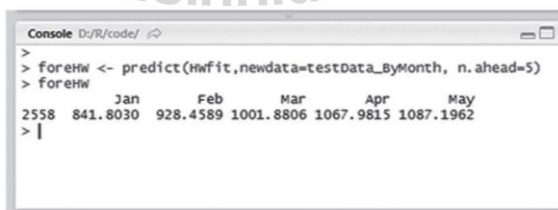
คำสั่งในภาษาอาร์ที่ใช้สำหรับพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าจากตัวแบบพยากรณ์ และค่าพยากรณ์ 5 คาบเวลาของแต่ละตัวแบบ เป็นดังนี้

1. ตัวแบบ glm มีคำสั่งพยากรณ์ข้อมูลและค่าพยากรณ์ เป็นดังรูปที่ 5



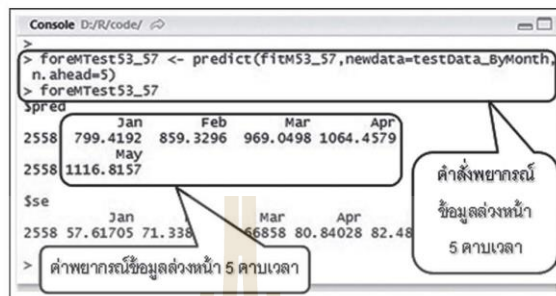
รูปที่ 5 คำสั่งพยากรณ์ข้อมูลและค่าพยากรณ์ของตัวแบบ glm

2. ตัวแบบ HoltWinters มีคำสั่งพยากรณ์ข้อมูลและค่าพยากรณ์ เป็นดังรูปที่ 6



รูปที่ 6 คำสั่งพยากรณ์ข้อมูลและค่าพยากรณ์ของตัวแบบ HoltWinters

3. ตัวแบบ ARIMA มีคำสั่งพยากรณ์ข้อมูลและค่าพยากรณ์ เป็นดังรูปที่ 7



รูปที่ 7 คำสั่งพยากรณ์ข้อมูลและค่าพยากรณ์ของตัวแบบ ARIMA

2. การเลือกตัวแบบพยากรณ์ที่เหมาะสม

การเลือกตัวแบบพยากรณ์ที่เหมาะสมมากที่สุดสำหรับการพยากรณ์ข้อมูล ใช้วิธีวัดค่าคลาดเคลื่อนทางสถิติ ได้แก่ ค่า RMSE และ MAPE ซึ่งค่าทั้ง 2 เป็นการวัดผลรวมของความแตกต่างระหว่างค่าพยากรณ์กับค่าสังเกตจริง โดยตัดเครื่องหมายบวกและลบออก เพื่อมิให้ค่าคลาดเคลื่อนที่เป็นบวกและลบหักล้างกันเอง ดังแสดงในสมการที่ 1 และ 2 ดังนั้น ค่า RMSE และค่า MAPE จึงมีความเหมาะสมในการนำมาใช้วัดความแม่นยำของการพยากรณ์ได้ดี ซึ่งตัวแบบที่เหมาะสมมากที่สุด คือ ตัวแบบที่มีความแม่นยำในการพยากรณ์มากที่สุด หรือให้ค่า RMSE และค่า MAPE ต่ำที่สุด

คำสั่งในภาษาอาร์สำหรับคำนวณค่า RMSE และค่า MAPE คือ การเรียกใช้ฟังก์ชัน accuracy() ซึ่งเป็นฟังก์ชันที่อยู่ในแพ็คเกจ forecast โดยมีการติดตั้งแพ็คเกจไว้ในขั้นตอนการสร้างตัวแบบ HoltWinters และตัวแบบ ARIMA เรียบร้อยแล้ว สำหรับการเรียกใช้ฟังก์ชัน accuracy() ต้องส่งค่าพยากรณ์และค่าจริงในรูปแบบของเวกเตอร์ให้กับฟังก์ชัน ดังนั้นจึงนำค่าจริงและค่าพยากรณ์ที่ได้จากตัวแบบทั้ง 3 มาสร้างเป็นไฟล์ชนิด CSV จากนั้นใช้คำสั่งอ่านข้อมูลจากไฟล์ แล้วใช้คำสั่งสร้างเวกเตอร์ขึ้นมา 4 ตัว เพื่อเก็บค่าจริงและค่าพยากรณ์ที่ได้จากตัวแบบทั้ง 3 โดยตั้งชื่อไฟล์ชนิด CSV ด้วยชื่อ "CompareModel" ซึ่งมีข้อมูลเป็นดังรูปที่ 8 และชุดคำสั่งในภาษาอาร์ สำหรับคำนวณค่า RMSE และ MAPE ของตัวแบบทั้ง 3 แสดงได้ดังรูปที่ 9

	A	B	C	D	E
1	Months	Actual_Value	glm	HoltWinter	ARIMA
2	1	745.82	986.13	841.8	799.42
3	2	853.6	988.18	928.46	859.33
4	3	1017.89	990.24	1001.88	969.05
5	4	1100.93	992.3	1067.98	1064.46
6	5	1216.77	994.36	1087.2	1116.82

รูปที่ 8 ไฟล์ชนิด CSV ชื่อ "CompareModel"



38 การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาตัวแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาอาร์

```

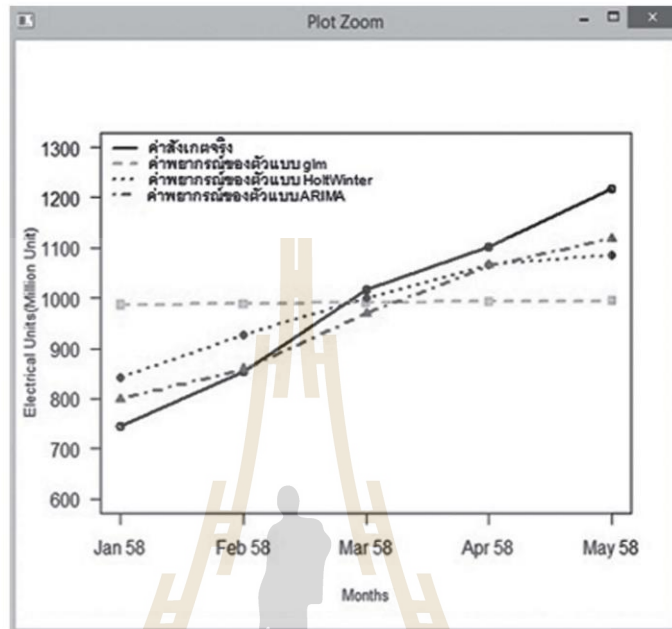
> library("forecast", lib.loc=~R/win-library/3.2")
> accDF <- read.csv(file="d:/DataForNewPaper1/comp
aremodel1.csv", header=TRUE, sep=",")
> act<-accDF[,2]
> glm<-accDF[,3]
> HoltWinters1<-accDF[,4]
> ARIMA1<-accDF[,5]
> accuracy(glm,act)
      ME      RMSE      MAE      MPE      MAPE
Test set -3.24 166.0674 146.716 -3.424968 15.76986
> accuracy(HoltWinters1,act)
      ME      RMSE      MAE      MPE      MAPE
Test set 2.132 81.47885 70.408 -1.229773 7.418652
> accuracy(ARIMA1,act)
      ME      RMSE      MAE      MPE      MAPE
Test set 24.858 57.182 48.518 1.666064 4.799628
  
```

รูปที่ 9 ชุดคำสั่งในภาษาอาร์ สำหรับคำนวณค่า RMSE และ MAPE

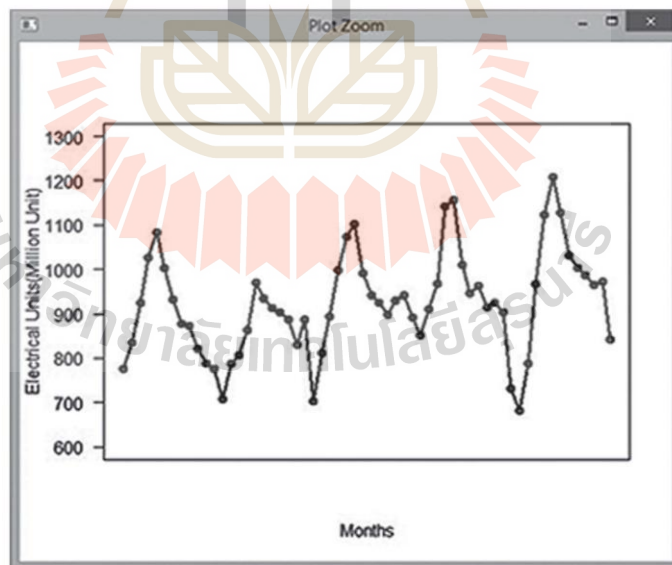
จากรูปที่ 9 ตัวแบบ ARIMA มีค่า RMSE และค่า MAPE ต่ำที่สุด ซึ่งมีค่าเท่ากับ 57.18 และ 4.80 ตามลำดับ ดังนั้น ตัวแบบ ARIMA เป็นตัวแบบที่มีความเหมาะสมที่สุดสำหรับการพยากรณ์ข้อมูลอนุกรมเวลา โดยสามารถแสดงค่าพยากรณ์ 5 คาบเวลา ค่า RMSE และค่า MAPE ของแต่ละตัวแบบได้ดังตารางที่ 1 นอกจากนี้ เมื่อนำคำสั่งเกิดจริงและค่าพยากรณ์ที่ได้จากตัวแบบทั้ง 3 มาสร้างเป็นกราฟเส้น ดังรูปที่ 10 ทำให้ทราบว่า ตัวแบบ glm ที่สร้างจากวิธีการวิเคราะห์ความถดถอยเชิงเส้นไม่เหมาะสำหรับนำมาใช้พยากรณ์อนุกรมเวลา เพราะเมื่อนำอนุกรมเวลามาสร้างกราฟอนุกรมเวลาดังรูปที่ 11 จากลักษณะของกราฟอนุกรมเวลา แสดงให้เห็นว่า อนุกรมเวลามีแนวโน้มเป็นเส้นตรงเชิงบวก และมีส่วนประกอบฤดูกาล เพราะมีรูปแบบการเกิดซ้ำตามรอบฤดูกาล ซึ่งตัวแบบ glm ที่สร้างจากวิธีการวิเคราะห์ความถดถอยเชิงเส้น ไม่สามารถตรวจจับความเคลื่อนไหวเนื่องจากแนวโน้มและฤดูกาลได้ โดยสอดคล้องกับค่า RMSE และค่า MAPE ของตัวแบบ glm ที่มีค่ามากที่สุด ส่วนตัวแบบ HoltWinters และตัวแบบ ARIMA สามารถตรวจจับความเคลื่อนไหวเนื่องจากแนวโน้มและฤดูกาลได้ แต่สำหรับอนุกรมเวลาชุดนี้ ตัวแบบ ARIMA มีความเหมาะสมมากกว่าตัวแบบ HoltWinters เนื่องจากให้ค่า RMSE และค่า MAPE ต่ำที่สุด

ตารางที่ 1 ค่าพยากรณ์ค่า RMSE และค่า MAPE ของแต่ละตัวแบบ

หน่วยจำหน่ายไฟฟ้า	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	RMSE	MAPE
ค่าสังเกตจริง	2558	2558	2558	2558	2558	-	-
ค่าพยากรณ์ของตัวแบบ glm	745.82	853.60	1017.89	1100.93	1216.77	166.07	15.77
ค่าพยากรณ์ของตัวแบบ HoltWinters	841.80	928.46	1001.88	1067.98	1087.20	81.48	7.42
ค่าพยากรณ์ของตัวแบบ ARIMA	799.24	859.33	969.05	1064.46	1116.82	57.18	4.80



รูปที่ 10 กราฟเปรียบเทียบระหว่างค่าสังเกตจริงและค่าพยากรณ์ 5 คาบเวลา



รูปที่ 11 กราฟอนุกรมเวลาสำหรับสร้างตัวแบบการพยากรณ์

40 การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาอาร์

### 3. การสร้างสมการพยากรณ์ของตัวแบบ ARIMA

เมื่อได้ตัวแบบ ARIMA เป็นตัวแบบการพยากรณ์ที่เหมาะสมที่สุด สามารถสร้างสมการพยากรณ์หน่วยจำหน่ายไฟฟ้าประเภทบ้านอยู่อาศัยของการไฟฟ้านครหลวง ที่ได้จากการแปลงรูปแบบทั่วไป (General Form) ของตัวแบบ ARIMA ด้วยการกำหนดรูปแบบที่เหมาะสมกับอนุกรมเวลา และประมาณค่าพารามิเตอร์ที่ไม่ทราบค่า โดยใช้ฟังก์ชัน `auto.arima()` ดังรูปที่ 12 ซึ่งทำให้ทราบว่า ตัวแบบ ARIMA สำหรับอนุกรมเวลาชุดนี้ ประกอบด้วย Autoregressive(AR) อันดับ  $p=1$  และ Seasonal Autoregressive(SAR) อันดับ  $P=1$  ซึ่งประมาณค่าสัมประสิทธิ์  $AR_1(\hat{\theta}_1)$  และ  $SAR_1(\hat{\Theta}_1)$  ได้เท่ากับ 0.7301 และ 0.6403 โดยที่อันดับ  $d=D=0$  และอันดับ  $q=Q=0$  ดังนั้น สามารถสร้างสมการพยากรณ์หน่วยจำหน่ายไฟฟ้าประเภทบ้านอยู่อาศัยของการไฟฟ้านครหลวง ได้ดังนี้

$$(1 - \hat{\theta}_1 B)(1 - \hat{\Theta}_1 B^{12})Y_t = \theta_0 \quad (8)$$

เมื่อ  $\theta_0 =$  ค่าคงที่

$$\begin{aligned} (1 - \hat{\theta}_1 B - \hat{\Theta}_1 B^{12} + \hat{\theta}_1 B \hat{\Theta}_1 B^{12})Y_t &= \theta_0 \\ Y_t - \hat{\theta}_1 Y_{t-1} - \hat{\Theta}_1 Y_{t-12} + \hat{\theta}_1 \hat{\Theta}_1 Y_{t-13} &= \theta_0 \\ Y_t &= \theta_0 + \hat{\theta}_1 Y_{t-1} + \hat{\Theta}_1 Y_{t-12} - \hat{\theta}_1 \hat{\Theta}_1 Y_{t-13} \\ Y_t &= \theta_0 + 0.73Y_{t-1} + 0.64Y_{t-12} - 0.47Y_{t-13} \end{aligned} \quad (9)$$

จากค่าสังเกตของอนุกรมเวลา ทำให้ทราบว่า ค่าคงที่  $\theta_0 = 92.42$  ดังนั้น ได้สมการพยากรณ์เป็นดังสมการที่ 10

$$Y_t = 92.42 + 0.73Y_{t-1} + 0.64Y_{t-12} - 0.47Y_{t-13} \quad (10)$$

```

Console D:/R/code/ <
> auto.arima(myts53_57)
Series: myts53_57
ARIMA(1,0,0)(1,0,0)[12] with non-zero mean
Coefficients:
ar1      sar1      intercept
0.7301   0.6403   929.9639
s.e.    0.0835   0.0984   57.2129

sigma^2 estimated as 3320: log likelihood=-331.93
AIC=671.85  AICC=672.58  BIC=680.23
> |
  
```

รูปที่ 12 ฟังก์ชัน `auto.arima()` สำหรับกำหนดรูปแบบของตัวแบบ ARIMA

เมื่อสามารถสร้างตัวแบบ ARIMA ซึ่งเป็นตัวแบบการพยากรณ์ที่เหมาะสมที่สุด ในรูปของสมการคณิตศาสตร์ ได้ดังสมการที่ (10) โดยอยู่ในรูปของสมการถดถอยเชิงเส้น ที่มีตัวแปรอิสระ 3 ตัว คือ  $Y_{t-1}$ ,  $Y_{t-12}$  และ  $Y_{t-13}$  หมายถึงค่าสังเกตที่ล่าช้าไป 1, 12 และ 13 คาบเวลา ตามลำดับ และตัวแปรตาม

คือ  $Y_t$  หมายถึง ค่าสังเกต ณ คาบเวลา  $t$  ใด ๆ ดังนั้น สามารถนำสมการที่ 10 ไปใช้สร้างโปรแกรม สำหรับพยากรณ์หน่วยจำหน่ายไฟฟ้ารายเดือน ประเภทบ้านพักอาศัยของการไฟฟ้านครหลวงได้ดังรูปที่ 13

เลือกเดือนพยากรณ์	เลือกปีพยากรณ์	จำนวน	จำนวน
มกราคม	2558	2557	841.36
มกราคม	2557	2557	681.77
มกราคม	2557	2557	730.90
ดำเนินการพยากรณ์			
ค่าพยากรณ์หน่วยจำหน่ายไฟฟ้าเดือน มกราคม ปี 2558	จำนวน	799.42	จำนวน

รูปที่ 13 ตัวอย่างโปรแกรมพยากรณ์ของตัวแบบ ARIMA

### การอภิปรายผล

จากผลการวิจัยพบว่า อนุกรมเวลาที่ศึกษามีการเคลื่อนไหวเนื่องจากแนวโน้มและฤดูกาล เพราะลักษณะของกราฟอนุกรมเวลามีแนวโน้มเป็นเส้นตรงเชิงบวก และมีรูปแบบการเกิดซ้ำตามรอบฤดูกาล เมื่อนำอนุกรมเวลามาสร้างตัวแบบการพยากรณ์ด้วยวิธีการวิเคราะห์ความถดถอยเชิงเส้น วิธีการปรับให้เรียบแบบเอ็กซ์โปเนนเชียล และวิธีของบ็อกซ์และเจนกินส์ แล้วทำการเปรียบเทียบตัวแบบการพยากรณ์ด้วยการวัดค่า RMSE และค่า MAPE พบว่า ตัวแบบ ARIMA มีค่า RMSE และค่า MAPE ต่ำที่สุด โดยมีค่า RMSE เท่ากับ 57.18 และค่า MAPE เท่ากับ 4.80 ซึ่งค่า MAPE มีค่าอยู่ในเกณฑ์ที่ต่ำ หมายความว่าตัวแบบการพยากรณ์มีความแม่นยำในการพยากรณ์สูง ดังนั้น ตัวแบบ ARIMA จึงเป็นตัวแบบที่เหมาะสมสำหรับการพยากรณ์ข้อมูลอนุกรมเวลาชุดดังกล่าวนี้ ซึ่งสอดคล้องกับงานวิจัยในอดีตของ Chujai, P. et al. (Chujai, P. et al., 2013) และงานวิจัยของ Wang, Y. et al. (Wang, Y. et al., 2012) ที่พบว่า ตัวแบบ ARIMA มีความเหมาะสมสำหรับการนำมาใช้พยากรณ์อนุกรมเวลาที่มีส่วนประกอบของแนวโน้มหรือฤดูกาล และจากการใช้ฟังก์ชัน auto.arima() เพื่อกำหนดรูปแบบของตัวแบบ ARIMA พบว่า ตัวแบบ ARIMA มีส่วนประกอบของ Autoregressive ที่อันดับ  $p=1$  และส่วนประกอบของ Seasonal Autoregressive ที่อันดับ  $P=1$  ซึ่งหมายความว่า ค่าสังเกต ณ เวลาใด ขึ้นอยู่กับค่าสังเกตก่อนหน้า 1 คาบเวลา และขึ้นอยู่กับค่าสังเกตก่อนหน้า 1 ฤดูกาล แต่ไม่มีส่วนประกอบของ Moving Average และ Seasonal Moving Average เพราะอันดับ  $q=Q=0$  หมายความว่า ค่าสังเกต ณ เวลาใด ๆ ไม่ขึ้นอยู่กับค่าคลาดเคลื่อนของการพยากรณ์ และอนุกรมเวลามีลักษณะนิ่ง (Stationary) เนื่องจากอันดับ  $d=D=0$  ทั้งนี้ จากรูปแบบของตัวแบบ ARIMA สามารถสร้างสมการพยากรณ์ของตัวแบบ ARIMA เพื่อใช้พยากรณ์ข้อมูลล่วงหน้า ได้ดังสมการที่ (10) และนำมาใช้พยากรณ์หน่วยจำหน่ายไฟฟ้ารายเดือนของผู้บริโภคประเภทบ้านพักอาศัย ประจำปี พ.ศ. 2558 ได้ดังตารางที่ 2



42 การวิเคราะห์อนุกรมเวลาของข้อมูลหน่วยจำหน่ายไฟฟ้า เพื่อค้นหาตัวแบบการพยากรณ์ที่เหมาะสม ด้วยภาษาอาร์

ตารางที่ 2 ค่าพยากรณ์หน่วยจำหน่ายไฟฟ้ารายเดือน ประจำปีพ.ศ. 2558 (หน่วยเป็น ล้านหน่วย)

ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.
799.42	859.33	969.05	1064.46	1116.82	1062.40	998.62	980.24	968.71	954.07	957.64	874.13

### บทสรุป

จากวัตถุประสงค์การวิจัย เพื่อศึกษาผลการวิเคราะห์ข้อมูลอนุกรมเวลาในการค้นหาตัวแบบที่เหมาะสมที่สุด โดยการเขียนโปรแกรมด้วยภาษาอาร์ พบว่า ข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้าานครหลวง สำหรับผู้บริโภคประเภทบ้านพักอาศัย เป็นอนุกรมเวลาที่มีส่วนประกอบของแนวโน้มและฤดูกาล ซึ่งตัวแบบ ARIMA เป็นตัวแบบที่มีความเหมาะสมมากที่สุด สำหรับนำมาใช้ในการพยากรณ์ข้อมูล เพราะสามารถตรวจจับการเคลื่อนไหวของอนุกรมเวลาเนื่องจากแนวโน้มหรือฤดูกาลได้ดี และจากรูปที่ 10 เมื่อพิจารณา ลักษณะของกราฟค่าพยากรณ์ของตัวแบบ glm สามารถสรุปเพิ่มเติมได้ว่า ตัวแบบ glm ที่สร้างจากวิธีการวิเคราะห์ความถดถอยเชิงเส้น ไม่เหมาะสมสำหรับนำมาใช้ในการพยากรณ์อนุกรมเวลาที่มีการเคลื่อนไหวเนื่องจากแนวโน้มหรือฤดูกาล เพราะไม่สามารถตรวจจับการเคลื่อนไหวของอนุกรมเวลา เนื่องจากแนวโน้มหรือฤดูกาลได้ นอกจากนี้ การวิจัยนี้ แสดงให้เห็นถึงแนวทางการวิเคราะห์อนุกรมเวลา โดยการเขียนโปรแกรมด้วยภาษาอาร์ ซึ่งช่วยให้นักวิจัยสามารถวิเคราะห์อนุกรมเวลา ได้ง่าย สะดวก และรวดเร็วขึ้น โดยลดขั้นตอนการวิเคราะห์ข้อมูลบางขั้นตอน เช่น ขั้นตอนการสร้างตัวแบบ ARIMA ต้องมีการตรวจสอบสภาวะหนึ่ง และกำหนดรูปแบบของตัวแบบ ARIMA แต่ในภาษาอาร์ มีฟังก์ชัน `auto.arima()` ที่ใช้กำหนดรูปแบบของตัวแบบ ARIMA โดยอัตโนมัติ ทำให้ผู้วิจัยไม่ต้องวิเคราะห์อนุกรมเวลา เพื่อกำหนดรูปแบบของตัวแบบ ARIMA อีกทั้งภาษาอาร์มีคำสั่งสร้างกราฟ ทำให้สามารถวิเคราะห์ข้อมูลจากกราฟได้ง่ายขึ้น

### References

- Bergmeir, C., and Berítez, J.M. (2012). On the Use of Cross-Validation for Time Series Predictor Evaluation. *Information Sciences*. Vol. 191. pp. 192-213
- Boldina, I., and Beninger, P.G. (2016). Strengthening Statistical Usage in Marine Ecology: Linear Regression. *Journal of Experimental Marine Biology and Ecology*. Vol. 474. pp. 81-91
- Bunn, D.W., and Farmer, E.D. (1985). *Comparative Models for Electrical Load Forecasting*. New York: John Wiley and Sons
- Chujai, P., Kerdprasop, N. and Kerdprasop, K. (2013). Time Series Analysis of Household Electric Consumption with ARIMA and ARMA Models. *Proceedings of the International MultiConference of Engineers and Computer Scientists*. China : Hong Kong. pp. 295-300.

- Keka, I., and Hamiti, M. (2013). Load profile analyses using R language. *Proceedings of the ITI 2013 35<sup>th</sup> International Conference on Information Technology Interfaces (ITI)*. Croatia : Cavtat. pp. 245-250
- Maçaira, P.M., Souza, R.C. and Oliveira, F.L.C. (2015). Modelling and Forecasting the Residential Electricity Consumption in Brazil with Pegels Exponential Smoothing Techniques. *Procedia Computer Science*. Vol. 55. p. 328-335
- Metropolitan Electricity Authority. (2015). *Report of Electrical Distribution*. Access (10 August 2015). Available (<http://www.meo.or.th/download/index.php#>)
- Pereira, C.M., Almeida de, N.N. and Velloso, M.L.F. (2015). Fuzzy Modeling to Forecast an Electric Load Times Series. *Procedia Computer Science*. Vol. 55. pp. 395-404
- Prema, V. and Rao, K.U. (2015). Development of Statistical Time Series Models for Solar Power Prediction. *Renewable Energy*. Vol. 83. pp. 100-109
- Wang, Y., Wang, J., Zhao, G. and Dong, Y. (2012). Application of Residual Modification Approach in Seasonal ARIMA for Electricity Demand Forecasting: A Case Study of China. *Energy Policy*. Vol. 48. pp. 284-294





แบบแจ้งผลการพิจารณาผลงานทางวิชาการ  
เพื่อตีพิมพ์เผยแพร่ในวารสารวิทยาศาสตร์และเทคโนโลยี  
มหาวิทยาลัยอุบลราชธานี

วันที่ 5 ก.ย. 2559

ตามที่ ผู้ช่วยศาสตราจารย์รณชัย ชื่นธวัช รองศาสตราจารย์กิตติศักดิ์ เกิดประสพ และ  
รองศาสตราจารย์นิตยา เกิดประสพ ได้ส่งผลงานวิชาการ เรื่อง “การพยากรณ์ความต้องการใช้งานหน่วย  
จำหน่ายไฟฟ้าด้วยซอฟต์แวร์วิเคราะห์โครงข่ายแบบตรวจสอบสลับ ๓ ส่วน” เพื่อตีพิมพ์เผยแพร่ในวารสาร  
วิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยอุบลราชธานี นั้น

มหาวิทยาลัยอุบลราชธานี ได้พิจารณานำผลงานดังกล่าวเผยแพร่ในวารสารวิทยาศาสตร์และ  
เทคโนโลยี มหาวิทยาลัยอุบลราชธานี ปีที่ ๑๙ ฉบับที่ ๓ กันยายน - ธันวาคม ๒๕๖๐

  
(ผู้ช่วยศาสตราจารย์อัญชลี สำเภา)

ผู้ช่วยอธิการบดีฝ่ายปฏิบัติการวิจัย ปฏิบัติราชการแทน  
อธิการบดีมหาวิทยาลัยอุบลราชธานี

สำนักงานส่งเสริมบริหารงานวิจัยฯ สำนักงานอธิการบดี มหาวิทยาลัยอุบลราชธานี

โทรศัพท์ ๐๔๕-๓๕๓๐๓๕ โทรสาร ๐๔๕-๓๕๓๐๔๒ Email address: [ubuscij@ubu.ac.th](mailto:ubuscij@ubu.ac.th)

การพยากรณ์ความต้องการใช้งานหน่วยจำหน่ายไฟฟ้าด้วยซัพพอร์ตเวกเตอร์เกรสชัน  
แบบตรวจสอบสลับ 3 ส่วน

The Demand Forecasting of Electrical Power Distribution Units by Support Vector  
Regression with 3 Folds Cross-Validation

รณชัย ชื่นรัชัช กิตติศักดิ์ เกิดประสพ และนิตยา เกิดประสพ

สาขาวิชาวิศวกรรมคอมพิวเตอร์ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จ. นครราชสีมา 30000

\*E-mail: c\_ronnachai@hotmail.com

บทคัดย่อ

การพยากรณ์ความต้องการใช้ไฟฟ้าได้อย่างแม่นยำ จะทำให้เกิดการวางแผนผลิตกระแสไฟฟ้า ได้อย่างมีประสิทธิภาพ ส่งผลให้ต้นทุนการผลิตกระแสไฟฟ้าลดลง และค่าใช้ไฟฟ้าของผู้บริโภคลดลงได้ ดังนั้น การวิจัยนี้ จึงมีวัตถุประสงค์เพื่อพยากรณ์หน่วยจำหน่ายไฟฟ้า ที่สะท้อนถึงความต้องการใช้ไฟฟ้า ด้วยการสร้างตัวแบบการพยากรณ์จากเทคนิคซัพพอร์ตเวกเตอร์เกรสชันแบบตรวจสอบสลับ 3 ส่วน (Support Vector Regression: SVR with 3 Folds Cross-Validation) จากนั้นเปรียบเทียบความแม่นยำของการพยากรณ์กับตัวแบบการถดถอยเชิงเส้นแบบพหุ (Multiple Linear Regression: MLR) และตัวแบบ ARIMA (Autoregressive Integrated Moving Average) ซึ่งข้อมูลที่น่าวิเคราะห์ มีลักษณะเป็นอนุกรมเวลา โดยเป็นข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง (กฟน.) ที่เก็บรวบรวมข้อมูลจากรายงานสถานการณ์การจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง ตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนพฤษภาคม ปีพ.ศ. 2558 และข้อมูลที่สนใจ มีขอบเขตเฉพาะจำนวนหน่วยจำหน่ายไฟฟ้าของผู้บริโภคประเภทบ้านอยู่อาศัย โดยนำข้อมูลตั้งแต่เดือนมกราคม ปีพ.ศ. 2553 ถึงเดือนธันวาคม ปีพ.ศ. 2557 ไปสร้างตัวแบบการพยากรณ์ด้วยภาษาอาร์ จากนั้น ใช้ข้อมูลตรวจสอบ ตั้งแต่เดือนมกราคมถึงเดือนพฤษภาคม ปีพ.ศ. 2558 เพื่อเปรียบเทียบความแม่นยำในการพยากรณ์ของตัวแบบทั้ง 3 ด้วยการวัดค่าคลาดเคลื่อน ได้แก่ ค่า Root Mean Square Error (RMSE) และค่า Mean Absolute Percentage Error (MAPE) จากผลการทดลอง พบว่า ตัวแบบ SVR เป็นตัวแบบที่ให้ค่า RMSE และค่า MAPE ต่ำที่สุด ดังนั้น ตัวแบบ SVR เป็นตัวแบบที่เหมาะสมที่สุดสำหรับนำมาใช้พยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวงกับผู้บริโภคประเภทบ้านอยู่อาศัย

คำสำคัญ: ซัพพอร์ตเวกเตอร์เกรสชัน การตรวจสอบสลับ การถดถอยเชิงเส้นแบบพหุ ตัวแบบอาร์มา อนุกรมเวลา  
การพยากรณ์ความต้องการใช้งานไฟฟ้า

Abstract

The accurate forecasting of electricity usage makes the efficiently planning about generating electricity, will reduce the electricity generating cost, moreover the electricity usage charge of customer will decrease as well. Therefore, this research applied the forecasting of electrical power distribution unit that indicated to the demand of electricity usage by creating the model of Support Vector Regression (SVR) with 3 folds cross-validation and compare the forecasting accuracy with the model of Multiple Linear Regression (MLR) and the model of Autoregressive Integrated Moving Average (ARIMA). The data set in this research is time series data in terms of units of household electricity usage were obtained from the Metropolitan Electricity Authority of Thailand. The data had been collected monthly from January 2010 to May 2015. Forecasting models with different parameters are generated from three techniques using the training data, which are the series from January 2010 to December 2014 and each model is implemented with R language. The remaining data from January 2015 to May 2015 are employed as the validating data. Forecasting performance of each model is measured by the root mean square

error (RMSE) and the mean absolute percentage error (MAPE) metrics. When we compare RMSE and MAPE of three models, the experimental result reveals that SVR model has lowest RMSE and MAPE. As a result, we can conclude that SVR model is the most powerful model to forecast this time series data.

**Keywords:** Support Vector Regression; Cross-validation; Multiple Linear Regression; ARIMA model; Time series; Electricity Usage Demand Forecasting

### 1. บทนำ

การพยากรณ์ปริมาณการใช้ไฟฟ้าของผู้บริโภคได้อย่างแม่นยำ จะช่วยวางแผนด้านทรัพยากร จัดสรรงบประมาณ และลดค่าใช้จ่ายในการดำเนินการลงได้ โดยสอดคล้องกับงานวิจัยของ Bunn และ Farmer [1] ที่ได้ศึกษาการพยากรณ์ความต้องการไฟฟ้า พบว่า ค่าใช้จ่ายในการดำเนินการจะเพิ่มขึ้น 10 ล้านปอนด์ ทุกปี เมื่อมีการพยากรณ์ผิดพลาดไปทุก 1% นอกจากนี้ ยังมีงานวิจัยในอดีต [2] [3] [7] ที่กล่าวว่า การพยากรณ์ปริมาณการใช้ไฟฟ้ามากกว่าค่าจริง อาจทำให้ผลิตพลังงานไฟฟ้ามากกว่าความต้องการจริง เป็นผลให้สูญเสียค่าใช้จ่ายโดยไม่จำเป็น แต่ถ้ายากรณ์ปริมาณการใช้ไฟฟ้าน้อยกว่าค่าจริง อาจทำให้ผลิตพลังงานไฟฟ้าไม่เพียงพอต่อความต้องการ เป็นผลให้เกิดความสูญเสียทางเศรษฐกิจ การดำเนินการที่สูญเสียเวลาและภาพลักษณ์ ดังนั้น งานวิจัยนี้ จึงนำเสนอการพยากรณ์หน่วยจำหน่ายไฟฟ้า ซึ่งเป็นข้อมูลการจำหน่ายไฟฟ้าของการไฟฟ้านครหลวงแห่งประเทศไทย ที่จำหน่ายให้กับผู้บริโภคที่อยู่ในเขตพื้นที่จังหวัด กรุงเทพมหานคร สมุทรปราการ และนนทบุรี ประเภทบ้านอยู่อาศัย โดยเก็บรวบรวมข้อมูลจากรายงานสถานการณ์การจำหน่ายไฟฟ้าประจำเดือน ตั้งแต่เดือนมกราคม 2553 ถึงเดือนพฤษภาคม 2558 [4] ซึ่งหน่วยจำหน่ายไฟฟ้า 1 หน่วย หมายถึง การใช้ไฟฟ้า 1,000 วัตต์ต่อชั่วโมง ดังนั้น หน่วยจำหน่ายไฟฟ้าจึงเป็นข้อมูลที่สะท้อนถึงความต้องการใช้ไฟฟ้าของผู้บริโภค

ข้อมูลหน่วยจำหน่ายไฟฟ้าที่ทำการศึกษา มีลักษณะเป็นอนุกรมเวลา ที่บันทึกข้อมูลเป็นรายเดือน ตามลำดับเวลา โดยการพยากรณ์ข้อมูลที่มีลักษณะเป็นอนุกรมเวลา จะดำเนินการสร้างตัวแบบการพยากรณ์ จากนั้นวัดความแม่นยำในการพยากรณ์ของตัวแบบ ด้วยการวัดค่าคลาดเคลื่อน (Residual) ซึ่งการสร้างตัวแบบการพยากรณ์มีหลากหลายเทคนิค และงานวิจัยในอดีต ที่ศึกษาเกี่ยวกับการสร้างตัวแบบการพยากรณ์ข้อมูลอนุกรมเวลา มีหลากหลายงานวิจัย ที่สามารถนำมากล่าวถึงได้ ดังนี้

Wang และคณะ [5] ศึกษาการปรับปรุงความแม่นยำของค่าพยากรณ์ความต้องการไฟฟ้าในเขตไฟฟ้าภาคตะวันตกเฉียงเหนือของประเทศจีน ที่ได้จากตัวแบบ Seasonal ARIMA ด้วยการปรับปรุงค่าคลาดเคลื่อน ผลการศึกษา พบว่า การปรับปรุงค่าคลาดเคลื่อนให้ความแม่นยำเพิ่มขึ้น

Lee และ Tong [6] ศึกษาการพยากรณ์อนุกรมเวลา ซึ่งเป็นข้อมูลการบริโภคพลังงานไฟฟ้ารายปีในจีน โดยนำเสนอตัวแบบผสมผสานระหว่างตัวแบบ ARIMA และ Genetic Programming (GP) ผลการศึกษา พบว่า วิธีการผสมผสานระหว่างตัวแบบ ARIMA และ GP มีความแม่นยำในการพยากรณ์มากกว่าการใช้ตัวแบบ ARIMA หรือตัวแบบ GP เพียงอย่างเดียว

Kaytez และคณะ [7] ศึกษาการพยากรณ์ปริมาณการบริโภคพลังงานไฟฟ้าในตุรกี ด้วยการเปรียบเทียบเทคนิค Artificial Neural Networks (ANN), Multiple Linear Regression (MLR) และ Least Squares Support Vector Machines (LSSVM) โดยวัดความแม่นยำของการพยากรณ์ด้วยค่า MAPE, MSE, RMSE และ SSE ผลการศึกษา พบว่า ตัวแบบที่สร้างจากเทคนิค LSSVM มีความแม่นยำในการพยากรณ์มากที่สุด

Ogcu, Demirel และ Zaim [8] ศึกษาการพยากรณ์ปริมาณการบริโภคพลังงานไฟฟ้าในประเทศตุรกี ซึ่งเป็นอนุกรมเวลาที่มีคาบเวลาเป็นรายเดือน โดยใช้เทคนิค ANN และ SVR จากนั้น เปรียบเทียบความแม่นยำของตัวแบบ ANN และ SVR ด้วยการวัดค่าคลาดเคลื่อน MAPE ผลการศึกษา พบว่า ตัวแบบ SVR มีความแม่นยำในการพยากรณ์มากกว่าตัวแบบ ANN

Wang และคณะ [9] ศึกษาการพยากรณ์โหลดไฟฟ้าประจำปีของเมืองปักกิ่ง ประเทศจีน ซึ่งเป็นอนุกรมเวลาที่มีคาบเวลาเป็นรายปี โดยใช้เทคนิค SVR ร่วมกับการคัดเลือกพารามิเตอร์ที่เหมาะสมด้วย Differential Evolution Algorithm และ เรียกว่า เทคนิค DESVR จากนั้น เปรียบเทียบความแม่นยำของการพยากรณ์กับตัวแบบที่

สร้างจากเทคนิค Back Propagation Neuron Network (BPNN) และเทคนิค Regression ผลการศึกษา พบว่า ตัวแบบ DESVR มีความแม่นยำในการพยากรณ์มากที่สุด

จากการทบทวนงานวิจัยในอดีต พบว่า วิธีการสร้างตัวแบบพยากรณ์ ด้วยเทคนิคซัพพอร์ตเวกเตอร์รีเกรสชัน (Support Vector Regression) ซึ่งเป็นวิธีการวิเคราะห์ความถดถอยระหว่างอินพุตที่อยู่ในรูปของเวกเตอร์กับตัวแปรเอาต์พุต ด้วยหลักการด้านแมชชีนเลิร์นนิง (Machine learning) ได้ถูกนำมาใช้ในงานวิจัยอย่างแพร่หลายมากขึ้น เพราะเป็นวิธีการสร้างตัวแบบการพยากรณ์ที่เป็นทางเลือกที่ดี (Alternative modeling) เนื่องจากให้ความแม่นยำสูง [7], [8], [9], [10] และเป็นวิธีการที่ประมวลผลได้รวดเร็ว เหมาะสมกับชุดข้อมูลที่มีขนาดเล็ก [10] ดังนั้นงานวิจัยนี้ จึงมีวัตถุประสงค์ เพื่อศึกษาการสร้างตัวแบบซัพพอร์ตเวกเตอร์รีเกรสชันแบบตรวจสอบสลับ 3 ส่วน สำหรับพยากรณ์หน่วยจำหน่ายไฟฟ้าประเภทบ้านอยู่อาศัยของการไฟฟ้านครหลวงแห่งประเทศไทย จากนั้นเปรียบเทียบความแม่นยำของการพยากรณ์กับตัวแบบการถดถอยเชิงเส้นแบบพหุและตัวแบบ ARIMA ด้วยการวัดค่าคลาดเคลื่อน RMSE และ MAPE

**2. วัสดุอุปกรณ์และวิธีดำเนินการวิจัย**

งานวิจัยนี้ มีการดำเนินการวิจัยด้วยการสร้างตัวแบบการพยากรณ์ โดยใช้วิธีการที่แตกต่างกัน 3 วิธี ดังนั้น จึงแบ่งวิธีการดำเนินการวิจัย เป็น 3 ส่วน ดังนี้

**2.1 ตัวแบบซัพพอร์ตเวกเตอร์รีเกรสชัน**

ซัพพอร์ตเวกเตอร์รีเกรสชัน (SVR) เป็นเทคนิคที่ใช้วิธีการของซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine: SVM) ที่นำเสนอโดย Vapnik [11] มาวิเคราะห์ความถดถอยระหว่างอินพุตเวกเตอร์ (Input vector) และตัวแปรเอาต์พุต (Output variables) ซึ่งนำมาใช้กับการพยากรณ์อนุกรมเวลาได้ โดยเปลี่ยนการจำแนกคลาสด้วย SVM เป็นการทำนายค่าด้วย SVR ที่มีเป้าหมาย คือ ต้องการค้นหาความสัมพันธ์เชิงเส้นระหว่างอินพุตเวกเตอร์ในมิติขนาด  $n$  ( $X \in R^n$ ) และตัวแปรเอาต์พุต ( $y \in R$ ) และเนื่องจาก SVR ตัดแปลงมาจาก SVM ดังนั้น สมการความถดถอยของ SVR จึงคล้ายคลึงกับสมการไฮเปอร์เพลน (Hyperplane) ของ SVM โดยสมการความถดถอยของ SVR แสดงได้ดังสมการที่ 1 [12]

$$f(x) = w^T x + b \tag{1}$$

เมื่อ  $w$  และ  $b$  เป็นความชันและออฟเซต (Offset) ของเส้นความถดถอย โดยการกำหนดค่าของ  $w$  และ  $b$  ทำได้ด้วยการหาค่าต่ำสุดของสมการที่ 2 [12]

$$R = \frac{1}{2} \|w\|^2 + \frac{C}{l} \sum_{i=1}^l |y_i - f(x_i)|_\epsilon \tag{2}$$

การใช้เทคนิค SVR ทำนายค่าเอาต์พุตจากอินพุตเวกเตอร์ จะมีการสร้างท่อเอปซิลอน (Epsilon tube) โดยใช้ฟังก์ชันสูญเสีย (Loss function) ที่มีหลากหลายรูปแบบ ในงานวิจัยนี้ใช้ฟังก์ชันสูญเสียแบบเอปซิลอนอินเซนซีฟ ( $\epsilon$ -insensitive loss function) ที่นำเสนอโดย Vapnik [11] เพราะเป็นฟังก์ชันสูญเสีย ที่นิยมใช้ในงานวิจัยอย่างแพร่หลาย [2] [9] [10] [12] [14] โดยแสดงได้ดังสมการที่ 3

$$|y_i - f(x_i)|_\epsilon = \begin{cases} 0 & \text{if } |y_i - f(x_i)| \leq \epsilon \\ |y_i - f(x_i)| - \epsilon & \text{Otherwise} \end{cases} \tag{3}$$

การแก้ปัญหาของสมการที่ 2 ที่มีเงื่อนไขตามสมการที่ 3 สามารถปรับให้อยู่ในรูปแบบการแก้ปัญหาแบบคู่อัล (Dual problem) ด้วยการใช้ตัวคูณลากรองจ์ (Lagrange multipliers) ดังสมการที่ 4-5 [12]

$$\begin{aligned} & \text{Maximize } L_p(\alpha_i, \alpha_i^*) \\ & = -\frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l (\alpha_i - \alpha_j^*) (\alpha_j - \alpha_i^*) x_i^T x_j \\ & - \epsilon \sum_{i=1}^l (\alpha_i + \alpha_i^*) + \sum_{i=1}^l (\alpha_i - \alpha_i^*) y_i \end{aligned} \tag{4}$$

$$\text{Subject to } \begin{cases} \sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0 \\ 0 \leq \alpha_i \leq C, & i = 1, \dots, l \\ 0 \leq \alpha_i^* \leq C, & i = 1, \dots, l \end{cases} \tag{5}$$

เมื่อ  $\alpha_i, \alpha_i^*$  เป็นตัวคูณลากรองจ์  $C$  เป็นจำนวนเต็มที่เป็นค่าคงที่ ซึ่งเป็นค่าใช้จ่าย (Cost) เมื่อมีข้อผิดพลาด (Error) เกิดขึ้น  $\epsilon$  เป็นความกว้างของท่อเอปซิลอน หรือเป็นความคลาดเคลื่อนของชุดข้อมูลฝึกสอน และ  $l$  คือจำนวนของซัพพอร์ตเวกเตอร์ (Support vector) ซึ่งอินพุตเวกเตอร์ที่เป็นซัพพอร์ตเวกเตอร์ จะมี  $\alpha_i, \alpha_i^* > 0$  ส่วนอินพุตเวกเตอร์ที่ไม่ใช่ซัพพอร์ตเวกเตอร์ จะมี  $\alpha_i, \alpha_i^* = 0$  และหลังจากที่คำนวณค่า  $\alpha_i$  และ  $\alpha_i^*$  จากชุดข้อมูลฝึกสอน จะสามารถสร้างสมการ SVR เพื่อใช้ทำนายค่าเอาต์พุตจากอินพุตเวกเตอร์ ได้ดังสมการที่ 6 [12]

$$f(x) = w_0^T x + b = \sum_{i=1}^l (\alpha_i - \alpha_i^*) x_i^T x + b \tag{6}$$

โดยที่ เวกเตอร์ถ่วงน้ำหนัก ( $w_0$ ) เป็นดังสมการที่ 7

$$w_0 = \sum_{i=1}^l (\alpha_i - \alpha_i^*) x_i \tag{7}$$

สมการที่ 6 อยู่ในรูปแบบสมการถดถอยเชิงเส้น แต่ในกรณีที่เป็นการถดถอยแบบไม่เป็นเชิงเส้น สามารถส่ง



อินพุตเวกเตอร์ไปยังมิติที่สูงขึ้น (High dimensional feature space) โดยใช้เคอร์เนลฟังก์ชัน (Kernel function) ซึ่งเคอร์เนลฟังก์ชันที่นิยมใช้ใน SVR มีดังนี้ [12]

(1) ลีเนียร์เคอร์เนล (Linear kernel):  $k(x_i, x) = x_i^T x$

(2) โพลีโนเมียลเคอร์เนล (Polynomial kernel):

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

(3) เกาส์เซียนเคอร์เนล (Gaussian (RBF) kernel):

$$k(x_i, x) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

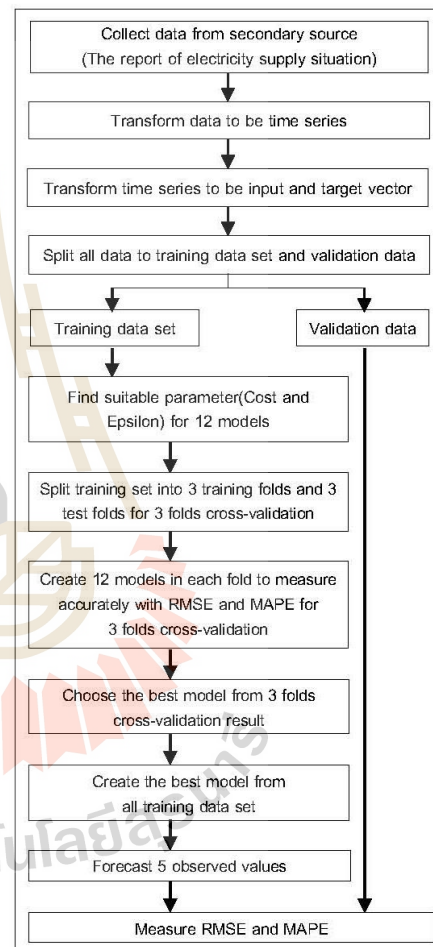
ดังนั้น สมการที่ 6 สามารถเขียนใหม่ในรูปแบบการถดถอยไม่เป็นเชิงเส้น โดยใช้เคอร์เนลฟังก์ชันได้ตั้งสมการที่ 8

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) k(x_i, x) + b \quad (8)$$

สำหรับงานวิจัยนี้ ดำเนินการเขียนโปรแกรมทุกขั้นตอน ด้วยการใช้ภาษา R โดยการสร้างตัวแบบพยากรณ์ด้วยเทคนิค SVR จะใช้ฟังก์ชัน svm() ในแพ็คเกจชื่อ e1071 ที่เผยแพร่โดย David Meyer และคณะ [13] ในส่วนเก็บข้อมูลของภาษา R (CRAN repository of R language) ซึ่งการนำ SVR ไปใช้กับการพยากรณ์อนุกรมเวลา จะต้องปรับชุดข้อมูลให้อยู่ในรูปแบบของอินพุตเวกเตอร์และทาร์เก็ต ( $D = \{(x_i, y_i)\}_{i=1}^n$ ) จากนั้นสร้างตัวแบบพยากรณ์จำนวน 12 ตัวแบบ โดยเหตุผลที่เลือกสร้างตัวแบบจำนวน 12 ตัวแบบ เนื่องจาก อนุกรมเวลาที่นำมาศึกษา มีคาบเวลาเป็นรายเดือน และมีจำนวนคาบเวลาใน 1 ฤดูกาล เท่ากับ 12 คาบเวลา ตามรอบปีปฏิทิน ซึ่งแต่ละตัวแบบ มีความแตกต่างที่จำนวนของอินพุตเวกเตอร์ นั่นคือ ตัวแบบที่ 1 มีอินพุตเวกเตอร์ 1 ตัว เป็นค่าสังเกตค่าหลัง 1 คาบเวลา ไล่ไปตามลำดับ จนถึงตัวแบบที่ 12 มีอินพุตเวกเตอร์ 12 ตัว เป็นค่าสังเกตค่าหลังตั้งแต่ 1 ถึง 12 คาบเวลา

การสร้างตัวแบบ SVR จะต้องกำหนดค่าพารามิเตอร์ C และ  $\epsilon$  โดยแพ็คเกจ e1071 มีฟังก์ชัน tune() [13] ที่ใช้สำหรับการกำหนดค่า C และ  $\epsilon$  ที่เหมาะสมที่สุด ดังนั้น จึงใช้ฟังก์ชัน tune() ในการกำหนดค่า C และ  $\epsilon$  ให้กับตัวแบบ SVR นอกจากนี้ ต้องระบุเคอร์เนลฟังก์ชันด้วย ซึ่งจากการสุ่มทดสอบสร้างตัวแบบ พบว่า ลีเนียร์เคอร์เนลให้ผลลัพธ์ที่ดีที่สุดกับข้อมูลชุดนี้ ดังนั้น งานวิจัยนี้ จึงใช้ลีเนียร์เคอร์เนลเป็นเคอร์เนลฟังก์ชัน และเพื่อการจัดความไม่แม่นยำในการวัดความแม่นยำของตัวแบบ จึงนำเสนอวิธีการตรวจสอบ

สลับ 3 ส่วน (3 folds cross-validation) สำหรับเลือกตัวแบบที่เหมาะสมที่สุด ในการนำมาใช้พยากรณ์อนุกรมเวลา โดยสรุปขั้นตอนการสร้างตัวแบบ SVR สำหรับพยากรณ์ข้อมูลอนุกรมเวลา เพื่อนำไปเปรียบเทียบกับตัวแบบ MLR และตัวแบบ ARIMA ได้ดังรูปที่ 1



รูปที่ 1. ขั้นตอนการสร้างตัวแบบ SVR ที่เหมาะสมที่สุด

จากรูปที่ 1. อธิบายแต่ละขั้นตอนได้ดังนี้

Step 1: Collect data from secondary source คือ ขั้นตอนรวบรวมข้อมูล จากข้อมูลทุติยภูมิที่ได้จากรายงานสถานการณ์การจำหน่ายไฟฟ้าประจำเดือน ของการไฟฟ้า

นครหลวง ตั้งแต่เดือนมกราคม 2553 ถึงเดือนพฤษภาคม 2558 ของผู้บริโภคประเภทบ้านอยู่อาศัย

Step 2: Transform data to be time series คือ ขั้นตอนให้นำข้อมูลทุติยภูมิที่รวบรวมได้ มาสร้างเป็นข้อมูลอนุกรมเวลา เรียงตามลำดับเวลา โดยมีเวลาเป็นรายเดือน แสดงได้ดังรูปที่ 2

Years	Months	Electric Units	No_Consumers	Mean_Units
2010	1	776.32	2420767	320.69
2010	2	836.16	2423046	345.09
2010	3	925.55	2433022	380.41
2010	4	1026.5	2439689	420.75
2010	5	1082.18	2448223	442.03
2010	6	1003.41	2456309	408.5
2010	7	930.91	2463398	377.9
2010	8	876.83	2473742	354.45

รูปที่ 2. ข้อมูลอนุกรมเวลาที่สร้างจากข้อมูลทุติยภูมิ

Step 3: Transform data to be input and target vector คือ ขั้นตอนการแปลงข้อมูลตามรูปที่ 2 ให้เป็นข้อมูลดังรูปที่ 3 โดยแต่ละคอลัมน์หมายถึงค่าสังเกตที่ล่าช้าไปตั้งแต่ 1 ถึง 12 คาบเวลา ( $y_{t-1}, y_{t-2}, \dots, y_{t-12}$ ) ตามลำดับ และคอลัมน์สุดท้ายคือ ค่าของทาร์เก็ต ซึ่งเป็นค่าสังเกต ณ เวลา  $t$  ( $y_t$ ) โดยค่าของทาร์เก็ตในแถวแรก คือ ค่าสังเกตที่ล่าช้าเวลา  $t=13$  เมื่อแปลงข้อมูลแล้ว จะได้ชุดข้อมูลจำนวน 53 ชุด จากอนุกรมเวลาเดิมที่มีค่าสังเกต 65 ค่า ตามจำนวนคาบเวลาตั้งแต่เดือนมกราคม 2553 ถึงเดือนพฤษภาคม 2558

ValT1	ValT2	ValT3	ValT4	ValT5	ValT6	ValT7	ValT8	ValT9	ValT10	ValT11	ValT12	Target
773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	1026.5	925.55	836.16	776.32	707.16
707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	1026.5	925.55	836.16	787.29
787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	1026.5	925.55	864.64
864.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	1026.5	864.58
864.58	864.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	968.86
968.86	864.58	864.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	933.59
933.59	968.86	864.58	864.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	912.8
912.8	933.59	968.86	864.58	864.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	902.59
902.59	912.8	933.59	968.86	864.58	864.64	787.29	707.16	773.99	787.33	820.6	873.47	886.08

รูปที่ 3. ลักษณะของชุดข้อมูลที่ใช้สร้างตัวแบบ SVR

Step 4: Split all data to training data set and validation data set คือ ขั้นตอนการแบ่งชุดข้อมูลที่ได้จากขั้นตอนที่ 3 ออกเป็น 2 ชุด ได้แก่ ชุดข้อมูลฝึกสอน (Training data set) จำนวน 48 ชุด และชุดข้อมูลตรวจสอบ (Validation data set) จำนวน 5 ชุด

Step 5: Find suitable parameter(Cost and Epsilon) for 12 models คือ ขั้นตอนการกำหนดค่า C และ  $\epsilon$  ที่เหมาะสมที่สุดของแต่ละตัวแบบ ซึ่งกำหนดตัวแบบไว้ 12

ตัวแบบ โดยตัวแบบที่ 1 มีอินพุตเวกเตอร์เป็นค่าสังเกตที่ล่าช้าไป 1 คาบเวลา ( $y_{t-1}$ ) , ตัวแบบที่ 2 มีอินพุตเวกเตอร์เป็นค่าสังเกตที่ล่าช้าไป 1 และ 2 คาบเวลา ( $y_{t-1}, y_{t-2}$ ) ได้ไปตามลำดับจนถึงตัวแบบที่ 12 มีอินพุตเวกเตอร์เป็นค่าสังเกตที่ล่าช้าไป 1, 2, ..., 12 คาบเวลา ( $y_{t-1}, y_{t-2}, \dots, y_{t-12}$ ) โดยการหาค่า C และ  $\epsilon$  ที่เหมาะสมที่สุดของแต่ละตัวแบบจะใช้ฟังก์ชัน tune() ซึ่งจากการทดลองสุ่มกับชุดข้อมูลฝึกสอน พบว่า ค่า C มีค่าอยู่ระหว่าง 1 ถึง 16 ส่วนค่า  $\epsilon$  มีค่าอยู่ระหว่าง 0 ถึง 1 ดังนั้น จึงกำหนดช่วงการค้นหาค่า C ตั้งแต่ 1 ถึง 16 และช่วงการค้นหาค่า  $\epsilon$  ตั้งแต่ 0 ถึง 1 เพิ่มขึ้นครั้งละ 0.001 จากนั้นเก็บค่า C และ  $\epsilon$  ที่เหมาะสมที่สุดของแต่ละตัวแบบไว้ใส่ไฟล์ เพื่อให้สามารถนำค่าไปใช้ต่อ ในขั้นตอนการเขียนคำสั่งสำหรับสร้างตัวแบบ SVR

Step 6: Split training set into 3 training folds and 3 test folds for 3 folds คือ ขั้นตอนการแบ่งชุดข้อมูลฝึกสอนจำนวน 48 ชุด ออกเป็น 3 ส่วน (Fold) ดังนั้น แต่ละส่วน จะมีจำนวนชุดข้อมูลเท่ากับ 16 ชุด สำหรับนำไปใช้เลือกโมเดลที่เหมาะสมที่สุดด้วยการตรวจสอบสลับแบบ 3 ส่วน และวิธีการแบ่งข้อมูลเป็น 3 ส่วน แสดงได้ดังรูปที่ 4



รูปที่ 4. การแบ่งชุดข้อมูลสำหรับการตรวจสอบสลับ 3 ส่วน

Step 7: Create 12 models in each fold to measure accurately with RMSE and MAPE for 3 folds cross-validation คือ ขั้นตอนการสร้างตัวแบบพยากรณ์ 12 ตัวแบบ โดยใช้ชุดข้อมูลที่แบ่งออกเป็น 3 ส่วน จากขั้นตอนที่ 6 จากนั้นวัดค่า RMSE และ MAPE กับชุดข้อมูลทดสอบ (Test data set) ของแต่ละตัวแบบ ในแต่ละส่วน เพื่อนำมาหาค่าเฉลี่ย และใช้เป็นเกณฑ์ในการเลือกตัวแบบที่เหมาะสมที่สุดสำหรับการนำมาพยากรณ์ค่าสังเกตล่วงหน้า 5 ช่วงเวลา

Step 8: Choose the best model from 3 folds cross-validation result คือ ขั้นตอนการเลือกตัวแบบที่เหมาะสมที่สุดด้วยการพิจารณาจากค่าเฉลี่ย RMSE และ MAPE ของทั้ง 3 ส่วน



Step 9: Create the best model from all training data set คือ ขั้นตอนการสร้างตัวแบบที่ดีที่สุดจากชุดข้อมูลฝึกสอนจำนวน 48 ชุด เพื่อนำไปใช้ในการพยากรณ์ค่าสังเกตล่วงหน้า 5 ชั่วโมง ด้วยอินพุตเวกเตอร์ของชุดข้อมูลตรวจสอบที่แบ่งไว้ จำนวน 5 ชุด

Step 10: Forecast 5 observed values คือ ขั้นตอนการใช้ตัวแบบที่ดีที่สุดพยากรณ์ค่าสังเกต 5 ค่า ซึ่งเป็นค่าสังเกตของเดือนมกราคมถึงเดือนพฤษภาคม 2558

Step 11: Measure RMSE and MAPE คือ ขั้นตอนการนำค่าพยากรณ์จากขั้นตอนที่ 10 ไปเปรียบเทียบกับชุดข้อมูลตรวจสอบ ที่แบ่งไว้ในขั้นตอนที่ 4 โดยเป็นค่าสังเกตในช่วงเวลาเดียวกับค่าพยากรณ์ เพื่อวัดค่า RMSE และ MAPE ของตัวแบบ SVR ที่เหมาะสมที่สุด

สำหรับบางส่วนของชุดค่าสังเกตที่สำคัญ ตามขั้นตอนการสร้างตัวแบบ SVR ที่เหมาะสมที่สุด (รูปที่ 1) แสดงได้ดังรูปที่ 5

```
AllData<-read.csv(file="d:/CodeSVR/Data/EuseTran.csv" , header=TRUE, sep=",")
RowTest=5
RowTrain=dim(AllData)[1] - RowTest
TrainSet<-matrix(nrow=RowTrain, ncol=13)
k=1
for(i in 1:RowTrain) {
  for(j in 1:13) {
    TrainSet[k,j]=AllData[i,j]
    k=k+1
  }
}
TrainSet<-data.frame(TrainSet)
names(TrainSet)<-c("ValT1","ValT2","ValT3","ValT4","ValT5","ValT6","ValT7","ValT8","ValT9",
,"ValT10","ValT11","ValT12","Target")
write.csv(TrainSet,file="d:/CodeSVR/Data/TrainSVR.csv", row.names =FALSE)
```

(a) Command set for transform data to training data set

```
InV2<-data.frame(Trainload$ValT1,Trainload$ValT2)
tune2 <- tune(svm,InV2, Trainload$Target, data = data,ranges = list(epsilon = seq(0,1,0.001), cost = 1:16))
```

(b) Command set with using tune() function for model2

```
InVM2F1 <- data.frame(TrainLoadF1$ValT1,TrainLoadF1$ValT2)
model2F1 <- svm(x=InVM2F1,y=TrainLoadF1$Target,kernel="linear",cost=TuneLoad[2,3],epsilon=TuneLoad[2,4])
newdataM2F1 = data.frame(Val1=TestLoadF1$ValT1,Val2=TestLoadF1$ValT2)
PredM2F1 <- predict(model2F1,newdata=newdataM2F1)
```

(c) Command set for create model2 in first fold by using svm() function

รูปที่ 5. บางส่วนของชุดค่าสังเกตสำหรับสร้างตัวแบบ SVR

## 2.2 ตัวแบบการถดถอยเชิงเส้นแบบพหุ

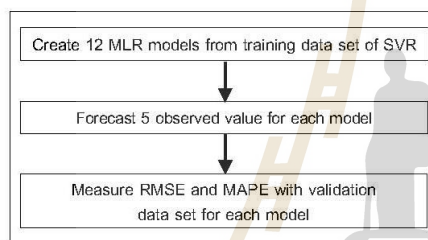
การวิเคราะห์การถดถอยเชิงเส้นแบบพหุ (Multiple Linear Regression: MLR) เป็นวิธีการวิเคราะห์ความสัมพันธ์ระหว่างกลุ่มของตัวแปรอิสระที่มีมากกว่า 1 ตัวแปรและตัวแปรตาม 1 ตัวแปร โดยตัวแบบพยากรณ์ที่สร้างจากวิธีการ MLR เป็นตัวแบบที่อยู่ในรูปแบบสมการเชิงเส้น ที่แสดง

ความสัมพันธ์ระหว่างกลุ่มของตัวแปรอิสระและตัวแปรตาม ดังสมการที่ 9 [14]

$$y = \beta_0 + \sum_{i=1}^k \beta_i x_i + \varepsilon \quad (9)$$

การนำวิธี MLR มาวิเคราะห์อนุกรมเวลาเชิงเดี่ยว ที่มีค่าสังเกตค่าเดียว ในแต่ละคาบเวลา ทำให้ตัวแปรตาม  $y$  หมายถึง ค่าสังเกต ณ.เวลา  $t$  ส่วนกลุ่มของตัวแปรอิสระ  $x_i$  ( $i=1, 2, \dots, k$ ) หมายถึง ค่าสังเกตที่ล่าช้าไป  $1, 2, \dots, k$

คาบเวลา ซึ่ง  $\beta_i$  ( $i=1, 2, \dots, k$ ) คือ ค่าสัมประสิทธิ์ความถดถอย (Regression Coefficient) ของ  $x_i$  และ  $\mathcal{E}$  คือค่าคลาดเคลื่อน (Residual) เมื่อ  $x_t$  หมายถึง ค่าสังเกตที่ลำดับที่  $t$  ดังนั้น ข้อมูลที่ใช้ในการสร้างตัวแบบ MLR จึงอยู่ในรูปแบบ  $D = \{(x_i, y_i)\}_{i=1}^n$  เช่นเดียวกับข้อมูลที่ใช้สร้างตัวแบบ SVR จึงใช้ชุดข้อมูลฝึกสอนของตัวแบบ SVR มาสร้างตัวแบบ MLR และใช้ชุดข้อมูลตรวจสอบของตัวแบบ SVR มาใช้วัดค่าคลาดเคลื่อน RMSE และ MAPE ของตัวแบบ MLR เพื่อนำไปเปรียบเทียบกับตัวแบบ SVR และตัวแบบ ARIMA โดยสรุปขั้นตอนการสร้างตัวแบบ MLR สำหรับพยากรณ์ข้อมูลอนุกรมเวลา 5 คาบเวลา เป็นดังรูปที่ 6



รูปที่ 6. ขั้นตอนการสร้างตัวแบบ MLR

จากรูปที่ 6 อธิบายแต่ละขั้นตอนได้ดังนี้

Step 1: Create 12 MLR models from training data set of SVR คือ ขั้นตอนการสร้างตัวแบบ MLR 12 ตัวแบบ โดยตัวแบบที่ 1 มีตัวแปรอิสระ 1 ตัวแปร และเป็นค่าสังเกตที่ลำดับที่ 1 คาบเวลา ( $y_{t-1}$ ) , ตัวแบบที่ 2 มีตัวแปรอิสระ 2 ตัวแปร และเป็นค่าสังเกตที่ลำดับที่ 1 และ 2 คาบเวลา ( $y_{t-1}, y_{t-2}$ ) ไปไปตามลำดับจนถึงตัวแบบที่ 12 มีตัวแปรอิสระ 12 ตัวแปร และเป็นค่าสังเกตที่ลำดับที่ 1, 2, ..., 12 คาบเวลา ( $y_{t-1}, y_{t-2}, \dots, y_{t-12}$ ) ซึ่งใช้ชุดข้อมูลฝึกสอนของตัวแบบ SVR ในการสร้างตัวแบบ MLR

Step 2: Forecast 5 observed values คือ ขั้นตอนการนำแต่ละตัวแบบ พยากรณ์ค่าสังเกตล่วงหน้า 5 คาบเวลา ซึ่งเป็นค่าสังเกตของเดือนมกราคมถึงเดือนพฤษภาคม 2558 สำหรับนำไปเปรียบเทียบกับชุดข้อมูลตรวจสอบ ซึ่งเป็นค่าสังเกตในช่วงเวลาเดียวกัน

Step 3: Measure RMSE and MAPE with validation data set for each model คือ ขั้นตอนการวัดค่า RMSE และ MAPE ของแต่ละตัวแบบกับชุดข้อมูลตรวจสอบ เพื่อเลือกตัวแบบ MLR ที่เหมาะสมที่สุด จากนั้น จะได้นำไปเปรียบเทียบกับความแม่นยำของการพยากรณ์กับตัวแบบ SVR และตัวแบบ ARIMA

ตัวอย่างชุดคำสั่งในการสร้างตัวแบบ MLR ตัวแบบที่ 2 ซึ่งมีตัวแปรอิสระ 2 ตัว คือ  $y_{t-1}$  และ  $y_{t-2}$  แสดงได้ดังรูปที่ 7

```

formularMLR<-Trainload$Target ~Trainload$ValT1+Trainload$ValT2
modelMLR <- glm(formularMLR,family = gaussian())
Testload<-read.csv(file="TesiData2.csv")
newdata = data.frame(Val1=Testload$ValT1,Val2=Testload$ValT2)
ForeMLR<-modelMLR$coefficients[1]+modelMLR$coefficients[2]*newdata[1]+modelMLR$coefficients[3]*newdata[2]
err1 <- Testload$ActualValue - ForeMLR
err2 <- err1/Testload$ActualValue
mseMLRtest <- sqrt(mean(err1[,1]^2))
mapeMLRtest<-mean(abs(100*err2[,1]))
mseMLRtest          # show RMSE
mapeMLRtest         # show MAPE
    
```

รูปที่ 7. ชุดคำสั่งสร้างตัวแบบ MLR ตัวแบบที่ 2

**2.3 ตัวแบบ ARIMA**

ตัวแบบ ARIMA เป็นตัวแบบที่สร้างจากวิธีของบอซซ์ และเจนกินส์ (Box and Jenkins method) [15] โดยพยากรณ์ข้อมูลจากรูปแบบหลัก 2 รูปแบบ ได้แก่ รูปแบบ

Autoregressive (AR) คือ รูปแบบที่กำหนดว่า ค่าพยากรณ์ที่เวลาใดๆ ขึ้นอยู่กับค่าสังเกตก่อนหน้า และรูปแบบ Moving average (MA) คือ รูปแบบที่กำหนดว่า ค่าพยากรณ์ที่เวลาใดๆ ขึ้นอยู่กับค่าคลาดเคลื่อนก่อนหน้า และเรียก

รูปแบบ AR ร่วมกับรูปแบบ MA ว่า รูปแบบ ARMA ซึ่งเป็นรูปแบบการวิเคราะห์อนุกรมเวลาที่เป็นสเตชันนารี ในกรณีที่อนุกรมเวลาไม่เป็นสเตชันนารี จะต้องแปลงอนุกรมเวลาให้เป็นสเตชันนารี และเรียกรูปแบบที่ใช้วิเคราะห์อนุกรมเวลาที่แปลงให้เป็นสเตชันนารี ได้ว่า รูปแบบ ARIMA โดยมีรูปแบบทั่วไปที่เขียนในรูปของตัวดำเนินการย้อนกลับ (Backward shift operator: B) ในรูปสมการ [5] ดังนี้

$$\theta_p(B)\phi_p(B^S)(1-B)^d(1-B^S)^D Y_t = w_q(B)W_Q(B^S)a_t$$

เมื่อ  $BY_t = Y_{t-1}$  และ  $B^k Y_t = Y_{t-k}$  โดยที่

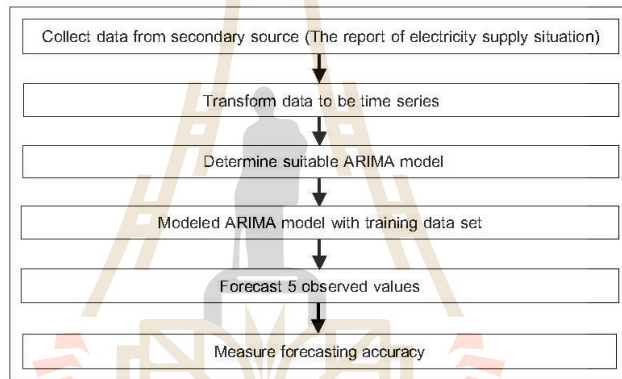
$$w_q(B) = 1 - w_1B - w_2B^2 - \dots - w_qB^q$$

$$\theta_p(B) = 1 - \theta_1B - \theta_2B^2 - \dots - \theta_pB^p$$

$$\phi_p(B^S) = 1 - \phi_1B^S - \phi_2B^{2S} - \dots - \phi_pB^{pS}$$

$$W_Q(B^S) = 1 - W_1B^S - W_2B^{2S} - \dots - W_QB^{QS}$$

สำหรับการสร้างตัวแบบ ARIMA ต้องวิเคราะห์อนุกรมเวลา เพื่อกำหนดรูปแบบ ARIMA(p, d, q)x(P, D, Q)<sub>s</sub> ที่เหมาะสมกับอนุกรมเวลา โดยเป็นขั้นตอนในการเลือกอันดับ p, d, q เนื่องจากแนวโน้ม และอันดับ P, D, Q เนื่องจากฤดูกาล ซึ่งภาษา R มีฟังก์ชัน auto.arima() ที่ใช้ในการระบุรูปแบบ ARIMA(p, d, q)x(P, D, Q)<sub>s</sub> ที่เหมาะสม ดังนั้น การพยากรณ์อนุกรมเวลาด้วยตัวแบบ ARIMA จึงมีขั้นตอนดังรูปที่ 8 และอธิบายแต่ละขั้นตอนได้ ดังนี้



รูปที่ 8. ขั้นตอนการสร้างตัวแบบ ARIMA

Step 1: Collect data from secondary source คือ ขั้นตอนรวบรวมข้อมูล จากข้อมูลทุติยภูมิที่ได้จากรายงานสถานการณ์การจำหน่ายไฟฟ้าประจำเดือนของการไฟฟ้านครหลวง เช่นเดียวกับขั้นตอนที่ 1 ของการสร้างตัวแบบ SVR

Step 2: Transform data to be time series คือ ขั้นตอนนำข้อมูลทุติยภูมิที่รวบรวมได้ มาสร้างเป็นข้อมูลอนุกรมเวลา เรียงตามลำดับเวลา โดยมีค่าเวลาเป็นรายเดือน เช่นเดียวกับขั้นตอนที่ 2 ของการสร้างตัวแบบ SVR ดังรูปที่ 2 จากนั้น แบ่งข้อมูลเป็น 2 ชุด ได้แก่ 1) ชุดข้อมูลฝึกสอน (Training data set) ซึ่งนำมาแปลงเป็นอนุกรมเวลาด้วยฟังก์ชัน ts() ในภาษา R เพื่อนำไปสร้างตัวแบบ ARIMA โดยใช้ข้อมูลตั้งแต่เดือนมกราคม 2553 ถึงเดือนธันวาคม 2557 สำหรับสร้างตัวแบบพยากรณ์ 2) ชุดข้อมูลตรวจสอบ (Validation data set) ซึ่งใช้ข้อมูลตั้งแต่เดือน

มกราคมถึงเดือนพฤษภาคม 2558 สำหรับวัดความถูกต้องของการพยากรณ์

Step 3: Determine suitable ARIMA คือขั้นตอนการกำหนดรูปแบบของตัวแบบ ARIMA ที่เหมาะสมกับชุดข้อมูลฝึกสอน โดยเป็นการกำหนดอันดับ p, d, q และอันดับ P, D, Q ของตัวแบบ ARIMA ด้วยฟังก์ชัน auto.arima()

Step 4: Modeled ARIMA model with training data set คือ ขั้นตอนการสร้างตัวแบบ ARIMA จากชุดข้อมูลฝึกสอน โดยนำค่าอันดับ p, d, q และอันดับ P, D, Q ที่ได้จากขั้นตอนที่ 3 มาใช้สร้างตัวแบบ ARIMA ด้วยฟังก์ชัน arima()

Step 5: Forecast 5 observed values คือ ขั้นตอนการนำตัวแบบ ARIMA พยากรณ์ค่าสังเกตล่วงหน้า 5 คาบเวลาด้วยฟังก์ชัน predict() ในภาษา R โดยค่าพยากรณ์เป็นค่าสังเกตของเดือนมกราคมถึงเดือนพฤษภาคม 2558 สำหรับ

นำไปเปรียบเทียบกันชุดข้อมูลตรวจสอบ ซึ่งเป็นค่าสังเกตในช่วงเวลาเดียวกัน

Step 6: Measure forecasting accuracy คือ ขั้นตอนการวัดความแม่นยำของการพยากรณ์ โดยนำค่าพยากรณ์ที่ได้จากตัวแบบมาเปรียบเทียบกับค่าจริงของชุดข้อมูลตรวจสอบ ด้วยการวัดค่าคลาดเคลื่อนทางสถิติ ได้แก่ ค่า RMSE (Root Mean Square Error) และค่า MAPE (Mean Absolute Percentage Error) [16] ซึ่งมีสมการดังนี้

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \quad (10)$$

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| 100 \frac{y_t - \hat{y}_t}{y_t} \right| \quad (11)$$

เมื่อ  $y_t$  = ค่าสังเกตที่เวลา  $t$ ,  $\hat{y}_t$  = ค่าพยากรณ์ที่เวลา  $t$  และ  $n$  = จำนวนช่วงเวลาที่ยพยากรณ์

สำหรับชุดคำสั่งทั้งหมดของภาษา R ตามขั้นตอนในรูปที่ 8 แสดงได้ดังรูปที่ 9.

```
library("forecast", lib.loc=~R/win-library/3.2")
arimaDF <- read.csv(file="d:/R/Euse2010_2014.csv", header=TRUE, sep=",")
arimaTS <- ts(arimaDF[,3], start=c(2010,1), end=c(2014,12), frequency=12)
auto.arima(arimaTS)
arimaM <- arima(arimaTS, order=c(p,d,q), list(order=c(P,D,Q), period=12))
# p,d,q and P,D,Q is an integer number, it's can be obtain by using auto.arima()
PredictARIMA <- predict(arimaM, n.ahead=5)
PredictARIMA # show forecasting value
TestSet<-read.csv(file="d:/R/Etest1_5_2015.csv")
ActualARIMA<-c(TestSet$Electric_Units)
ForecastARIMA<-c(PredictARIMA$pred[1:5])
err1 <- ActualARIMA-ForecastARIMA
err2 <- (ActualARIMA-ForecastARIMA)/ActualARIMA
mseARIMA <- sqrt(mean(err1^2))
mapeARIMA<-mean(abs(100*err2))
mseARIMA # show RMSE
mapeARIMA # show MAPE
```

รูปที่ 9. ชุดคำสั่งของภาษา R ตามขั้นตอนการสร้างตัวแบบ ARIMA

### 3. ผลการวิจัย

งานวิจัยนี้ ได้ศึกษาการวิเคราะห์อนุกรมเวลา เพื่อพยากรณ์ข้อมูลหน่วยจำหน่ายไฟฟ้าประเภทบ้านอยู่อาศัยของการไฟฟ้านครหลวงแห่งประเทศไทย ซึ่งได้ดำเนินการทดลอง ด้วยวิธีการสร้างตัวแบบพยากรณ์ที่แตกต่างกัน 3 วิธี ได้แก่ วิธีการสร้างตัวแบบ SVR วิธีการสร้างตัวแบบ MLR และวิธีการสร้างตัวแบบ ARIMA โดยใช้การเขียนโปรแกรมด้วยภาษา R เป็นเครื่องมือดำเนินการวิจัย ดังนั้นจึงแบ่งผลการวิจัยตามวิธีดำเนินการวิจัย จากนั้นเปรียบเทียบผลการวิจัยที่ได้จากวิธีดำเนินการวิจัยทั้ง 3 วิธี ซึ่งผลการวิจัยแต่ละส่วน เป็นดังนี้

#### 3.1 ผลการวิจัยจากตัวแบบ SVR

เทคนิค SVR เป็นวิธีการที่มีพารามิเตอร์ ซึ่งพารามิเตอร์ที่สำคัญคือ  $C$  และ  $\mathcal{E}$  โดย  $C$  คือ ค่าคงที่ ที่ใช้กำหนดความเหมาะสมระหว่างค่าคลาดเคลื่อนที่ยอมรับได้และขนาดของเวกเตอร์ถ่วงน้ำหนัก ( $\|w\|$ ) ส่วน  $\mathcal{E}$  คือ การกำหนดค่าความกว้างของท่อเอปซิลอน หรือค่าคลาดเคลื่อนที่ยอมรับได้ของชุดข้อมูลฝึกสอน ซึ่งงานวิจัยนี้ ใช้ฟังก์ชัน tune() ค้นหาค่า  $C$  และ  $\mathcal{E}$  ที่เหมาะสม โดยฟังก์ชัน tune() จะค้นหาค่าที่เหมาะสมแบบกริดเสิร์จ (Grid search) ตลอดช่วงของค่าพารามิเตอร์ที่กำหนดไว้ และประเมินความเหมาะสมด้วยการวัดค่า MSE (Mean Square Error) [13] ได้ผลลัพธ์ดังตารางที่ 1



ตารางที่ 1. ค่า C และ  $\epsilon$  ที่เหมาะสมจากการค้นหาด้วยฟังก์ชัน tune() ของตัวแบบ SVR

Model	C	$\epsilon$	Model	C	$\epsilon$
1	1	0.433	7	2	0.001
2	1	0	8	2	0.024
3	5	0.567	9	2	0.081
4	2	0.686	10	3	0.193
5	2	0.263	11	2	0.011
6	2	0.325	12	3	0.135

ตารางที่ 2. ผลการตรวจสอบสลับ 3 ส่วนของตัวแบบ SVR

Model	RMSE	RMSE	RMSE	RMSE	MAPE	MAPE	MAPE	MAPE
	fold 1	fold 2	fold 3	Average	fold 1	fold 2	fold 3	Average
1	90.13	66.06	74.43	76.88	7.17	5.28	6.97	6.47
2	80.02	56.49	89.28	75.26	6.55	4.03	8.12	6.24
3	75.45	65.62	88.10	76.39	6.45	5.50	8.20	6.72
4	77.03	70.52	80.04	75.86	6.70	5.99	7.29	6.66
5	75.15	63.70	103.87	80.91	6.74	5.09	9.34	7.06
6	80.58	65.95	104.83	83.79	7.02	5.38	9.75	7.38
7	64.73	54.29	110.28	76.43	5.85	4.07	11.05	6.99
8	62.73	56.17	119.19	79.36	5.63	4.80	12.41	7.61
9	56.61	57.00	106.40	73.34	5.22	4.81	10.84	6.96
10	49.99	55.80	99.17	68.32	4.70	4.61	9.91	6.41
11	49.19	49.22	96.53	64.98	4.65	4.14	9.69	6.16
12	54.20	54.89	93.60	67.56	5.01	4.38	9.26	6.22

เมื่อได้ค่า C และ  $\epsilon$  ที่เหมาะสมสำหรับแต่ละตัวแบบ จะนำไปสร้างตัวแบบ SVR จำนวน 12 ตัวแบบ และเลือกตัวแบบที่ดีที่สุดด้วยการวัดค่า RMSE และ MAPE โดยวิธีตรวจสอบสลับ 3 ส่วน ซึ่งผลการวัดค่า RMSE และ MAPE ด้วยวิธีตรวจสอบสลับ 3 ส่วนของแต่ละตัวแบบ เป็นดังตารางที่ 2 จากตารางที่ 2 ตัวแบบที่ 11 เป็นตัวแบบที่ให้ค่าเฉลี่ย RMSE และ MAPE ต่ำที่สุด จึงเป็นตัวแบบ SVR ที่ดีที่สุด

สำหรับนำมาใช้ในการพยากรณ์อนุกรมเวลาชุดนี้ ดังนั้น ทำการสร้างตัวแบบ SVR ด้วยชุดข้อมูลฝึกสอน ที่มีอินพุต 11 ตัว คือ ค่าสังเกตค่าหลังตั้งแต่ 1 ถึง 11 คาบเวลา จากนั้นนำไปเปรียบเทียบกับค่าสังเกตจริงในช่วงเวลาเดียวกันของชุดข้อมูลตรวจสอบ เพื่อวัดค่า RMSE และ MAPE ได้ผลลัพธ์เป็นดังตารางที่ 3

ตารางที่ 3. ค่าพยากรณ์และค่าคลาดเคลื่อนของตัวแบบ SVR ที่เหมาะสมที่สุด

Value	Jan	Feb	Mar	Apr	May	RMSE	MAPE
	2015	2015	2015	2015	2015		
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
SVR model 11	749.60	795.30	960.09	1144.16	1197.00	42.46	3.71

### 3.2 ผลการวิจัยจากตัวแบบ MLR

สำหรับการสร้างตัวแบบพยากรณ์ด้วยเทคนิค MLR จะใช้ชุดข้อมูลฝึกสอนที่มีลักษณะเช่นเดียวกับชุดข้อมูลฝึกสอนที่ใช้สร้างตัวแบบ SVR โดยทำการสร้างตัวแบบ MLR จำนวน 12 ตัวแบบ จากชุดข้อมูลฝึกสอน ใน

ลักษณะเช่นเดียวกับตัวแบบ SVR จากนั้น นำตัวแบบ MLR ทั้ง 12 ตัวแบบ ไปพยากรณ์ค่าสังเกต 5 คาบเวลา แล้วนำไปเปรียบเทียบกับค่าสังเกตจริงในช่วงเวลาเดียวกัน ของชุดข้อมูลทดสอบ เพื่อวัดค่า RMSE และ MAPE ได้ ผลลัพธ์เป็นดังตารางที่ 4

ตารางที่ 4. ค่าพยากรณ์และค่าคลาดเคลื่อนของตัวแบบ MLR

Value	Jan 2015	Feb 2015	Mar 2015	Apr 2015	May 2015	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
MLR model 1	865.83	795.18	874.89	996.39	1057.80	122.04	11.91
MLR model 2	813.97	768.57	929.83	1060.45	1076.56	90.47	8.59
MLR model 3	821.41	759.93	917.04	1068.54	1090.44	91.29	8.87
MLR model 4	821.31	760.18	916.76	1068.13	1090.64	91.26	8.87
MLR model 5	824.85	762.87	911.22	1074.87	1100.06	89.62	8.73
MLR model 6	814.68	755.67	908.80	1085.66	1089.13	92.47	8.66
MLR model 7	754.15	726.63	881.39	1070.78	1119.97	94.98	8.02
MLR model 8	772.42	744.48	901.54	1091.95	1127.63	82.68	7.18
MLR model 9	785.11	792.74	935.99	1129.05	1174.92	53.84	5.29
MLR model 10	782.15	797.80	943.23	1131.58	1184.00	49.03	4.84
MLR model 11	760.70	789.74	950.92	1143.49	1190.37	47.52	4.42
MLR model 12	762.18	792.08	952.35	1143.28	1188.78	46.74	4.40

จากตารางที่ 4 ตัวแบบ MLR ที่ดีที่สุด คือ ตัวแบบที่ 12 เพราะให้ค่า RMSE และ MAPE ต่ำที่สุด โดยมีค่า RMSE = 46.74 และค่า MAPE = 4.40 ดังนั้น ตัวแบบ MLR ที่ดีที่สุด คือ ตัวแบบที่มีตัวแปรอิสระ 12 ตัวแปร ซึ่งเป็นค่าสังเกตหลังตั้งแต่ 1 ถึง 12 คาบเวลา ( $y_{t-1}, y_{t-2}, \dots, y_{t-12}$ ) ส่วนตัวแปรตาม คือ ค่าสังเกต ณ เวลา  $t$  ( $y_t$ ) นอกจากนี้ พิจารณาผลการทดลองตามตารางที่ 4. พบว่าการเพิ่มตัวแปรอิสระ จะทำให้ค่าคลาดเคลื่อนลดลง ซึ่งสอดคล้องกับหลักการวิเคราะห์ความถดถอยเชิงเส้นแบบพหุ แต่การเพิ่มตัวแปรอิสระเข้าไป เพื่อให้ค่าคลาดเคลื่อนลดลง จะต้องใช้ข้อมูลจำนวนมาก และเป็นข้อดีของหลักการวิเคราะห์ความถดถอยเชิงเส้นแบบพหุ

### 3.3 ผลการวิจัยจากตัวแบบ ARIMA

การสร้างตัวแบบ ARIMA จะต้องกำหนดรูปแบบของตัวแบบ ARIMA ด้วยการใช้พารามิเตอร์ ( $p, d, q$ ) และอันดับ ( $P, D, Q$ )<sub>S</sub> ซึ่งภาษา R มีฟังก์ชัน `auto.arima()` สำหรับกำหนดอันดับ ( $p, d, q$ ) และอันดับ ( $P, D, Q$ )<sub>S</sub> ที่เหมาะสม

โดยผลจากการใช้ฟังก์ชัน `auto.arima()` เพื่อกำหนดรูปแบบของตัวแบบ ARIMA เป็นดังรูปที่ 10 ซึ่งทำให้ทราบว่า ตัวแบบ ARIMA มีรูปแบบเป็น  $ARIMA(1,0,0) \times (1,0,0)_{12}$  โดยมีคาบเวลา 1 ฤดูกาลเท่ากับ 12 ( $S=12$ ) และสรุปได้ว่าตัวแบบ ARIMA ประกอบด้วย Autoregressive (AR) อันดับ  $p=1$  และ Seasonal autoregressive (SAR) อันดับ  $P=1$  โดยประมาณค่าสัมประสิทธิ์ AR, ( $\theta_1$ ) และ SAR, ( $\Theta_1$ ) ได้เท่ากับ 0.7301 และ 0.6403 ตามลำดับ โดยที่อันดับ  $d=D=0$  และอันดับ  $q=Q=0$  ดังนั้น สามารถสร้างสมการพยากรณ์ ได้ดังนี้

$$\begin{aligned}(1 - \theta_1 B)(1 - \Theta_1 B^{12})Y_t &= \theta_0 \quad \text{เมื่อ } \theta_0 = \text{ค่าคงที่} \\ (1 - \theta_1 B - \Theta_1 B^{12} + \theta_1 \Theta_1 B^{12})Y_t &= \theta_0 \\ Y_t - \theta_1 Y_{t-1} - \Theta_1 Y_{t-12} + \theta_1 \Theta_1 Y_{t-13} &= \theta_0 \\ Y_t &= \theta_0 + \theta_1 Y_{t-1} + \Theta_1 Y_{t-12} - \theta_1 \Theta_1 Y_{t-13} \\ Y_t &= \theta_0 + 0.73Y_{t-1} + 0.64Y_{t-12} - 0.47Y_{t-13}\end{aligned}$$

จากค่าสังเกตของอนุกรมเวลา ทำให้ทราบว่า ค่าคงที่  $\theta_0 = 92.42$  ดังนั้น ได้สมการพยากรณ์จากตัวแบบ ARIMA เป็นดังนี้



$Y_t = 92.42 + 0.73Y_{t-1} + 0.64Y_{t-12} - 0.47Y_{t-13}$  จากนั้น นำไปเปรียบกับค่าสังเกตจริงในช่วงเวลาเดียวกัน  
 เมื่อนำตัวแบบ ARIMA พยากรณ์ค่าสังเกต 5 ของชุดข้อมูลตรวจสอบ เพื่อวัดค่า RMSE และ MAPE จะ  
 ตามเวลา ตั้งแต่เดือนมกราคมถึงเดือนพฤษภาคม 2558 ได้ผลลัพธ์เป็นดังตารางที่ 5

```

Console D:/R/code/ ↵
>
> auto.arima(arimats)
Series: arimats
ARIMA(1,0,0)(1,0,0)[12] with non-zero mean

Coefficients:
      ar1      sar1  intercept
      0.7314  0.6425  929.9975
s.e.  0.0833  0.0980   57.5709

sigma^2 estimated as 3300: log likelihood=-331.78
AIC=671.56  AICC=672.29  BIC=679.94
>
    
```

รูปที่ 10 ผลลัพธ์ของการใช้ฟังก์ชัน auto.arima()

ตารางที่ 5. ค่าพยากรณ์และค่าคลาดเคลื่อนของตัวแบบ ARIMA

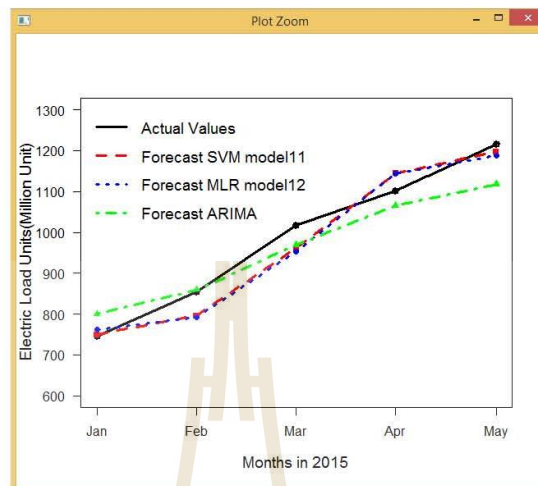
Value	Jan 2015	Feb 2015	Mar 2015	Apr 2015	May 2015	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
ARIMA	799.24	859.33	969.40	1065.12	1117.64	57.18	4.80

**3.4 เปรียบเทียบผลการวิจัยของตัวแบบ SVR ตัวแบบ MLR และตัวแบบ ARIMA**

จากการทดลองสร้างตัวแบบพยากรณ์อนุกรมเวลาของ หน่วยจำหน่ายไฟฟ้าในเขตพื้นที่กรุงเทพมหานคร ของ ประเทศไทย พบว่า ตัวแบบพยากรณ์ที่มีความแม่นยำมากที่สุด คือ ตัวแบบที่สร้างจากเทคนิค SVR ซึ่งวัดความแม่นยำ ส่วนค่าพยากรณ์และค่าคลาดเคลื่อนของ 3 ตัวแบบ สรุปได้ดังตารางที่ 6 จากนั้น นำมาสร้างกราฟเส้นเปรียบเทียบกับค่าสังเกตจริงในช่วงเวลาเดียวกัน ได้ผลดังรูปที่ 11

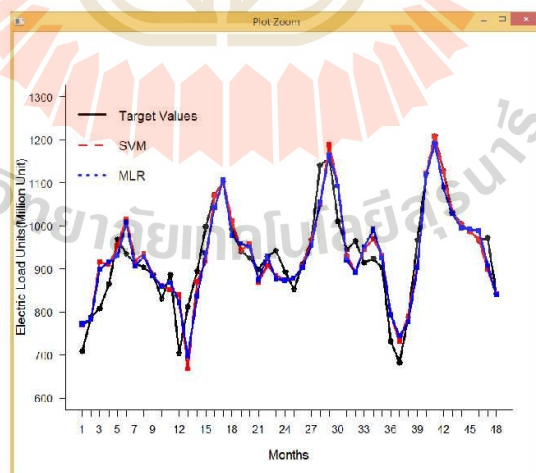
ตารางที่ 6. ค่าพยากรณ์และค่าคลาดเคลื่อนของ 3 ตัวแบบ

Value	Jan 2015	Feb 2015	Mar 2015	Apr 2015	May 2015	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
SVR model 11	749.60	795.30	960.09	1144.16	1197.00	42.46	3.71
MLR model 12	752.18	792.08	952.35	1143.28	1188.78	46.74	4.40
ARIMA	799.24	859.33	969.40	1065.12	1117.64	57.18	4.80



รูปที่ 11. กราฟเปรียบเทียบค่าพยากรณ์ของ 3 ตัวแบบ

จากรูปที่ 11 พบว่า กราฟค่าพยากรณ์ของตัวแบบ SVR และตัวแบบ MLR มีลักษณะใกล้เคียงกันมาก นอกจากนี้ เมื่อนำค่าสังเกตจริงของชุดข้อมูลฝึกสอน และค่าเอาต์พุต ที่ได้จากขั้นตอนการฝึกสอนของตัวแบบ SVR และ MLR มาสร้างกราฟตามลำดับเวลา ดังรูปที่ 12 พบว่า กราฟเอาต์พุตของตัวแบบ SVR และ MLR มีลักษณะใกล้เคียงกันด้วย ดังนั้น จากผลการทดลอง แสดงให้เห็นว่า เทคนิคการพยากรณ์ด้วยตัวแบบ SVR ที่มีเคอร์เนลฟังก์ชันเป็นลิเนียร์เคอร์เนล มีความคล้ายคลึงกับเทคนิค MLR แต่ตัวแบบ SVR มีค่าคลาดเคลื่อนต่ำกว่า และใช้จำนวนตัวแปรที่ใช้เป็นอินพุตน้อยกว่า ซึ่งแสดงให้เห็นว่า เทคนิค SVR เป็นเทคนิคที่เป็นทางเลือกที่ดี สำหรับนำมาพยากรณ์อนุกรมเวลา โดยให้ความแม่นยำของการพยากรณ์ในระดับสูง



รูปที่ 12. กราฟเปรียบเทียบเอาต์พุตของตัวแบบ SVR และ MLR ของชุดข้อมูลฝึกสอน

#### 4. สรุปและเสนอแนะ

จากการศึกษาการวิเคราะห์หอนุกรมเวลา เพื่อพยากรณ์ข้อมูลหอนุกรมเวลา ซึ่งเป็นข้อมูลหน่วยจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง (กฟน.) แห่งประเทศไทย โดยเก็บรวบรวมข้อมูลจากรายงานสถานการณ์การจำหน่ายไฟฟ้าของการไฟฟ้านครหลวง ตั้งแต่เดือนมกราคม 2553 ถึงเดือนพฤษภาคม 2558 และข้อมูลที่สนใจ มีขอบเขตเฉพาะจำนวนหน่วยจำหน่ายไฟฟ้าให้กับผู้บริโภคประเภทบ้านอยู่อาศัย โดยสร้างตัวแบบการพยากรณ์ด้วยเทคนิคที่แตกต่างกัน 3 วิธี ได้แก่ ตัวแบบ SVR ตัวแบบ MLR และตัวแบบ ARIMA ซึ่งตัวแบบ ARIMA เป็นตัวแบบพยากรณ์หอนุกรมเวลาเชิงเดี่ยว ส่วนตัวแบบ SVR และตัวแบบ MLR เป็นตัวแบบที่มีชุดข้อมูลในรูปแบบของอินพุตเวกเตอร์ และทาร์เก็ต ( $D = \{(x_i, y_i)\}_{i=1}^n$ ) โดยอินพุตเวกเตอร์ คือ ค่าสังเกตค่าหลังตั้งแต่ 1-12 คาบเวลา ( $y_{t-1}, y_{t-2}, \dots, y_{t-12}$ ) ส่วนทาร์เก็ต คือ ค่าสังเกต ( $y_t$ ) ณ เวลา  $t$  ในส่วนของการสร้างตัวแบบ SVR ได้สร้างตัวแบบขึ้นมา 12 ตัวแบบ จากนั้นคัดเลือกตัวแบบที่ดีที่สุดด้วยวิธีการตรวจสอบสลับ 3 ส่วน ผลการทดลอง พบว่า ตัวแบบ SVR ที่ดีที่สุด คือตัวแบบที่ 11 ซึ่งมีอินพุต 11 ตัว คือ ค่าสังเกตค่าหลังตั้งแต่ 1 ถึง 11 คาบเวลา และในส่วนของ การสร้างตัวแบบ MLR ได้สร้างตัวแบบขึ้นมา 12 ตัวแบบ จากนั้นเลือกตัวแบบที่ดีที่สุดด้วยการวัดค่าคลาดเคลื่อน RMSE และ MAPE กับชุดข้อมูลตรวจสอบ ผลการทดลอง พบว่า ถ้ามีจำนวนตัวแปรอิสระหรือตัวแปรอินพุตมากขึ้น จะทำให้ค่าคลาดเคลื่อนลดลง ดังนั้น ตัวแบบ MLR ที่ดีที่สุด คือตัวแบบที่ 12 มีตัวแปรอิสระ 12 ตัวแปร คือ ค่าสังเกตค่าหลังตั้งแต่ 1 ถึง 12 คาบเวลา ในส่วนสุดท้าย เมื่อนำค่าพยากรณ์ 5 ช่วงเวลาของเดือนมกราคมถึงเดือนพฤษภาคม 2558 มาวัดค่าคลาดเคลื่อน RMSE และ MAPE ผลการทดลอง พบว่า ตัวแบบ SVR เป็นตัวแบบที่ให้ค่าคลาดเคลื่อนต่ำที่สุด คือค่า RMSE เท่ากับ 42.46 และค่า MAPE = 3.71 ดังนั้น ตัวแบบ SVR เป็นตัวแบบที่มีความแม่นยำในการพยากรณ์หอนุกรมเวลาชุดนี้มากที่สุด นอกจากนี้ ผลการทดลอง พบว่า ตัวแบบ SVR ที่มีเคอร์เนลฟังก์ชันเป็นลิเนียร์เคอร์เนล และตัวแบบ MLR มีความคล้ายคลึงกันมาก แต่ตัวแบบ SVR ให้ความแม่นยำในการพยากรณ์สูงกว่า ด้วยจำนวนตัวแปรอินพุตน้อยกว่า แสดงให้เห็นว่า ตัวแบบ SVR เป็นตัวแบบที่สามารถนำมาใช้ในการพยากรณ์หอนุกรมเวลาได้ดี แต่ถ้าเปรียบเทียบในด้านความซับซ้อนของตัวแบบ SVR กับตัวแบบ ARIMA พบว่า ตัวแบบ SVR เป็นตัวแบบที่มีความ

ซับซ้อนมากกว่าตัวแบบ ARIMA เพราะใช้อินพุต 11 ตัว ในขณะที่ ตัวแบบ ARIMA ใช้อินพุต 3 ตัว สำหรับการพยากรณ์ ดังนั้น ข้อเสนอแนะจากงานวิจัยนี้ คือ ค้นหาวิธีการลดจำนวนอินพุตของ SVR เพื่อลดความซับซ้อนของตัวแบบ SVR และให้ความแม่นยำของการพยากรณ์ในระดับสูงเช่นเดิม

#### 5. บรรณานุกรม

- [1] Bunn, D.W. and Farmer, E.D. 1985. **Comparative Models for Electrical Load Forecasting**. New York: John Wiley and Sons.
- [2] Hong, W. C. 2009. "Electric load forecasting by support vector model". **Applied Mathematical Modelling**. 33: 2444-2454.
- [3] Ismail, Z., Yahya, A. and Mahpol, K. A. 2009. "Forecasting Peak Load Electricity Demand Using Statistics and Rule Based Approach". **American Journal of Applied Sciences**. 6(8): 1618-1625.
- [4] การไฟฟ้านครหลวงแห่งประเทศไทย, 2558. รายงานสถานการณ์การจำหน่ายไฟฟ้า. <http://www.mea.or.th/download/index.php#>. 20 สิงหาคม.
- [5] Wang, Y., Wang, J., Zhao, G. and Dong Y. 2012. "Application of residual modification approach in seasonal ARIMA for electricity demand forecasting: A case study of China". **Energy Policy**. 48: 284-294.
- [6] Lee, Y. S. and Tong, L. I. 2011. "Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming". **Knowledge-Based Systems**. 24: 66-72.
- [7] Kaytez, F., Taplamacioglu, M. C., Cam, E. and Hardalac, F. 2015. "Forecasting electricity consumption: A comparison of regression analysis, neural networks and least squares support vector machines". **Electrical Power and Energy Systems**. 67: 431-438.

- [8] Ogcü, G., Demirel, O.F. and Zaim, S. 2012. "Forecasting electricity consumption with neural networks and support vector regression" **Procedia-Social and Behavioral Sciences.** 58: 1576-1585.
- [9] Wang, J., Li, L., Niu, D. and Tan, Z. 2012. "An annual load forecasting model based on support vector regression with differential evolution algorithm". **Applied Energy.** 94: 65-70.
- [10] Fan, G. F., Peng, L. L., Hong, W. C. and Fan, S. 2016. "Electric load forecasting by the SVR model with differential empirical mode decomposition and autoregression" **Neurocomputing.** 173: 958-970.
- [11] Vapnik, V. 1995. **The Nature of Statistical Learning Theory.** Springer-Verlag.
- [12] Bagheripour, P., Gholami, A., Asoodeh, M. and Asadi, M. V. 2015. "Support vector regression based determination of shear wave velocity". **Journal of Petroleum Science and Engineering.** 125: 95-99.
- [13] Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., Chang, C.C., and Lin, C.C. 2015. **Misc Functions of the Department of Statistics, Probability Theory Group.** <http://cran.r-project.org/web/packages/le1071/index.html>. 5 December.
- [14] Sajjan, K.S., Kumar, V. and Tyagi, B. 2015. "Genetic algorithm based support vector machine for on-line voltage stability monitoring" **Electrical Power and Energy Systems.** 73: 200-208.
- [15] Box, G.E.P. and Jenkins, G. 1976. **Times series analysis forecasting and control.** San Francisco, CA: Holden-Day.
- [16] Bergmeir, C. and Benítez, J. M. 2012. "On the use of cross-validation for time series predictor evaluation". **Information Sciences.** 191: 192-213.



บันทึกข้อความ  
มหาวิทยาลัยเทคโนโลยีสุรนารี

หน่วยงาน Suranaree Journal of Science and Technology, Tel. 4756 Fax : 4750

ที่ No. 5621/ 18๕๑

วันที่ ๖ October 2016

เรื่อง Letter of acceptance

Dear Mr. Ronnachai Chuentawat,

I am pleased to inform you that your article entitled "THE FORECAST OF ELECTRICAL POWER DISTRIBUTION UNIT USING SUPPORT VECTOR REGRESSION OPTIMIZED WITH GENETIC ALGORITHM" authored Ronnachai Chuentawat, Kittisak Kerdprasop and Nittaya Kerdprasop has been accepted to be published in Suranaree Journal of Science and Technology. The galley proof will be later sent to you for the final corrections.

Thank you for publishing with Suranaree Journal of Science and Technology.

Sincerely,

Peerapong Uthansakul, Ph.D.

Associate Professor

Editor in Chief

มหาวิทยาลัยเทคโนโลยีสุรนารี





**Author's Accepted Manuscript**

**The Forecast of Electrical Power Distribution Unit Using Support Vector Regression Optimized with Genetic Algorithm**

**Ronnachai Chuentawat, Kittisak Kerdprasop, and Nittaya Kerdprasop**

**Reference:** ENG16010954  
**www:** <http://ird.sut.ac.th/e-journal/Journal/pdf/16010954.pdf>

**To appear in:** Suranaree Journal of Science and Technology  
**ISSN:** 0858-849X

**Received date:** June 17, 2016  
**Revised date:** October 05, 2016  
**Accepted date:** October 06, 2016

**Cite this article as:**

Chuentawat, R., Kerdprasop, K., and Kerdprasop, N. (2016). The forecast of electrical power distribution unit using support vector regression optimized with genetic algorithm. Suranaree J. Sci. Technol. Available from: <http://ird.sut.ac.th/e-journal/Journal/pdf/16010954.pdf>.

This is an early version of an unedited manuscript that has been accepted for publication. The manuscript will undergo copyediting and review of the resulting galley proof before it is published in its final form.

It should be noted that, during the production process, errors may be found which may affect the content. All disclaimers relating to the opinions and statements contained in the journal remain in force.



## THE FORECAST OF ELECTRICAL POWER DISTRIBUTION UNIT USING SUPPORT VECTOR REGRESSION OPTIMIZED WITH GENETIC ALGORITHM

**Running head: The forecast of EPDU by Support Vector Regression**

**Ronnachai Chuentawat<sup>1\*</sup>, Kittisak Kerdprasop<sup>1</sup>,  
and Nittaya Kerdprasop<sup>1</sup>**

<sup>1</sup> *School of Computer Engineering, Institute of Engineering, Suranaree University of Technology, Nakhon Ratchasima, 30000, Thailand. Tel. 0-4422-4422; Fax. 0-4422-4602; e-mail: c\_ronnachai@hotmail.com*

*\* Corresponding author*

### **Abstract**

This research applied time series forecasting with a proposed algorithm: the genetic algorithm optimizing support vector regression (GASVR). The forecasting accuracy performance of the GASVR has been compared with the techniques of an artificial neural network and an autoregressive integrated moving average to forecast the power consumption of Bangkok's metropolitan area. Time series data in terms of the electrical power distribution unit for household electricity usage were obtained from the Metropolitan Electricity Authority of Thailand. The forecasting performance of each model is measured by the root mean square error (RMSE) and the mean absolute percentage error (MAPE) metrics. The experimental results of the RMSE and MAPE comparisons between the 3 models reveal that the GASVR model has the lowest RMSE and MAPE. Based on such results, we can conclude that the proposed GASVR algorithm, which is the support vector regression with parameter optimization by the genetic algorithm, is the most powerful model to forecast time series data in the specific domain of household power consumption.

**Keywords:** Support vector regression, genetic algorithm, artificial neural network, ARIMA model, time series

### **Introduction**

This research studies univariate time series data having a single observed variable that changes its value by time order. This data set is the electrical power distribution unit (EPDU) of the Metropolitan Electricity Authority of Thailand (MEA) covering household consumers in Bangkok and the provinces of Samutprakan and Nonthaburi. The time series data were collected from the MEA's monthly reports during the period from January 2010 to May 2015. The data are presented in an electrical power distribution unit metric in which 1 unit refers to 1,000 kilowatts per hour. These data

in a monthly period scale are expected to contain a time series pattern regarding the demand for electricity. An accurate capturing of such a pattern is obviously beneficial to the MEA in that an accurate forecast provides precise information which is advantageous in resource planning, management of funding, and reducing the operational costs. Bunn and Farmer (1985) found and reported that 1% of electrical load forecasting errors raised the operational costs by 10 million pounds. Therefore, forecasting precision is a challenging problem, especially the demand for electricity supply.

The data used in this study are of limited size containing only 65 observed values. However, there has been much research in the past confirming that electricity consumption data have underlying patterns that can be captured with a low error rate regardless of their size. This research work to generate the forecasting model from a small data set can be examined as follows. Wang *et al.* (2012) used 31 data sets of the annual total electricity consumption of Beijing city between 1978 and 2008 to generate a forecasting model using support vector regression (SVR) optimized with differential evolution. Then, comparing the forecasting power with a back propagation neural network and regression technique, they found that their method yielded the best model showing the lowest mean absolute percentage error (MAPE) at 4.8%. Wang *et al.* (2012) used 35 data sets of the monthly electricity consumption of northwest China from March 2007 to Jan 2010 for training a forecasting model using a seasonal autoregressive integrated moving average (ARIMA), particle swarm optimization, and hybrid method. The hybrid method was the best model with the MAPE value at 2.38%. Kaytez *et al.* (2015) used 40 data sets of the annual electricity consumption of Turkey during the period from 1970 to 2009 to generate a forecasting model using regression analysis, neural networks, and least squares support vector machines. The best model was the least squares support vector machines which achieved the lowest MAPE value at 1.004%. Therefore, we are confident that the data set containing 65 observed values used in this research is sufficient in its size for generating a reliable forecasting model, if we use the appropriate technique to generate the model.

The appropriate strategy for forecasting time series data is to apply several techniques to generate the forecasting models. After that the best model can be chosen by the measurement of the forecasting errors. Previous research studies that explored many possible techniques to find the best model for forecasting time series data in the specific domain of electricity consumption can be reviewed as follows.

Wang *et al.* (2012) tried to increase the forecasting accuracy of the model to predict the electrical load in China. They created a seasonal ARIMA model for the forecasting task. They also modified the residuals to increase the accuracy using the seasonal ARIMA, particle swarm optimization, and hybrid method. They found that the residual modification could increase the forecasting accuracy and the hybrid method was the best method for electricity forecasting.

Lee and Tong (2011) studied a time series that was the yearly data of electricity consumption in China. They presented a hybrid model of the ARIMA and genetic programming methods for forecasting. The model's accuracy was measured by 3 metrics: root mean square error (RMSE), mean absolute error, and mean absolute percentage error (MAPE). They reported that the hybrid model could

generate a forecast more accurately than a single ARIMA or single genetic programming model.

Wang and Meng (2012) studied the forecasting of the energy consumption in China using the ARIMA, artificial neural network (ANN), and hybrid ARIMA-ANN models. They compared the accuracy using the RMSE, mean absolute error, and MAPE. The result of their study showed that a hybrid model provided better accuracy in forecasting than either the ARIMA or ANN models.

Ogcu *et al.* (2012) did an experiment on forecasting electricity consumption in Turkey. The data set was a time series that had a monthly time period. They used the ANN and SVR techniques to model the data set and compared the accuracy by means of the MAPE. The result was that the model using SVR had more accuracy than the model generated from the ANN.

Jirong *et al.* (2011) did an experiment to forecast housing prices in China. The data set was a time series that had a yearly time period between 1993 and 2002. They applied a genetic algorithm (GA) to optimize the SVR parameters and called their method G-SVR. After building the model, they compared the accuracy between their own G-SVR and a grey model. The result was that the G-SVR showed more accuracy than the grey model.

From the literature review, it can be concluded that existing methods to model a time series can be divided into 2 groups. The first group is traditional time series modeling such as regression analysis, exponential smoothing, or the Box and Jenkins method (Box and Jenkins, (1990). The second group is the machine learning method such as an ANN or SVR. In traditional time series modeling, the most popular model is the ARIMA model from the Box and Jenkins method. The popularity is due to its high precision (Lee and Tong, 2011; Wang *et al.*, 2012; Wang and Meng, 2012). In the machine learning method, the SVR model has recently been used extensively because it shows high precision in predicting various data sets (Wang and Meng, 2012; Ogcu *et al.*, 2012; Jirong *et al.*, 2011; Fan *et al.*, 2016). SVR was also found to be the suitable technique to model small sized time series data (Fan *et al.*, 2016).

On applying the SVR technique to model a time series' data, the model's accuracy depends on the appropriate setting of the parameters for the SVR. Therefore, the objective of this research is modeling the EPDU data to forecast future values as accurately as possible. We propose a novel technique using a genetic algorithm to optimize the parameters of the SVR and the proposed technique is called the genetic algorithm optimizing support vector regression (GASVR). After presenting the GASVR technique, we experimentally compare its forecasting accuracy against 2 well-known techniques: the ANN and ARIMA. The comparison is based on the measurement of the RMSE and MAPE.

## Materials and Methods

This research adopts the EPDU data of the MEA with a particular sector of household consumers. The EPDU data set is a monthly time series collected from the monthly reports of the MEA from January 2010 to May 2015. In this research, we apply the GASVR to model the electricity consumption pattern of the population in the Bangkok metropolitan and surrounding areas. The forecasting efficiency of the GASVR model



is compared with the ANN and ARIMA models. The GASVR technique is explained in the following sub-section with its conceptual framework illustrated in Figure 1. The concepts of the ANN and ARIMA are also presented at the end of this section.

### GASVR Model

Support vector regression (SVR) was introduced as a machine learning technique by Vapnik (1995). SVR is an extension of support vector machine (SVM), which is a classification method, to analyze the regression between the input vector and the output. SVR is thus used to forecast numeric values instead of the categorical classification, as traditionally has been done by the SVM. Therefore, SVR focuses on finding a linear relationship mapping the input vector  $X$  in  $n$ - dimensions ( $X \in \mathbb{R}^n$ ) to the output  $y$  ( $y \in \mathbb{R}$ ). Due to the fact that SVR has been modified from the SVM, the regression equation of SVR is therefore similar to a hyperplane equation of the SVM. The regression equation of SVR can be shown in Equation 1 (Bagheripour *et al.*, 2016).

$$f(x) = w^T x + b \quad \text{----- (1)}$$

where,  $w$  is a slope and  $b$  is an offset of the regression line. We can define  $w$  and  $b$  by minimizing Equation 2 (Bagheripour *et al.*, 2016).

$$R = \frac{1}{2} \|w\|^2 + \frac{C}{l} \sum_{i=1}^l |y_i - f(x_i)|_\varepsilon \quad \text{----- (2)}$$

When we use SVR to predict the output from the input vector, we will define the epsilon tube by the loss function. In our research, we use an  $\varepsilon$ -insensitive loss function that can be shown in Equation 3 (Vapnik, 1995).

$$|y_i - f(x_i)|_\varepsilon = \begin{cases} 0 & \text{if } |y_i - f(x_i)|_\varepsilon \leq \varepsilon \\ |y_i - f(x_i)|_\varepsilon - \varepsilon & \text{Otherwise} \end{cases} \quad \text{----- (3)}$$

To solve Equation 2 that has Equation 3 as a constraint, we can reformulate the problem to the dual problem by using Lagrange multipliers, as shown in Equations 4 and 5 (Bagheripour *et al.*, 2016).

$$\text{Maximize } L_p(\alpha_i, \alpha_i^*) = -\frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) x_i^T x_j - \varepsilon \sum_{i=1}^l (\alpha_i + \alpha_i^*) + \sum_{i=1}^l (\alpha_i - \alpha_i^*) y_i \quad \text{----- (4)}$$

$$\text{Subject to } \begin{cases} \sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0 \\ 0 \leq \alpha_i \leq C, & i = 1, \dots, l \\ 0 \leq \alpha_i^* \leq C, & i = 1, \dots, l \end{cases} \quad \text{----- (5)}$$

where  $\alpha_i, \alpha_i^* > 0$  are the positive Lagrange multipliers,  $C$  is the cost of error and it is a positive parameter,  $\varepsilon$  is the width of the epsilon tube, and  $l$  is the number of the support vector. The support vector is an input vector that has  $\alpha_i, \alpha_i^* > 0$  and when we calculate  $\alpha_i, \alpha_i^*$  from the training set, we can formulate the equation of the SVR to predict the output from the input vector as shown in Equation 6 (Bagheripour *et al.*, 2016).

$$f(x) = w_0^T x + b = \sum_{i=1}^l (\alpha_i - \alpha_i^*) x_i^T x + b \quad \text{----- (6)}$$

The weight vector of the regression hyperplane ( $w_0$ ) is given by Equation 7 (Bagheripour *et al.*, 2016).

$$w_0 = \sum_{i=1}^l (\alpha_i - \alpha_i^*) x_i \quad \text{----- (7)}$$

Equation 6 is the format of linear regression. For the non-linear case, we can map the input vector in high dimensional feature space by using a kernel function.

The mapping can be the multiplication of the vectors  $x_i$  and  $x_j$ , and the popular kernel functions can be listed as follows (Sajan *et al.*, 2015).

- (1) Linear kernel:  $k(x_i, x) = x_i^T x$
- (2) Polynomial kernel:  $k(x_i, x) = (1 + x_i \cdot x_j)^d$
- (3) Gaussian (RBF) kernel:  $k(x_i, x) = \exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2})$

Therefore, Equation 6 can be reformulated in the format of non-linear regression by using the kernel function as shown in Equation 8 (Bagheripour *et al.*, 2016).

$$f(x) = \sum_{i=1}^l (\alpha_i - \alpha_i^*) k(x_i, x) + b \quad \text{----- (8)}$$

This research created the model of SVR with R language and used the “svm()” function available in the “e1071” package. This package was introduced by Meyer *et al.* (2015) and is publicly available in the CRAN repository. To generate the SVR model, we need to define the kernel function. From a series of experiments, we found that the linear kernel gave the best result, so we used the linear kernel in our research. For the parameters C and  $\epsilon$ , their optimal values were to be searched for with the genetic algorithm (GA), and the final model would be called the GASVR model. From the conceptual framework, as shown in Figure 1, we can describe the steps to generate the GASVR model as follows.

1. Collect the data from the monthly reports of the MEA. The data were obtained from January 2010 to May 2015. The data are the household electrical usage of residents in the MEA area. We then order the data to build up a time series as shown in Figure 2.

2. Divide the data into 2 data sets. The first data set is the training set to create the forecasting model. Our training set is the data from January 2010 to December 2014. The second data set is the validating set to measure the accuracy of the forecasting model. The validating set is the data from January to May 2015.

3. For the model creation of the GASVR and ANN algorithms, the data set that is used to build the model must be in the form of a pair between the input vectors and corresponding targets ( $D = \{(x_i, y_i)\}_{i=1}^n$ ). Therefore, the data in Figure 2 would be transformed to the data as shown in Figure 3 before the data can be used to build the model. Each column of the data set in Figure 3 is the lag time ranging from the 1 to 12 time periods ( $y_{t-1}, y_{t-2}, \dots, y_{t-12}$ ). The last column is a target, which is the observed value at time t. The first value of the target is the observed value that has the time index equal to 13 ( $t=13$ ). After transforming the data, we will get 53 rows from 65 observed values that have the time period from January 2010 to May 2015.

4. Use the simple linear regression to define the input vectors of the GASVR by using the data set as shown in the Figure 3. We created 12 models of simple linear regression and each model used the observed values at the lag times 1 to 12 ( $y_{t-1}, y_{t-2}, \dots, y_{t-12}$ ) as independent variables. All 12 models have the same observed value at time t ( $y_t$ ) as a dependent variable.

5. Select the input vectors of the GASVR by considering the value of  $R^2$  (coefficient of determination). It must be greater than or equal to 0.5 ( $R^2 \geq 0.5$ ).

6. Define the optimal C and  $\epsilon$  parameters by applying the GA. In the R language, it has the “rgba()” function in the “genalg” package that was introduced by Willighagen and Ballings (2015) and which is available in the CRAN repository

of R language for finding the optimal solution by the GA. The process of the GA can be described as follows.

6.1 *Define the chromosome representation.* This step defines the pattern of the genes in a chromosome. This research uses a GA to find the optimal C and  $\epsilon$ , so that each chromosome consists of 2 genes that represent C and  $\epsilon$ . The type of chromosome encoding is a float chromosome and the setting of the GA can be shown as in Figure 4. We randomly set the initial population with 200 chromosomes.

6.2 *Evaluate the fitness value.* The evaluation of the fitness in each chromosome is used to choose the offspring that will be the next generation. In our research, we define an evaluation function to be the RMSE, defined as shown in Equation 9 (Bergmeir and Benítez, 2012).

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \quad \text{----- (9)}$$

6.3 *Perform the selection operation.* This step is the selection of the offspring that will be the next generation. This research uses the roulette wheel method for the selection operation.

6.4 *Perform the crossover operation.* This step is to build the offspring. This research uses 2 parent chromosomes for the single-point crossover operation and sets the rate of crossover to be 0.8.

6.5 *Perform the mutation operation.* This step is to mutate the offspring by using only a single parent chromosome. The mutation is used to avoid the problem of local optimum and the rate of mutation is set to be 0.01.

6.6 *Perform the replacement operation.* This step is the replacement over the existing population by using the new set of the population that has the fitness value better than the old population set. We replace the parents with the offspring at the rate of 0.8 (replace 160 parent chromosomes from 200).

The whole process of the GA to find the optimal C and  $\epsilon$  parameters is shown in Figure 5.

7. After obtaining the optimal C and  $\epsilon$  parameters from the GA, apply the 2 parameters to create the model through the SVR algorithm, and the final product is the GASVR model.

8. Test the accuracy of the GASVR model by predicting the 5 observed values in January to May 2015. The forecasting values are compared against the actual values of the validation set to compute forecasting errors.

9. Measure the forecasting accuracy by using the RMSE in Equation 9 and the MAPE as shown in Equation 10 (Bergmeir and Benítez, 2012). The RMSE and MAPE are also used to compare with the ANN and ARIMA in later steps of the experimentation.

$$\text{MAPE} = \frac{1}{n} \sum_{t=1}^n \left| 100 \frac{y_t - \hat{y}_t}{y_t} \right| \quad \text{----- (10)}$$

#### ANN Model

An artificial neural network (ANN) is the simulated network of neurons in the human brain. We can apply an ANN to forecast time series data. The ANN performs learning from the existing data by analyzing the correlation between the observed values at the current time with previously observed values. After getting an ANN



model from the training set, we can use such a model to forecast the values of new observations and can measure the accuracy by comparing with the actual values in the validation set.

In this research, we use an ANN model from the network architecture called three-layer feed-forward back-propagation neural networks with 1 hidden layer. The output from a model is a forecasting value at the current time ( $y_t$ ). The input to a model is the previously observed values at 1 to  $p$  time intervals ( $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ ) and can be represented as a vector. The correlation between the input and output can be shown as in Equation 11 (Wang and Meng, 2012).

$$y_t = \omega_0 + \sum_{j=1}^q \omega_j g(\omega_{0j} + \sum_{i=1}^p \omega_{ij} y_{t-i}) + e_t \quad \text{----- (11)}$$

where  $\omega_j$  ( $j=1, \dots, q$ ) and  $\omega_{ij}$  ( $i=0, \dots, p; j=1, \dots, q$ ) are the model parameters, which are called weights,  $p$  is the number of neurons in the input layer, and  $q$  is the number of neurons in the hidden layer. Each neuron uses a sigmoid function (presented in Equation 12) as a transfer function (Wang and Meng, 2012).

$$sig(x) = \frac{1}{(1+exp(-x))} \quad \text{----- (12)}$$

From the ANN model shown in Equation 11, we can transform to a non-linear function to represent the relationship between previously observed values ( $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ ) and the forecasting value ( $y_t$ ), as in Equation 13 (Wang and Meng, 2012).

$$y_t = f(y_{t-1}, \dots, y_{t-p}, \omega) + e_t \quad \text{----- (13)}$$

where  $\omega$  is a vector of all the parameters, and  $f()$  is a function used for determining the network structure and the connecting of weights. As a result, the neural network model is equivalent to and can be expressed in terms of a non-linear autoregressive model. The output layer will have 1 neuron because it will be used to predict 1 future value in a single time period. In our research, the input of the ANN will be analyzed by simple linear regression.

This research defines the optimal ANN model by testing with 4 ANN models. Each model has a different number of neurons in a hidden layer varying from 1 to 4 neurons. The structure of the final ANN model used for the experiment in this section can be shown as in Figure 6.

We then implement our ANN structures using Matlab software and the steps of implementation can be described as follows:

1. Use the data in Figure 3 to train the 4 ANN models.
  2. The inputs of all the ANN models come from the simple linear regression analysis.
  3. Normalize the data set to be in the range of 0 to 1 to avoid the data overflow because the process of the ANN is a time-consuming iterative process.
  4. Define the structure of the ANN by using the "newff()" function.
  5. Fix the initial weight and bias instead of randomizing them. The reason for having specific values is that there will be a rapid convergence to a good solution.
  6. Train the 4 ANN models by using the training data in Figure 3.
  7. Predict the 5 EPDU values from January to May 2015 from all the ANN models for accuracy measurement with the RMSE and MAPE metrics.
  8. Choose the optimal ANN model that has the minimum RMSE and MAPE.
- Later on we will use the optimal ANN to compare its forecasting performance with the GASVR and ARIMA.

### ARIMA Model

The ARIMA model is a model derived from the Box and Jenkins method (Box and Jenkins, 1990). The ARIMA model can predict the data through the conformation of autoregressive (AR) and moving average (MA); the compound of these 2 conformations is thus called ARMA. The ARMA is used for the stationary process of a time series if that series is non-stationary. The transformation to a stationary one is a necessary process before generating the model, called the ARIMA model. The general term of the ARIMA (Wang *et al.*, 2012) can be presented with a backward shift operator (B) in the following equation.

$$\theta_p(B)\Theta_p(B^S)(1-B)^d(1-B^S)^D Y_t = w_q(B)W_q(B^S)a_t \quad \text{----- (14)}$$

where  $BY_t = Y_{t-1}$  and  $B^k Y_t = Y_{t-k}$

$$w_q(B) = 1 - w_1 B - w_2 B^2 - \dots - w_q B^q$$

$$\theta_p(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_p B^p$$

$$\Theta_p(B^S) = 1 - \Theta_1 B^S - \Theta_2 B^{2S} - \dots - \Theta_p B^{pS}$$

$$W_q(B^S) = 1 - W_1 B^S - W_2 B^{2S} - \dots - W_q B^{qS}$$

In order to generate an ARIMA model with R language, we used the “arima()” function in the “forecast” package that was introduced by Hyndman (2015) and which is available in the CRAN repository to create the ARIMA model. We have to analyze the time series data for defining the suitable parameters of the ARIMA(p, d, q)x(P, D, Q)s. That means selecting the suitable p, d, q from trend and P, D, Q from the seasonal. In the “forecast” package, it has the “auto.arima()” function to assign a suitable ARIMA(p, d, q)x(P, D, Q)s. From our conceptual framework, we design the time series forecasting with an ARIMA model as shown in Figure 7. Each step can be described as follows.

1. Collect the 65 data instances from the monthly reports of the MEA.
2. Transfer data to be a time series by using the “ts()” function and split it into 2 parts. The first part is the training data: January 2010 to December 2014. The second part is the validation data: January to May 2015.
3. Define the suitable parameters of the ARIMA (the parameters p, d, q and P, D, Q) by using the “auto.arima()” function.
4. Generate the ARIMA model by using the “arima()” function.
5. Predict the 5 observed values from January to May 2015 by using the “predict()” function.
6. Measure the forecasting accuracy with the RMSE and MAPE metrics.

The command set in R according to the steps presented in Figure 7 can be shown as in Figure 8.

### Results and Discussion

Due to this research having 3 different models, we divided the experimental results into 4 parts. The first 3 parts are the results obtained from the 3 models (GASVR, ANN, and ARIMA). The last part of the results section is the comparison between the 3 models.

#### Results of GASVR Model

The input vectors of the GASVR model were defined by the simple linear regression analysis of the 1 to 12 lag time observed values and use the  $R^2 \geq 0.5$  to

determine the input vectors. The result of the input vector determination can be shown in Figure 9.

From Figure 9, the observed values at the 1 and 12 lag times have  $R^2$  greater than 0.5, so we use  $y_{t-1}$  and  $y_{t-12}$  to be the input of the GASVR model. Later on, we use the GA to find the optimal C and  $\varepsilon$  parameters that can be shown in Figure 10.

When we know the optimal C and  $\varepsilon$  parameters, we can thus use them to generate the optimal GASVR model by using the command set in Figure 11. We then predict the 5 observed values and measure the RMSE and MAPE error values that can be shown in Table 1.

### Results of ANN Models

For the experiment with the ANN model, we use the structure of the ANN, as shown in Figure 6, and use  $y_{t-1}$  and  $y_{t-12}$  as an input, which is the same as the GASVR. To avoid over-fitting, we design the ANN structure to contain a few neurons (from 1 to 4) in the hidden layer. The 4 ANN models are named ANN1 to ANN4. In addition, we found that by fixing the values of the initial weight and bias, the ANN shows a better performance than by randomizing them. From our experiment, the initial weight and bias of each ANN model that minimizes the RMSE and MAPE can be shown as in Table 2.

When we use the initial weight and bias for training the ANN models and then measure their RMSE and MAPE values with the validation data set, the results are as shown in Table 3. The ANN3 model has a minimum MAPE value and when we measure the correlation of each model, we found that the ANN3 model has a maximum correlation coefficient equal to 0.88868. Its regression graph can be shown as in Figure 12. Therefore, we can conclude that the ANN3 model is the optimal ANN model for this research and it has a structure as shown in Figure 13.

### Results of ARIMA Model

After exploring with the `auto.arima()` function to find the suitable parameters for the  $ARIMA(p, d, q) \times (P, D, Q)_s$ , we found that the order of the autoregressive (AR) should be  $p = 1$ , the order of the seasonal autoregressive (SAR) should be  $P = 1$ , the order of the difference should be  $d = D = 0$ , and the order of the moving average should be  $q = Q = 0$ . When we estimate the coefficient of AR1 ( $\hat{\theta}_1$ ) and SAR1 ( $\hat{\Theta}_1$ ) by the `auto.arima()` function, we get the values of 0.7301 and 0.6403, respectively. Therefore, the forecasting equations can be derived as follows:

$$\begin{aligned} (1 - \hat{\theta}_1 B)(1 - \hat{\Theta}_1 B^{12})Y_t &= \theta_0 && \text{while } \theta_0 \text{ is constant} \\ (1 - \hat{\theta}_1 B - \hat{\Theta}_1 B^{12} + \hat{\theta}_1 \hat{\Theta}_1 B^{12})Y_t &= \theta_0 \\ Y_t - \hat{\theta}_1 Y_{t-1} - \hat{\Theta}_1 Y_{t-12} + \hat{\theta}_1 \hat{\Theta}_1 Y_{t-13} &= \theta_0 \\ Y_t = \theta_0 + \hat{\theta}_1 Y_{t-1} + \hat{\Theta}_1 Y_{t-12} - \hat{\theta}_1 \hat{\Theta}_1 Y_{t-13} \\ Y_t = \theta_0 + 0.73Y_{t-1} + 0.64Y_{t-12} - 0.47Y_{t-13} \end{aligned}$$

From the existing observed values of the time series,  $\theta_0 = 92.42$  and the the forecasting computation of the ARIMA can be presented as in Equation 15.

$$Y_t = 92.42 + 0.73Y_{t-1} + 0.64Y_{t-12} - 0.47Y_{t-13} \quad \text{----- (15)}$$

We use the ARIMA model to predict the 5 observed values for the measurement of the RMSE and MAPE. The results of the experiment are shown in Table 4.



### The comparison of the 3 models: GASVR, ANN, and ARIMA

From our experiment, we found that the input vector selection for the GASVR and ANN models by means of simple regression analysis resulted in the input vectors of the observed values at the 1 and 12 lag time ( $y_{t-1}$  and  $y_{t-12}$ ). For building the ARIMA model, the conformation of the ARIMA model is  $(p, d, q) \times (P, D, Q) = (1, 0, 0) \times (1, 0, 0)$ . This means that the output value at time ( $y_t$ ) depends on 2 inputs that are the observed values at the 1 and 12 lag time ( $y_{t-1}$  and  $y_{t-12}$ ). Therefore, the input of the 3 different models generated from 3 different methods in our experiment has the same set of input vectors.

By predicting the electricity usage during the period from January to May 2015, we found that the GASVR model yielded the lowest RMSE and MAPE (RMSE = 53.79 and MAPE = 4.40). This accuracy performance is about 8.33% better than the ARIMA model. Therefore, we can conclude that the GASVR model is the most suitable model for power consumption forecasting. The comparative results of the 3 models are summarized in Table 5, and also are graphically shown in Figure 14. From Figure 14, we found that the forecasting trends of the GASVR and ARIMA models are similar and align to the actual values, whereas the forecasting graph of the ANN3 model is worse than the other 2 models. But the graph of the ANN3 model can rapidly adjust the irregular error in March to a normal error.

From the accuracy comparison results of the GASVR, ANN, and ARIMA models, the GASVR has the lowest RMSE and MAPE. This may be due to the fact that the power consumption time series consists of complex linear and non-linear patterns, which are difficult to forecast correctly. The ARIMA model is good at capturing linear patterns, but it cannot easily capture the non-linear patterns (Pai and Lin, 2005; Wang and Meng, 2012). Conversely, the machine learning techniques (ANN and GASVR) can capture non-linear patterns quite well, but they may not easily capture the linear patterns (Wang and Meng, 2012; Zhang et al., 2016). Neither the ARIMA nor machine learning technique alone is adequate in modeling and predicting time series data that consist of linear and non-linear patterns. Therefore, we integrate both techniques by determining the inputs of the machine learning techniques using the lag time observed values that have  $R^2$  greater than 0.5, which can represent linear patterns. We then generate the forecasting model by using the machine learning technique that can efficiently capture the non-linear patterns.

With this underlying assumption, the ANN and GASVR models should capture both linear and non-linear patterns. The correctness of this assumption has been experimentally confirmed through the lower MAPE values in the ANN and GASVR models, as compared with the ARIMA model. When we compare the ANN with the GASVR, the GASVR has a lower MAPE than the ANN because the GASVR is more suitable to the small data size than the ANN. This observation is in accordance with the result observed by other researchers (Zhang *et al.*, 2016).

### Conclusions

We studied the time series analysis for forecasting the electrical power distribution units (EPDU) using monthly data reported by the Metropolitan Electricity

Authority from January 2010 to May 2015. The specific part of the data used in this research is the EPDU of household consumers. We generated 3 different models to predict the future EPDU by using 3 different techniques: genetic algorithm optimized support vector regression (GASVR), artificial neural network (ANN), and autoregressive integrated moving average (ARIMA). For the modeling of the GASVR and ANN, the data set must be in the form of a pair between the input vectors and targets ( $D = \{(x_i, y_i)\}_{i=1}^n$ ). The input vectors are the observed values at the lag time 1 to 12 ( $y_{t-1}, y_{t-2}, \dots, y_{t-12}$ ) and the targets are the observed values at time  $t$  ( $y_t$ ). We defined the input vectors of the GASVR and ANN models by using simple linear regression analysis and the result reveals that the observed values at the 1 and 12 lag time ( $y_{t-1}$  and  $y_{t-12}$ ) are the appropriate inputs for the GASVR and ANN models. For the ANN model, the result reveals that the ANN3 model that has 3 neurons in the hidden layer is the most suitable model. For the ARIMA model, the suitable conformation is ARIMA(1, 0, 0)x(1, 0, 0)<sub>12</sub> that means the output  $y_t$  depends on the 2 inputs at the 1 and 12 lag time ( $y_{t-1}$  and  $y_{t-12}$ ). These lagging periods are similar to the inputs of the GASVR and ANN models. Finally, we compared the performance of the 3 models using the same validation data set; we found that the GASVR model has the lowest RMSE and MAPE values because the GASVR model can capture both linear and non-linear patterns of the time series; this is the reason for its better performance when being compared with the ARIMA model. The GASVR model is also suitable to a small data set when compared with the ANN model that shows a lower performance than the GASVR model using the same training and testing data sets. Therefore, we can conclude that the idea proposed in this paper to extend the SVR technique by using the GA to optimize the parameters  $C$  and  $\epsilon$  prior to the application of SVR yields the best method for modelling the power consumption time series.

### Acknowledgements

This work is the part of learning in the doctoral program of computer engineering in the School of Computer Engineering, Suranaree University of Technology, Thailand. We are grateful to the Metropolitan Electricity Authority of Thailand for publicizing the data that had been used in this study.

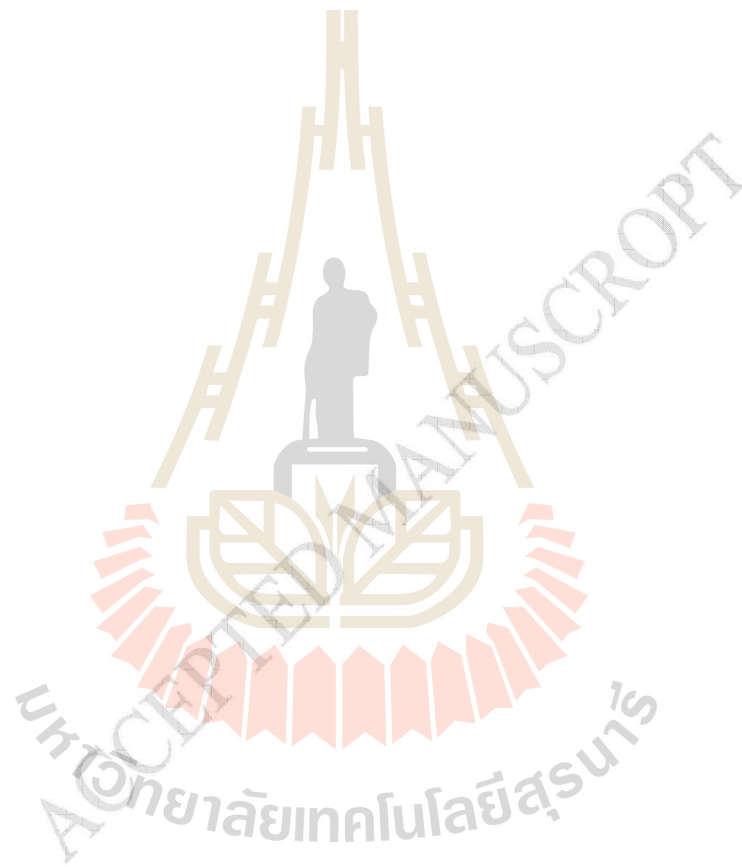
### References

- Bagheripour, P., Gholami, A., Asoodeh, M., and Asadi, M.V. (2015). Support vector regression based determination of shear wave velocity. *J. Petrol. Sci. Eng.*, 125:95-99.
- Bergmeir, C. and Benítez, J.M. (2012). On the use of cross-validation for time series predictor evaluation. *Inform. Sciences*, 191:192-213.
- Box, G.E.P. and Jenkins, G. (1990). *Time Series Analysis, Forecasting and Control*. Holden-Day, San Francisco, CA, USA, 500p.
- Bunn, D.W. and Farmer, E.D. (1985). *Comparative Models for Electrical Load Forecasting*. John Wiley and Sons, New York, NY, USA, 232p.
- Fan, G.F., Peng, L.L., Hong, W.C., and Fan, S. (2016). Electric load forecasting by the SVR model with differential empirical mode decomposition and autoregression. *Neurocomputing*, 173:958-970.

- Hyndman, R.J. (2015). *Forecasting Functions for Time Series and Linear Models*. Melbourne, Australia: Monash University. Available from: <http://github.com/robjhyndman/forecast>. Accessed date: Sep 3, 2015.
- Jirong, G., Mingcang, Z., and Liuguangyan, J. (2011). Housing price forecasting based on genetic algorithm and support vector machine. *Expert Syst. Appl.*, 38:3,383-3,386.
- Kaytez, F., Taplamacioglu, M.C., Cam, E., and Hardalac, F. (2015). Forecasting electricity consumption: A comparison of regression analysis, neural networks and least squares support vector machines. *Electrical Power and Energy Systems*, 67:431-438.
- Lee, Y.S. and Tong, L.I. (2011). Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming. *Knowledge-Based Systems*, 24:66-72.
- Metropolitan Electricity Authority. (2015). Reports of electrical distribution. Bangkok, Thailand. Available from: <http://www.me.a.or.th/download/index.php#>. Accessed date: Aug 12, 2015.
- Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., Chang, C.C., and Lin, C.C. (2015). *Misc Functions of the Department of Statistics, Probability Theory Group*. Vienna, Austria: Vienna University of Technology. Available from: <http://cran.r-project.org/web/packages/e1071/index.html>. Accessed date: Dec 5, 2015.
- Ogcu, G., Demirel, O.F., and Zaim, S. (2012). Forecasting electricity consumption with neural networks and support vector regression. *Procedia-Social and Behavioral Sciences*, 58:1,576-1,585.
- Pai, P.F. and Lin, C.S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega*, 33:497-505.
- Sajan, K.S., Kumar, V., and Tyagi, B. (2015). Genetic algorithm based support vector machine for on-line voltage stability monitoring. *Electrical Power and Energy Systems*, 73:200-208.
- Wang, J., Li, L., Niu, D., and Tan, Z. (2012a). An annual load forecasting model based on support vector regression with differential evolution algorithm. *Appl. Energ.*, 94:65-70.
- Wang, X. and Meng, M. (2012). A Hybrid Neural Network and ARIMA Model for Energy Consumption Forecasting. *Journal of Computers*, 7(5):1,184-1,190.
- Wang, Y., Wang, J., Zhao, G., and Dong, Y. (2012b). Application of residual modification approach in seasonal ARIMA for electricity demand forecasting: A case study of China. *Energ. Policy*, 48:284-294.
- Willighagen, E. and Ballings, M. (2015). R Based Genetic Algorithm. CRAN Repository. Available from: <http://github.com/egonw/genalg>. Accessed date: Dec 5, 2015.
- Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. 2nd ed. Springer-Verlag, New York, Ny, USA, 314p.

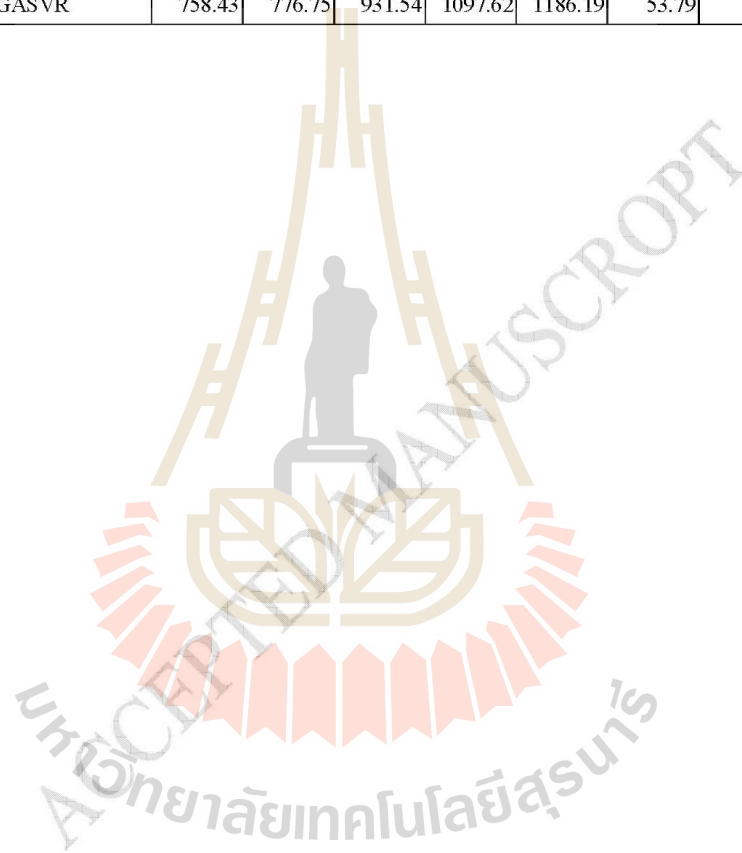


Zhang, F., Deb, C., Lee, S.E., , Yang, J., and Shah, K.W. (2016). Time series forecasting for building energy consumption using weighted Support Vector Regression with differential evolution optimization technique. *Energ. Buildings*, 126:94-103.



**Table 1. Predicted values and error of the GASVR model**

Value	Jan 2015	Feb 2015	Mar 2015	Apr 2015	May 2015	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
GASVR	758.43	776.75	931.54	1097.62	1186.19	53.79	4.40



**Table 2. The initial weight and bias of each ANN model**

Model	net.iw{1,1}	net.lw{2,1}	net.b{1}	net.b{2}
ANN1	[0.5 0.5]	[0.5]	[0.5]	[0.5]
ANN2	[0.5 0.5;0.1 0.1]	[0.4 0.4]	[0.5 0.5]	[0.5]
ANN3	[0.5 0.4;0.5 0.5;0.5 0.5]	[0.5 0.5 0.5]	[0.5 0.5 0.5]	[0.5]
ANN4	[0.5 0.4;0.5 0.5;0.5 0.5;0.5 0.5]	[0.5 0.5 0.5 0.5]	[0.5 0.5 0.5 0.5]	[0.5]



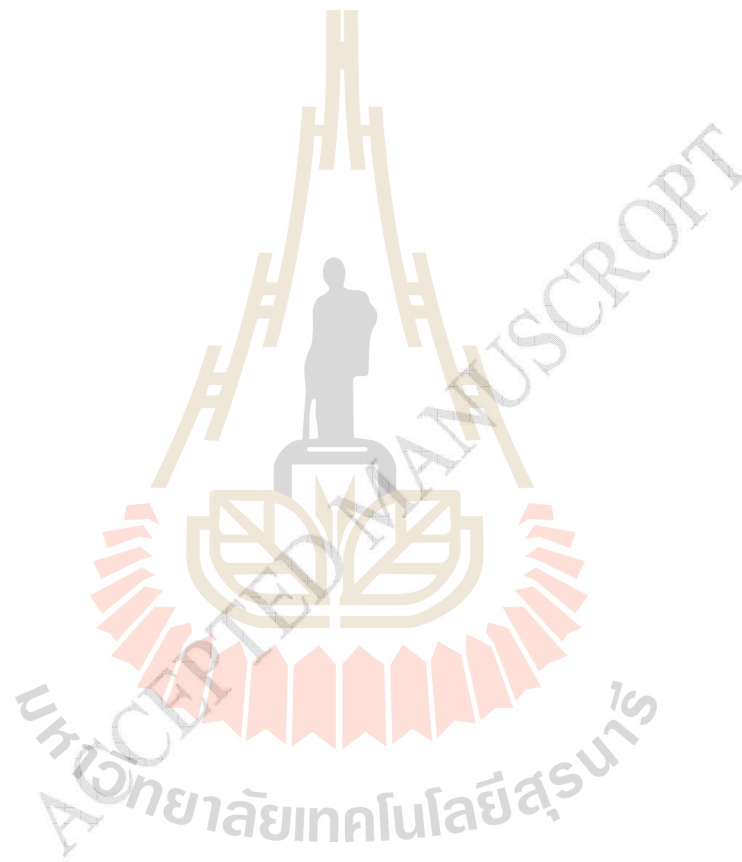
**Table 3. Predicted values and errors for the ANN models**

Value	Jan	Feb	Mar	Apr	May	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
ANN1	786.94	789.21	911.85	1096.59	1190.53	59.67	5.21
ANN2	785.26	809.44	905.99	1123.58	1193.65	58.47	5.08
ANN3	753.33	816.63	890.15	1148.87	1207.41	63.46	4.58
ANN4	759.61	760.58	913.55	1089.92	1184.74	64.66	5.33



**Table 4. Predicted values and errors of the ARIMA model**

Value	Jan	Feb	Mar	Apr	May	RMSE	MAPE
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-
ARIMA	799.24	859.33	969.4	1065.12	1117.63	57.18	4.80



**Table 5. Predicted values and errors of the 3 models**

Value	Jan	Feb	Mar	Apr	May	RMSE	MAPE	Increasing Accuracy
Actual Value	745.82	853.60	1017.89	1100.93	1216.77	-	-	-
GASVR	758.43	776.75	931.54	1097.62	1186.19	53.79	4.40	+8.33%
ANN3	753.33	816.63	890.15	1148.87	1207.41	63.46	4.58	+4.58%
ARIMA	799.24	859.33	969.4	1065.12	1117.63	57.18	4.80	0%





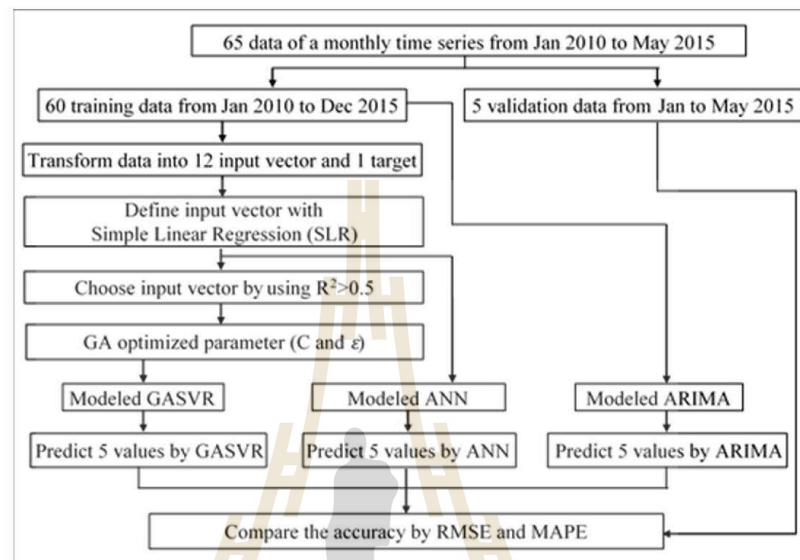
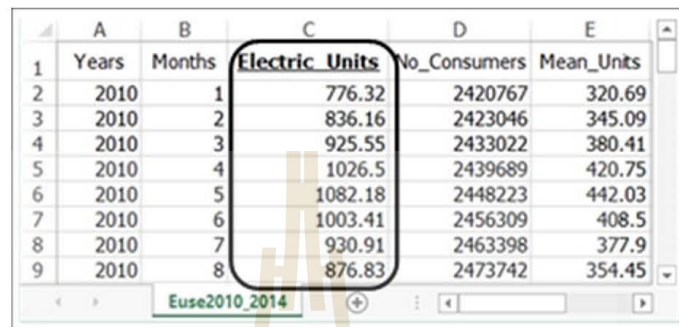


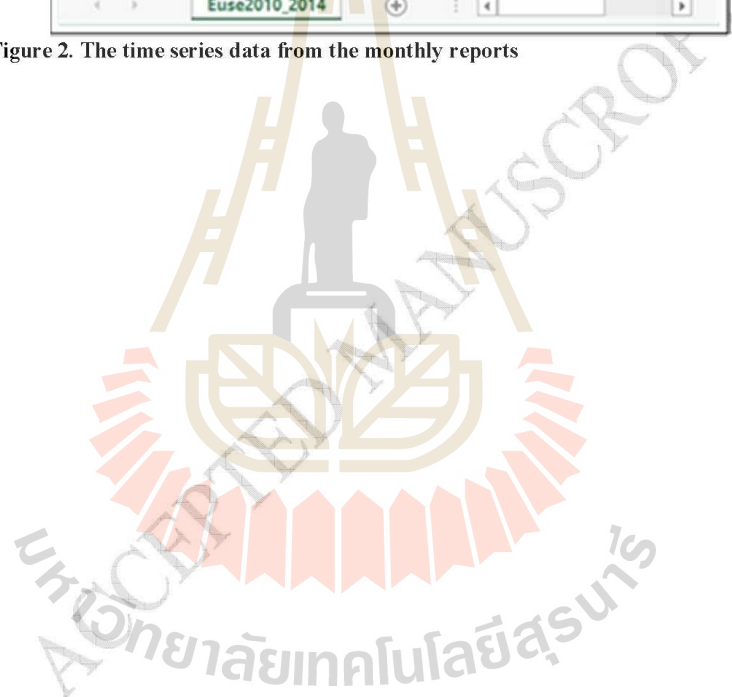
Figure 1. The conceptual framework of this research





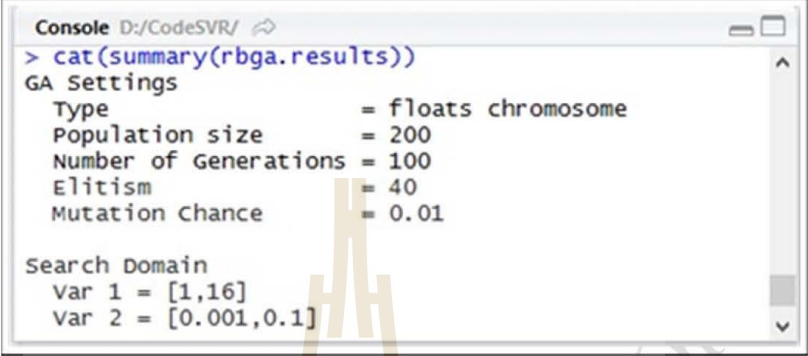
	A	B	C	D	E
1	Years	Months	Electric Units	No_Consumers	Mean_Units
2	2010	1	776.32	2420767	320.69
3	2010	2	836.16	2423046	345.09
4	2010	3	925.55	2433022	380.41
5	2010	4	1026.5	2439689	420.75
6	2010	5	1082.18	2448223	442.03
7	2010	6	1003.41	2456309	408.5
8	2010	7	930.91	2463398	377.9
9	2010	8	876.83	2473742	354.45

Figure 2. The time series data from the monthly reports



A	B	C	D	E	F	G	H	I	J	K	L	M
ValT1	ValT2	ValT3	ValT4	ValT5	ValT6	ValT7	ValT8	ValT9	ValT10	ValT11	ValT12	Target
773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	1026.5	925.55	836.16	776.32	707.16
707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	1026.5	925.55	836.16	787.29
787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	1026.5	925.55	806.64
806.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	1026.5	864.58
864.58	806.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	1082.2	968.86
968.86	864.58	806.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	1003.4	933.59
933.59	968.86	864.58	806.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	930.91	912.8
912.8	933.59	968.86	864.58	806.64	787.29	707.16	773.99	787.33	820.6	873.47	876.83	902.59
902.59	912.8	933.59	968.86	864.58	806.64	787.29	707.16	773.99	787.33	820.6	873.47	886.08

Figure 3. The data set for modeling with the GASVR



```
Console D:/CodeSVR/
> cat(summary(rbga.results))
GA Settings
Type = floats chromosome
Population size = 200
Number of Generations = 100
Elitism = 40
Mutation Chance = 0.01

Search Domain
Var 1 = [1,16]
Var 2 = [0.001,0.1]
```

Figure 4. The settings of the genetic algorithm

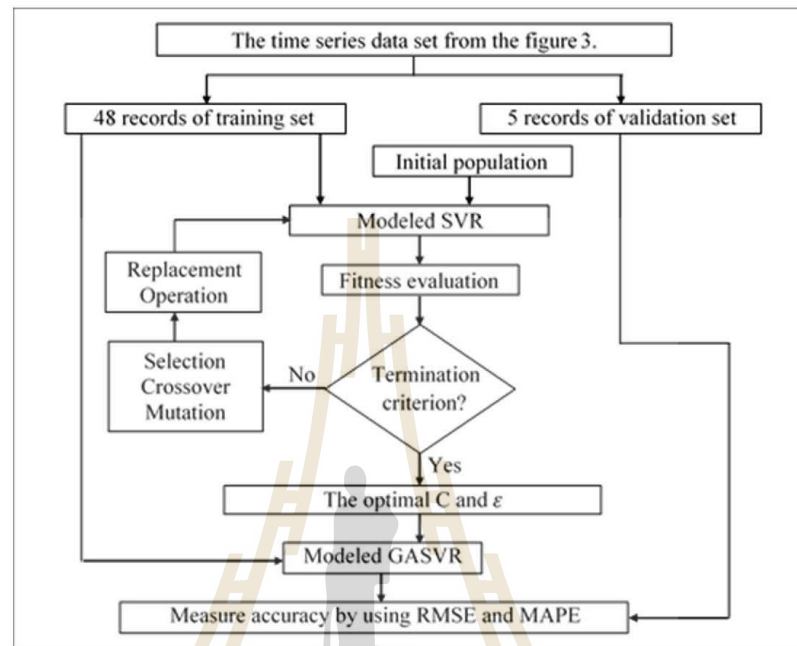


Figure 5. The operational process of the GASVR



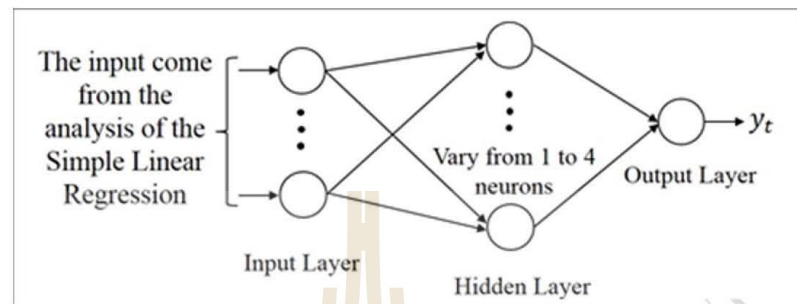
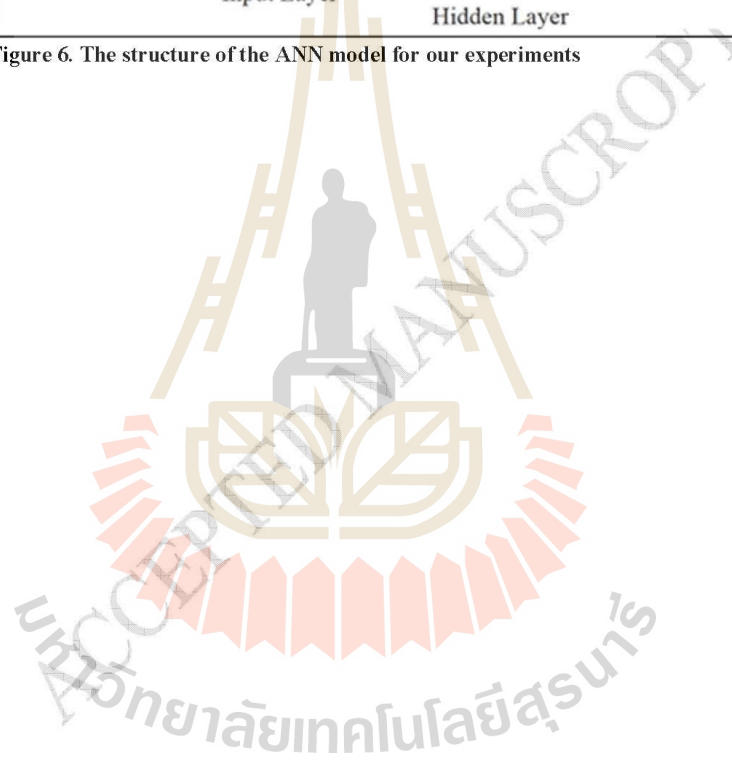


Figure 6. The structure of the ANN model for our experiments





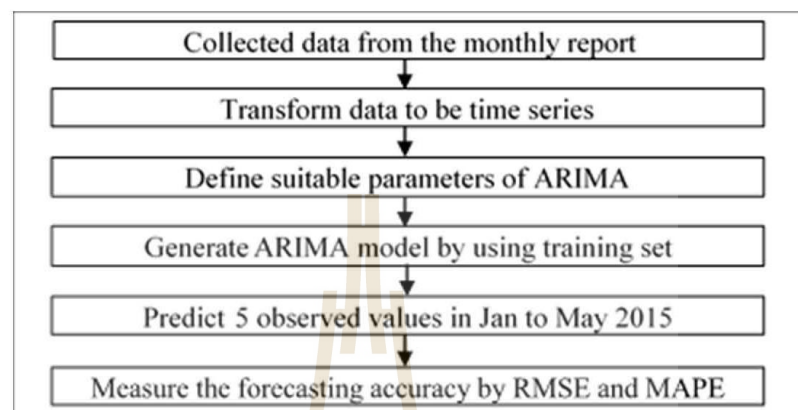
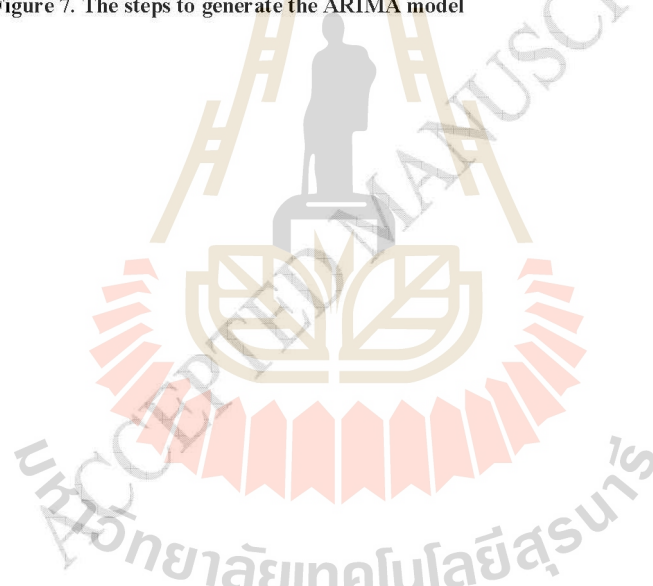


Figure 7. The steps to generate the ARIMA model

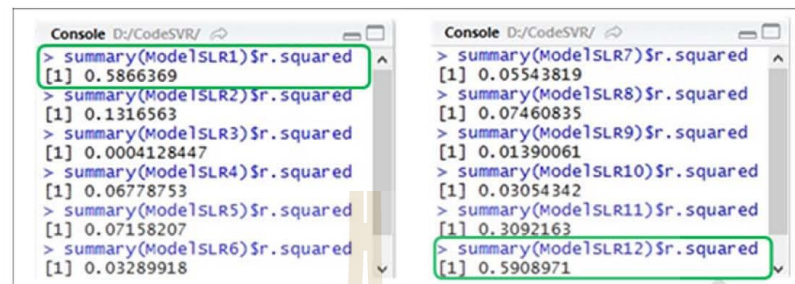


```

library("forecast", lib.loc=~R/win-library/3.2")
arimaDF <- read.csv(file="d:/R/Euse2010_2014.csv", header=TRUE, sep=",")
arimaTS <- ts(arimaDF[,3], start=c(2010,1), end=c(2014,12), frequency=12)
auto.arima(arimaTS)
arimaM <- arima(arimaTS, order=c(p,d,q), list(order=c(P,D,Q), period=12))
      # p,d,q and P,D,Q is an integer number, it's can be obtain by using
      # auto.arima()
PredictARIMA <- predict(arimaM, n.ahead=5)
PredictARIMA # show forecasting value
TestSet <- read.csv(file="d:/R/Etest1_5_2015.csv")
ActualARIMA <- c(TestSet$Electric_Units)
ForecastARIMA <- c(PredictARIMA$pred[1:5])
err1 <- ActualARIMA - ForecastARIMA
err2 <- (ActualARIMA - ForecastARIMA) / ActualARIMA
rmseARIMA <- sqrt(mean(err1^2))
mapeARIMA <- mean(abs(100*err2))
rmseARIMA      # show RMSE
mapeARIMA      # show MAPE

```

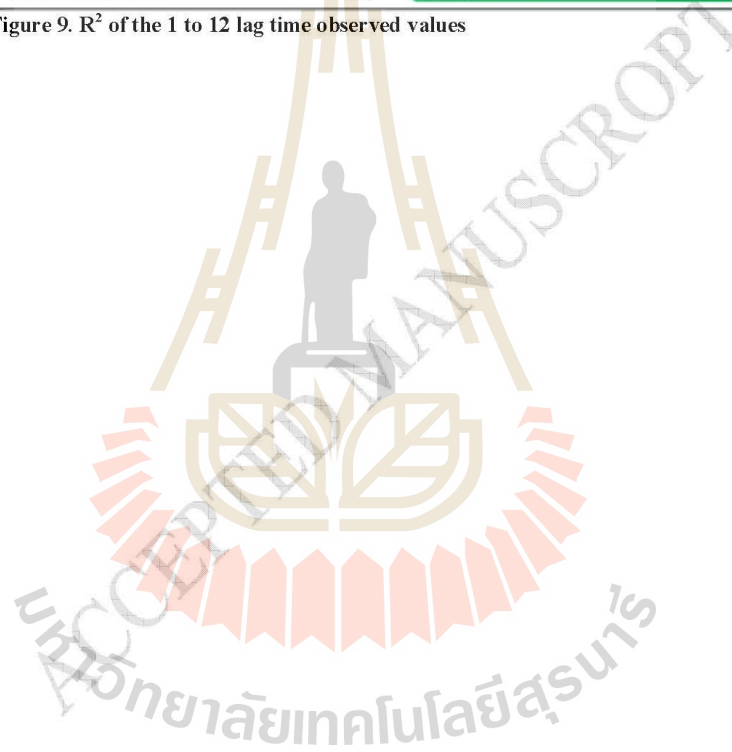
Figure 8. The command set in R language to generate the ARIMA model

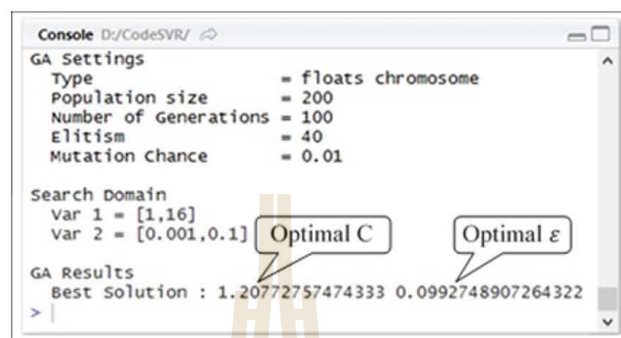


```
Console D:/CodeSVR/
> summary(ModelSLR1)$r.squared
[1] 0.5866369
> summary(ModelSLR2)$r.squared
[1] 0.1316563
> summary(ModelSLR3)$r.squared
[1] 0.0004128447
> summary(ModelSLR4)$r.squared
[1] 0.06778753
> summary(ModelSLR5)$r.squared
[1] 0.07158207
> summary(ModelSLR6)$r.squared
[1] 0.03289918

Console D:/CodeSVR/
> summary(ModelSLR7)$r.squared
[1] 0.05543819
> summary(ModelSLR8)$r.squared
[1] 0.07460835
> summary(ModelSLR9)$r.squared
[1] 0.01390061
> summary(ModelSLR10)$r.squared
[1] 0.03054342
> summary(ModelSLR11)$r.squared
[1] 0.3092163
> summary(ModelSLR12)$r.squared
[1] 0.5908971
```

Figure 9.  $R^2$  of the 1 to 12 lag time observed values



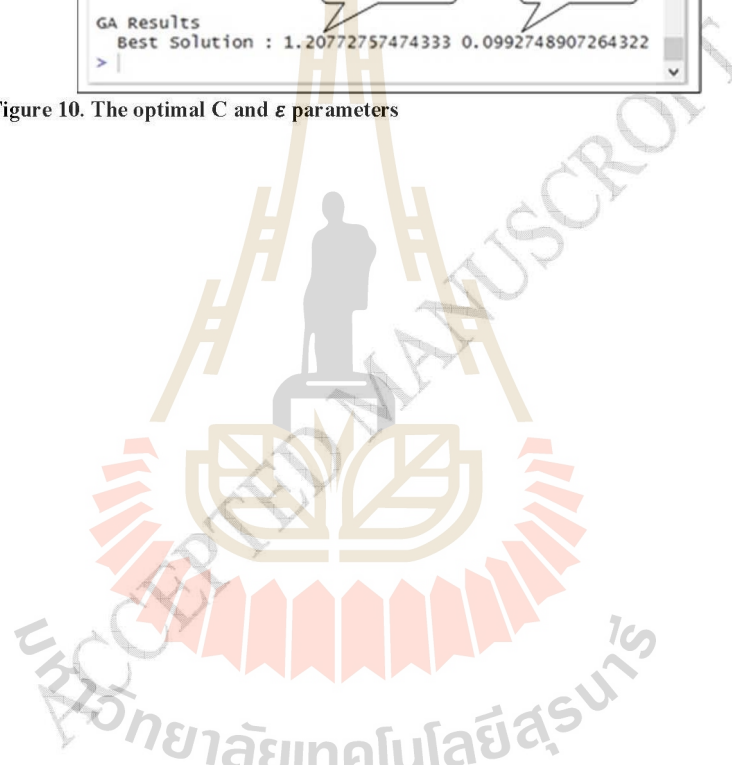


```
Console D:/CodeSVR/ ↗
GA Settings
Type           = floats chromosome
Population size = 200
Number of Generations = 100
Elitism        = 40
Mutation Chance = 0.01

Search Domain
var 1 = [1,16]
var 2 = [0.001,0.1]

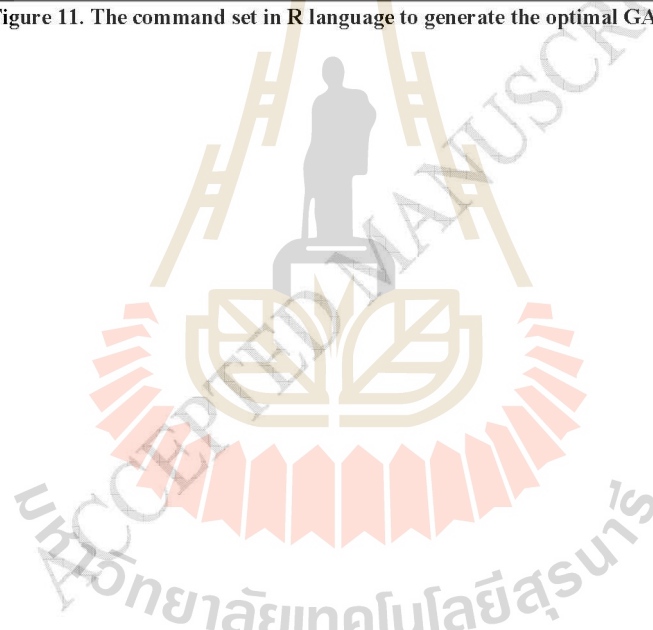
GA Results
Best Solution : 1.20772757474333 0.0992748907264322
> |
```

Figure 10. The optimal  $C$  and  $\varepsilon$  parameters



```
Console D:/CodeSVR/ ↵
> Trainload<-read.csv(file="TrainingData.csv")
> Inv<-data.frame(Trainload$valT1,Trainload$valT2)
> model <- svm(x=Inv,y=Trainload$Target,kernel="linear",cost=1.2
0772757474333,epsilon=0.0992748907264322)
> Testload<-read.csv(file="TestData2.csv")
> newdata = data.frame(val1=Testload$valT1,val2=Testload$valT2)
> ForesVM <- predict(model,newdata=newdata)
> error <- Testload$ActualValue - ForesVM
> err2 <- (Testload$ActualValue - ForesVM)/Testload$Actualvalue
> RMSE_GASVR <- sqrt(mean(error^2))
> RMSE_GASVR
[1] 53.79114
> MAPE_GASVR<-mean(abs(100*err2))
> MAPE_GASVR
[1] 4.39831
```

Figure 11. The command set in R language to generate the optimal GASVR model



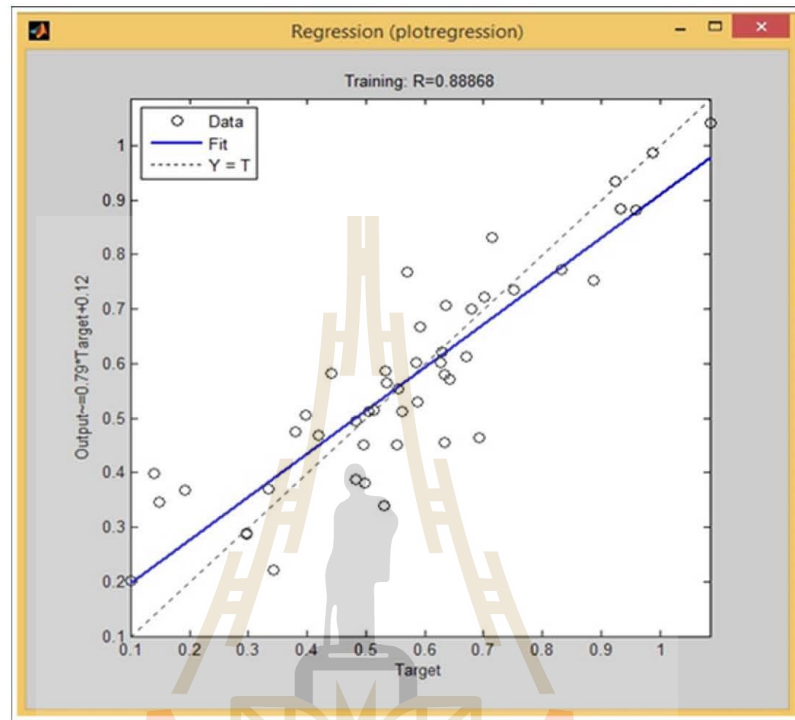


Figure 12. Regression graph of ANN3



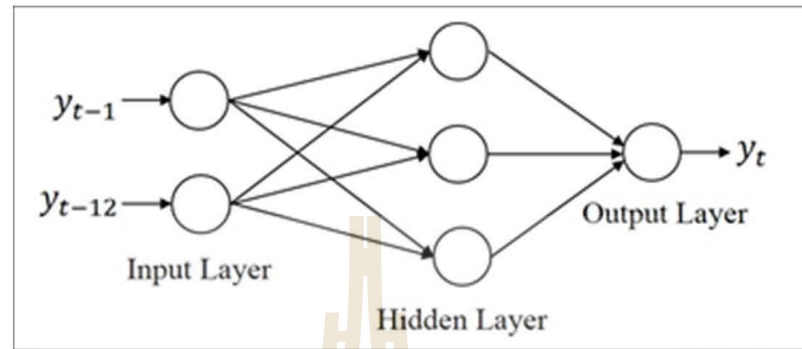


Figure 13. The structure of ANN3



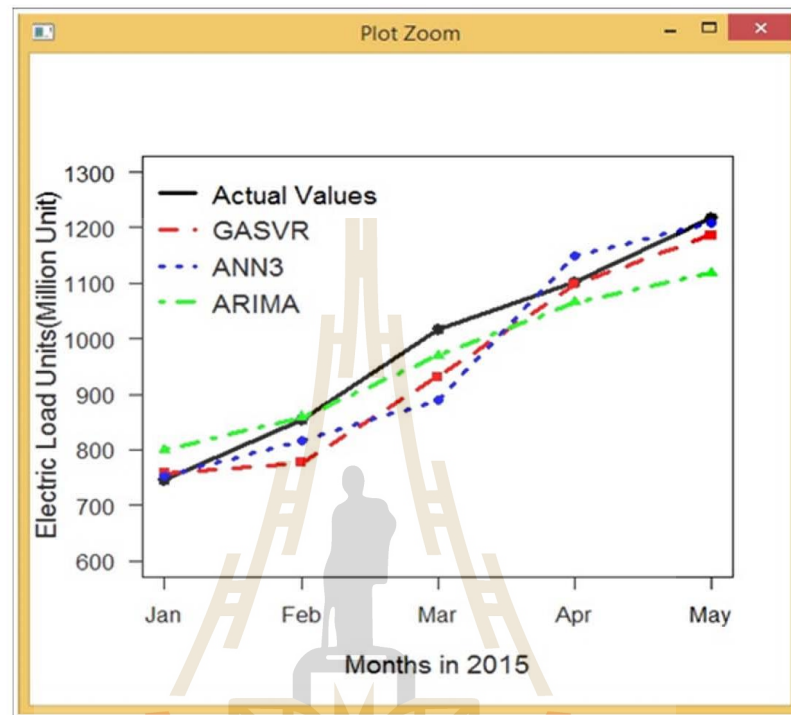
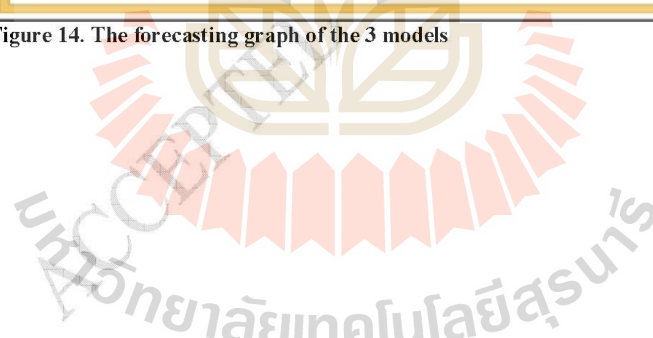


Figure 14. The forecasting graph of the 3 models



## ประวัติผู้เขียน

นายรัชชัช ชื่นรัชช เกิดเมื่อวันที่ 3 กันยายน 2516 ที่อำเภอเมือง จังหวัดนครราชสีมา เริ่มเข้าศึกษาระดับชั้นประถมศึกษาปีที่ 1 ถึง 6 ที่โรงเรียนอัสสัมชัญนครราชสีมา อำเภอเมือง จังหวัดนครราชสีมา จากนั้นศึกษาต่อในระดับมัธยมตอนต้นและตอนปลายที่โรงเรียนราชสีมาวิทยาลัย อำเภอเมือง จังหวัดนครราชสีมา ในปีการศึกษา 2533 ได้เข้าศึกษาต่อระดับปริญญาตรีในสาขาวิชาวิศวกรรมไฟฟ้า ที่มหาวิทยาลัยขอนแก่น และสำเร็จการศึกษาในปีการศึกษา 2539 หลังจากสำเร็จการศึกษาได้เข้าทำงานในบริษัทที่เกี่ยวข้องกับการรับเหมาติดตั้งระบบไฟฟ้าภายในโรงงาน เป็นเวลา 4 ปี จากนั้น ได้เข้าศึกษาต่อระดับปริญญาโทสาขาวิชาการจัดการระบบสารสนเทศ สถาบันบัณฑิตพัฒนบริหารศาสตร์ เมื่อปีพ.ศ. 2543 และสำเร็จการศึกษาในปีการศึกษา 2545 หลังจากสำเร็จการศึกษาระดับปริญญาโท ได้ทำงานทางสายงานวิชาการ ตำแหน่งอาจารย์ มหาวิทยาลัยวงษ์ชวลิตกุล ถึงปีพ.ศ. 2548 โดยมีตำแหน่งสุดท้ายเป็นหัวหน้าสาขาวิชาคอมพิวเตอร์ธุรกิจ ต่อมา ในปีพ.ศ. 2553 ได้มาดำรงตำแหน่งอาจารย์ที่โปรแกรมวิชาวิทยาการสารสนเทศ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏนครราชสีมา และในปีพ.ศ. 2557 ได้รับตำแหน่งทางวิชาการ ระดับผู้ช่วยศาสตราจารย์ จากนั้น ในปีพ.ศ. 2558 ได้รับทุนการศึกษาจากมหาวิทยาลัยราชภัฏนครราชสีมา ให้เข้าศึกษาต่อระดับปริญญาเอกในสาขาวิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีสุรนารี ในระหว่างการศึกษามีความรู้ และความอนุเคราะห์ในด้านต่าง ๆ จากอาจารย์ที่ปรึกษาและอาจารย์ประจำวิชา เป็นอย่างดี โดยมีบทความวิจัยที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างการศึกษารายละเอียดดังภาคผนวก ข