



การศึกษาการพยากรณ์ยอดขายสำหรับประกันชีวิตกรมธรรม์หลักและสัญญาเพิ่มเติม
โดยใช้วิธีการพยากรณ์แบบฤดูกาล และการพยากรณ์แนวโน้ม



วิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตาม
หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
วิทยาลัยวิศวกรรมศาสตร์

บัณฑิตวิทยาลัย มหาวิทยาลัยรังสิต
ปีการศึกษา 2567



**A STUDY OF SALES FORECASTING FOR LIFE INSURANCE ON PRIMARY
POLICY AND SUPPLEMENTARY POLICY USING SEASONAL
FORECASTING AND TREND FORECASTING METHODS**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF ENGINEERING
IN ELECTRICAL AND COMPUTER ENGINEERING
COLLEGE OF ENGINEERING**

**GRADUATE SCHOOL, RANGSIT UNIVERSITY
ACADEMIC 2024**

วิทยานิพนธ์เรื่อง
การศึกษาการพยากรณ์ยอดขายสำหรับประกันชีวิตกรมธรรม์หลักและสัญญาเพิ่มเติมโดยใช้
วิธีการพยากรณ์แบบฤดูกาล และการพยากรณ์แนวโน้ม

โดย
ปานรวี บุญสม

ได้รับการพิจารณาให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

มหาวิทยาลัยรังสิต
ปีการศึกษา 2567

รศ. ดร. เอกรัฐ บุญงา
ประธานกรรมการสอบ

ผศ. ดร. ไพบูรณ์ ช้อยหยด
กรรมการ

รศ. ดร. ดวงอาทิตย์ ศรีมูล
กรรมการและอาจารย์ที่ปรึกษา

บัณฑิตวิทยาลัยรับรองแล้ว

(ศ. ดร. สৌจิตต์ เพ็ชรประสาน)

คณบดีบัณฑิตวิทยาลัย

15 พฤษภาคม 2568

Thesis entitled

**A STUDY OF SALES FORECASTING FOR LIFE INSURANCE ON PRIMARY
POLICY AND SUPPLEMENTARY POLICY USING SEASONAL
FORECASTING AND TREND FORECASTING METHODS**

by
PANRAWE BOONSOM

was submitted in partial fulfillment of the requirements
for the degree of Master of Engineering in Electrical and Computer Engineering

Rangsit University
Academic Year 2024

Assoc. Prof. Akkarat Boonpoonga, D.Eng.
Examination Committee Chairperson

Asst. Prof. Paiboon Yoiod, D.Eng.
Member

Assoc. Prof. Duang-arthit Srimoon, D.Eng.
Member and Advisor

Approved by Graduate School

(Prof. Suejit Pechprasarn, Ph.D.)

Dean of Graduate School

May 15, 2025

กิตติกรรมประกาศ

การทำวิจัยฉบับนี้สำเร็จลุล่วงไปด้วยความกรุณาและการสนับสนุนจาก รศ.ดร. ดวงอาทิตย์ ศรีมูล อาจารย์ที่ปรึกษาวิทยานิพนธ์ ข้าพเจ้าขอขอบพระคุณที่ท่านได้สละเวลาให้คำปรึกษาอันเป็นประโยชน์อย่างยิ่งทั้งในด้านวิชาการและการแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้นระหว่างการทำวิจัยและให้คำแนะนำที่มีคุณค่าเสมอมา ข้าพเจ้าขอขอบพระคุณอาจารย์ทุกท่านในหลักสูตรที่ได้ให้คำแนะนำที่เป็นประโยชน์ในทุกขั้นตอนของการทำวิทยานิพนธ์และคณะกรรมการสอบวิทยานิพนธ์ทุกท่านที่ได้กรุณาให้ข้อเสนอแนะที่สำคัญซึ่งทำให้สามารถทำวิจัยนี้สำเร็จได้อย่างสมบูรณ์ ข้าพเจ้าขอแสดงความขอบคุณต่อคณาจารย์คณะวิศวกรรมศาสตร์ และคณาจารย์นอกภาควิชาทุกท่าน ที่ได้ให้ความรู้ ข้อเสนอแนะ และการสนับสนุนในทุกด้าน อันเป็นส่วนสำคัญที่ทำให้การทำวิจัยฉบับนี้สำเร็จไปได้ด้วยดี ข้าพเจ้าขอขอบพระคุณทุกท่านอย่างยิ่งสำหรับทุกการสนับสนุนที่ได้รับในการทำงานวิจัยครั้งนี้

สุดท้ายนี้ความมุ่งมั่นตั้งใจของผู้วิจัยเป็นสิ่งสำคัญและกำลังใจจากเพื่อนเป็นส่วนประกอบสำหรับการสำเร็จการศึกษาหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปานรวี บุญสม
ผู้วิจัย

6405537 : ปานรวี บุญสม
 ชื่อวิทยานิพนธ์ : การศึกษาการพยากรณ์ยอดขายสำหรับประกันชีวิตกรมธรรม์หลักและสัญญา
 เพิ่มเติมโดยใช้วิธีการพยากรณ์แบบฤดูกาล และการพยากรณ์แนวโน้ม
 หลักสูตร : วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
 อาจารย์ที่ปรึกษา : รศ. ดร. ดวงอาทิตย์ ศรีมูล

บทคัดย่อ

งานวิจัยนี้นำเสนอการพยากรณ์ยอดขายประกันชีวิตโดยใช้วิธีการ 4 วิธี ได้แก่ การบวก
 โซลท์วินเทอร์ การคูณโซลท์วินเทอร์ ปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย และปรับเรียบเอ็กซ์
 โพเนนเชียลสองเท่า โดยจะใช้วิธีคำนวณค่าเฉลี่ย 3 วิธีในการประเมินค่าความคลาดเคลื่อนสำหรับ
 การพยากรณ์ ได้แก่ ค่าความผิดพลาดสัมบูรณ์เฉลี่ย ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย และ
 ค่าความผิดพลาดกำลังสองเฉลี่ย จากนั้นทำการวัดค่าความแม่นยำของการพยากรณ์แต่ละวิธีเพื่อ
 เลือกวิธีการพยากรณ์ที่แม่นยำและเหมาะสมที่สุด ผลการวิจัยพบว่าการพยากรณ์ยอดขายประกัน
 ชีวิตจากการประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป การใช้การคูณโซลท์วิน
 เทอร์ มีความแม่นยำสูงสุดที่ 98.43% การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก วิธีการคูณ
 โซลท์วินเทอร์มีความแม่นยำสูงสุดถึง 86.12% การพยากรณ์ยอดขายประกันชีวิตสัญญาเพิ่มเติมวิธี
 ที่แม่นยำที่สุดคือการบวกโซลท์วินเทอร์มีเปอร์เซ็นต์ความแม่นยำสูงสุดถึง 94.61% และการ
 พยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ วิธีการคูณโซลท์วินเทอร์มัลติพลิเคทีฟ มีความ
 แม่นยำสูงสุดที่ 98.72% ซึ่งสูงกว่าวิธีการอื่น ๆ อย่างชัดเจน

(วิทยานิพนธ์มีจำนวนทั้งสิ้น 191 หน้า)

คำสำคัญ : การทำนาย, การบวกโซลท์วินเทอร์, การคูณโซลท์วินเทอร์, ปรับเรียบเอ็กซ์
 โพเนนเชียลอย่างง่าย, ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า, ค่าเบี่ยงเบน
 สัมบูรณ์เฉลี่ย, เปอร์เซ็นต์สัมบูรณ์เฉลี่ย, ข้อผิดพลาดกำลังสองเฉลี่ย

6405537 : Panrawe Boonsom

Thesis Title : A Study of Sales Forecasting for Life Insurance on Primary Policy and Supplementary Policy Using Seasonal Forecasting and Trend Forecasting Methods

Program : Master of Engineering in Electrical and Computer Engineering

Thesis Advisor : Assoc. Prof. Duang-arthit Srimoon, D.Eng.

Abstract

This research explores the application of four time series forecasting methods—Holt-Winters' Additive, Holt-Winters' Multiplicative, Exponential Smoothing, and Double Exponential Smoothing—to life insurance sales data. The forecasting accuracy of each method was evaluated using three standard error metrics: Mean Absolute Deviation (MAD), Mean Absolute Percentage Error (MAPE), and Mean Square Error (MSE). The most accurate and appropriate forecasting method was selected based on performance assessment. The research findings suggest that the Holt-Winters' Multiplicative method is the most effective technique for forecasting life insurance premiums. This method yielded the highest accuracy, with a maximum of 98.43% for general life insurance products and 86.12% for primary policy premiums. For supplementary policy premiums, the Holt-Winters' Additive method proved most accurate, achieving a maximum accuracy of 94.61%. The Holt-Winters' Multiplicative method yielded the highest accuracy (98.72%) in forecasting total sales across all policies.

(Total 191 pages)

Keywords : Forecasting, Holt-Winters' Additive, Holt-Winters' Multiplicative, Simple Exponential Smoothing, Double Exponential Smoothing, Mean Absolute Deviation, Mean Absolute Percentage, Mean Square Error

สารบัญ

	หน้า
กิตติกรรมประกาศ	ก
บทคัดย่อภาษาไทย	ข
บทคัดย่อภาษาอังกฤษ	ค
สารบัญ	ง
สารบัญตาราง	ญ
สารบัญรูป	ฉ
บทที่ 1	บทนำ
	1
1.1	ความเป็นมาและความสำคัญของโครงการ
	1
1.2	วัตถุประสงค์ของงานวิจัย
	3
1.3	สมมติฐานการวิจัย
	4
1.4	ขอบเขตของการศึกษา
	5
1.5	ประโยชน์ที่จะได้รับจากการศึกษา
	5
1.6	แนวทางการดำเนินงาน
	5
1.7	แผนการดำเนินงาน
	6
บทที่ 2	หลักการและทฤษฎีที่เกี่ยวข้อง
	9
2.1	นิยามและความหมายคำศัพท์ในการพยากรณ์ยอดขายประกันชีวิต
	9
2.2	ทฤษฎีการพยากรณ์ยอดขายประกันชีวิต
	10
2.2.1	การพยากรณ์โดยวิธีการบวกโฮสต์-วินเทอร์
	11
2.2.2	การพยากรณ์โดยวิธีการคูณโฮสต์-วินเทอร์
	12
2.2.3	การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย
	14
2.2.4	การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า
	15
2.3	การคำนวณค่าพยากรณ์
	16
2.3.1	ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD)
	16

สารบัญ (ต่อ)

	หน้า
2.3.2 ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย (Mean Absolute Percentage Error, MAPE)	17
2.3.3 ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE)	17
2.4 ข้อมูลยอดขายประกันภัยที่เกี่ยวข้องกับงานวิจัย	18
2.5 ภาษาที่ใช้เขียนโปรแกรม	19
2.5.1 การพัฒนาเว็บฝั่งเซิร์ฟเวอร์	19
2.5.2 ระบบอัตโนมัติด้วยสคริปต์ Python	20
2.5.3 วิทยาศาสตร์ข้อมูลและแมชชีนเลิร์นนิง	20
2.5.4 การพัฒนาซอฟต์แวร์	21
2.5.5 ระบบทดสอบซอฟต์แวร์อัตโนมัติ	21
2.6 การเขียน Python ด้วย Google Colab	22
บทที่ 3 ขั้นตอนการดำเนินงาน	28
3.1 การสร้างไฟล์ CSV สำหรับยอดขายประกันชีวิตด้วยข้อมูลที่รวบรวมจากเว็บไซต์สำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย (คปภ.)	31
3.2 การโหลดและเตรียมข้อมูล โค้ดสำหรับโหลดและเตรียมข้อมูล	31
3.3 พยากรณ์ยอดขายประกันชีวิตด้วย 4 วิธีการ	32
3.4 คำนวณค่าพยากรณ์ด้วย 3 วิธีการ	32
3.5 คำนวณความแม่นยำ สรุปผลและนำเสนอ	33
3.6 เปรียบเทียบผลลัพธ์และเลือกวิธีที่ดีที่สุด	34
3.7 การ Optimize ค่าพารามิเตอร์	35
3.8 โค้ดโปรแกรมสำหรับการพยากรณ์	36
3.8.1 โค้ดการพยากรณ์ด้วยวิธีการบวกโฮสต์-วินเทอร์	36
3.8.2 โค้ดการพยากรณ์ด้วยวิธีการคูณโฮสต์-วินเทอร์	42

สารบัญ (ต่อ)

	หน้า
3.8.3 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย	48
3.8.4 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า	51
3.9 โค้ดคำนวณความแม่นยำ	56
3.10 โค้ดการ Optimize ค่าพารามิเตอร์	58
3.10.1 โค้ดการ Optimize ค่าพารามิเตอร์วิธีการบวกโฮสต์-วินเทอร์	59
3.10.2 โค้ดการ Optimize ค่าพารามิเตอร์วิธีการคูณโฮสต์-วินเทอร์	61
3.10.3 โค้ดการ Optimize ค่าวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย	63
3.10.4 โค้ดการ Optimize ค่าวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า	65
บทที่ 4 ผลการวิจัย	68
4.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป	69
4.1.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีการบวกโฮสต์-วินเทอร์	70
4.1.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีการคูณโฮสต์-วินเทอร์	73
4.1.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย	76
4.1.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า	78

สารบัญ (ต่อ)

	หน้า
4.1.5 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีต่าง ๆ	81
4.1.6 ผลลัพธ์การวัดข้อผิดพลาดการพยากรณ์กรมธรรม์แบบทั่วไป	82
4.1.7 ผลลัพธ์ความแม่นยำของการพยากรณ์กรมธรรม์แบบทั่วไป	86
4.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก	87
4.2.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีการบวกโสมท์-วินเทอร์	88
4.2.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีการคูณโสมท์-วินเทอร์	90
4.2.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย	93
4.2.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า	96
4.2.5 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีต่าง ๆ	99
4.2.6 ผลลัพธ์การวัดข้อผิดพลาดการพยากรณ์กรมธรรม์หลัก	100
4.2.7 ผลลัพธ์ความแม่นยำของการพยากรณ์กรมธรรม์หลัก	104
4.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม	105
4.3.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมด้วยวิธีการบวกโสมท์-วินเทอร์	106
4.3.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมด้วยวิธีการคูณโสมท์-วินเทอร์	109
4.3.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย	112

สารบัญ (ต่อ)

	หน้า
4.3.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญา เพิ่มเติมด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า	115
4.3.5 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญา เพิ่มเติมด้วยวิธีต่าง ๆ	118
4.3.6 ผลลัพธ์ความแม่นยำของการพยากรณ์กรมธรรม์เพิ่มเติม	119
4.3.7 ผลลัพธ์ความแม่นยำของการพยากรณ์กรมธรรม์เพิ่มเติม	123
4.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์	124
4.4.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ ด้วยวิธีการบวกโสมท์-วินเทอร์	124
4.4.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ ด้วยวิธีการคูณโสมท์-วินเทอร์	127
4.4.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย	130
4.4.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า	134
4.4.5 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ ด้วยวิธีต่าง ๆ	137
4.4.6 ผลลัพธ์การวัดข้อผิดพลาดการพยากรณ์รวมทุกกรมธรรม์	139
4.4.7 ผลลัพธ์ความแม่นยำของการพยากรณ์รวมทุกกรมธรรม์	143
บทที่ 5 บทสรุปและข้อเสนอแนะ	145
5.1 บทสรุป	145
5.2 ข้อเสนอแนะ	146
บรรณานุกรม	147

สารบัญ (ต่อ)

	หน้า
ภาคผนวก	149
ภาคผนวก ก	โค้ดโปรแกรมภาษา Python
ภาคผนวก ข	ค่าการเพิ่มประสิทธิภาพข้อผิดพลาดของการพยากรณ์
	ค่าการพยากรณ์ยอดขายประกันชีวิต ค่าการคำนวณข้อผิดพลาด
	การพยากรณ์เปอร์เซ็นต์ความแม่นยำสำหรับวิธีการพยากรณ์
ประวัติผู้วิจัย	191



สารบัญตาราง

ตารางที่	หน้า
1.1 แผนการดำเนินงาน	6
1.1 แผนการดำเนินงาน (ต่อ)	7
1.1 แผนการดำเนินงาน (ต่อ)	8
2.1 มูลค่ายอดขายประกันชีวิตกรมธรรม์แบบทั่วไป กรมธรรม์หลัก สัญญา เพิ่มเติมจากกรมธรรม์หลัก และรวมทุกกรมธรรม์	18
2.1 มูลค่ายอดขายประกันชีวิตกรมธรรม์แบบทั่วไป กรมธรรม์หลัก สัญญา เพิ่มเติมจากกรมธรรม์หลัก และรวมทุกกรมธรรม์ (ต่อ)	19
4.1 ผลการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีต่าง ๆ	70
4.2 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีต่าง ๆ	87
4.2 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีต่าง ๆ (ต่อ)	88
4.3 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธี ต่าง ๆ	105
4.3 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธี ต่าง ๆ (ต่อ)	106
4.4 ผลการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีต่าง ๆ	124

สารบัญรูป

รูปที่	หน้า
2.1	เว็บไซต์ Google Colab 22
2.2	การ Sign ต้องมีบัญชีของ Google 23
2.3	การนำบัญชี Gmail มา Sign In ในการใช้งาน Colab 23
2.4	หน้าจอจากเว็บไซต์ Google Colab สำหรับ เขียนโปรแกรมภาษา Python 24
2.5	เริ่มการเขียนโปรแกรมภาษา Python ด้วย Google Colab 24
2.6	ทดสอบ Code จากโปรแกรมภาษา Python ที่ Google Colab 25
2.7	การเพิ่มช่องใหม่สำหรับเขียน Code โปรแกรมภาษา Python ที่ Google Colab 25
2.8	การแทรกข้อความหรือบทความที่ Google Colab 26
2.9	การเขียน Code และแทรกข้อความสลับกันที่ Google Colab 26
2.10	การเปลี่ยนชื่อ Notebook ที่ Google Colab 27
3.1	กระบวนการวิเคราะห์และพยากรณ์ยอดขายกรมธรรม์ประกันภัย พร้อม การประเมินความแม่นยำของแบบจำลอง กรณีศึกษาปี ค.ศ. 2022 28
3.2	การโหลดและเตรียมข้อมูลสำหรับการพยากรณ์ยอดขายประกัน 31
3.3	ฟังก์ชันสำหรับการประมวลผลการพยากรณ์ยอดขายประกันชีวิตในการ พยากรณ์ด้วยโปรแกรมภาษา Python 35
3.4	โค้ดการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนฟังก์ชันคำนวณค่า MSE MAD MAPE และฟังก์ชัน Holt-Winters Additive 37
3.5	โค้ดการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนลูป คำนวณค่า เริ่มต้นของ Level Trend และ Seasonality 38
3.5	โค้ดการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนลูป คำนวณค่า เริ่มต้นของ Level Trend และ Seasonality (ต่อ) 39
3.6	โค้ดการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนคำนวณค่าที่ พยากรณ์ คำนวณค่าที่พยากรณ์ล่วงหน้า และคำนวณค่าความคลาดเคลื่อน (Errors) 40

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.6	ได้ผลการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนคำนวณค่าที่พยากรณ์ คำนวณค่าที่พยากรณ์ล่วงหน้า และคำนวณค่าความคลาดเคลื่อน (Errors) (ต่อ)	41
3.7	ได้ผลการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนรวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์ และคืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน	42
3.8	ได้ผลการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนฟังก์ชันคำนวณค่า MSE MAD MAPE และ ฟังก์ชัน Holt-Winters Multiplicative	42
3.8	ได้ผลการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนฟังก์ชันคำนวณค่า MSE MAD MAPE และ ฟังก์ชัน Holt-Winters Multiplicative (ต่อ)	43
3.9	ได้ผลการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนรูป และ คำนวณค่า Level	44
3.10	ได้ผลการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนคำนวณค่า Trend Seasonality ค่าที่พยากรณ์ (Forecast) และคำนวณค่าพยากรณ์ล่วงหน้า	45
3.10	ได้ผลการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนคำนวณค่า Trend Seasonality ค่าที่พยากรณ์ (Forecast) และคำนวณค่าพยากรณ์ล่วงหน้า (ต่อ)	46
3.11	ได้ผลการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนคำนวณค่าความคลาดเคลื่อน (Errors) รวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์ และคืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน	47
3.12	ได้ผลการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ส่วน Import Numpy ฟังก์ชันคำนวณค่า Mean Squared Error (MSE) และ ค่า Mean Absolute Deviation (MAD)	48
3.13	ได้ผลการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ส่วนฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE) ฟังก์ชัน Simple Exponential Smoothing (SES) รูป คำนวณค่าที่พยากรณ์ และคำนวณค่าพยากรณ์ล่วงหน้า	49

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.13	โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ส่วนฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE) ฟังก์ชัน Simple Exponential Smoothing (SES) ดูป คำนวณค่าที่พยากรณ์ และ คำนวณค่าพยากรณ์ล่วงหน้า (ต่อ)	50
3.14	โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ส่วน คำนวณค่าความคลาดเคลื่อน (Errors) เริ่มจากข้อมูลจุดที่ 9	50
3.15	โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนรวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์ คำนวณค่าที่พยากรณ์และค่าความคลาดเคลื่อน และฟังก์ชันคำนวณค่า Mean Squared Error (MSE) และฟังก์ชันคำนวณค่า Mean Absolute Deviation (MAD)	51
3.16	โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE) ฟังก์ชัน Double Exponential Smoothing (DES) สำหรับพยากรณ์ข้อมูลอนุกรมเวลา และ ดูป	52
3.16	โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE) ฟังก์ชัน Double Exponential Smoothing (DES) สำหรับพยากรณ์ข้อมูลอนุกรมเวลา และ ดูป (ต่อ)	53
3.17	โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนกรณีข้อมูลจุดที่สอง กำหนดค่าเริ่มต้นสำหรับ Level Trend คำนวณค่า Level คำนวณค่าที่พยากรณ์ คำนวณค่าพยากรณ์ล่วงหน้า และ สร้างลิสต์ว่างสำหรับเก็บค่าพยากรณ์ในอนาคต	54
3.18	โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนเก็บค่าของช่วงการพยากรณ์ คำนวณค่าความคลาดเคลื่อน (Errors) เริ่มจากข้อมูลจุดที่ 9	55

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.18	โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนเก็บค่าของช่วงการพยากรณ์ คำนวณค่าความคลาดเคลื่อน (Errors) เริ่มจากข้อมูลจุดที่ 9 (ต่อ)	56
3.19	นำเข้าไลบรารี Seaborn สำหรับการวาดกราฟ และ นำเข้าไลบรารี Matplotlib สำหรับการวาดกราฟ และสร้างลิสต์ของชื่อวิธีการพยากรณ์	56
3.19	นำเข้าไลบรารี Seaborn สำหรับการวาดกราฟ และ นำเข้าไลบรารี Matplotlib สำหรับการวาดกราฟ และสร้างลิสต์ของชื่อวิธีการพยากรณ์ (ต่อ)	57
3.20	สร้างลิสต์ของค่า Accuracy สร้างกราฟแท่ง (Bar Plot) เพิ่มข้อความแสดงค่าบนกราฟแท่ง และ แสดงกราฟ	57
3.20	สร้างลิสต์ของค่า Accuracy สร้างกราฟแท่ง (Bar Plot) เพิ่มข้อความแสดงค่าบนกราฟแท่ง และ แสดงกราฟ (ต่อ)	58
3.21	เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับ Optimize ค่าพารามิเตอร์วิธีการบวกไฮลท์-วินเทอร์ และสร้างกราฟ 3 มิติ	59
3.22	แยกค่าของ Alpha Beta Gamma ค่า MSE สำหรับใช้ในการพล็อตกราฟ สร้างกราฟแบบ Scatter Plot ในรูปแบบ 3 มิติ และเพิ่มแถบสี (Colorbar)	60
3.23	เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับ Optimize ค่าพารามิเตอร์วิธีการคูณไฮลท์-วินเทอร์	61
3.24	แสดงผลลัพธ์ของค่า MSE MAD และ MAPE ที่น้อยที่สุด สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ และเพิ่มแถบสี (Colorbar)	62
3.24	แสดงผลลัพธ์ของค่า MSE MAD และ MAPE ที่น้อยที่สุด สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ และเพิ่มแถบสี (Colorbar) (ต่อ)	63

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.25	เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับ Optimize ค่าพารามิเตอร์วิธีการคูณโฮสต์-วินเทอร์ สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ
3.25	เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับ Optimize ค่าพารามิเตอร์วิธีการคูณโฮสต์-วินเทอร์ สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ (ต่อ)
3.26	เพิ่มแถบสี (Colorbar) เพื่อแสดงระดับของค่า MSE ในกราฟ และแสดงกราฟ
3.27	เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับการ Optimize ค่าวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ
3.28	สร้างกราฟแบบ Scatter Plot ในรูปแบบ 3 มิติ แสดงค่าของ MSE และเพิ่มแถบสี (Colorbar) เพื่อแสดงระดับของค่า MSE ในกราฟ
4.1	ตัวอย่างการอัปโหลดไฟล์ Sales_Primary-General.csv ที่ Colab
4.2	คำสั่งการโหลด
4.3	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีการบวกโฮสต์-วินเทอร์ระหว่างปี ค.ศ. 2018 - 2022
4.4	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีการคูณโฮสต์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022
4.5	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายระหว่างปี ค.ศ. 2018 - 2022
4.6	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าระหว่างปี ค.ศ. 2018 - 2022

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.7	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022
4.8	แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) สำหรับการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป
4.9	แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) สำหรับการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป
4.10	แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) สำหรับการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป
4.11	แผนภูมิแท่งแสดงผลลัพธ์ความแม่นยำของการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป
4.12	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีการบวกไฮลท์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022
4.13	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีการคูณไฮลท์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022
4.14	แผนภูมิเส้นแสดงการพยากรณ์ประกันชีวิตกรมธรรม์หลักโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ระหว่างปี ค.ศ. 2018 - 2022
4.15	แผนภูมิเส้นแสดงการพยากรณ์ประกันชีวิตกรมธรรม์หลักโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ระหว่างปี ค.ศ. 2018 - 2022
4.16	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022
4.17	แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก
4.18	แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.19	แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก	104
4.20	แผนภูมิแท่งแสดงผลลัพธ์ความแม่นยำของการพยากรณ์กรมธรรม์หลัก	105
4.21	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีการบวกโฮสต์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022	108
4.22	ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีการคูณโฮสต์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022	109
4.23	ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายระหว่างปี ค.ศ. 2018 - 2022	112
4.24	ผลการพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่ายอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม ระหว่างปี ค.ศ. 2018 - 2022	115
4.25	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022	118
4.26	แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม	120
4.27	แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม	121
4.28	แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม	122
4.29	แผนภูมิแท่งแสดงผลลัพธ์ความแม่นยำของการพยากรณ์กรมธรรม์เพิ่มเติม	123
4.30	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีการบวกโฮสต์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022	126

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.31	แผนภูมิเส้นการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีการคูณไฮลท์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022	128
4.32	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ระหว่างปี ค.ศ. 2018 - 2022	131
4.33	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ระหว่างปี ค.ศ. 2018 - 2022	135
4.34	แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022	138
4.35	แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) สำหรับยอดขายประกันชีวิตรวมทุกกรมธรรม์	140
4.36	แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) สำหรับยอดขายประกันชีวิตรวมทุกกรมธรรม์	141
4.37	แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) สำหรับยอดขายประกันชีวิตรวมทุกกรมธรรม์	142
4.38	แผนภูมิแท่งแสดงผลลัพธ์ความแม่นยำของการพยากรณ์รวมทุกกรมธรรม์	143

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

ปัจจุบันธุรกิจประกันภัยมีการเติบโตอย่างต่อเนื่อง ซึ่ง 10 อันดับต้นของบริษัทประกันชีวิตในประเทศไทยสำหรับปี พ.ศ.2565 ที่มียอดขายประกันชีวิตมากที่สุด ข้อมูลจาก “สำนักข่าวอีไฟแนนซ์ไทย” (สำนักข่าวอีไฟแนนซ์ไทย, 2565) คือ เอไอเอ ประเทศไทย บริษัท เอฟดับบลิวดี ประกันชีวิต จำกัด (มหาชน) บริษัท ไทยประกันชีวิต จำกัด (มหาชน) บริษัท เมืองไทยประกันชีวิต จำกัด (มหาชน) บริษัท กรุงไทย-แอกซ่า ประกันชีวิต จำกัด (มหาชน) บริษัท กรุงเทพประกันชีวิต จำกัด (มหาชน) บริษัท อลิอันซ์ อยุธยา ประกันชีวิต จำกัด (มหาชน) บริษัท พูเด็นเซียล ประกันชีวิต (ประเทศไทย) จำกัด (มหาชน) บริษัท ไทยสมุทรประกันชีวิต จำกัด (มหาชน) และ บริษัท เจเนอราลี่ ประกันชีวิต (ไทยแลนด์) จำกัด (มหาชน) ซึ่งมีส่วนแบ่งการตลาด(มาร์เก็ตแชร์) รวม 93.7% และยอดขายประกันภัยรับรวม 179,138.60 ล้านบาท

การทำประกันภัยโดยเฉพาะประกันชีวิตเป็นส่วนที่ผู้คนตระหนักถึงการทำกรรมธรรม์เพื่อคุ้มครองชีวิต เนื่องจากเหตุการณ์ในอนาคตเป็นสิ่งที่ไม่สามารถคาดคะเนถึงความแน่นอนได้ ซึ่งเหตุการณ์จากภัยธรรมชาติ อุบัติเหตุ โรคภัยไข้เจ็บส่วนบุคคลนั้นสามารถเกิดขึ้นได้อยู่ตลอดเวลา เมื่อผู้บริโภคได้ทำประกันภัยไว้และหากกรณีเหตุการณ์หรืออุบัติเหตุที่ไม่คาดคิดนั้นเกิดขึ้น บริษัทประกันภัยก็จะมาดำเนินการดูแล ชดใช้ค่าเสียหาย และชดใช้ค่าชดเชยต่าง ๆ ให้ ในวงเงินคุ้มครองที่ครอบคลุมตามที่ระบุในกรมธรรม์ ประกันชีวิตนั้นยังมีส่วนในการบรรเทาภาระค่าใช้จ่ายที่เกิดจากเหตุการณ์ที่ไม่ได้คาดการณ์ไว้ล่วงหน้าโดยไม่กระทบเงินออมส่วนบุคคล หรือกระทบเงินออมส่วนบุคคลน้อยที่สุด

ยอดขายประกันชีวิตจากสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย (คปภ., 2565) จัดเก็บข้อมูลยอดขายประกันชีวิตและผลิตภัณฑ์ประกันชีวิตจากบริษัทประกันชีวิต 15 บริษัท ซึ่งมีความเกี่ยวข้องกับงานวิจัยนี้มี 10 ประเภท ได้แก่ การประกันชีวิต

กรรมกรรมหลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไปสำหรับสามัญ การประกันชีวิตกรรมกรรมหลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไปสำหรับอุตสาหกรรม การประกันชีวิตกรรมกรรมหลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไปสำหรับกลุ่ม ผลิตภัณฑ์ประกันชีวิตแบบบำนาญ ผลิตภัณฑ์ประกันชีวิตแบบยูนิคัล ผลิตภัณฑ์ประกันชีวิตแบบยูนิเวอร์แซลไลฟ์ การประกันภัยอุบัติเหตุส่วนบุคคล สัญญาเพิ่มเติมอุบัติเหตุ สัญญาเพิ่มเติมสุขภาพ และสัญญาเพิ่มเติมอื่น ๆ

สำหรับงานวิจัยนี้ได้ศึกษาเกี่ยวกับยอดขายประกันภัย โดยทำการรวบรวมเก็บข้อมูลยอดขายประกันชีวิตจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย (คปภ., 2565) ตั้งแต่ปี พ.ศ. 2561 - 2565 (ค.ศ. 2018 - 2022) ซึ่งการพยากรณ์ยอดขายสำหรับประกันชีวิตใช้วิธีการพยากรณ์ 4 วิธี ได้แก่ การบวกโฮลท์-วินเทอร์ (Holt Winters' Additive) การคูณโฮลท์-วินเทอร์ (Holt Winters' Multiplicative) ปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (Simple Exponential Smoothing) และปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า (Double Exponential Smoothing) และทำการคำนวณจากภาษาโปรแกรม Python และนำข้อมูลที่ได้จากการคำนวณมาสร้างกราฟ ซึ่งวิธีการพยากรณ์เหล่านี้จะนำไปพยากรณ์แนวโน้มของยอดขายรับประกันภัยในอนาคตในอีก 1 ปีข้างหน้า ซึ่งผลการพยากรณ์จะสามารถทำให้ผู้ที่ทำกรรมกรรมเพื่อคุ้มครองชีวิตได้รู้ถึงแนวโน้มของยอดขายรับประกันภัยแต่ละประเภทซึ่งมีผลต่อการซื้อประกันภัย และผลการพยากรณ์สำหรับเจ้าของบริษัทประกันชีวิตจะสามารถนำมากำหนดกลยุทธ์ทางการตลาดและเพิ่มประสิทธิภาพในการจำหน่ายผลิตภัณฑ์ประกันภัยในอนาคตได้ ซึ่งการพยากรณ์ทั้ง 4 วิธีสำหรับงานวิจัยนี้เป็นวิธีที่เหมาะสมสำหรับข้อมูลยอดขายประกันชีวิตที่แตกต่างกันออกไปดังนี้

วิธีที่ 1 คือ การพยากรณ์โดยวิธีการบวกโฮลท์-วินเทอร์ ที่สามารถคำนวณข้อมูลในอดีตโดยอาศัยวิธีการทางคณิตศาสตร์ในการคำนวณข้อมูลหรือสถิติ และวิเคราะห์อนุกรมเวลาแบบบวก ที่มีแนวโน้มและฤดูกาล ระบุในแต่ละช่วงหรือฤดูกาลมาพยากรณ์ข้อมูลในอนาคต (เรวัฒน์ ไทยทอง, 2560, น. 1449-1458; Pleños, 2022, pp. 17-29; Nurhamidah, Wan, & Faisol, 2020, pp. 151-157; Suwanvijit, Lumley, Choonpradub, & McNeil, 2011, pp. 87-102; Sidqi & Sumitra, 2019, pp. 1-5) วิธีนี้เหมาะที่สุดสำหรับข้อมูลที่มีแนวโน้มและฤดูกาลที่ไม่เพิ่มขึ้นเมื่อเวลาผ่านไป ผลลัพธ์ที่ได้เป็นการคาดการณ์แบบโค้งซึ่งแสดงการเปลี่ยนแปลงตามฤดูกาลของข้อมูล

วิธีที่ 2 คือ การพยากรณ์ด้วยวิธีโฮลท์-วินเทอร์ เป็นการพยากรณ์ข้อมูลในอดีตโดยอาศัยหลักคณิตศาสตร์ในการคำนวณสถิติและวิเคราะห์ข้อมูลอนุกรมเวลา โดยใช้วิธีการที่เรียกว่าการคูณ

ที่มีแนวโน้มและฤดูกาล ซึ่งการวิเคราะห์แนวโน้มและฤดูกาลที่มีประสิทธิภาพ จะทำให้ธุรกิจ ประกันภัยมองเห็นภาพการเกิดขึ้นของแนวโน้มการซื้อประกันภัยจากผู้บริโภค (เรวัฒน์ ไทยทอง, 2560, น. 1449-1458; Swapnarekha, Behera, Nayak, Naik, & Kumar, 2021, pp. 1-15; Suwanvijit et al., 2011; Sidqi & Sumitra, 2019, pp. 1-5) วิธีนี้เหมาะที่สุดสำหรับข้อมูลที่มีแนวโน้มและฤดูกาลที่ เพิ่มขึ้นเมื่อเวลาผ่านไป ผลลัพธ์ที่ได้เป็นการคาดการณ์แบบโค้งซึ่งจำลองการเปลี่ยนแปลงตาม ฤดูกาลของข้อมูล

วิธีที่ 3 คือ การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย เป็นวิธีการคาดการณ์ ความต้องการในช่วงเวลาหนึ่ง ๆ ซึ่งประเมินว่าความต้องการซื้อประกันภัยจะเท่ากับค่าเฉลี่ยของ การบริโภคในอดีตจากช่วงเวลาที่กำหนด กระบวนการนี้จะเป็นไปอย่างต่อเนื่องโดยที่ใช้ข้อมูลใน อดีตเพื่อคำนวณสิ่งที่คาดหวังว่าความต้องการขายสินค้าประเภทประกันชีวิตจะเป็นอย่างไร (ธีระพงษ์ ทับพร, ยอดคนภา เกษเมือง, เอกพล ทับพร, และภชรดิษฐ์ แสงจิตต์, 2561, น. 28-40; Rahman, Salma, Hossain, & Khan, 2016, pp. 19-25; Ostertagová & Ostertag, 2012, pp. 62-66)

วิธีที่ 4 คือ การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลแบบสองเท่า ซึ่งจะคล้ายกับการ พยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย สิ่งที่แตกต่างกันคือ วิธีปรับเรียบเอ็กซ์โพเนนเชียลแบบสองเท่ามีองค์ประกอบในการรับเทรนด์ และคำนึงถึงแนวโน้มของข้อมูลยอดขาย ประกันภัยในอนาคตที่จะเพิ่มขึ้นหรือลดลงเมื่อเวลาผ่านไปโดยไม่เกิดซ้ำ ทำได้โดยใช้สมการ 2 สมการที่มีค่าคงที่ 2 ตัว (ธีระพงษ์ ทับพร และคณะ, 2561, น. 28-40; Rahman et al., 2016; Ostertagová & Ostertag, 2012, pp. 62-66) เหตุผลในการเลือกวิธีการพยากรณ์ 4 วิธี สำหรับยอดขาย ประกันชีวิตโดยไม่เลือกเพียง 1 วิธีสำหรับการพยากรณ์คือ วิธีการพยากรณ์ทั้ง 4 วิธีมีความ เหมาะสมที่จะนำมาพยากรณ์ข้อมูลยอดขายรับประกันชีวิต ซึ่งงานวิจัยนี้จะทำการคำนวณค่า พยากรณ์จากทั้งหมด 4 วิธี และเลือกวิธีการพยากรณ์ที่เหมาะสมที่สุดสำหรับยอดขายรับประกัน ชีวิต

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 เพื่อทำการวิเคราะห์การพยากรณ์สำหรับยอดขายประกันชีวิตสำหรับอนาคตจาก เว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย (คปภ., 2565) โดย 4 วิธีการพยากรณ์ ได้แก่ วิธีการบวกโฮลท์-วินเทอร์ วิธีการคูณโฮลท์-วินเทอร์ วิธีปรับเรียบ

เอ็กซ์โพเนนเชียลอย่างง่าย และวิธีปรับเรียบเอ็กซ์โพเนนเชียลแบบสองเท่า ซึ่งผลการวิเคราะห์จะเป็นขอยอดขายประกันชีวิตในอนาคต ผลการวิเคราะห์สามารถนำไปเป็นข้อมูลประกอบสำหรับการตัดสินใจในการทำธุรกรรมเพื่อคุ้มครองชีวิต และเจ้าของบริษัทประกันชีวิตจะสามารถสร้างกลยุทธ์ทางการตลาดเพื่อเพิ่มประสิทธิภาพในการจำหน่ายผลิตภัณฑ์ประกันภัยในอนาคตได้

1.2.2 เพื่อศึกษาวิธีการพยากรณ์ที่เหมาะสมที่สุดสำหรับขอยอดขายประกันชีวิตจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย จากการคำนวณค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) ค่าผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย (Mean Absolute Percentage Error, MAPE) และค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE)

1.2.3 เพื่อเลือกวิธีการพยากรณ์สำหรับขอยอดขายประกันชีวิตจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย เพื่อทำการพยากรณ์ขอยอดขายประกันที่แม่นยำในปีต่อไป

1.3 สมมติฐานการวิจัย

1.3.1 ทำการจัดเก็บข้อมูลขอยอดขายประกันจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย ตั้งแต่ปี พ.ศ. 2561 - 2565 (ค.ศ. 2018 - 2022) เพื่อมาวิเคราะห์ขอยอดขายประกัน ทำให้สามารถทราบได้ว่าแนวโน้มสำหรับขอยอดขายประกันจากในอีก 1 ปีข้างหน้าจะเพิ่มมากขึ้นหรือลดลง

1.3.2 คำนวณการพยากรณ์สำหรับข้อมูลขอยอดขายประกัน โดย 4 วิธี ได้แก่ การบวกโฮลท์-วินเทอร์ การคูณโฮลท์-วินเทอร์ ปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย และวิธีปรับเรียบเอ็กซ์โพเนนเชียลแบบสองเท่า ซึ่งการพยากรณ์ทั้ง 4 วิธีนี้ สามารถนำไปพยากรณ์ข้อมูลขอยอดขายประกันในอนาคตในอีก 1 ปีข้างหน้า ซึ่งผู้ที่ทำประกันหรือกำลังตัดสินใจทำประกันก็สามารถเลือกประเภทของประกันที่เหมาะสมและมีแนวโน้มข้อมูลรับขอยอดขายประกันที่ดี และเจ้าของบริษัทประกันชีวิตสามารถนำผลการพยากรณ์สำหรับข้อมูลขอยอดขายประกันไปสร้างและกำหนดกลยุทธ์ทางการตลาดได้

1.3.3 ทำการวิเคราะห์เพื่อเลือกวิธีที่ดีที่สุดสำหรับการพยากรณ์ข้อมูลขอยอดขายประกันจากการหาค่าพยากรณ์ด้วยวิธีค่าเฉลี่ยความผิดพลาดสัมบูรณ์ ค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ และค่าเฉลี่ยความผิดพลาดกำลังสอง โดยใช้ภาษาโปรแกรม Python การวิเคราะห์นี้ทำให้สามารถประเมินความแม่นยำของแต่ละวิธีการพยากรณ์และคัดเลือกวิธีที่ดีที่สุด

1.4 ขอบเขตของการศึกษา

1.4.1 พยากรณ์ข้อมูลยอดขายประกันจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย เป็นเวลา 5 ปี โดยใช้การคำนวณทางคณิตศาสตร์และวิศวกรรมศาสตร์ 4 วิธี ได้แก่ วิธีการบวกโสมท์-วินเทอร์ วิธีการคูณโสมท์-วินเทอร์ วิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย และวิธีปรับเรียบเอ็กซ์โพเนนเชียลแบบสองเท่า โดยใช้ข้อมูลยอดขายประกัน 4 ปี สำหรับพยากรณ์ข้อมูลยอดขายประกันในอีก 1 ปีข้างหน้า

1.4.2 ทำการหาค่าการพยากรณ์จากค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย ค่าผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย และค่าผิดพลาดกำลังสองเฉลี่ย เพื่อนำค่าการพยากรณ์ที่ได้มาเปรียบกับการพยากรณ์แต่ละวิธี ซึ่งค่าความเคลื่อนที่น้อยที่สุด จะเป็นวิธีที่เหมาะสมที่สุดสำหรับการพยากรณ์ข้อมูลยอดขายประกัน

1.4.3 การพยากรณ์ข้อมูลยอดขายประกันโดย 4 วิธี ได้แก่ วิธีการบวกโสมท์-วินเทอร์ วิธีการคูณโสมท์-วินเทอร์ วิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย และวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า และการหาค่าการพยากรณ์จากค่าเฉลี่ยความผิดพลาดสัมบูรณ์ ค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ และค่าเฉลี่ยความผิดพลาดกำลังสอง ทำการคำนวณโดยใช้ภาษาโปรแกรม Python และเปรียบเทียบผลระหว่างค่าการพยากรณ์ที่ได้และข้อมูลยอดขายประกันจริง

1.5 ประโยชน์ที่จะได้รับการศึกษา

1.5.1 ได้จัดเก็บข้อมูลยอดขายประกันจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย จากข้อจริง ซึ่งข้อมูลเป็นยอดขายประกันจากประกันชีวิตแต่ละแบบที่มีแนวโน้มมากและน้อยตามความเป็นจริงจากลูกค้าที่ซื้อผลิตภัณฑ์ประกันชีวิต

1.5.2 ได้รับข้อมูลการวิเคราะห์จากการพยากรณ์ข้อมูลยอดขายประกันทำให้สามารถพยากรณ์แนวโน้มยอดขายประกันชีวิตแต่ละแบบว่ามีแนวโน้มเพิ่มขึ้นหรือลดลงในอนาคต เพื่อเป็นประโยชน์ต่อการตัดสินใจของผู้ซื้อ และผู้ขายในการวางแผนการตลาด

1.6 แนวทางการดำเนินงาน

1.6.1 จัดเก็บและรวบรวมข้อมูลยอดขายประกันจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย เป็นเวลา 5 ปีให้ครบทุกไตรมาส ข้อมูลยอดขาย

ตารางที่ 1.1 แผนการดำเนินงาน (ต่อ)

ลำดับ	กิจกรรม	ช่วงเวลาการดำเนินงาน (2566 - 2567)											
		พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.
7	แก้ไขเล่มวิทยานิพนธ์ที่ได้รับการตรวจจากอาจารย์ที่ปรึกษา												
8	เตรียมเล่มวิทยานิพนธ์ที่สมบูรณ์ เตรียมสไลด์สำหรับการสอบปิดเล่ม และนัดหมายคณะกรรมการเพื่อสอบปิดเล่ม												
9	สอบปิดเล่ม วิทยานิพนธ์ แก้ไขค้าง คณะกรรมการได้ทำการตรวจและส่งเล่ม วิทยานิพนธ์												

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 นิยามและความหมายคำศัพท์ในการพยากรณ์ยอดขายประกันชีวิต

การพยากรณ์ คือการคาดการณ์โดยศึกษาจากข้อมูลเก่าในอดีต และนำระบบซอฟต์แวร์มาเพื่อทำการวิเคราะห์ข้อมูลจำนวนมากที่เก็บรวบรวมมาเป็นระยะเวลานาน จากนั้นซอฟต์แวร์จะคาดการณ์ความต้องการและแนวโน้มในอนาคต

การพยากรณ์โดยวิธีการบวกโฮลท์-วินเทอร์ (Holt Winters' Additive) (Pleños, 2022, pp. 17-29; Rahman et al., 2016) เป็นส่วนขยายของการปรับให้เรียบแบบเอกซ์โพเนนเชียลของ Holt ที่ให้ความสำคัญกับข้อมูลของฤดูกาล วิธีการนี้จะสร้างค่าที่ปรับให้เรียบแบบเอกซ์โพเนนเชียลสำหรับระดับการคาดการณ์ แนวโน้มของการคาดการณ์ และการปรับเปลี่ยนตามฤดูกาลของการคาดการณ์ วิธีการบวกตามฤดูกาลนี้จะเพิ่มปัจจัยตามฤดูกาลให้กับการพยากรณ์ตามแนวโน้ม ทำให้เกิดการคาดการณ์เพิ่มเติมของ Holt-Winters วิธีนี้เหมาะที่สุดสำหรับข้อมูลที่มีแนวโน้มและฤดูกาลที่ไม่เพิ่มขึ้นเมื่อเวลาผ่านไป ผลลัพธ์ที่ได้คือการคาดการณ์แบบโค้งซึ่งแสดงการเปลี่ยนแปลงตามฤดูกาลของข้อมูล การพยากรณ์โดยวิธีการคูณโฮลท์-วินเทอร์จึงสามารถสะท้อนลักษณะการเปลี่ยนแปลงของข้อมูลที่มีรูปแบบตามสัดส่วน (Proportional Seasonality) ได้อย่างเหมาะสม โดยเฉพาะในกรณีที่ฤดูกาลมีการเปลี่ยนแปลงสัมพันธ์กับระดับของข้อมูล

การพยากรณ์โดยวิธีการคูณโฮลท์-วินเทอร์ (Holt Winters' Multiplicative) (Pleños, 2022, pp. 17-29; Rahman et al., 2016) การพยากรณ์นี้คล้ายกับวิธีการพยากรณ์ของการบวกโฮลท์-วินเทอร์ ซึ่งการพยากรณ์โดยวิธีการคูณโฮลท์-วินเทอร์นี้ยังคำนวณค่าที่ปรับให้เรียบแบบเอกซ์โพเนนเชียลสำหรับระดับ แนวโน้ม และการปรับเปลี่ยนตามฤดูกาลในการคาดการณ์ วิธีการคูณตามฤดูกาลนี้จะคูณการคาดการณ์แนวโน้มตามฤดูกาล ทำให้เกิดการคาดการณ์แบบทวีคูณของโฮลท์-วินเทอร์ วิธีนี้เหมาะที่สุดสำหรับข้อมูลที่มีแนวโน้มและฤดูกาลที่เพิ่มขึ้นเมื่อเวลาผ่านไป ผลลัพธ์ที่ได้คือการคาดการณ์แบบโค้งซึ่งจำลองการเปลี่ยนแปลงตามฤดูกาลของข้อมูล

การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (Simple Exponential Smoothing) (Ostertagová & Ostertag, 2012, pp. 62-66) คือ เป็นวิธีการพยากรณ์ข้อมูลแบบอนุกรมเวลาที่ใช้กับข้อมูลที่ไม่มีแนวโน้ม (Trend) หรือฤดูกาล (Seasonality) และมีการเปลี่ยนแปลงในระดับคงที่ (Stationary Level) เท่านั้น วิธีนี้ถือว่าเป็นวิธีพื้นฐานของการพยากรณ์เชิงปริมาณ โดยการทำให้ข้อมูลในอดีตมีการให้น้ำหนักโดยการลดทอนเชิงเอ็กซ์โพเนนเชียล

การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า (Double Exponential Smoothing) (Sidqi & Sumitra, 2019, pp. 1-5) เป็นวิธีการพยากรณ์แบบอนุกรมเวลาที่ใช้กับข้อมูลที่มีแนวโน้ม (Trend) แต่ไม่มีฤดูกาล (Seasonality) วิธีนี้เป็นการต่อยอดจาก Simple Exponential Smoothing โดยเพิ่มการคำนวณส่วนของแนวโน้มเข้าไปในโมเดล

ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) (Sidqi & Sumitra, 2019, pp. 1-5) เป็นวิธีการวัดความแม่นยำ โดยการแก้ปัญหาค่าเฉลี่ยของความผิดพลาด ซึ่งจะพิจารณาความแตกต่างค่าจริงกับค่าพยากรณ์

ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย (Mean Absolute Percentage Error, MAPE) (Ostertagová & Ostertag, 2012, p. 63) หมายถึงข้อผิดพลาดร้อยละสัมบูรณ์ อีกทั้งยังเป็นที่รู้จักกันเป็นค่าเฉลี่ยร้อยละส่วนเบี่ยงเบนสัมบูรณ์ ที่เป็นตัวชี้วัดของความถูกต้องทำนายของวิธีการพยากรณ์ในสถิติ มักจะแสดงความแม่นยำเป็นอัตราส่วน

ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) (Ostertagová & Ostertag, 2012, p. 63) จะมีการทำให้ค่าความผิดพลาด กลายเป็นบวกก่อนโดยการนำค่าความผิดพลาด มายกกำลังสอง ก่อนที่จะนำค่าความผิดพลาด มาหาค่าเฉลี่ย

2.2 ทฤษฎีการพยากรณ์ยอดขายประกันชีวิต

การพยากรณ์คือการคาดการณ์โดยศึกษาจากข้อมูลเก่าและรูปแบบต่าง ๆ ในอดีต งานวิจัยนี้ใช้ระบบซอฟต์แวร์ด้วยภาษาโปรแกรม Python เพื่อวิเคราะห์ข้อมูลยอดขายประกันจำนวนมากที่เก็บรวบรวมมาเป็นระยะเวลานานจำนวน 4 ปี (ค.ศ. 2018 - 2021) จากนั้นซอฟต์แวร์จะคาดการณ์

ความต้องการและแนวโน้มในอนาคตสำหรับปีที่ 5 (ค.ศ. 2022) เพื่อเป็นข้อมูลประกอบการตัดสินใจด้านการเงิน การตลาด และการดำเนินงานของบริษัทประกันภัยต่าง ๆ อย่างแม่นยำยิ่งขึ้น

2.2.1 การพยากรณ์โดยวิธีการบวกโฮสต์-วินเทอร์

การบวกโฮสต์-วินเทอร์ (Pleños, 2022, pp. 17-29; Rahman et al., 2016) เป็นวิธีการพยากรณ์ข้อมูลที่มีทั้ง แนวโน้ม (Trend) และ ฤดูกาล (Seasonality) โดยรูปแบบนี้จะใช้ในกรณีที่ผลกระทบของฤดูกาลมีลักษณะแบบ เพิ่มขึ้นหรือลดลงคงที่ (Additive) หมายความว่า ฤดูกาลส่งผลต่อข้อมูลโดยการเพิ่มหรือลดค่าข้อมูลโดยตรง โมเดลนี้เป็นการขยายจาก โฮสต์-วินเทอร์ แบบพื้นฐาน ที่มีการรวมองค์ประกอบ 3 ส่วน คือ ระดับพื้นฐาน (Level) แนวโน้ม (Trend) และฤดูกาล (Seasonality) เข้าด้วยกัน ส่วนประกอบของการบวกโฮสต์-วินเทอร์ ระดับพื้นฐาน (Level) ค่าข้อมูลที่ถูกปรับแล้ว ซึ่งรวมถึงฤดูกาลและแนวโน้ม แนวโน้ม (Trend) การเปลี่ยนแปลงของข้อมูลเมื่อเวลาผ่านไป เช่น ข้อมูลเพิ่มขึ้นหรือลดลงอย่างต่อเนื่อง ฤดูกาล (Seasonality) รูปแบบการเปลี่ยนแปลงที่เกิดขึ้นซ้ำตามช่วงเวลา (เช่น ฤดูกาลของปีที่มีการเปลี่ยนแปลงซ้ำกันทุกปี) มีสมการดังนี้

$$\hat{Y}_{t+n} = E_t + nT_t + S_{t+n+p} \quad (2-1)$$

$$\text{Level } E_t = \alpha(Y_t - S_{t-p}) + (1 - \alpha)(E_{t-1} + T_{t-1})$$

$$\text{Trend } T_t = \beta(E_t - E_{t-1}) + (1 - \beta)T_{t-1}$$

$$\text{Seasonal } S_t = \gamma(Y_t - E_t) + (1 - \gamma)S_{t-p}$$

โดยที่

$\hat{Y}_{t+n}$: ค่าพยากรณ์ ณ เวลา $t+n$ (เช่น หาก t คือปัจจุบัน และ n คือจำนวนช่วงเวลาในอนาคตที่ต้องการคาดการณ์)

Y_t : ค่าพยากรณ์ ณ เวลา t

E_t : ระดับคาดหวังของอนุกรมเวลาในช่วงเวลา t ซึ่งสะท้อนค่าที่ปรับแล้วจากข้อมูลที่มีความผันผวนหรือแนวโน้มอื่น ๆ

E_{t-1} : คาดหวังระดับของอนุกรมเวลาในช่วงเวลา $t-1$

T_t : ค่าแนวโน้ม ณ เวลา t

T_{t-1} : ค่าแนวโน้ม ณ เวลา $t-1$

S_t : ปัจจัยตามฤดูกาลในช่วง t

- S_{t-p} : ปัจจัยตามฤดูกาลในช่วง $t-p$
- S_{t+n+p} : ส่วนของฤดูกาล (Seasonality) ซึ่งสะท้อนถึงรูปแบบที่เกิดขึ้นซ้ำตามฤดูกาลในอนาคต โดย S คือค่าของฤดูกาลที่เวลา $t+n+p$ ซึ่ง p อาจใช้ปรับตามรูปแบบของฤดูกาล (เช่น ถ้าเป็นฤดูกาลที่หมุนเวียนทุกเดือนหรือทุกปี)
- n : ระยะเวลาที่จะหาในอนาคตเมื่อเทียบกับข้อมูลล่าสุดที่มีอยู่
- p : จำนวนงวดที่พิจารณา (ไตรมาส)
- α : ค่าคงที่ปรับเรียบ ($0 \leq \alpha \leq 1$)
- β : ค่าคงที่ปรับเรียบ ($0 \leq \beta \leq 1$)
- γ : ค่าคงที่ปรับเรียบ ($0 \leq \gamma \leq 1$)

โมเดลการบวกโฮลท์-วินเทอร์ เหมาะสำหรับข้อมูลที่มีแนวโน้มและฤดูกาลที่คงที่ (Additive) เช่น การพยากรณ์ยอดขายสินค้า ที่มีการเติบโตและความผันผวนตามฤดูกาล การคาดการณ์ปริมาณการใช้พลังงาน ที่มีการเติบโตพร้อมกับการเปลี่ยนแปลงตามฤดูกาล เช่น ความต้องการพลังงานในหน้าร้อนและหน้าหนาว

การบวกโฮลท์-วินเทอร์ เป็นวิธีการพยากรณ์ข้อมูลที่มีทั้งแนวโน้มและฤดูกาล โดยรูปแบบนี้ใช้ในกรณีที่ฤดูกาลมีผลกระทบที่เพิ่มขึ้นหรือลดลงอย่างคงที่ เป็นวิธีที่ใช้กันอย่างแพร่หลายในการพยากรณ์ข้อมูลธุรกิจหรือเศรษฐกิจที่มีลักษณะตามฤดูกาล

2.2.2 การพยากรณ์โดยวิธีการคูณโฮลท์-วินเทอร์

การคูณโฮลท์-วินเทอร์ (Pleños, 2022, pp. 17-29; Rahman et al., 2016) เป็นวิธีการพยากรณ์ข้อมูลที่มีทั้งแนวโน้ม (Trend) และ ฤดูกาล (Seasonality) เช่นเดียวกับโฮลท์-วินเทอร์แบบแอดดิทีฟ แต่แตกต่างกันตรงที่รูปแบบฤดูกาลในกรณีนี้จะเป็นลักษณะ การคูณ (Multiplicative) นั่นหมายความว่า ผลกระทบของฤดูกาลจะมีการปรับตามสัดส่วนของระดับข้อมูล ไม่ได้เพิ่มขึ้นหรือลดลงแบบคงที่ โมเดลนี้ถูกใช้ในกรณีที่ข้อมูลมีการเปลี่ยนแปลงทั้งแนวโน้มและฤดูกาล แต่ผลของฤดูกาลไม่ได้คงที่ตามจำนวน แต่จะเพิ่มหรือลดตามระดับข้อมูลในช่วงเวลานั้น เช่น ฤดูกาลอาจส่งผลให้ยอดขายเพิ่มขึ้น 20% ทุกปี ไม่ใช่เพิ่มขึ้นเป็นจำนวนคงที่ ส่วนประกอบของการคูณโฮลท์-วินเทอร์ ได้แก่ระดับพื้นฐาน (Level) ค่าข้อมูลที่ถูกปรับแล้วรวมถึงฤดูกาลและแนวโน้ม ถัดไปเป็นแนวโน้ม (Trend) การเปลี่ยนแปลงของข้อมูลเมื่อเวลาผ่านไป เช่น ข้อมูลที่เพิ่มขึ้น

หรือลดลงอย่างต่อเนื่อง และฤดูกาล (Seasonality) ซึ่งเป็นรูปแบบการเปลี่ยนแปลงซ้ำตามช่วงเวลาที่มีลักษณะเป็นการคูณ ปรับตามระดับข้อมูลในเวลานั้น ๆ มีสมการดังนี้

$$\hat{Y}_{t+n} = (E_t + nT_t)S_{t+n+p} \quad (2-2)$$

$$\text{Level } E_t = \alpha(Y_t/S_{t-p}) + (1 - \alpha)(E_{t-1} + T_{t-1})$$

$$\text{Trend } T_t = \beta(E_t - E_{t-1}) + (1 - \beta)T_{t-1}$$

$$\text{Seasonal } S_t = \gamma(Y_t/E_t) + (1 - \gamma)S_{t-p}$$

โดยที่

$\hat{Y}_{t+n}$: ค่าพยากรณ์ ณ เวลา $t+n$ (การพยากรณ์อนาคตหลังจากเวลา t ไปอีก n ช่วงเวลา)

E_t : ระดับคาดหวังของอนุกรมเวลาในช่วงเวลา t ซึ่งแสดงถึงค่าของข้อมูลที่ไม่มีปัจจัยของแนวโน้มหรือฤดูกาลเข้ามาเกี่ยวข้อง

nT_t : ส่วนของแนวโน้ม (Trend) ซึ่งคำนวณจากค่าสัมประสิทธิ์ของแนวโน้ม ณ เวลา t (T_t) คูณด้วยจำนวนช่วงเวลาที่คาดการณ์ (n) แสดงถึงการเปลี่ยนแปลงของข้อมูลเนื่องจากแนวโน้มเมื่อคาดการณ์ไปในอนาคต

T_t : ค่าแนวโน้มเมื่อเวลาผ่านไป t

S_t : ปัจจัยตามฤดูกาลในช่วง t

S_{t+n+p} : ส่วนของฤดูกาล (Seasonality) ซึ่งเป็นค่าสัมประสิทธิ์ของฤดูกาลในช่วงเวลาที่ $t+n+p$ โดย p คือ การปรับค่าฤดูกาลเพื่อให้ตรงกับลักษณะรูปแบบของฤดูกาล (เช่น ถ้ารูปแบบฤดูกาลเกิดขึ้นทุกปี)

α : ค่าคงที่ปรับเรียบ ($0 \leq \alpha \leq 1$)

β : ค่าคงที่ปรับเรียบ ($0 \leq \beta \leq 1$)

γ : ค่าคงที่ปรับเรียบ ($0 \leq \gamma \leq 1$)

การบวกโฮลท์-วินเทอร์ ผลกระทบของฤดูกาลเป็นแบบคงที่ (เพิ่มหรือลดค่าข้อมูลในแต่ละช่วงเวลาตามฤดูกาลแบบคงที่ ในขณะที่โฮลท์-วินเทอร์แบบมัลติพลีเคทีฟ ฤดูกาลจะเป็นการคูณ ซึ่งหมายความว่าฤดูกาลจะส่งผลตามสัดส่วนของระดับข้อมูล เช่น ถ้าข้อมูลในปัจจุบันเพิ่มขึ้นมาก ผลของฤดูกาลจะยิ่งใหญ่ขึ้นตามไปด้วย ซึ่งการคูณโฮลท์-วินเทอร์ เหมาะสำหรับข้อมูลที่มีลักษณะฤดูกาลที่ผลกระทบเพิ่มขึ้นหรือลดลงตามขนาดของข้อมูล ตัวอย่างเช่น การพยากรณ์ยอดขายสินค้าที่มียอดขายเพิ่มขึ้นตามฤดูกาล และผลของฤดูกาลเพิ่มขึ้นตามระดับยอดขาย การคาดการณ์ความต้องการพลังงาน ที่มีความต้องการเพิ่มขึ้นตามแนวโน้มและฤดูกาล เช่น ฤดูกาลร้อน

ที่ทำให้มีการใช้ไฟฟ้าเพิ่มขึ้นมาก การคูณโวลต์-วินเทอร์ ใช้สำหรับการพยากรณ์ข้อมูลที่มีแนวโน้มและฤดูกาล โดยฤดูกาลจะส่งผลแบบการคูณตามสัดส่วนของข้อมูล วิธีนี้เหมาะสำหรับข้อมูลที่ฤดูกาลไม่ได้มีผลแบบคงที่ แต่จะส่งผลเพิ่มขึ้นหรือลดลงตามระดับของข้อมูล

2.2.3 การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

ปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (Ostertagová & Ostertag, 2012, pp. 62-66) เป็นวิธีการพยากรณ์สำหรับข้อมูลที่ไม่มีแนวโน้มหรือฤดูกาล เป็นวิธีการปรับค่าให้นุ่มนวลเพียงหนึ่งระดับ โดยอาศัยข้อมูลในอดีตทั้งหมดและให้น้ำหนักที่ลดลงตามเวลา วิธีนี้เหมาะสำหรับข้อมูลที่มีความแปรปรวนในระยะสั้นหรือข้อมูลที่ไม่ซับซ้อน เช่น ข้อมูลที่มีแนวโน้มคงที่ ปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายจะให้น้ำหนักกับค่าข้อมูลในอดีตที่ใกล้เคียงกับเวลาปัจจุบันมากกว่า โดยค่าที่ใกล้เคียงจะได้รับน้ำหนักสูงกว่า และค่าที่อยู่ห่างออกไปจะได้รับน้ำหนักน้อยลง มีสมการดังนี้

$$Y_t = Y_{t-1} + \alpha(Y_{t-1} - Y_{t-1}^*) \quad (2-3)$$

$$Y_t = \alpha Y_{t-1} + (1 - \alpha) Y_{t-1}^*$$

โดยที่

Y_t : ค่าพยากรณ์ ณ เวลา t ซึ่งเป็นค่าของตัวแปรที่สนใจ (อาจเป็นราคา ปริมาณ หรือค่าที่ต้องการคาดการณ์) ณ เวลา t (ปัจจุบัน)

Y_{t-1} : ค่า ณ เวลา $t-1$ (ค่าของตัวแปรเดียวกันจากช่วงเวลาก่อนหน้า)

Y_{t-1}^* : ค่าพยากรณ์ ณ เวลา $t-1$ หรือค่าที่คาดหวังหรือตัวแปรเป้าหมาย (Target) ณ เวลา $t-1$ ซึ่งเป็นค่าที่ใช้เป็นเกณฑ์ในการปรับค่า

α : ค่าคงที่ปรับเรียบ ($0 \leq \alpha \leq 1$) ซึ่งเป็นค่าสัมประสิทธิ์การปรับ (Adjustment Factor) หรือที่เรียกว่าค่าเรียนรู้ (Learning Rate) ซึ่งเป็นค่าระหว่าง 0 และ 1 โดยที่

α ควบคุมความเร็วในการปรับค่า Y_t เข้าหาค่าเป้าหมาย Y_{t-1}^*

ปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายเหมาะกับข้อมูลที่ไม่มีแนวโน้มหรือฤดูกาล เช่น การพยากรณ์ปริมาณสินค้าในคลัง สำหรับสินค้าไม่มีการเปลี่ยนแปลงชัดเจน การพยากรณ์จำนวนผู้ให้บริการ ที่มีความคงที่ในระยะเวลานาน การคาดการณ์อุปสงค์ ของสินค้าอุปโภคบริโภคที่มีความต้องการคงที่ โดยที่ข้อดี ได้แก่ สะดวกต่อการใช้งานและคำนวณ และเหมาะกับข้อมูลที่ไม่ซับซ้อนและไม่มีแนวโน้มหรือฤดูกาล ในส่วนของข้อจำกัดนั้นจะไม่สามารถใช้กับข้อมูลที่มีแนวโน้ม

หรือฤดูกาลได้ดี เนื่องจากการไม่มีการพิจารณาการเปลี่ยนแปลงในระยะยาว ซึ่งปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย เป็นวิธีการพยากรณ์ที่ง่ายและรวดเร็ว เหมาะสำหรับข้อมูลที่ไม่มีแนวโน้มหรือฤดูกาล การให้น้ำหนักที่ลดลงตามเวลา ส่งผลให้ค่าพยากรณ์มีความแม่นยำมากขึ้นในกรณีที่ข้อมูลในอดีตไม่มีการเปลี่ยนแปลงมากนัก ซึ่งเหมาะสำหรับสถานการณ์ที่ต้องการพยากรณ์ระยะสั้นและมีข้อมูลที่เสถียรต่อเนื่องในช่วงเวลาใกล้เคียง

2.2.4 การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า (Sidqi & Sumitra, 2019, pp. 1-5) หรือที่เรียกว่า โฮลท์ (Holt's Linear Trend Model) เป็นวิธีการพยากรณ์ที่ใช้สำหรับข้อมูลที่มี แนวโน้ม (Trend) แต่ไม่มีฤดูกาล (Seasonality) กล่าวคือ วิธีนี้เหมาะสำหรับข้อมูลที่มีการเปลี่ยนแปลงในระดับ (Level) และการเพิ่มขึ้นหรือลดลงตามแนวโน้ม แต่ไม่ซับซ้อนถึงขั้นมีลักษณะตามฤดูกาล วิธีนี้ประกอบด้วย การคำนวณสองขั้นตอน ได้แก่ ระดับพื้นฐาน (Level) และแนวโน้ม (Trend) ซึ่งเอื้อต่อการพยากรณ์ แนวโน้มข้อมูลที่เพิ่มขึ้นหรือลดลงได้อย่างมีประสิทธิภาพ ส่วนประกอบของปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ได้แก่ระดับพื้นฐาน (Level) ค่าข้อมูลที่ถูกปรับเพื่อให้แสดงค่าในปัจจุบันรวมถึงการเปลี่ยนแปลงของแนวโน้ม และแนวโน้ม (Trend) การเปลี่ยนแปลงของข้อมูลเมื่อเวลาผ่านไป เช่น การเติบโตหรือการลดลงของข้อมูลอย่างต่อเนื่อง มีสมการดังนี้

$$Y_{t+1} = L_t + (h)T_t \quad (2-4)$$

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

$$T_t = \beta(L_t - L_{t-1}) + (1 - \beta)T_{t-1}$$

โดยที่

Y_{t+1} : ค่า ณ เวลา $t+1$ ซึ่งเป็นค่าที่คาดการณ์หรือพยากรณ์สำหรับช่วงเวลาถัดไป (อนาคต)

L_t : ค่าระดับ ณ เวลา t ซึ่งสะท้อนค่าพื้นฐานของข้อมูลในช่วงเวลาปัจจุบัน (ข้อมูลที่ไม่ได้รับผลกระทบจากแนวโน้มหรือฤดูกาล)

L_{t-1} : ค่าระดับ ณ เวลา $t-1$

H : ขอบเขตการพยากรณ์ จำนวนช่วงเวลาที่ต้องการคาดการณ์ไปข้างหน้า (เช่น ถ้าต้องการพยากรณ์สำหรับช่วงเวลา $t+1$ ค่า h จะเท่ากับ 1 แต่ถ้าต้องการคาดการณ์สำหรับ $t+2$ แล้ว h จะเท่ากับ 2 เป็นต้น)

- T_t : ค่าสัมประสิทธิ์ของแนวโน้ม (Trend) ที่เวลา t ซึ่งแสดงถึงการเปลี่ยนแปลงของข้อมูลตามแนวโน้มที่เกิดขึ้น
- α : ค่าคงที่ปรับเรียบ ($0 \leq \alpha \leq 1$)
- β : ค่าคงที่ปรับเรียบ ($0 \leq \beta \leq 1$)

สมการปรับเรียบเอ็กซ์โพเนนเชียลสองเท่ามักใช้ในกรณีที่ข้อมูลมีแนวโน้มชัดเจน แต่ไม่มีลักษณะฤดูกาล ตัวอย่างเช่น การพยากรณ์ยอดขาย ของสินค้าในอนาคตที่มียอดขายเพิ่มขึ้นหรือลดลงตามแนวโน้ม การคาดการณ์อัตราการเติบโตของจำนวนผู้ใช้ บริการที่มีแนวโน้มการเพิ่มขึ้นหรือลดลง การพยากรณ์ราคา สินค้าหรือบริการที่มีการเคลื่อนไหวตามแนวโน้มในอดีต ดังนั้นปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าเป็นวิธีการที่มีประสิทธิภาพในการพยากรณ์ข้อมูลที่มีแนวโน้มชัดเจน โดยใช้ทั้งระดับพื้นฐานและแนวโน้มของข้อมูลเพื่อคาดการณ์อนาคต วิธีนี้เหมาะสมสำหรับข้อมูลที่ไม่ซับซ้อนเกินไป แต่ต้องการพยากรณ์ข้อมูลที่มีการเติบโตหรือลดลงอย่างต่อเนื่อง

2.3 การคำนวณค่าพยากรณ์

2.3.1 ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD)

ตัวชี้วัดที่ใช้วัดความแม่นยำของแบบจำลองการพยากรณ์ โดยการคำนวณค่าผิดพลาดของการพยากรณ์แต่ละจุดข้อมูล และนำมาหาค่าเฉลี่ย ความผิดพลาดสัมบูรณ์จะไม่คำนึงถึงทิศทางของความผิดพลาด (ไม่ว่าจะเกินหรือลดต่ำกว่าค่าจริง) ดังนั้นค่าผิดพลาดที่เป็นบวกหรือเป็นลบจะถูกแปลงเป็นค่าบวกทั้งหมด ทำให้ค่า MAD เป็นการวัดที่สะท้อนความแตกต่างระหว่างค่าที่พยากรณ์กับค่าจริงในรูปแบบที่ไม่เกี่ยวกับทิศทางของข้อผิดพลาด มีสมการดังนี้

$$MAD = \frac{\sum |A_t - F_t|}{n} \quad (2-5)$$

ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ยเป็นการวัดความผิดพลาดสัมบูรณ์ระหว่างค่าพยากรณ์และค่าจริง โดยให้ความสำคัญกับขนาดของข้อผิดพลาดโดยไม่สนใจทิศทาง การเบี่ยงเบน (บวกหรือลบ) ใช้เพื่อประเมินความแม่นยำของแบบจำลองการพยากรณ์และเปรียบเทียบผลลัพธ์จากโมเดลต่าง ๆ โดยการคำนวณค่าเบี่ยงเบนสัมบูรณ์เฉลี่ยจะทำให้สามารถประเมินประสิทธิภาพของโมเดลได้อย่างชัดเจน โดยไม่ถูกผลกระทบจากการเบี่ยงเบนในทิศทางต่าง ๆ การใช้ค่านี้เหมาะสำหรับการวัดผลในกรณีที่ต้องการให้ความสำคัญกับความผิดพลาดที่มีขนาดใหญ่

2.3.2 ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย (Mean Absolute Percentage Error, MAPE)

ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย (Mean Absolute Percentage Error, MAPE) เป็นเครื่องมือที่ใช้วัดความแม่นยำของแบบจำลองการพยากรณ์ โดยจะคำนวณค่าผิดพลาดในรูปแบบของเปอร์เซ็นต์ระหว่าง ค่าที่พยากรณ์และค่าจริง แล้วนำมาหาค่าเฉลี่ย เพื่อให้ได้ผลลัพธ์ที่สามารถเปรียบเทียบได้ในเชิงเปอร์เซ็นต์ มีสมการดังนี้

$$MAPE = \frac{\sum \left(\frac{A_t - F_t}{A_t} \right)}{n} \times 100\% \quad (2-6)$$

ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ยใช้ในการวัดความผิดพลาดของโมเดลการพยากรณ์ในรูปแบบเปอร์เซ็นต์ เป็นเครื่องมือที่นิยมใช้เพื่อประเมินความแม่นยำของแบบจำลอง โดยคำนึงถึงความผิดพลาดในแต่ละจุดข้อมูลและทำให้เป็นเปอร์เซ็นต์ ซึ่งทำให้สามารถเปรียบเทียบโมเดลต่าง ๆ ได้

2.3.3 ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE)

ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) คือ เครื่องมือที่ใช้วัดความแม่นยำของโมเดลการพยากรณ์ โดยการคำนวณค่าผิดพลาด (Error) ระหว่างค่าที่พยากรณ์และค่าจริง แล้วนำค่าผิดพลาดที่ได้มาทำการยกกำลังสองและหาค่าเฉลี่ย ค่า MSE จะให้ความสำคัญกับค่าผิดพลาดที่ใหญ่ (ความผิดพลาดที่มาก) เนื่องจากการยกกำลังสองทำให้ค่าผิดพลาดที่มีขนาดใหญ่มีผลมากกว่าค่าผิดพลาดที่เล็ก มีสมการดังนี้

$$MSE = \frac{\sum (A_t - F_t)^2}{n} \quad (2-7)$$

ค่าความผิดพลาดกำลังสองเฉลี่ย เป็นเครื่องมือที่ใช้วัดความผิดพลาดระหว่างค่าที่พยากรณ์กับค่าจริงในรูปแบบของค่าผิดพลาดกำลังสอง โดยมุ่งเน้นให้ความสำคัญกับค่าผิดพลาดที่ใหญ่ การใช้ค่าเฉลี่ยความผิดพลาดกำลังสองเพื่อการประเมินความแม่นยำของโมเดลการพยากรณ์ และสามารถเปรียบเทียบประสิทธิภาพของโมเดลต่าง ๆ ได้ โดยจะทำให้เห็นผลกระทบของความผิดพลาดที่มีขนาดใหญ่ต่อความแม่นยำของการพยากรณ์อย่างชัดเจน ซึ่งเป็นการเน้นย้ำจุดอ่อนของโมเดลในการจัดการกับข้อผิดพลาดที่สำคัญ

2.4 ข้อมูลยอดขายประกันภัยที่เกี่ยวข้องกับงานวิจัย

ข้อมูลยอดขายประกันจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย (คปภ., 2565) ตั้งแต่ปี พ.ศ. 2561 - 2565 (ค.ศ. 2018 - 2022) ซึ่งข้อมูลยอดขายประกันตั้งแต่ปี พ.ศ. 2561 - 2564 (ค.ศ. 2018 - 2021) เป็นข้อมูลที่นำมาเพื่อทำการพยากรณ์และเปรียบเทียบค่าที่ได้จากการพยากรณ์กับข้อมูลยอดขายประกัน พ.ศ. 2565 (ค.ศ. 2022) เพื่อทำการวิเคราะห์หว่ายอดขายประกันจากการพยากรณ์มีความตรงหรือใกล้เคียงกับค่าจริงมากเพียงใด โดยมูลค่ากรมธรรม์ได้ถูกแบ่งออกเป็น 4 รูปแบบ คือ

- 1) ยอดขายประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป
- 2) ยอดขายประกันชีวิตกรมธรรม์หลัก
- 3) ยอดขายประกันชีวิตสัญญาเพิ่มเติมจากกรมธรรม์หลัก
- 4) ยอดขายประกันชีวิตรวมทุกกรมธรรม์

โดยมูลค่ายอดขายประกันชีวิตกรมธรรม์แบบต่าง ๆ แสดงไว้ดังตารางที่ 2.1

ตารางที่ 2.1 มูลค่ายอดขายประกันชีวิตกรมธรรม์แบบทั่วไป กรมธรรม์หลัก สัญญาเพิ่มเติมจากกรมธรรม์หลัก และรวมทุกกรมธรรม์

Period	Year	Quarter	มูลค่ายอดขายประกันชีวิต (บาท)			
			กรมธรรม์ทั่วไป	กรมธรรม์หลัก	สัญญาเพิ่มเติม	รวมทุกกรมธรรม์
1	2018	1	120,957,430.86	12,064,969.75	1,459,909.47	134,482,310.08
2	2018	2	237,176,683.26	22,415,287.00	3,135,763.08	262,727,733.34
3	2018	3	348,315,999.34	38,032,011.99	66,552,399.05	452,900,410.38
4	2018	4	473,640,932.31	53,146,719.25	90,247,932.04	617,035,583.60
5	2019	1	111,696,754.49	7,409,569.12	24,861,304.10	143,967,627.71
6	2019	2	223,261,561.44	17,086,034.17	48,072,989.46	288,420,585.07
7	2019	3	334,183,042.80	27,396,807.51	72,838,021.79	434,417,872.10
8	2019	4	456,844,390.25	42,522,874.74	98,665,579.77	598,032,844.76
9	2020	1	107,824,503.86	8,213,632.78	28,520,342.94	144,558,479.58
10	2020	2	207,898,856.27	17,182,364.88	52,265,650.40	277,346,871.55
11	2020	3	311,040,107.44	27,695,833.73	78,844,226.85	417,580,168.02

ตารางที่ 2.1 มูลค่ายอดขายประกันชีวิตกรมธรรม์แบบทั่วไป กรมธรรม์หลัก สัญญาเพิ่มเติมจาก
กรมธรรม์หลัก และรวมทุกกรมธรรม์ (ต่อ)

Period	Year	Quarter	มูลค่ายอดขายประกันชีวิต (บาท)			
			กรมธรรม์ทั่วไป	กรมธรรม์หลัก	สัญญาเพิ่มเติม	รวมทุกกรมธรรม์
12	2020	4	432,801,114.79	45,026,611.16	107,022,301.12	584,850,027.07
13	2021	1	103,863,695.65	14,702,125.45	28,992,683.84	147,558,504.94
14	2021	2	200,041,391.53	27,753,519.57	55,743,671.59	283,538,582.69
15	2021	3	297,642,101.18	43,886,109.98	83,800,510.25	425,328,721.41
16	2021	4	413,999,403.31	67,116,178.95	114,310,417.96	595,426,000.22
17	2022	1	98,786,804.70	14,242,461.68	31,691,762.82	144,721,029.20
18	2022	2	191,643,743.94	26,229,579.05	60,425,681.52	278,299,004.51
19	2022	3	292,947,075.59	38,337,511.45	90,518,857.46	421,803,444.50
20	2022	4	408,846,655.82	57,804,859.63	122,999,750.74	589,651,266.19

2.5 ภาษาที่ใช้เขียนโปรแกรม

Python เป็นภาษาการเขียนโปรแกรมที่ได้รับความนิยมอย่างแพร่หลายในด้านเว็บแอปพลิเคชัน การพัฒนาซอฟต์แวร์ วิทยาศาสตร์ข้อมูล และแมชชีนเลิร์นนิง (ML) เนื่องจาก Python มีประสิทธิภาพสูง เรียนรู้ได้สะดวก และสามารถทำงานได้บนหลายแพลตฟอร์ม นักพัฒนาจึงเลือกใช้ภาษา Python เป็นอย่างมาก นอกจากนี้ ซอฟต์แวร์ Python ยังสามารถดาวน์โหลดได้ฟรี รองรับการทำงานร่วมกับระบบต่าง ๆ และส่งเสริมการเร่งกระบวนการพัฒนาได้อย่างมีประสิทธิภาพ (แอมะซอนเว็บเซอร์วิส, 2567) ตัวอย่างมีดังนี้

2.5.1 การพัฒนาเว็บฝั่งเซิร์ฟเวอร์

การพัฒนาเว็บฝั่งเซิร์ฟเวอร์ (Server) เกี่ยวข้องกับฟังก์ชันแบ็กเอนด์ (Backend) ที่ซับซ้อนซึ่งทำให้เว็บไซต์สามารถทำงานและแสดงข้อมูลให้กับผู้ใช้ได้ ตัวอย่างเช่น เว็บไซต์อาจต้องทำงานร่วมกับฐานข้อมูล ติดต่อกับเว็บไซต์อื่น ๆ และรักษาความปลอดภัยของข้อมูลระหว่างการส่งผ่านเครือข่าย การพัฒนาในส่วนนี้มักเกี่ยวข้องกับภาษาโปรแกรม เช่น PHP Python หรือ Node.js และใช้ร่วมกับระบบจัดการฐานข้อมูลอย่าง MySQL หรือ MongoDB เพื่อจัดเก็บและเรียกคืนข้อมูลอย่างมีประสิทธิภาพ

Python มีประโยชน์อย่างยิ่งในการเขียนโค้ดฝั่งเซิร์ฟเวอร์เนื่องจากมีไลบรารี (Library) จำนวนมากที่มีโค้ดสำเร็จรูปสำหรับฟังก์ชันเบ็คเอนด์ที่ซับซ้อนอยู่แล้ว และยังมีเฟรมเวิร์ก (Framework) หลากหลายที่สนับสนุนให้นักพัฒนาสร้างเว็บแอปพลิเคชันได้อย่างรวดเร็ว เช่น สามารถสร้างโครงสร้างแอปพลิเคชันได้โดยไม่ต้องเริ่มจากศูนย์ และใช้เครื่องมือทดสอบภายในเฟรมเวิร์กโดยไม่ต้องพึ่งพาเครื่องมือจากภายนอก

2.5.2 ระบบอัตโนมัติด้วยสคริปต์ Python

ภาษาการเขียนสคริปต์เป็นภาษาการเขียนโปรแกรมที่ทำให้การทำงานที่ปกติมนุษย์ทำได้สามารถดำเนินการได้โดยอัตโนมัติ โปรแกรมเมอร์จึงนิยมใช้สคริปต์ Python เพื่อทำให้งานประจำวันหลายประเภทเป็นไปโดยอัตโนมัติ เช่น

- 1) การเปลี่ยนชื่อไฟล์จำนวนมากในครั้งเดียว
- 2) การแปลงไฟล์เป็นรูปแบบอื่น
3. การลบคำซ้ำในไฟล์ข้อความ
- 4) การดำเนินการทางคณิตศาสตร์พื้นฐาน
- 5) การส่งข้อความอีเมล
- 6) การดาวน์โหลดเนื้อหาจากอินเทอร์เน็ต
- 7) การวิเคราะห์บันทึกพื้นฐาน
- 8) การค้นหาข้อผิดพลาดในหลายไฟล์

2.5.3 วิทยาศาสตร์ข้อมูลและแมชชีนเลิร์นนิง

วิทยาศาสตร์ข้อมูลมุ่งเน้นการดึงความรู้จากข้อมูล ในขณะที่แมชชีนเลิร์นนิง (ML) ทำให้คอมพิวเตอร์เรียนรู้จากข้อมูลโดยอัตโนมัติและสามารถทำการคาดการณ์ได้อย่างแม่นยำ นักวิทยาศาสตร์ข้อมูลใช้ Python สำหรับงานด้านวิทยาศาสตร์ข้อมูลหลายประเภท เช่น

- 1) การทำความสะอาดข้อมูล ซึ่งหมายถึงการแก้ไขและลบข้อมูลที่ไม่ถูกต้อง
- 2) การแยกและเลือกคุณสมบัติต่าง ๆ ของข้อมูล
- 3) การระบุประเภทข้อมูล โดยการเพิ่มชื่อที่มีความหมายให้กับข้อมูล
- 4) การค้นหาสถิติต่าง ๆ จากข้อมูล
- 5) การแสดงข้อมูลด้วยภาพ เช่น แผนภูมิเส้น กราฟแท่ง ฮิสโทแกรม และแผนภูมิ

นักวิทยาศาสตร์ข้อมูลยังใช้ไลบรารี Python สำหรับแมชชีนเลิร์นนิงเพื่อฝึกฝนโมเดล ML และสร้างตัวจำแนกประเภทที่สามารถจำแนกข้อมูลได้อย่างแม่นยำ ตัวจำแนกประเภทแบบ Python ถูกใช้ในการจำแนกประเภทต่าง ๆ เช่น รูปภาพ ข้อความ การรับส่งข้อมูลทางเครือข่าย การรู้จำเสียง และการจดจำใบหน้า นอกจากนี้ นักวิทยาศาสตร์ข้อมูลยังใช้ Python ในการทำดีปเลิร์นนิง ซึ่งเป็นเทคนิคขั้นสูงในแมชชีนเลิร์นนิง โดยดีปเลิร์นนิงทำให้ระบบสามารถเรียนรู้จากข้อมูลจำนวนมากและสร้างแบบจำลองที่มีความซับซ้อนสูงได้อย่างมีประสิทธิภาพ

2.5.4 การพัฒนาซอฟต์แวร์

นักพัฒนาซอฟต์แวร์มักใช้ Python สำหรับงานด้านการพัฒนาและการประยุกต์ใช้ซอฟต์แวร์ในหลายด้าน เช่น

- 1) การติดตามข้อผิดพลาด (บั๊ก) ในโค้ดของซอฟต์แวร์
- 2) การสร้างซอฟต์แวร์โดยอัตโนมัติ
- 3) การจัดการ โครงการซอฟต์แวร์
- 4) การพัฒนาต้นแบบซอฟต์แวร์
- 5) การพัฒนาแอปพลิเคชันบนเดสก์ท็อป โดยใช้ไลบรารีสำหรับส่วนติดต่อผู้ใช้

แบบกราฟิก (GUI)

- 6) การพัฒนาเกม ตั้งแต่เกมที่ใช้ข้อความพื้นฐาน ไปถึงวิดีโอเกมที่ซับซ้อน

2.5.5 ระบบทดสอบซอฟต์แวร์อัตโนมัติ

การทดสอบซอฟต์แวร์เป็นกระบวนการที่ตรวจสอบว่าผลลัพธ์ที่ได้รับจากซอฟต์แวร์ตรงกับผลลัพธ์ที่คาดหวังหรือไม่ เพื่อให้แน่ใจว่าซอฟต์แวร์ไม่มีข้อผิดพลาด นักพัฒนามักใช้เฟรมเวิร์กต่าง ๆ สำหรับการทดสอบหน่วยใน Python เช่น Unit Test Robot และ PyUnit เพื่อทดสอบการทำงานของโค้ดที่เขียนขึ้น ผู้ทดสอบซอฟต์แวร์ใช้ Python ในการเขียนกรณีทดสอบสำหรับสถานการณ์ต่าง ๆ ตัวอย่างเช่น การทดสอบอินเทอร์เฟซผู้ใช้ของเว็บแอปพลิเคชัน การทดสอบส่วนประกอบซอฟต์แวร์หลายตัว และการทดสอบคุณสมบัติใหม่ นักพัฒนาสามารถใช้เครื่องมือหลายตัวในการเรียกใช้สคริปต์ทดสอบโดยอัตโนมัติ เครื่องมือเหล่านี้เรียกว่าเครื่องมือรวบรวมและปรับใช้อัตโนมัติ (Continuous Integration/Continuous Deployment หรือ CI/CD) เช่น Travis CI และ Jenkins ซึ่งทำให้การทดสอบเป็นไปโดยอัตโนมัติ เครื่องมือ CI/CD จะเรียกใช้สคริปต์ทดสอบ Python โดยอัตโนมัติ และรายงานผลการทดสอบทุกครั้งที่นักพัฒนานำการเปลี่ยนแปลงโค้ดใหม่เข้ามาใช้

2.6 การเขียน Python ด้วย Google Colab

การเขียนโปรแกรมภาษา Python ด้วย Google Colab สามารถเริ่มต้นได้โดยเข้าไปที่เว็บไซต์ <https://colab.research.google.com>

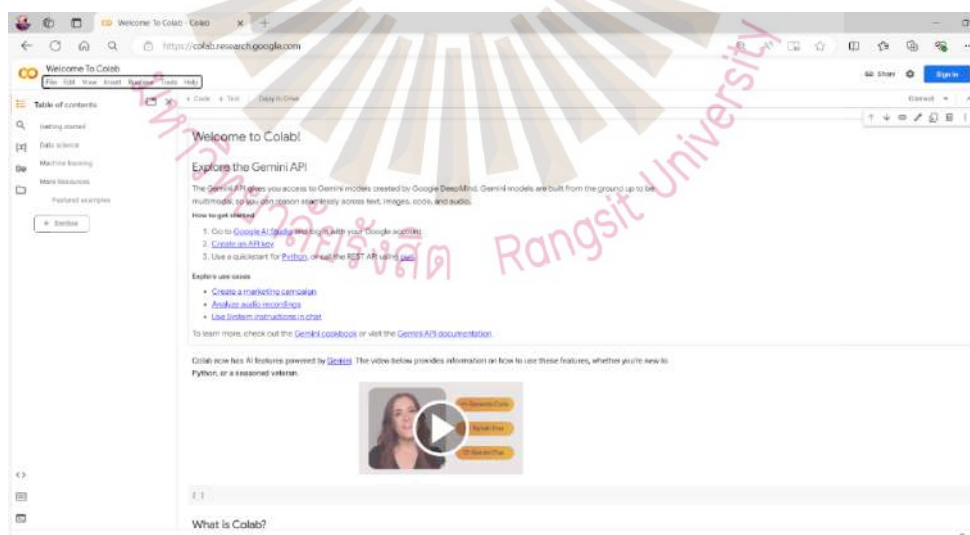
Google Colab เป็นเครื่องมือที่ทำให้สามารถเขียนและรันโค้ด Python ได้บนเบราว์เซอร์โดยไม่ต้องติดตั้งโปรแกรมเพิ่มเติม

นอกจากนี้ยังรองรับการใช้งานร่วมกับ Google Drive ทำให้สามารถบันทึกและแชร์ไฟล์งานได้อย่างสะดวก

2.6.1 รูปที่ 2.1 คลิกเข้าไปที่เว็บไซต์ Google Colab และคลิกไปที่เมนู File และเลือก New Notebook ใน Drive เพื่อเริ่มต้นการเขียนโปรแกรมภาษา Python ด้วย Google Colab

Google Colab เป็นเครื่องมือที่สนับสนุนการเขียนและรันโค้ดได้ทันทีบนเบราว์เซอร์โดยไม่ต้องติดตั้งโปรแกรมเพิ่มเติม

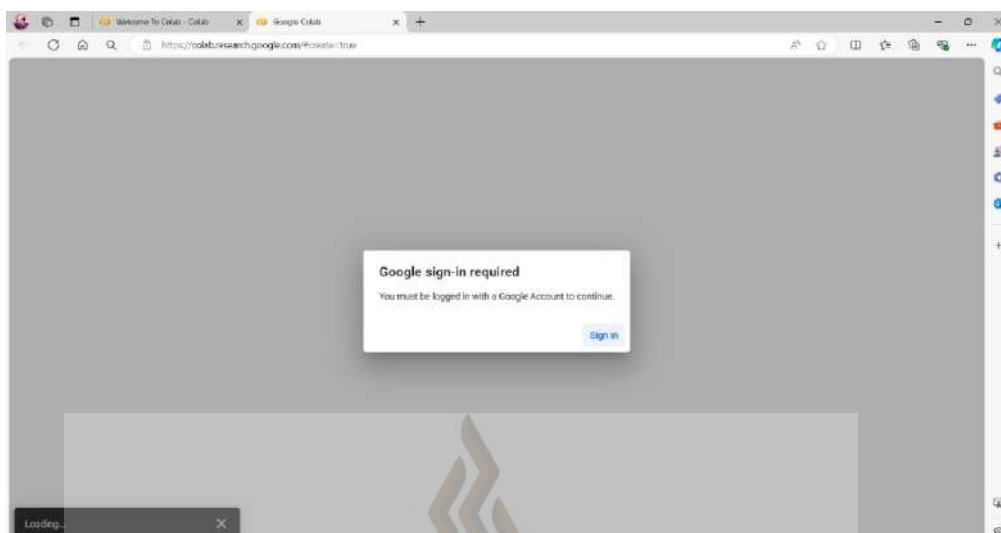
นอกจากนี้ยังสามารถเชื่อมต่อกับ Google Drive เพื่อจัดเก็บไฟล์และแชร์งานร่วมกับผู้อื่นได้อย่างสะดวก



รูปที่ 2.1 เว็บไซต์ Google Colab

ที่มา: Google, 2023a

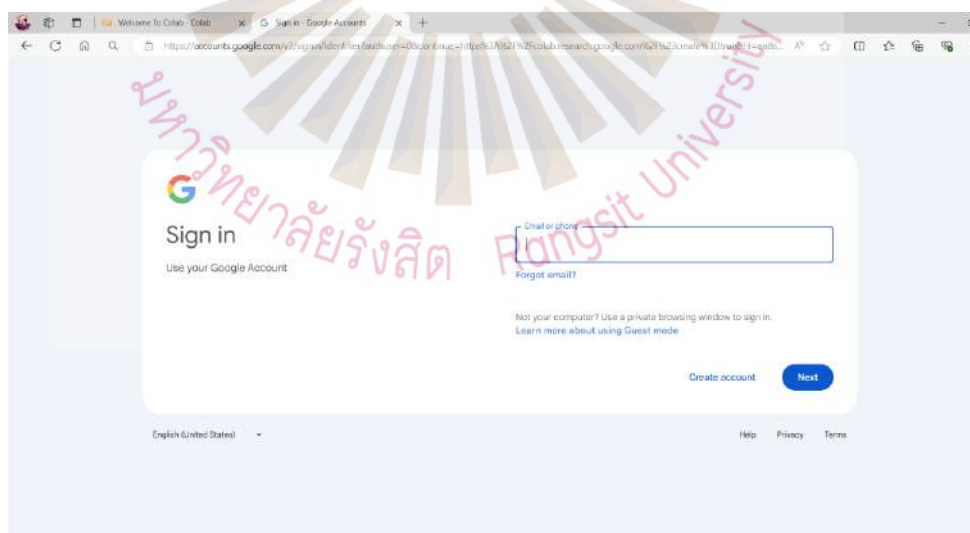
2.6.2 รูปที่ 2.2 การจะใช้ Google Colab ต้องมีบัญชีของ Google ดังนั้นผู้ใช้สามารถใช้บัญชี Gmail ในการ Sign In Google Colab ได้



รูปที่ 2.2 การ Sign ต้องมีบัญชีของ Google

ที่มา: Google, 2023b

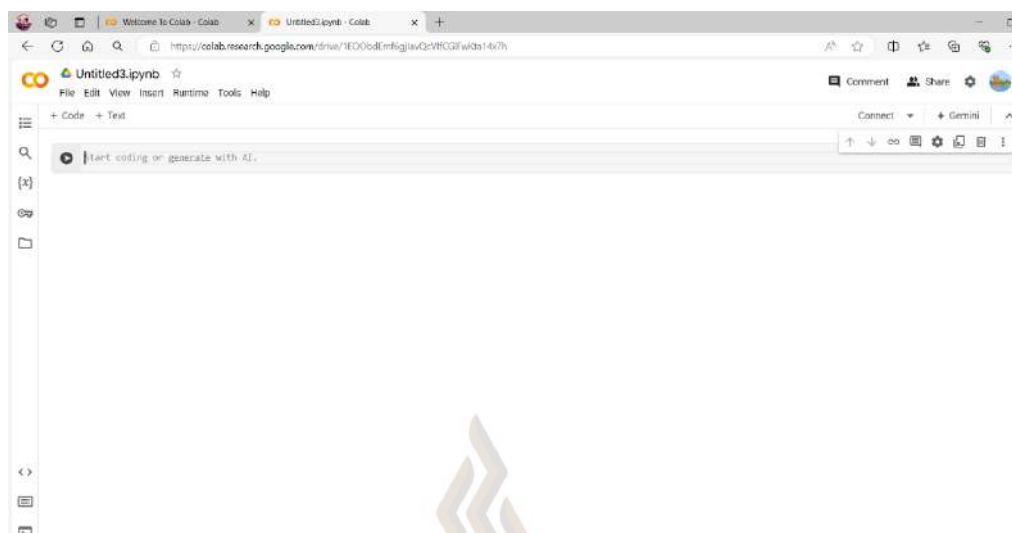
2.6.3 รูปที่ 2.3 กรอกชื่อ Gmail และ Password ไปตามขั้นตอน กรณียังไม่มี Gmail ก็สามารถสมัครบัญชี และนำบัญชี Gmail มา Sign In ในการใช้งาน Google Colab เพื่อเข้าสู่ระบบ และเริ่มต้นใช้งานแพลตฟอร์มสำหรับการเขียนและรันโค้ดบนระบบคลาวด์



รูปที่ 2.3 การนำบัญชี Gmail มา Sign In ในการใช้งาน Colab

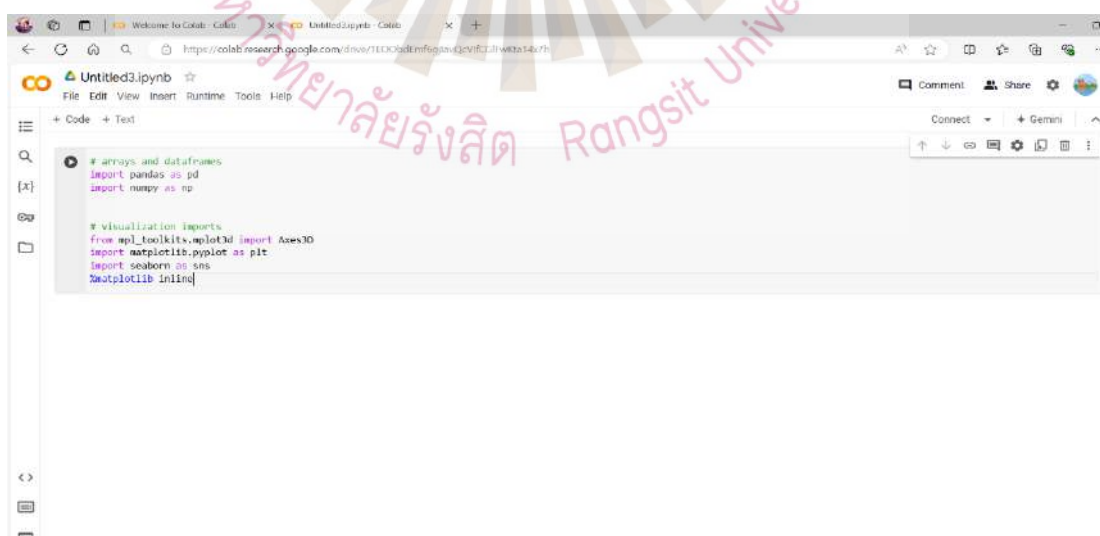
ที่มา: Google, 2023b

2.6.4 รูปที่ 2.4 ช่องว่างที่ปรากฏบนหน้าจอจากเว็บไซต์ Google Colab แสดงความพร้อมใช้งานเขียนโปรแกรมภาษา Python



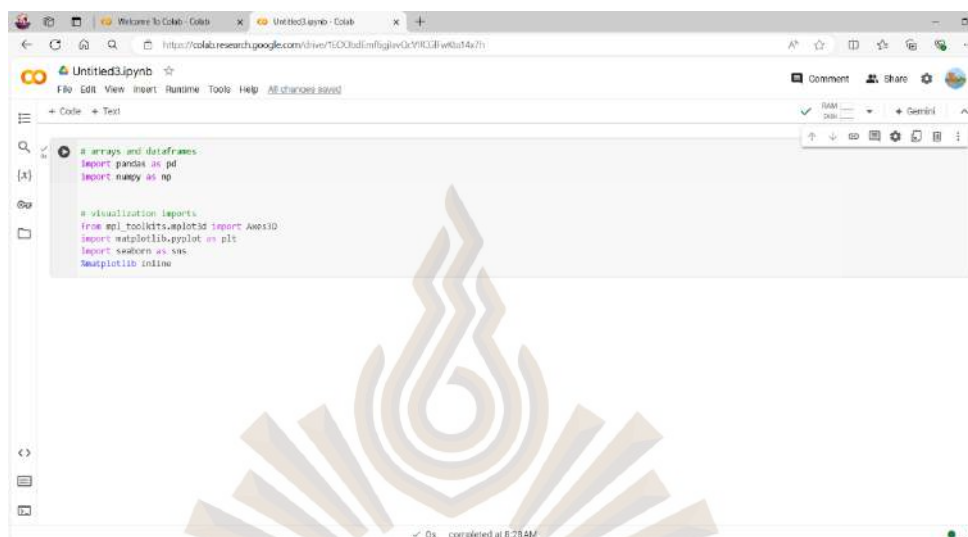
รูปที่ 2.4 หน้าจอจากเว็บไซต์ Google Colab สำหรับ เขียนโปรแกรมภาษา Python
ที่มา: Google, 2023a

2.6.5 รูปที่ 2.5 พิมพ์ Import Pandas As Pd และ Import Numpy As Np ต่อด้วยคำสั่งสำหรับการนำเข้าเครื่องมือด้านการแสดงผล (Visualization Imports) ซึ่งเป็นการเตรียมความพร้อมในการเขียนโปรแกรมด้วยภาษา Python สำหรับการวิเคราะห์และพยากรณ์ข้อมูล พร้อมทั้งสร้างพื้นฐานที่จำเป็นต่อกระบวนการวิเคราะห์ข้อมูล การเรียกใช้ไลบรารีเหล่านี้เป็นขั้นตอนสำคัญในการเตรียมสิ่งแวดล้อมสำหรับการจัดการข้อมูลและการแสดงผลข้อมูล



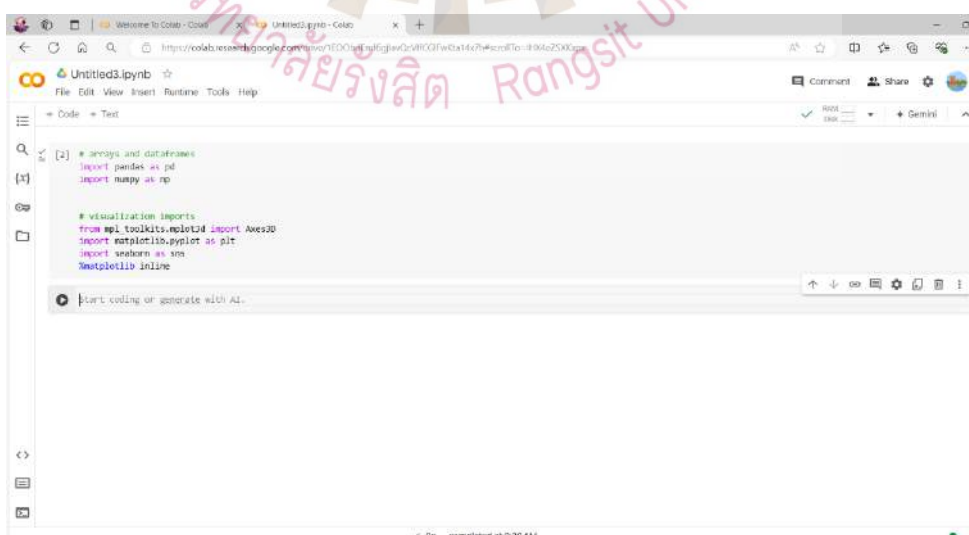
รูปที่ 2.5 เริ่มการเขียนโปรแกรมภาษา Python ด้วย Google Colab
ที่มา: Google, 2023a

2.6.6 รูปที่ 2.6 คลิกตรงปุ่มสามเหลี่ยมที่เหมือนปุ่ม Play ของวิดีโอ หรือกด Ctrl-Enter โดยกดปุ่ม Ctrl ค้างไว้ แล้วกด Enter เพื่อทดสอบ Code เมื่อรันคำสั่งแล้ว ระบบจะแสดงผลลัพธ์หรือกราฟที่ได้จากโค้ดที่เขียนไว้ใน Cell นั้นทันที



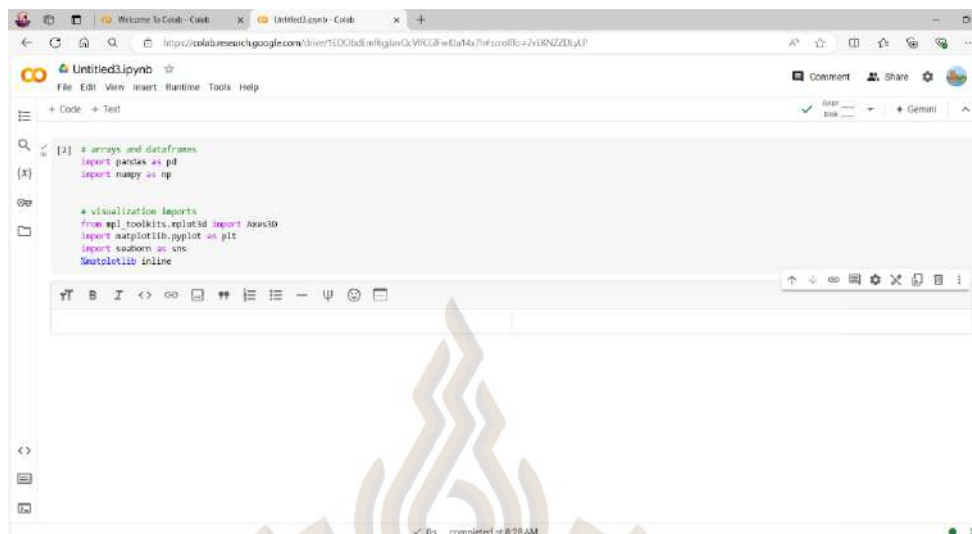
รูปที่ 2.6 ทดสอบ Code จากโปรแกรมภาษา Python ที่ Google Colab
ที่มา: Google, 2023a

2.6.7 รูปที่ 2.7 คลิก +Code จะมีช่องขึ้นมาใหม่สำหรับเขียน Code ทำให้เขียนโปรแกรมอื่นเพิ่มขึ้นได้



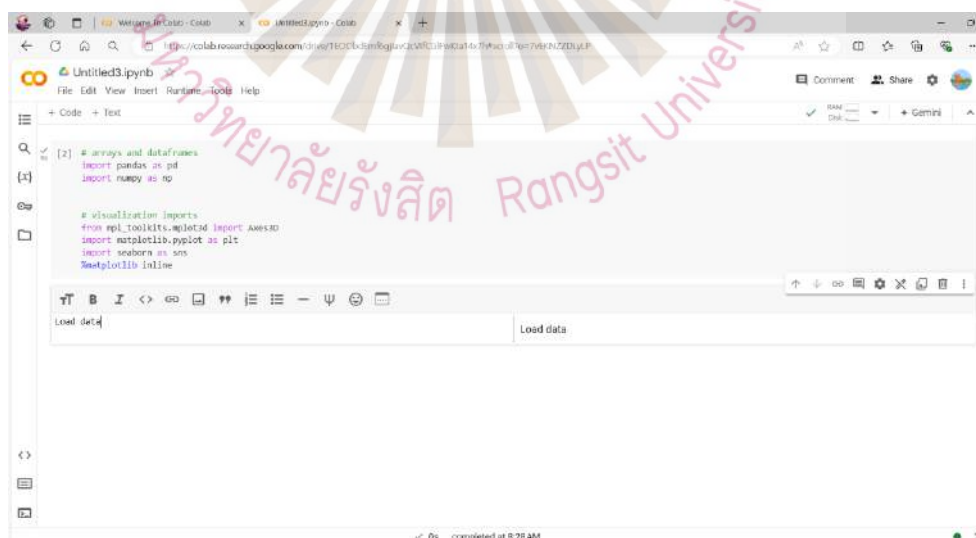
รูปที่ 2.7 การเพิ่มช่องใหม่สำหรับเขียน Code โปรแกรมภาษา Python ที่ Google Colab
ที่มา: Google, 2023a

2.6.8 รูปที่ 2.8 กรณีต้องการแทรกข้อความหรือบทความ คลิก +Text จะได้ช่องสำหรับเขียนข้อความ



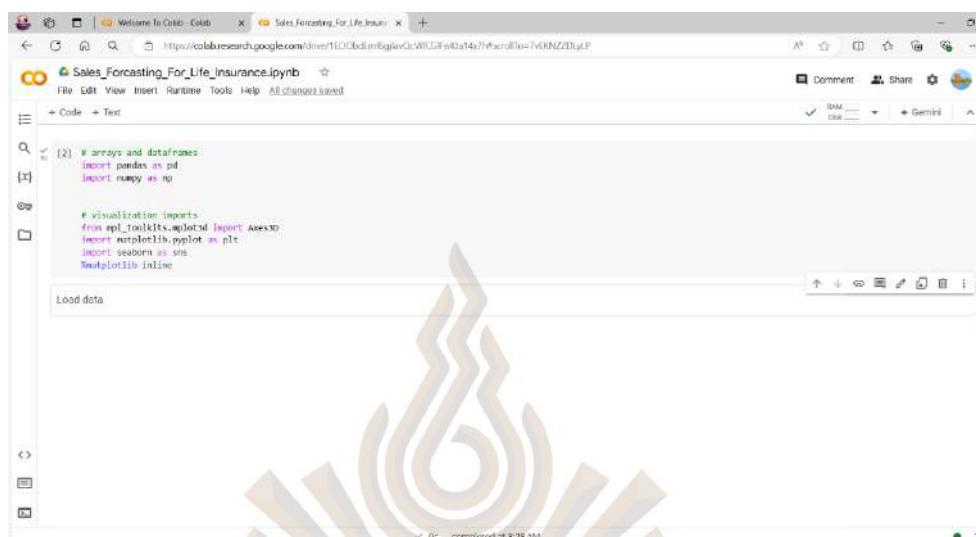
รูปที่ 2.8 การแทรกข้อความหรือบทความที่ Google Colab
ที่มา: Google, 2023a

2.6.9 รูปที่ 2.9 ผู้เขียนโปรแกรมภาษา Python สามารถเขียนโปรแกรมและทดสอบ กรณีจะเขียน Code เพิ่ม คลิก +Code กรณีจะแทรกข้อความคลิก +Text สลับกันได้



รูปที่ 2.9 การเขียน Code และแทรกข้อความสลับกันที่ Google Colab
ที่มา: Google, 2023a

2.6.10 ในรูปที่ 2.10 เปลี่ยนชื่อ Notebook ตรงช่องที่เขียนว่า Untitled.Ipynb เป็น Sales_Forecasting_For_Life_Insurance.Ipynb



รูปที่ 2.10 การเปลี่ยนชื่อ Notebook ที่ Google Colab
ที่มา: Google, 2023a

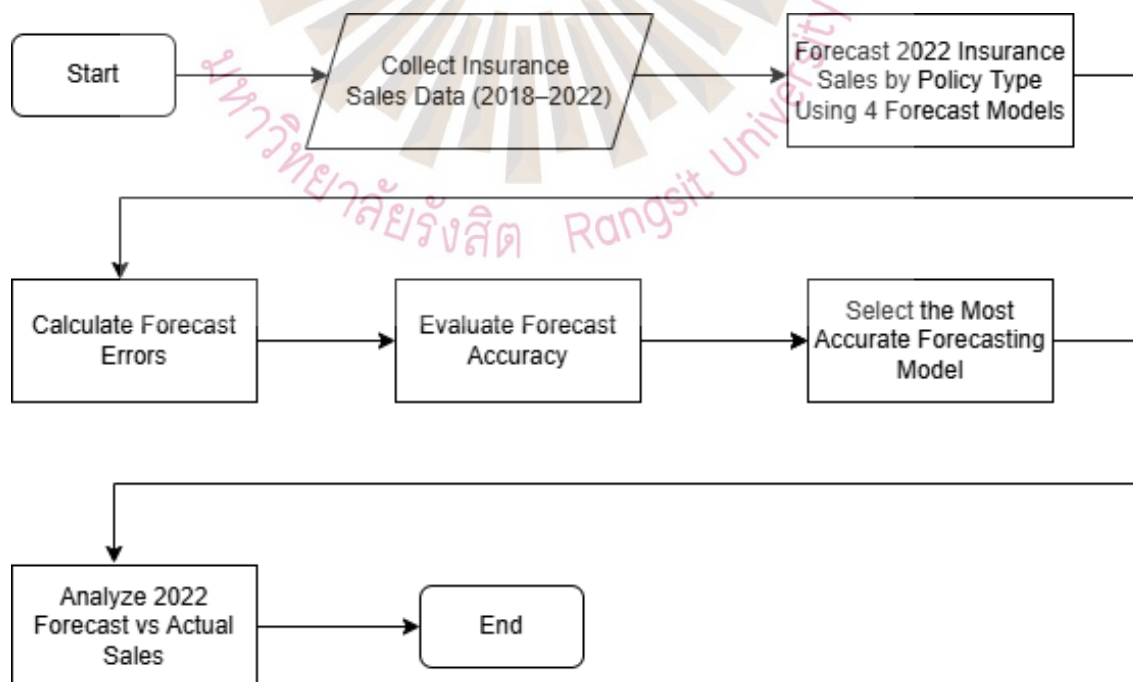
เมื่อแก้ไขชื่อเรียบร้อยแล้ว โปรแกรมจะถูก Save อัตโนมัติ กรณีเข้ามาใช้งาน Colab ใหม่อีกครั้ง สามารถคลิก File – Open Notebook แล้วหาชื่อที่ Save ไว้เพื่อเปิดมาใช้งาน

มหาวิทยาลัยรังสิต Rangsit University

บทที่ 3

ขั้นตอนการดำเนินงาน

ขั้นตอนการพยากรณ์ยอดขายประกันชีวิตอย่างเป็นระบบ เริ่มจากการรวบรวมข้อมูลยอดขายประกันในช่วงปี พ.ศ. 2561 - 2565 (ค.ศ. 2018 - 2022) เป็นระยะเวลา 5 ปี แยกตามประเภทกรรมธรรม์ ได้แก่ กรรมธรรม์ทั่วไป กรรมธรรม์หลัก สัญญาเพิ่มเติม และยอดรวมทั้งหมด จากนั้นดำเนินการพยากรณ์ยอดขายสำหรับปี พ.ศ. 2565 (ค.ศ. 2022) ในแต่ละประเภท โดยใช้วิธีการทางสถิติเพื่อคำนวณค่าความคลาดเคลื่อน เช่น MAD MAPE และ MSE เพื่อประเมินความแม่นยำของผลการพยากรณ์ จากผลการวิเคราะห์จะเลือกวิธีที่มีความแม่นยำสูงสุดสำหรับแต่ละประเภทกรรมธรรม์ และในขั้นสุดท้ายจะเปรียบเทียบค่าที่พยากรณ์ได้กับยอดขายจริงในปี พ.ศ. 2565 พร้อมทั้งวิเคราะห์ปัจจัยหรือสาเหตุที่อาจส่งผลกระทบต่อความแตกต่างของแต่ละประเภทประกัน กระบวนการนี้ทำให้การพยากรณ์มีความน่าเชื่อถือและใช้ข้อมูลอย่างมีประสิทธิภาพ กระบวนการพยากรณ์นี้สามารถเลือกวิธีการที่เหมาะสมที่สุดในการคาดการณ์ยอดขาย และระบุปัจจัยที่มีผล



รูปที่ 3.1 กระบวนการวิเคราะห์และพยากรณ์ยอดขายกรรมธรรม์ประกันภัย พร้อมการประเมินความแม่นยำของแบบจำลอง กรณีศึกษาปี ค.ศ. 2022

จากรูปที่ 3.1 แผนผังขั้นตอน (Flowchart) นี้นำเสนอขั้นตอนเชิงระบบของกระบวนการพยากรณ์ยอดขายกรมธรรม์ประกันชีวิตประจำปี พ.ศ. 2565 โดยอิงจากข้อมูลยอดขายย้อนหลังระหว่างปี พ.ศ. 2561 - 2565 ซึ่งจำแนกตามประเภทของกรมธรรม์อย่างชัดเจน กระบวนการประกอบด้วยการเก็บรวบรวมข้อมูล การพยากรณ์ด้วยเทคนิคทางสถิติต่าง ๆ การคำนวณค่าความคลาดเคลื่อนของผลลัพธ์ การประเมินประสิทธิภาพของแต่ละวิธี และการเลือกวิธีที่แม่นยำที่สุดเพื่อนำไปเปรียบเทียบกับข้อมูลจริงในปีเป้าหมาย ทั้งนี้ เพื่อใช้ผลลัพธ์ในการวิเคราะห์แนวโน้ม และพัฒนาแบบจำลองให้เหมาะสมกับพฤติกรรมของข้อมูลในอนาคตอย่างมีหลักวิชาการ โดยสามารถอธิบายเป็นลำดับขั้นได้ดังนี้

1) Start เริ่มต้นกระบวนการวิเคราะห์และพยากรณ์ยอดขายกรมธรรม์ประกันภัย พร้อมการประเมินความแม่นยำของแบบจำลอง กรณีศึกษาปี ค.ศ. 2022

2) Collect Insurance Sales Data (2018 - 2022) รวบรวมข้อมูลยอดขายประกันชีวิตย้อนหลัง 5 ปี ตั้งแต่ปี พ.ศ. 2561 - 2565 (ค.ศ. 2018 - 2022) โดยแยกตามประเภทของกรมธรรม์ 4 กลุ่ม ดังนี้

- 2.1) กรมธรรม์ทั่วไป
- 2.2) กรมธรรม์หลัก
- 2.3) สัญญาเพิ่มเติม
- 2.4) รวมทุกกรมธรรม์

ข้อมูลเหล่านี้จะถูกใช้พื้นฐานสำหรับการพยากรณ์ในขั้นตอนถัดไป

3) Forecast 2022 Insurance Sales by Policy Type Using 4 Forecast Models พยากรณ์ยอดขายประกันชีวิตในปีที่ 5 (ปี ค.ศ. 2022) สำหรับแต่ละประเภทของกรมธรรม์ โดยใช้ 4 วิธีทางสถิติ ดังนี้

3.1) Simple Exponential Smoothing

ใช้สำหรับข้อมูลที่ไม่มีแนวโน้มหรือฤดูกาล โดยจะให้น้ำหนักกับข้อมูลในอดีตที่อยู่ใกล้ปัจจุบันมากกว่าข้อมูลที่อยู่ห่างออกไป เพื่อให้การพยากรณ์สะท้อนค่าล่าสุดได้แม่นยำยิ่งขึ้น

3.2) Double Exponential Smoothing

ใช้เมื่อต้องการพิจารณาแนวโน้มในข้อมูล โดยสามารถจัดการกับข้อมูลที่มีการเปลี่ยนแปลงอย่างต่อเนื่องในช่วงเวลาได้อย่างมีประสิทธิภาพ

3.3) Holt-Winters Additive Method

ใช้เมื่อข้อมูลมีแนวโน้มและฤดูกาลที่มีการเปลี่ยนแปลงเท่า ๆ กันในทุกช่วงเวลา ไม่มีการเพิ่มหรือลดทวิคูณในฤดูกาล

3.4) Holt-Winters Multiplicative Method

ใช้เมื่อข้อมูลมีแนวโน้มและฤดูกาลแบบ “คูณ” (เช่น ความผันผวนเปลี่ยนแปลงตามขนาดยอดขาย) การใช้หลายวิธีทำให้สามารถเปรียบเทียบและเลือกวิธีที่ให้ผลลัพธ์แม่นยำที่สุดในบริบทของข้อมูลจริง

4) Calculate Forecast Errors เมื่อได้ผลลัพธ์การพยากรณ์จากแต่ละวิธีแล้ววัดค่าความคลาดเคลื่อนของแต่ละโมเดล โดยใช้ 3 ตัวชี้วัดมาตรฐาน ดังนี้

4.1) MAD (Mean Absolute Deviation) วัดความเบี่ยงเบนโดยเฉลี่ย

4.2) MAPE (Mean Absolute Percentage Error) วัดข้อผิดพลาดในรูปเปอร์เซ็นต์

4.3) MSE (Mean Square Error) วัดความคลาดเคลื่อนแบบยกกำลังสอง

5) Evaluate Forecast Accuracy นำค่าความคลาดเคลื่อนจากแต่ละวิธีมาวิเคราะห์และประเมินความแม่นยำของผลการพยากรณ์ เพื่อพิจารณาว่าวิธีใดให้ผลที่ใกล้เคียงกับยอดขายจริงมากที่สุด โดยคำนวณความแม่นยำของการพยากรณ์จากค่า MAPE ที่ได้ ซึ่งทำให้สามารถเลือกวิธีที่มีความแม่นยำสูงสุดในการพยากรณ์ได้

6) Select the Most Accurate Forecasting Model จากการเปรียบเทียบค่าความคลาดเคลื่อน จะทำการ เลือกวิธีการพยากรณ์ที่มีประสิทธิภาพสูงสุด สำหรับแต่ละประเภทกรรมธรรม์ โดยพิจารณาจากค่าความแม่นยำที่ดีที่สุดในแต่ละกรณี

7) Analyze 2022 Forecast vs Actual Sales เปรียบเทียบยอดขายที่พยากรณ์ไว้กับยอดขายจริงในปี ค.ศ. 2022 โดยแยกตามแต่ละประเภทกรรมธรรม์ พร้อมทั้ง วิเคราะห์เหตุผลที่อาจทำให้เกิดความแตกต่าง เช่น สถานการณ์เศรษฐกิจ พฤติกรรมผู้บริโภค หรือการเปลี่ยนแปลงด้านนโยบาย เพื่อสรุปแนวโน้มและแนวทางในการพัฒนาโมเดล

8) End สิ้นสุดกระบวนการวิเคราะห์ โดยสรุปผลการพยากรณ์และประเมินเพื่อใช้เป็นแนวทางในการปรับปรุงโมเดลหรือวางแผนกลยุทธ์

ในการพยากรณ์ยอดขายประกันชีวิต ได้นำเข้าข้อมูลยอดขายรายปีจากไฟล์ CSV ครอบคลุมช่วงเวลา 5 ปี (ค.ศ. 2018 - 2022) โดยใช้ข้อมูล 4 ปีแรกในการพยากรณ์ยอดขายปีที่ 5 (ค.ศ. 2022) ด้วย 4 วิธี ได้แก่ วิธีบวกลบคูณหาร-วินเทอร์ วิธีคูณลบคูณหาร-วินเทอร์ การปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย และการปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า พร้อมปรับแต่งค่าพารามิเตอร์ เช่น ค่า Smoothing Level เพื่อเพิ่มความแม่นยำ และเปรียบเทียบผลลัพธ์เพื่อเลือกวิธีที่ให้ค่าความผิดพลาดต่ำที่สุดสำหรับการสนับสนุนการตัดสินใจทางธุรกิจ ประกอบด้วยขั้นตอนดังนี้

3.1 การสร้างไฟล์ CSV สำหรับยอดขายประกันชีวิตด้วยข้อมูลที่รวบรวมจากเว็บไซต์สำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย (คปภ.)

เพื่อให้สามารถนำข้อมูลยอดขายประกันชีวิตมาใช้ในการวิเคราะห์หรือประมวลผลได้อย่างสะดวกและเป็นระบบ จึงได้มีการจัดทำไฟล์ข้อมูลในรูปแบบ CSV (Comma - Separated Values) โดยใช้ข้อมูลจากเว็บไซต์ของสำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย (คปภ.) ซึ่งเป็นหน่วยงานหลักที่ดูแลและเผยแพร่ข้อมูลสถิติเกี่ยวกับธุรกิจประกันภัยในประเทศไทย โดยจะดำเนินการจัดสร้างไฟล์ CSV สำหรับยอดขายประกันชีวิต แบ่งตามประเภทของกรรมธรรม์และลักษณะของข้อมูลที่เผยแพร่ไว้ในเว็บไซต์ของ คปภ. แบ่งออกเป็น 4 ไฟล์ ดังนี้

- 1) Sales_Primary-General.csv
- 2) Sales_Primary.csv
- 3) Sales_Additional.csv
- 4) Sales_Total.csv

3.2 การโหลดและเตรียมข้อมูล โค้ดสำหรับโหลดและเตรียมข้อมูล ดังนี้

จากรูปที่ 3.2 โค้ดนี้ใช้ไลบรารี Pandas เพื่อจัดการข้อมูลในรูปแบบตาราง โดยนำเข้าไลบรารีด้วยชื่อย่อว่า Pd แล้วอ่านไฟล์ Data.Csv เข้าเป็น DataFrame ชื่อ Data จากนั้นใช้ Print(Data.Head()) เพื่อแสดง 5 แถวแรกของข้อมูล เพื่อตรวจสอบชื่อคอลัมน์ รูปแบบข้อมูล หรือค่าผิดปกติก่อนนำไปวิเคราะห์หรือสร้างโมเดล

```
import pandas as pd

# โหลดข้อมูลจากไฟล์ CSV (สมมติว่าไฟล์ชื่อว่า Data.csv)
data = pd.read_csv('data.csv')

# ตรวจสอบข้อมูล
print(data.head())
```

รูปที่ 3.2 การโหลดและเตรียมข้อมูลสำหรับการพยากรณ์ยอดขายประกัน

3.3 พยากรณ์ยอดขายประกันชีวิตด้วย 4 วิธีการ

เพื่อให้สามารถวางแผนธุรกิจและประเมินแนวโน้มยอดขายประกันชีวิตในอนาคตได้อย่างมีประสิทธิภาพ จึงมีความจำเป็นต้องใช้เทคนิคการพยากรณ์ (Forecasting) ที่เหมาะสม โดยในหัวข้อนี้จะนำเสนอ 4 วิธีการพื้นฐานที่สามารถประยุกต์ใช้กับข้อมูลยอดขายแบบอนุกรมเวลา ซึ่งจะทำให้สามารถคาดการณ์ทิศทางของยอดขายได้แม่นยำมากขึ้น และนำไปสู่การตัดสินใจเชิงกลยุทธ์ที่ดีขึ้นในภาคธุรกิจประกันภัย ประกอบด้วย

1) การบวกโฮลท์-วินเทอร์

หมายถึงการใช้วิธีการบวกค่าคาดการณ์ตามแบบจำลอง Holt-Winters ซึ่งเป็นเทคนิคพยากรณ์แบบอนุกรมเวลา (Time Series Forecasting) ที่รวมทั้งแนวโน้มและฤดูกาล

2) การคูณโฮลท์-วินเทอร์

เป็นการใช้รูปแบบ Holt-Winters เช่นกัน แต่ในแบบคูณ ซึ่งเหมาะกับข้อมูลที่มีความแปรผันแบบสัดส่วน เช่น ยอดขายที่เพิ่มขึ้นตามฤดูกาลในอัตราคงที่

3) การปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

คือวิธีการทำให้ข้อมูลเรียบขึ้น (ลดการสวิงของข้อมูล) ด้วยเทคนิค Exponential Smoothing แบบพื้นฐาน เพื่อหาค่าคาดการณ์ในอนาคต

4) การปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

เป็นการต่อยอดจากข้อ 3 โดยใช้ Double Exponential Smoothing ที่สามารถคำนึงถึงแนวโน้มของข้อมูลด้วย ทำให้เหมาะกับข้อมูลที่มีลักษณะเปลี่ยนแปลงตามกาลเวลา

3.4 คำนวณค่าพยากรณ์ด้วย 3 วิธีการ

ในการทำแบบจำลองการพยากรณ์ (Forecasting Model) ไม่ว่าจะใช้วิธีทางสถิติ หรือการเรียนรู้ของเครื่อง (Machine Learning) สิ่งสำคัญที่ต้องทำหลังจากการพยากรณ์คือ การวัดประสิทธิภาพของค่าพยากรณ์ เพื่อดูว่าค่าที่ได้มีความแม่นยำเพียงใดเมื่อเปรียบเทียบกับค่าจริง

การวัดความแม่นยำหรือความคลาดเคลื่อนของค่าพยากรณ์ (Forecast Error) เป็นกระบวนการสำคัญที่สนับสนุนให้สามารถปรับปรุงโมเดลให้ดีขึ้น รวมทั้งใช้ในการเปรียบเทียบประสิทธิภาพระหว่างวิธีการพยากรณ์ต่าง ๆ ได้อย่างมีประสิทธิภาพ ประกอบด้วย 3 วิธีการวัดความคลาดเคลื่อนที่นิยมใช้ ดังนี้

1) ค่าความผิดพลาดกำลังสองเฉลี่ย (MSE)

MSE เป็นวิธีที่นำความคลาดเคลื่อนแต่ละค่ามายกกำลังสองก่อนนำมาเฉลี่ย เหมาะสำหรับกรณีที่ต้องการให้ค่าคลาดเคลื่อนที่มากผิดปกติ (Outlier) มีผลกระทบต่อการประเมินมากขึ้น

1.1) ข้อดี เน้นความผิดพลาดที่รุนแรง เพราะการยกกำลังสองจะขยายค่าความคลาดเคลื่อนที่มาก

1.2) ข้อเสีย อ่อนไหวต่อ Outlier มากเกินไปในบางกรณี

2) ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (MAD)

MAD เป็นการนำค่าความคลาดเคลื่อนมาหาค่าสัมบูรณ์ (Absolute) แล้วนำมาเฉลี่ย ทำให้เหมาะสำหรับการประเมินความผิดพลาดโดยไม่ให้ค่าผิดปกติมีอิทธิพลมากนัก

2.1) ข้อดี เข้าใจชัดเจน ไม่อ่อนไหวต่อ Outlier มาก

2.2) ข้อเสีย อาจไม่เหมาะหากต้องการให้ความสำคัญกับความผิดพลาดที่รุนแรง

3) ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย (MAPE)

MAPE แสดงความผิดพลาดในรูปแบบเปอร์เซ็นต์ ทำให้สามารถเข้าใจได้ชัดเจนและเปรียบเทียบระหว่างชุดข้อมูลต่าง ๆ ได้ดี

3.1) ข้อดี ให้มุมมองเชิงเปรียบเทียบระหว่างหน่วยหรือขนาดข้อมูลต่างกัน

3.2) ข้อเสีย ไม่เหมาะกับข้อมูลที่มีค่าใกล้ศูนย์ เพราะจะทำให้ค่า MAPE พุ่งสูงหรือไม่มีความหมาย

ทั้งสามวิธีนี้มีจุดเด่นและข้อจำกัดที่แตกต่างกัน การเลือกใช้ควรพิจารณาตามบริบทของข้อมูลและเป้าหมายของการพยากรณ์ เช่น ถ้าต้องการหลีกเลี่ยงผลกระทบจาก Outlier อาจเลือกใช้ MAD แต่หากต้องการวิเคราะห์ความคลาดเคลื่อนในเชิงเปรียบเทียบอาจเลือกใช้ MAPE เป็นต้น โดยการเลือกวิธีที่เหมาะสมจะส่งผลให้ความแม่นยำและประสิทธิภาพในการพยากรณ์เพิ่มขึ้น

3.5 คำนวณความแม่นยำ สรุปผลและนำเสนอ

หลังจากที่ได้ทำการคำนวณค่าความคลาดเคลื่อนของการพยากรณ์ด้วยวิธีต่าง ๆ เช่น MSE, MAD และ MAPE แล้ว ขั้นตอนถัดไปคือการประเมินระดับความแม่นยำของแบบจำลองพยากรณ์แต่ละวิธี ซึ่งจะทำให้สามารถเปรียบเทียบประสิทธิภาพและเลือกวิธีที่เหมาะสมที่สุดสำหรับการนำไปใช้งานจริง

1) การคำนวณค่าความแม่นยำ

ในการวัดความแม่นยำของการพยากรณ์ ใช้ค่าความคลาดเคลื่อนประเภท MAPE มาเป็นเกณฑ์ โดยใช้สูตร

$$\text{Accuracy} = 100\% - \text{MAPE} \quad (3-1)$$

แนวคิดคือ หากค่า MAPE ต่ำ แสดงว่าค่าพยากรณ์มีความคลาดเคลื่อนน้อย ความแม่นยำก็จะสูงตามไปด้วย การหาค่าความแม่นยำในลักษณะนี้ทำให้สามารถแปลผลออกมาในรูปแบบที่เข้าใจได้ชัดเจน และใช้สื่อสารผลการวิเคราะห์ได้อย่างมีประสิทธิภาพ

2) การสรุปผล

เมื่อได้ค่าความแม่นยำของแต่ละวิธีการพยากรณ์แล้ว สามารถจัดเรียงและเปรียบเทียบเพื่อดูว่าวิธีใดมีประสิทธิภาพดีที่สุด วิธีใดแม่นยำน้อยที่สุด ข้อมูลนี้จะเป็นประโยชน์อย่างมากในการตัดสินใจเลือกวิธีที่เหมาะสมสำหรับการนำไปใช้ในสถานการณ์จริง

3) การนำเสนอผลในรูปแบบกราฟแท่ง

การนำเสนอข้อมูลในรูปแบบ กราฟแท่ง (Bar Chart) เป็นวิธีที่ทำให้เข้าใจผลลัพธ์ได้ชัดเจน โดยใช้กราฟแสดงค่าความแม่นยำของแต่ละวิธีการพยากรณ์บนแกนแนวนอน (Y-Axis) และชื่อของแต่ละวิธีบนแกนแนวนอน (X-Axis) ซึ่งจุดเด่นของการใช้กราฟแท่งมีดังนี้

3.1) เห็นความแตกต่างของค่าความแม่นยำระหว่างแต่ละวิธีได้อย่างชัดเจน

3.2) สนับสนุนการวิเคราะห์เชิงเปรียบเทียบ

3.3) สามารถใช้ประกอบการสื่อสารหรือรายงานผลการพยากรณ์ได้อย่างมีประสิทธิภาพ

ประสิทธิภาพ

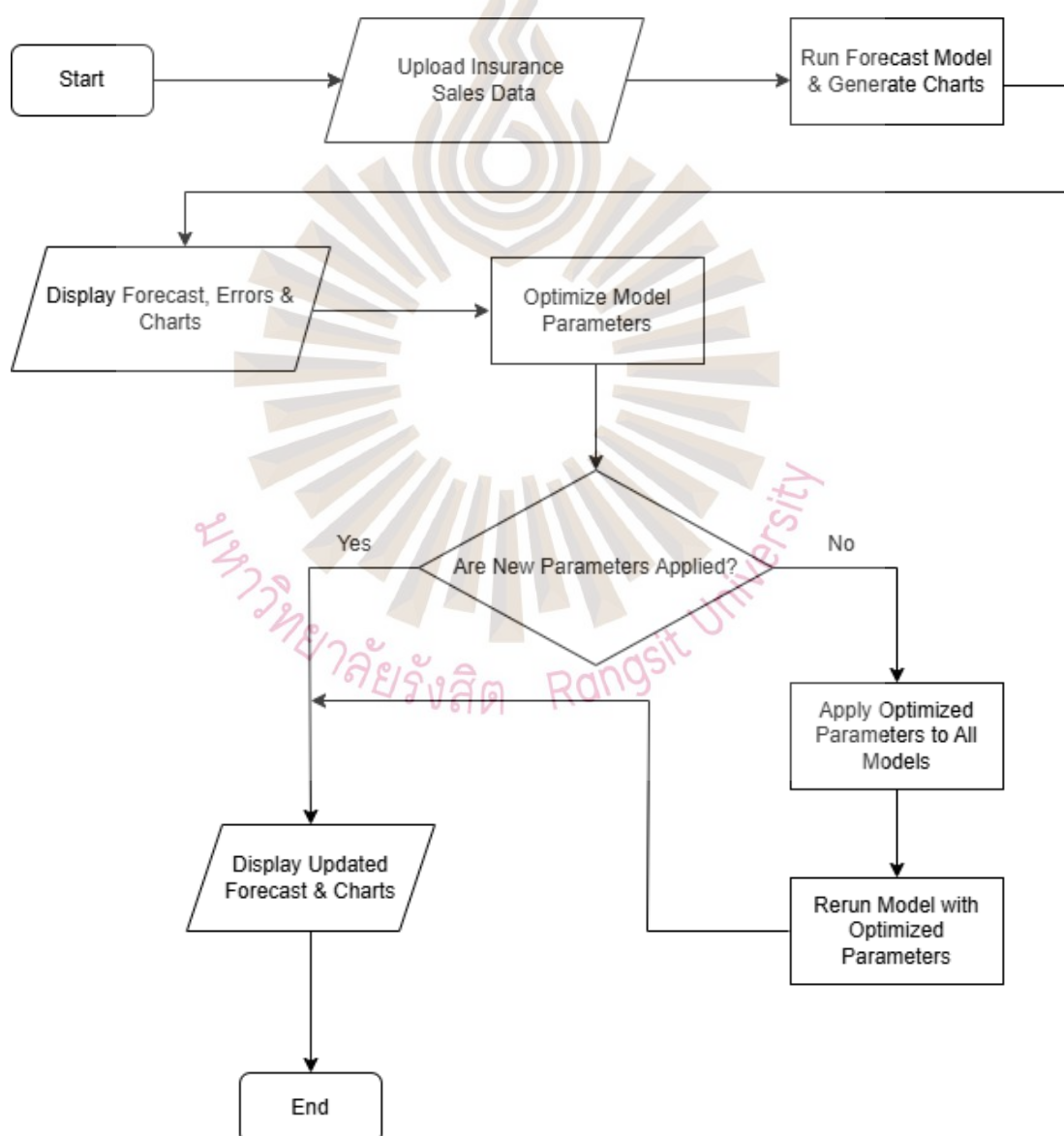
3.6 เปรียบเทียบผลลัพธ์และเลือกวิธีที่ดีที่สุด

การเปรียบเทียบผลลัพธ์ของแต่ละวิธีการพยากรณ์จะเลือกวิธีที่มีค่าความผิดพลาดต่ำสุดจากค่า MAPE และมีความแม่นยำในการพยากรณ์สูงที่สุด โดยวิเคราะห์ข้อมูลทั้ง 4 ชุดแยกกันอย่างอิสระ วิธีที่ให้ค่าความผิดพลาดต่ำสุดในแต่ละชุดจะถูกนำมาเปรียบเทียบและสรุปผล เพื่อเพิ่มประสิทธิภาพในการพยากรณ์ยอดขายประกันชีวิต โดยใช้ภาษา Python และการปรับค่าพารามิเตอร์ (Parameter Optimization) เพื่อเพิ่มความแม่นยำของแบบจำลอง กระบวนการนี้จะทำให้แบบจำลองมีความยืดหยุ่นและสามารถปรับตัวให้เหมาะสมกับลักษณะของข้อมูลในแต่ละกลุ่มได้อย่างมีประสิทธิภาพ และสามารถนำไปประยุกต์ใช้กับข้อมูลจริงในอนาคตได้อย่างแม่นยำ

3.7 การ Optimize ค่าพารามิเตอร์

วิธีในการพยากรณ์ด้วยวิธีการที่ต่างกันสามารถทำได้โดยการหาค่าที่ดีที่สุดของพารามิเตอร์ที่เกี่ยวข้อง โดยใช้ Grid Search หรือวิธีอื่น ๆ ที่สนับสนุนในการหาค่าที่เหมาะสม เพื่อให้ได้ค่าพยากรณ์ที่แม่นยำที่สุด ในที่นี้เราจะใช้ α β และ γ สำหรับการพยากรณ์โดยวิธีต่าง ๆ ดังนี้

ผังงาน (Flow Chart) สำหรับการประมวลผลการพยากรณ์ยอดขายประกันชีวิตในการพยากรณ์ด้วยโปรแกรมภาษา Python แสดงดังรูปที่ 3.3



รูปที่ 3.3 ผังงานสำหรับการประมวลผลการพยากรณ์ยอดขายประกันชีวิตในการพยากรณ์ด้วยโปรแกรมภาษา Python

จากรูปที่ 3.3 ฟังงาน (Flowchart) นี้แสดงขั้นตอนการทำงานของระบบที่ใช้ ภาษา Python บน Google Colab สำหรับการ พยากรณ์ยอดขายประกันภัย โดยใช้ข้อมูลจากไฟล์ CSV พร้อมการ วัดค่าความคลาดเคลื่อนและการสร้างกราฟต่าง ๆ โดยสามารถอธิบายเป็นลำดับขั้นได้ดังนี้

- 1) Start เริ่มต้นกระบวนการพยากรณ์ยอดขายประกันภัย
- 2) Upload Insurance Sales Data ผู้ใช้ทำการ อัปโหลดไฟล์ CSV ที่มีข้อมูลยอดขายประกันภัย
- 3) Run Forecast Model & Generate Charts ระบบจะใช้ Python บน Google Colab เพื่อ พยากรณ์ยอดขาย คำนวณค่าความคลาดเคลื่อน (Error Measurements) และ สร้างกราฟเส้น (Line Chart) และ กราฟแท่ง (Bar Graph)
- 4) Display Forecast, Errors & Charts ระบบจะแสดงผลลัพธ์ สำหรับค่าพยากรณ์ยอดขาย ของปี ค.ศ. 2022 ค่าความคลาดเคลื่อน กราฟเส้น และ กราฟแท่ง
- 5) Optimize Model Parameters ดำเนินการปรับค่าพารามิเตอร์ในโมเดลพยากรณ์เพื่อให้ผล แม่นยำขึ้น
- 6) Are New Parameters Applied? Decision Point โมเดลนำค่าพารามิเตอร์ใหม่มาใช้หรือไม่
 - 6.1) Yes แสดงผลเป็น Display Updated Forecast & Charts ระบบจะแสดง ค่า พยากรณ์รอบใหม่ ค่าความคลาดเคลื่อนใหม่ กราฟเส้น และ กราฟแท่งรอบใหม่
 - 6.2) No ดำเนินการต่อด้วย Apply Optimized Parameters to All Models ระบบจะ บังคับใช้ค่าพารามิเตอร์ที่ได้รับการปรับ กับโมเดลพยากรณ์ทั้งหมด
- Rerun Model with Optimized Parameters ดำเนินการ รันโปรแกรม Python ซ้ำ โดยใช้ค่าพารามิเตอร์ที่ได้รับการปรับแล้ว คำนวณค่าความคลาดเคลื่อนใหม่ และสร้างกราฟใหม่ จากไฟล์ CSV
- Display Updated Forecast & Charts ระบบจะแสดง ค่าพยากรณ์รอบใหม่ ค่า ความคลาดเคลื่อนใหม่ กราฟเส้น และ กราฟแท่งรอบใหม่
- 7) End สิ้นสุดกระบวนการ หลังจากได้ผลลัพธ์ที่ผ่านการปรับค่าพารามิเตอร์และแสดง กราฟผลลัพธ์แล้ว กระบวนการจะสิ้นสุดลงโดยสามารถนำผลลัพธ์ไปใช้ในการวิเคราะห์เชิงธุรกิจ

3.8 โค้ดโปรแกรมสำหรับการพยากรณ์

3.8.1 โค้ดการพยากรณ์ด้วยวิธีการบวกโฮสต์-วินเทอร์

โค้ดการพยากรณ์ด้วยวิธีการบวกโฮสต์-วินเทอร์ มีรายละเอียดดังนี้

```

import numpy as np # นำเข้าไลบรารี Numpy สำหรับการคำนวณเชิงตัวเลข

# ฟังก์ชันคำนวณค่า Mean Squared Error (MSE)
def mse(actual, predicted):
    return np.square(np.subtract(np.array(actual), np.array(predicted))).mean()

# ฟังก์ชันคำนวณค่า Mean Absolute Deviation (MAD)
def mad(y_true, predictions):
    y_true, predictions = np.array(y_true), np.array(predictions)
    return np.mean(np.abs(y_true - predictions))

# ฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE)
def mape(y_test, pred):
    y_test, pred = np.array(y_test), np.array(pred)
    mape = np.mean(np.abs((y_test - pred) / y_test))
    return mape * 100

# ฟังก์ชัน Holt-Winters Additive สำหรับพยากรณ์ข้อมูลอนุกรมเวลา
def HWA(data_input, alpha=0.1, beta=0.1, gamma=0.1, predict_period=1):
    Y = data_input.tolist() # แปลงข้อมูลอินพุตเป็นลิสต์
    E = [] # ค่า Level
    T = [] # ค่า Trend
    S = [] # ค่า Seasonality เริ่มต้น
    countS = 0 # ตัวนับสำหรับฤดูกาล
    Yhat = [] # ลิสต์สำหรับเก็บค่าที่พยากรณ์

```

รูปที่ 3.4 โค้ดการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนฟังก์ชันคำนวณค่า MSE MAD MAPE และฟังก์ชัน Holt-Winters Additive

จากรูปที่ 3.4 โค้ดนี้ใช้ในการวิเคราะห์ความแม่นยำของการพยากรณ์ข้อมูลอนุกรมเวลา และเตรียมใช้โมเดล Holt-Winters Additive สำหรับพยากรณ์ข้อมูลที่มีลักษณะเฉพาะ

ในตอนต้นมีการนำเข้าไลบรารี Numpy ซึ่งทำให้การจัดการและคำนวณข้อมูลเชิงตัวเลขเป็นไปอย่างมีประสิทธิภาพ โดยในโค้ดมีการนิยามฟังก์ชันสำหรับคำนวณค่าความผิดพลาดในการพยากรณ์ 3 สมการ คือ Mean Squared Error (MSE) Mean Absolute Deviation (MAD) และ Mean Absolute Percentage Error (MAPE) ซึ่งแต่ละตัวใช้วิธีวัดความแตกต่างระหว่างค่าจริง (Actual) กับค่าที่โมเดลพยากรณ์ออกมา ด้วยสูตรทางคณิตศาสตร์ที่แตกต่างกัน เพื่อประเมินว่าโมเดลพยากรณ์ได้แม่นยำเพียงใดในเชิงปริมาณ

ถัดไปเริ่มประกาศฟังก์ชัน HWA ซึ่งย่อมาจาก Holt-Winters Additive Model ซึ่งเป็นโมเดลอนุกรมเวลาที่ใช้ส่วนประกอบสามส่วน ได้แก่ Level (ค่าระดับพื้นฐาน) Trend (แนวโน้ม) และ Seasonality (ฤดูกาล/ความเป็นคาบ) สำหรับการพยากรณ์ข้อมูล โดยโมเดลนี้เหมาะกับข้อมูลที่มีแนวโน้มและมีรูปแบบเป็นคาบฤดูกาลแบบ เพิ่มเข้าไป (Additive) เช่น ยอดขายที่เพิ่มขึ้นตามเวลา แต่มีลักษณะวนลูปในแต่ละปี โค้ดในส่วนนี้ยังไม่เสร็จสมบูรณ์ แต่ได้เริ่มต้นกำหนดตัวแปรสำคัญ เช่น E สำหรับ Level T สำหรับ Trend และ S สำหรับ Seasonal Components และ Yhat สำหรับเก็บค่าที่พยากรณ์ในแต่ละช่วงเวลา

โค้ดนี้วางโครงสร้างเบื้องต้นสำหรับการประเมินและเปรียบเทียบความแม่นยำของโมเดลพยากรณ์ โดยเน้นไปที่การใช้งานโมเดล Holt-Winters ในรูปแบบ Additive และวัดผลด้วยตัวชี้วัดมาตรฐานของการพยากรณ์ข้อมูลเชิงเวลา การประเมินจะทำให้เข้าใจถึงประสิทธิภาพของโมเดลในแง่ของการจับแนวโน้มและฤดูกาลของข้อมูล นอกจากนี้ยังสามารถนำไปปรับใช้เพื่อเลือกโมเดลที่มีความแม่นยำสูงสุดตามลักษณะของข้อมูลที่ใช้ในการพยากรณ์ การเลือกโมเดลที่เหมาะสมจะส่งผลต่อความแม่นยำและประสิทธิภาพของการพยากรณ์ในขั้นตอนถัดไป

```
# ลูปผ่านข้อมูล
for idx, value in enumerate(Y):
    if idx < predict_period - 1: # กรณีที่ยังไม่ครบช่วงฤดูกาล
        E.append(None)
        T.append(None)
        S.append(None)
        Yhat.append(None)
    elif idx == predict_period - 1: # คำนวณค่าเริ่มต้นของ Level Trend และ Seasonality
```

รูปที่ 3.5 โค้ดการพยากรณ์ด้วยวิธีการบวกลบ-คูณ ส่วนลูป คำนวณค่าเริ่มต้นของ Level Trend และ Seasonality

```

E.append(np.mean(Y[:idx + 1]))
T.append(0)
S[countS] = Y[countS] - E[countS]
countS += 1
Yhat.append(None)
else:
    # คำนวณค่า Level
    cal_E = (alpha * (Y[idx] - S[idx - predict_period])) + ((1 - alpha) * (E[idx - 1] + T[idx - 1]))
    E.append(cal_E)

    # คำนวณค่า Trend
    cal_T = (beta * (E[idx] - E[idx - 1])) + ((1 - beta) * T[idx - 1])
    T.append(cal_T)

    # คำนวณค่า Seasonality
    if countS < predict_period:
        S[countS] = Y[countS] - E[countS + predict_period - 1]
        countS += 1
        cal_S = (gamma * (Y[idx] - E[idx])) + ((1 - gamma) * S[idx - predict_period])
        S.append(cal_S)

```

รูปที่ 3.5 โค้ดการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนรูป คำนวณค่าเริ่มต้นของ Level Trend และ Seasonality (ต่อ)

จากรูปที่ 3.5 โค้ดในส่วนนี้เป็นแกนหลักของฟังก์ชัน Holt-Winters Additive (HWA) ซึ่งทำหน้าที่วนลูปผ่านข้อมูลอนุกรมเวลา (Time Series) เพื่อคำนวณค่าพยากรณ์ในแต่ละช่วงเวลา โดยมีการประมวลผลข้อมูลออกเป็น 3 ช่วงหลัก ได้แก่ ช่วงเริ่มต้นก่อนมีข้อมูลพอสำหรับวิเคราะห์ฤดูกาล (Predict_Period) ช่วงที่ใช้สำหรับคำนวณค่าเริ่มต้นของระดับ (Level) แนวโน้ม (Trend) และฤดูกาล (Seasonality) และช่วงที่เข้าสู่การคำนวณปกติแบบ Recursive ซึ่งทำให้การพยากรณ์มีความแม่นยำมากขึ้นตามลำดับของข้อมูล

ในช่วงแรกของลูป (ก่อนครบจำนวนรอบฤดูกาล) โค้ดจะใส่ค่า None ลงในลิสต์ต่าง ๆ เพื่อสงวนที่ว่างให้กับค่าที่จะคำนวณภายหลัง เมื่อถึงรอบที่ข้อมูลเพียงพอ ($\text{Idx} == \text{Predict_Period} - 1$) โปรแกรมจะเริ่มต้นคำนวณค่าเริ่มต้นของระดับ (โดยใช้ค่าเฉลี่ยของข้อมูลช่วงต้น) กำหนด Trend เป็น 0 (เพราะยังไม่มีแนวโน้มในช่วงต้น) และคำนวณ Seasonality เป็นส่วนต่างระหว่างค่าจริงกับค่าเฉลี่ย จากนั้นจะเริ่มนับฤดูกาลผ่านตัวแปร CountS

สำหรับช่วงเวลาหลังจากนั้น โค้ดจะเข้าสู่ขั้นตอนการพยากรณ์จริง โดยใช้สมการ Holt-Winters Additive ในการคำนวณค่าระดับ (E) แนวโน้ม (T) และฤดูกาล (S) จากค่าก่อนหน้าและพารามิเตอร์ที่กำหนดไว้ (α, β, γ) โดยค่าระดับจะอิงจากผลรวมของค่าจริงลบ Seasonal Component และค่าพยากรณ์จากรอบก่อนหน้า ส่วนค่าแนวโน้มจะสะท้อนการเปลี่ยนแปลงของระดับเมื่อเทียบกับรอบก่อน ส่วนค่าฤดูกาลจะคำนวณแบบผสมระหว่างค่าจริงกับส่วนประกอบฤดูกาลก่อนหน้า ทั้งหมดนี้เป็นกระบวนการ Recursive ที่ทำให้โมเดลสามารถเรียนรู้รูปแบบจากข้อมูลในอดีตแล้วนำมาพยากรณ์ข้อมูลในอนาคตได้อย่างต่อเนื่อง

โค้ดนี้เป็นขั้นตอนการพยากรณ์ที่ละเอียดและแม่นยำ ซึ่งเหมาะสำหรับข้อมูลที่มีทั้งแนวโน้มและลักษณะเป็นฤดูกาลที่มีผลรวมแบบ Additive

```
# คำนวณค่าที่พยากรณ์ (Forecast)
cal_Yhat = E[idx - 1] + T[idx - 1] + S[idx - predict_period]
Yhat.append(cal_Yhat)

# คำนวณค่าพยากรณ์ล่วงหน้า
lastValueY = Y[len(Y)-1] # ดึงค่าข้อมูล Y สุดท้ายในลิสต์ (ค่าจริงของข้อมูลอนุกรมเวลา)
lastValueE = E[len(E)-1] # ดึงค่า Smoothing Level สุดท้ายในลิสต์ E (ค่าพยากรณ์ระดับ
# สุดท้าย)
lastValueT = T[len(T)-1] # ดึงค่า Trend สุดท้ายในลิสต์ T (ค่าพยากรณ์แนวโน้มสุดท้าย)
lastValueS = S[len(S)-1] # ดึงค่า Seasonal สุดท้ายในลิสต์ S (ค่าฤดูกาลสุดท้าย)
lastValueYhat = Yhat[len(Yhat)-1] # ดึงค่าพยากรณ์สุดท้ายในลิสต์ Yhat (ค่าที่พยากรณ์
# สำหรับอนุกรมเวลาถัดไป)
predictList = [] # สร้างลิสต์ว่างสำหรับเก็บค่าพยากรณ์ในอนาคต
```

รูปที่ 3.6 โค้ดการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนคำนวณค่าที่พยากรณ์ คำนวณค่าที่พยากรณ์ล่วงหน้า และคำนวณค่าความคลาดเคลื่อน (Errors)

```

for i in range(predict_period):
    Y.append(None)
    E.append(None)
    T.append(None)

    cal_predict = lastValueE+((i+1)*lastValueT)+S[len(S)-predict_period]

    predictList.append(cal_predict)

# คำนวณค่าความคลาดเคลื่อน (Errors)
MSE = round(mse(data_input[8:], Yhat[8:len(Yhat)]), 2)
MAD = round(mad(data_input[8:], Yhat[8:len(Yhat)]), 2)
MAPE = round(mape(data_input[8:], Yhat[8:len(Yhat)]), 2)

```

รูปที่ 3.6 โค้ดการพยากรณ์ด้วยวิธีการบอกโฮลท์-วินเทอร์ ส่วนคำนวณค่าที่พยากรณ์ คำนวณค่าที่พยากรณ์ล่วงหน้า และคำนวณค่าความคลาดเคลื่อน (Errors) (ต่อ)

จากรูปที่ 3.6 โค้ดส่วนนี้เป็นการดำเนินการต่อเนื่องจากฟังก์ชัน Holt-Winters Additive (HWA) โดยช่วงแรกเป็นการวนลูปผ่านข้อมูล (For Idx, Value In Enumerate(Y)) เพื่อคำนวณค่าระดับ (Level) แนวโน้ม (Trend) และฤดูกาล (Seasonality) ในแต่ละช่วงเวลา โดยเริ่มจากการกำหนดค่าเริ่มต้นในช่วงที่ข้อมูลยังไม่ครบสำหรับการพยากรณ์ตามฤดูกาล (Predict_Period) และจากนั้นจะเริ่มคำนวณค่าที่จำเป็นทั้งหมดโดยใช้สูตรของ Holt-Winters ซึ่งพิจารณาความเปลี่ยนแปลงในอดีต รวมถึงรูปแบบฤดูกาลแบบบวก (Additive Seasonality)

หลังจากสิ้นสุดการวนลูปเพื่อประมวลผลข้อมูลจริงแล้ว โค้ดจะเข้าสู่การพยากรณ์ล่วงหน้า โดยใช้ค่าล่าสุดของระดับ (E) แนวโน้ม (T) และ Seasonal Component (S) ในการคำนวณค่าที่คาดการณ์ไว้ล่วงหน้าจำนวนเท่ากับ Predict_Period โดยเก็บไว้ใน PredictList ซึ่งทำให้สามารถสร้างการพยากรณ์ได้แม้ไม่มีข้อมูลจริงในช่วงนั้น กระบวนการนี้ใช้ค่าพยากรณ์ที่ได้จากการคำนวณเพื่อสร้างค่าคาดการณ์ที่สามารถนำไปใช้ในการตัดสินใจหรือวางแผน

โค้ดได้คำนวณค่าความคลาดเคลื่อนของผลการพยากรณ์ด้วยตัวชี้วัดมาตรฐาน ได้แก่ MSE (Mean Squared Error) MAD (Mean Absolute Deviation) และ MAPE (Mean Absolute Percentage Error) โดยใช้ข้อมูลจริงกับค่าที่โมเดลพยากรณ์ไว้ในช่วงเวลาที่ข้อมูลเพียงพอ (Index 8 เป็นต้นไป) เพื่อประเมินความแม่นยำของโมเดล Holt-Winters Additive ในการพยากรณ์อนุกรมเวลา โดยค่าทั้งสองจะถูกปัดเศษเป็นทศนิยมสองตำแหน่งเพื่อให้สะดวกต่อการแสดงผลหรือเปรียบเทียบ

```
Yhat.extend(predictList) # รวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์
```

```
return predictList, MSE, MAD, MAPE, Yhat, Y # คืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน
```

รูปที่ 3.7 โค้ดการพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ ส่วนรวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์ และคืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน

จากรูปที่ 3.7 โค้ดส่วนนี้เป็นขั้นตอนสุดท้ายของฟังก์ชัน Holt-Winters Additive โดยทำหน้าที่รวมค่าที่พยากรณ์ล่วงหน้า (PredictList) เข้ากับค่าพยากรณ์ทั้งหมด (Yhat) ผ่านคำสั่ง Yhat.Extend (PredictList) เพื่อให้ผลลัพธ์รวมอยู่ในลิสต์เดียวกัน ซึ่งสะดวกต่อการนำไปแสดงกราฟหรือวิเคราะห์ภาพรวมของการพยากรณ์ทั้งในอดีตและอนาคต จากนั้นฟังก์ชันจะคืนค่าผลลัพธ์ออกมาในรูปแบบหลายตัวแปร ได้แก่ ค่าที่พยากรณ์ล่วงหน้า (PredictList) ค่าความคลาดเคลื่อนแบบ MSE MAD และ MAPE ที่คำนวณไว้ก่อนหน้านี้ ลิสต์ของค่าที่พยากรณ์ทั้งหมด (Yhat) และค่าจริงของข้อมูล (Y) ที่ใช้ในการวิเคราะห์ ซึ่งข้อมูลเหล่านี้สามารถนำไปใช้เพื่อประเมินประสิทธิภาพของโมเดล หรือใช้เปรียบเทียบกับโมเดลอื่นได้อย่างยืดหยุ่นและครบถ้วน

3.8.2 โค้ดการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์

โค้ดการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ดังนี้

```
import numpy as np # นำเข้าไลบรารี Numpy สำหรับการคำนวณเชิงตัวเลข

# ฟังก์ชันคำนวณค่า Mean Squared Error (MSE)
def mse(actual, predicted):
    return np.square(np.subtract(np.array(actual), np.array(predicted))).mean()

# ฟังก์ชันคำนวณค่า Mean Absolute Deviation (MAD)
def mad(y_true, predictions):
    y_true, predictions = np.array(y_true), np.array(predictions)
```

รูปที่ 3.8 โค้ดการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนฟังก์ชันคำนวณค่า MSE MAD MAPE และ ฟังก์ชัน Holt-Winters Multiplicative

```

return np.mean(np.abs(y_true - predictions))

# ฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE)
def mape(y_test, pred):
    y_test, pred = np.array(y_test), np.array(pred)
    mape = np.mean(np.abs((y_test - pred) / y_test))
    return mape * 100

# ฟังก์ชัน Holt-Winters Multiplicative สำหรับพยากรณ์ข้อมูลอนุกรมเวลา
def HWM(data_input, alpha=0.2, beta=0.6, gamma=0.6, predict_period=1):
    Y = data_input.tolist() # แปลงข้อมูลอินพุตเป็นลิสต์
    E = [] # ค่า Level
    T = [] # ค่า Trend
    S = [] # ค่า Seasonality
    countS = 0 # ตัวนับสำหรับฤดูกาล
    Yhat = [] # ลิสต์สำหรับเก็บค่าที่พยากรณ์

```

รูปที่ 3.8 โค้ดการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนฟังก์ชันคำนวณค่า MSE MAD MAPE และ ฟังก์ชัน Holt-Winters Multiplicative (ต่อ)

จากรูปที่ 3.8 โค้ดชุดนี้เริ่มต้นจากการนำเข้าไลบรารี Numpy ซึ่งเป็นเครื่องมือหลักในการจัดการและคำนวณข้อมูลเชิงตัวเลขใน Python ต่อมา มีการนิยามฟังก์ชันสำหรับวัดค่าความคลาดเคลื่อนระหว่างค่าจริงและค่าที่พยากรณ์ ได้แก่ Mse Mad และ Mape ซึ่งเป็นตัวชี้วัดมาตรฐานในงานด้านการพยากรณ์ข้อมูลอนุกรมเวลา (Time Series Forecasting)

ฟังก์ชัน Mse ใช้คำนวณค่า Mean Squared Error ซึ่งวัดความคลาดเคลื่อนโดยเฉลี่ยแบบยกกำลังสอง เพื่อเน้นการลงโทษความผิดพลาดที่มีขนาดใหญ่ ส่วน Mad ใช้คำนวณค่า Mean Absolute Deviation ซึ่งวัดความคลาดเคลื่อนเฉลี่ยแบบสัมบูรณ์ และ Mape ใช้คำนวณค่า Mean Absolute Percentage Error ซึ่งแสดงความคลาดเคลื่อนในรูปแบบร้อยละของค่าจริง ทำให้เข้าใจชัดเจนในการเปรียบเทียบผลพยากรณ์กับค่าจริงในแต่ละช่วงเวลา

ในส่วนของฟังก์ชัน HWM ซึ่งย่อมาจาก Holt-Winters Multiplicative เป็นการเตรียมสำหรับสร้างแบบจำลองพยากรณ์ที่เหมาะสมกับข้อมูลที่มีแนวโน้ม (Trend) และฤดูกาล (Seasonality)

ในลักษณะ แบบคูณ (Multiplicative) ซึ่งหมายถึงอิทธิพลของฤดูกาลจะเปลี่ยนแปลงตามระดับของข้อมูล ตัวแปรภายในเช่น E T และ S จะใช้เก็บค่าระดับ แนวโน้ม และฤดูกาลตามลำดับ ข้อมูลอินพุตจะถูกแปลงเป็นลิสต์เพื่อให้สามารถประมวลผลแบบวนลูปได้สะดวก โดยเตรียมตัวแปร Yhat เพื่อเก็บค่าพยากรณ์ที่ได้จากการคำนวณในแต่ละรอบ ฟังก์ชันนี้ยังไม่สมบูรณ์ในส่วนของการพยากรณ์ แต่โครงสร้างพื้นฐานได้ถูกกำหนดไว้เรียบร้อยแล้วสำหรับการต่อยอด

```
# ลูปผ่านข้อมูล
for idx, value in enumerate(Y):
    if idx < predict_period - 1: # กรณีที่ยังไม่ครบช่วงฤดูกาล
        E.append(None)
        T.append(None)
        S.append(None)
        Yhat.append(None)
    elif idx == predict_period - 1: # คำนวณค่าเริ่มต้นของ Level Trend และ Seasonality
        E.append(np.mean(Y[:idx + 1]))
        T.append(0)
        S.append(None)
        S[countS] = Y[countS] / E[countS] # คำนวณค่า Seasonality เริ่มต้น
        countS += 1
        Yhat.append(None)
    else:

        # คำนวณค่า Level
        cal_E = (alpha * (Y[idx] / S[idx - predict_period])) + ((1 - alpha) * (E[idx - 1] + T[idx - 1]))
        E.append(cal_E)
```

รูปที่ 3.9 โค้ดการพยากรณ์ด้วยวิธีการคูณ โฮลต์-วินเทอร์ ส่วนลูป และ คำนวณค่า Level

จากรูปที่ 3.9 โค้ดนี้เป็นส่วนหนึ่งของฟังก์ชัน Holt-Winters Multiplicative ที่วนลูปผ่านข้อมูลเพื่อตรวจสอบและคำนวณค่าที่จำเป็น โดยใช้ตัวแปรดัชนี Idx เข้าถึงข้อมูลจริง (Y)

ในช่วงต้นของข้อมูล (ก่อนจำนวนรอบฤดูกาลจะครบตามที่กำหนดไว้ด้วย Predict_Period) โปรแกรมจะไม่สามารถคำนวณค่าพยากรณ์ได้ จึงกำหนดค่า None สำหรับลิสต์ที่เก็บค่าต่าง ๆ ได้แก่ Level (E) Trend (T) Seasonality (S) และค่าพยากรณ์ (Yhat) เพื่อเก็บตำแหน่งไว้ชั่วคราวและหลีกเลี่ยงการคำนวณที่ไม่ถูกต้องในระยะเริ่มต้น โดยกระบวนการนี้สามารถป้องกันการเกิดข้อผิดพลาดในระหว่างขั้นตอนการประมวลผลข้อมูล

เมื่อถึงตำแหน่งที่ข้อมูลเริ่มเพียงพอ ($\text{Idx} == \text{Predict_Period} - 1$) โปรแกรมจะเริ่มคำนวณค่าเริ่มต้นของ Level และ Seasonality โดย Level ถูกกำหนดเป็นค่าเฉลี่ยของข้อมูลก่อนหน้านั้น ส่วนค่า Seasonality เริ่มต้นจะถูกคำนวณจากอัตราส่วนระหว่างค่าจริง (Y) กับค่าระดับ (E) ซึ่งเป็นลักษณะของโมเดลแบบคูณ (Multiplicative)

หลังจากนั้นในรอบถัดไป โปรแกรมจะเริ่มคำนวณค่าต่าง ๆ ที่จำเป็นต่อการพยากรณ์ โดยเริ่มจากการคำนวณค่า Level (Cal_E) ด้วยสูตรที่ผสมผสานระหว่างข้อมูลจริงที่ถูกปรับฤดูกาล กับค่าระดับและแนวโน้มก่อนหน้านี้ โดยใช้พารามิเตอร์ Alpha ในการควบคุมน้ำหนักของข้อมูลใหม่ และข้อมูลเก่า ซึ่งเป็นหัวใจของวิธี Holt-Winters ในการคาดการณ์ข้อมูลอนุกรมเวลาที่มีฤดูกาลในเชิงคูณ

```
# คำนวณค่า Trend
cal_T = (beta * (E[idx] - E[idx - 1])) + ((1 - beta) * T[idx - 1])
T.append(cal_T)

# คำนวณค่า Seasonality
if countS < predict_period:
    S[countS] = Y[countS] / E[countS + predict_period - 1]
    countS += 1

cal_S = (gamma * (Y[idx] / E[idx])) + ((1 - gamma) * S[idx - predict_period])
S.append(cal_S)

# คำนวณค่าที่พยากรณ์ (Forecast)
cal_Yhat = (E[idx - 1] + T[idx - 1]) * S[idx - predict_period]
```

รูปที่ 3.10 โค้ดการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนคำนวณค่า Trend Seasonality ค่าที่พยากรณ์ (Forecast) และคำนวณค่าพยากรณ์ล่วงหน้า

```

Yhat.append(cal_Yhat)

# คำนวณค่าพยากรณ์ล่วงหน้า
lastValueY = Y[len(Y)-1] # ดึงค่าข้อมูล Y สุดท้ายในลิสต์ (ค่าจริงของข้อมูลอนุกรมเวลา)
lastValueE = E[len(E)-1] # ดึงค่า Smoothing Level สุดท้ายในลิสต์ E (ค่าพยากรณ์ระดับ
สุดท้าย)
lastValueT = T[len(T)-1] # ดึงค่า Trend สุดท้ายในลิสต์ T (ค่าพยากรณ์แนวโน้มสุดท้าย)
lastValueS = S[len(S)-1] # ดึงค่า Seasonal สุดท้ายในลิสต์ S (ค่าฤดูกาลสุดท้าย)
lastValueYhat = Yhat[len(Yhat)-1] # ดึงค่าพยากรณ์สุดท้ายในลิสต์ Yhat (ค่าที่พยากรณ์
สำหรับอนุกรมเวลาถัดไป)

predictList = [] # สร้างลิสต์ว่างสำหรับเก็บค่าพยากรณ์ในอนาคต
for i in range(predict_period):
    Y.append(None)
    E.append(None)
    T.append(None)
    cal_predict = (lastValueE+((i+1)*lastValueT))*S[len(S)-predict_period]
    predictList.append(cal_predict)

```

รูปที่ 3.10 โค้ดการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ส่วนคำนวณค่า Trend Seasonality ค่าที่พยากรณ์ (Forecast) และคำนวณค่าพยากรณ์ล่วงหน้า (ต่อ)

จากรูปที่ 3.10 โค้ดนี้เป็นส่วนหนึ่งของกระบวนการพยากรณ์ข้อมูลอนุกรมเวลา (Time Series Forecasting) โดยใช้วิธี Triple Exponential Smoothing (Holt-Winters Method) ซึ่งรวมองค์ประกอบของระดับข้อมูล (Level) แนวโน้ม (Trend) และฤดูกาล (Seasonality) เข้าด้วยกันเพื่อพยากรณ์ค่าที่จะเกิดขึ้นในอนาคต

เริ่มจากการคำนวณค่าแนวโน้ม (Trend) ใหม่ในแต่ละรอบโดยอิงจากค่าความแตกต่างของระดับ (Level) ปัจจุบันกับรอบก่อนหน้า และค่าความเรียบของแนวโน้ม (Beta) จากนั้นคำนวณค่าฤดูกาล (Seasonality) โดยเริ่มเก็บข้อมูลฤดูกาลรอบแรกจากการหารค่าจริงกับค่าระดับในรอบที่สอดคล้องกับฤดูกาลก่อนหน้า และคำนวณค่าฤดูกาลใหม่ในรอบถัด ๆ ไปด้วยค่า Gamma เพื่อปรับฤดูกาลให้สอดคล้องกับข้อมูลปัจจุบัน หลังจากนั้นจึงนำค่าระดับ แนวโน้ม และฤดูกาลมาคำนวณค่าที่พยากรณ์ (Forecast) สำหรับรอบนั้น ๆ และเก็บไว้ในลิสต์ Yhat

เมื่อคำนวณค่าจากข้อมูลจริงจบแล้ว จะเข้าสู่ขั้นตอนการพยากรณ์ล่วงหน้า (Forecast Future Values) โดยนำค่าระดับ แนวโน้ม และฤดูกาลล่าสุดที่ได้จากรอบสุดท้าย มาใช้คำนวณค่าที่จะเกิดขึ้นในอนาคตตามจำนวนช่วงเวลาที่กำหนด (Predict_Period) โดยใช้สูตร $(Level + Trend \times \text{จำนวนงวดล่วงหน้า}) \times Seasonality$ ซึ่งค่าฤดูกาลจะดึงจากตำแหน่งที่สัมพันธ์กับรอบฤดูกาลก่อนหน้า จากนั้นนำค่าที่ได้จากการพยากรณ์ในอนาคตเก็บไว้ใน PredictList ซึ่งใช้เป็นผลลัพธ์สุดท้ายของการคาดการณ์

```
# คำนวณค่าความคลาดเคลื่อน (Errors)
MSE = round(mse(data_input[8:], Yhat[8:len(Yhat)]), 2)
MAD = round(mad(data_input[8:], Yhat[8:len(Yhat)]), 2)
MAPE = round(mape(data_input[8:], Yhat[8:len(Yhat)]), 2)

Yhat.extend(predictList) # รวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์
return predictList, MSE, MAD, MAPE, Yhat, Y # คืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน
```

รูปที่ 3.11 โค้ดการพยากรณ์ด้วยวิธีการคูณ Holt-Winters ส่วนคำนวณค่าความคลาดเคลื่อน (Errors) รวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์ และคืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน

จากรูปที่ 3.11 โค้ดส่วนนี้ทำหน้าที่คำนวณค่าความคลาดเคลื่อนของการพยากรณ์เพื่อประเมินความแม่นยำของโมเดล โดยใช้ค่าข้อมูลจริง (Data_Input) เปรียบเทียบกับค่าที่พยากรณ์ได้ (Yhat) ตั้งแต่ตำแหน่งที่ 8 เป็นต้นไป ซึ่งอาจเป็นการหลีกเลี่ยงค่าช่วงเริ่มต้นที่ไม่เสถียรในกระบวนการทำ Smooth หรือการเตรียมค่าฤดูกาล จากนั้นจะคำนวณค่าความคลาดเคลื่อน 3 ค่าหลัก ได้แก่ MSE (Mean Squared Error) คือค่าเฉลี่ยของกำลังสองของความคลาดเคลื่อน MAD (Mean Absolute Deviation) คือค่าเฉลี่ยของความคลาดเคลื่อนสัมบูรณ์ และ MAPE (Mean Absolute Percentage Error) คือเปอร์เซ็นต์ของความคลาดเคลื่อนสัมบูรณ์เทียบกับค่าจริง ซึ่งทั้งหมดนี้ถูกปิดเศษทศนิยมให้เหลือ 2 ตำแหน่งเพื่อความกระชับของผลลัพธ์

หลังจากนั้นจะนำค่าพยากรณ์ล่วงหน้าที่อยู่ใน PredictList มาต่อท้ายใน Yhat เพื่อรวมค่าพยากรณ์ทั้งหมดไว้ด้วยกัน แล้วจึงส่งค่ากลับออกไปเป็นผลลัพธ์ของฟังก์ชัน ได้แก่ ค่าพยากรณ์อนาคต (PredictList) ค่าความคลาดเคลื่อนทั้ง 3 ชนิด (MSE MAD และ MAPE) ค่าพยากรณ์รวม (Yhat) และข้อมูลอนุกรมเวลาทั้งหมด (Y) ที่มีการต่อค่า None สำหรับอนาคตไว้ด้วย โค้ดนี้ยังเอื้อต่อการนำผลลัพธ์ไปใช้ในการวิเคราะห์หรือสร้างกราฟเปรียบเทียบค่าจริงกับค่าพยากรณ์ต่อไป

3.8.3 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ดังนี้

```
import numpy as np # นำเข้าไลบรารี Numpy สำหรับการคำนวณเชิงตัวเลข

# ฟังก์ชันคำนวณค่า Mean Squared Error (MSE)
def mse(actual, predicted):
    return np.square(np.subtract(np.array(actual), np.array(predicted))).mean()

# ฟังก์ชันคำนวณค่า Mean Absolute Deviation (MAD)
def mad(y_true, predictions):
    y_true, predictions = np.array(y_true), np.array(predictions)
    return np.mean(np.abs(y_true - predictions))
```

รูปที่ 3.12 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ส่วน Import Numpy ฟังก์ชันคำนวณค่า Mean Squared Error (MSE) และ ค่า Mean Absolute Deviation (MAD)

จากรูปที่ 3.12 โค้ดส่วนนี้เป็นการกำหนดฟังก์ชันสำหรับคำนวณค่าความคลาดเคลื่อนระหว่างข้อมูลจริงกับค่าที่พยากรณ์ได้ โดยใช้ไลบรารี NumPy ซึ่งทำให้การคำนวณเชิงตัวเลขทำได้มีประสิทธิภาพ ฟังก์ชันแรก Mse ใช้สำหรับคำนวณค่า Mean Squared Error (MSE) ซึ่งเป็นค่าเฉลี่ยของกำลังสองของผลต่างระหว่างค่าจริง (Actual) กับค่าพยากรณ์ (Predicted) โดยค่าที่ได้จะบ่งบอกว่าค่าพยากรณ์เบี่ยงเบนจากค่าจริงมากน้อยเพียงใดในภาพรวม ส่วนฟังก์ชันที่สอง Mad ใช้สำหรับคำนวณค่า Mean Absolute Deviation (MAD) ซึ่งเป็นค่าเฉลี่ยของค่าสัมบูรณ์ของผลต่างระหว่างค่าจริงกับค่าที่พยากรณ์ โดยค่าที่ได้จะเน้นให้ความสำคัญกับความคลาดเคลื่อนโดยไม่คำนึงถึงทิศทาง ทั้งสองฟังก์ชันนี้มักถูกใช้ร่วมกันเพื่อตรวจสอบความแม่นยำของโมเดลการพยากรณ์ในลักษณะที่ต่างกัน นอกจากจะทำให้สามารถวัดความแม่นยำของโมเดลแล้ว ค่าความคลาดเคลื่อนอย่าง MSE และ MAD ยังสามารถนำมาใช้เปรียบเทียบประสิทธิภาพระหว่างโมเดลต่าง ๆ ได้อย่างเป็นระบบ ค่า MSE จะให้ความสำคัญกับความคลาดเคลื่อนที่มีขนาดใหญ่ เนื่องจากการยกกำลังสอง จึงเหมาะสำหรับกรณีที่ต้องการลงโทษความผิดพลาดที่รุนแรง ส่วน MAD นั้นเหมาะกับสถานการณ์ที่ต้องการความยืดหยุ่นและไม่ต้องการให้ความคลาดเคลื่อนสูง

```

# ฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE)
def mape(y_test, pred):
    y_test, pred = np.array(y_test), np.array(pred)
    mape = np.mean(np.abs((y_test - pred) / y_test))
    return mape * 100

# ฟังก์ชัน Simple Exponential Smoothing (SES) สำหรับพยากรณ์ข้อมูลอนุกรมเวลา
def SES(data_input, alpha=0.0, predict_period=1):
    Y = data_input.tolist() # แปลงข้อมูลอินพุตเป็นลิสต์
    Yt = [] # ลิสต์สำหรับเก็บค่าที่พยากรณ์

    # ดูผ่านข้อมูล
    for idx, value in enumerate(Y):
        if idx == 0: # กรณีจุดเริ่มต้น ใช้ค่าจริงเป็นค่าพยากรณ์
            Yt.append(Y[idx])
        else:
            # คำนวณค่าที่พยากรณ์
            cal_Yt = round((alpha * Y[idx - 1]) + ((1 - alpha) * Yt[idx - 1]), 2)
            Yt.append(cal_Yt)

    # คำนวณค่าพยากรณ์ล่วงหน้า
    lastValueY = Y[len(Y)-1] # ค่า Y สุดท้าย
    lastValueYt = Yt[len(Yt)-1] # ค่า Y ที่พยากรณ์สุดท้าย
    predictList = []

    for i in range(predict_period):
        Y.append(None)
        cal_predict = round((alpha * lastValueY) + ((1 - alpha) * lastValueYt), 2)

```

รูปที่ 3.13 โคลการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ส่วนฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE) ฟังก์ชัน Simple Exponential Smoothing (SES) ดูป คำนวณค่าที่พยากรณ์ และคำนวณค่าพยากรณ์ล่วงหน้า

```

if i == 0: # สำหรับช่วงการพยากรณ์ครั้งแรก
    predictList.append(cal_predict)
else: # สำหรับช่วงการพยากรณ์ครั้งถัดไป
    cal_predict = round((alpha * lastValueY) + ((1 - alpha) * predictList[i - 1]), 2)
    predictList.append(cal_predict)

```

รูปที่ 3.13 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ส่วนฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE) ฟังก์ชัน Simple Exponential Smoothing (SES) คูณค่าจำนวนค่าที่พยากรณ์ และคำนวณค่าพยากรณ์ล่วงหน้า (ต่อ)

จากรูปที่ 3.13 โค้ดนี้ประกอบด้วยฟังก์ชันสองส่วนที่ทำงานเกี่ยวกับการพยากรณ์ข้อมูลอนุกรมเวลา โดยเริ่มจากฟังก์ชัน Mape สำหรับคำนวณค่าความคลาดเคลื่อนแบบสัมบูรณ์ในเชิงเปอร์เซ็นต์ หรือ Mean Absolute Percentage Error (MAPE) ซึ่งใช้วัดความแม่นยำของการพยากรณ์ในลักษณะที่แสดงเป็นเปอร์เซ็นต์ของข้อผิดพลาดเมื่อเทียบกับค่าจริง โดยเหมาะสำหรับการเปรียบเทียบผลการพยากรณ์ในชุดข้อมูลที่มีขนาดแตกต่างกัน ส่วนอีกฟังก์ชันคือ SES หรือ Simple Exponential Smoothing เป็นเทคนิคพื้นฐานในการพยากรณ์ข้อมูลอนุกรมเวลา โดยใช้พารามิเตอร์ Alpha เป็นค่าน้ำหนักที่ควบคุมระดับความไวต่อข้อมูลใหม่ ฟังก์ชันจะเริ่มจากการแปลงข้อมูลอินพุตให้อยู่ในรูปของลิสต์ แล้วทำการพยากรณ์ค่าในแต่ละจุดตามสูตร SES ที่ผสมค่าจริงจากรอบก่อนหน้ากับค่าพยากรณ์รอบก่อนหน้า จากนั้นจะคำนวณค่าพยากรณ์ล่วงหน้าในอนาคตตามจำนวนรอบที่ต้องการ (Predict_Period) โดยใช้ค่าจริงสุดท้าย (LastValueY) และค่าพยากรณ์สุดท้าย (LastValueYt) เป็นฐานในการพยากรณ์รอบแรก และใช้ค่าที่พยากรณ์ได้ในรอบก่อนหน้ามาคำนวณรอบถัดไป ทำให้ได้ชุดค่าพยากรณ์อนาคตที่มีลักษณะราบเรียบและสะท้อนแนวโน้มล่าสุดของข้อมูล โดยค่าที่ได้จะถูกเก็บไว้ใน PredictList เพื่อใช้งานต่อหรือแสดงผล

```

# คำนวณค่าความคลาดเคลื่อน (Errors) เริ่มจากข้อมูลจุดที่ 9
MSE = round(mse(data_input[8:], Yt[8:len(Yt)]), 2)
MAD = round(mad(data_input[8:], Yt[8:len(Yt)]), 2)
MAPE = round(mape(data_input[8:], Yt[8:len(Yt)]), 2)

```

รูปที่ 3.14 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ส่วนคำนวณค่าความคลาดเคลื่อน (Errors) เริ่มจากข้อมูลจุดที่ 9

จากรูปที่ 3.14 โค้ดนี้เป็นการคำนวณค่าความคลาดเคลื่อนของโมเดลหรือระบบพยากรณ์ โดยใช้ข้อมูลจริง (Y_t) และค่าที่โมเดลคาดการณ์ไว้ ($Data_Input$) โดยเริ่มพิจารณาตั้งแต่จุดข้อมูลที่ 9 เป็นต้นไป (Index 8 เนื่องจากเริ่มนับจาก 0) การเลือกเริ่มจากจุดที่ 9 อาจเป็นเพราะข้อมูลช่วงต้นยังไม่เสถียรหรือมีการใช้ในการตั้งค่าเริ่มต้นบางอย่าง ค่าความคลาดเคลื่อนที่คำนวณมี 3 ค่า ได้แก่ MSE (Mean Squared Error) ซึ่งเป็นค่าเฉลี่ยของผลต่างระหว่างค่าที่คาดการณ์กับค่าจริงในรูปของกำลังสอง ใช้วัดความรุนแรงของข้อผิดพลาด MAD (Mean Absolute Deviation) ซึ่งเป็นค่าเฉลี่ยของค่าผิดพลาดแบบสัมบูรณ์ ใช้วัดขนาดของข้อผิดพลาดเฉลี่ยอย่างตรงไปตรงมา และ MAPE (Mean Absolute Percentage Error) ซึ่งวัดข้อผิดพลาดในรูปเปอร์เซ็นต์ของค่าจริง เพื่อให้เข้าใจง่ายและสามารถเปรียบเทียบได้ในหลายบริบท ทั้งสามค่าถูกปัดเป็นเลขทศนิยม 2 ตำแหน่งเพื่อความกระชับและสะดวกในการแสดงผลหรือรายงานผลการวิเคราะห์

ค่าความคลาดเคลื่อนเหล่านี้สามารถใช้เป็นเกณฑ์ในการประเมินและเปรียบเทียบประสิทธิภาพของแบบจำลองพยากรณ์ เพื่อเลือกวิธีที่เหมาะสมที่สุดกับลักษณะข้อมูล

3.8.4 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ดังนี้

```
Yt.extend(predictList) # รวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์
return predictList, MSE, MAD, MAPE, Yt, Y # คืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน

# ฟังก์ชันคำนวณค่า Mean Squared Error (MSE)
def mse(actual, predicted):
    return np.square(np.subtract(np.array(actual), np.array(predicted))).mean()

# ฟังก์ชันคำนวณค่า Mean Absolute Deviation (MAD)
def mad(y_true, predictions):
    y_true, predictions = np.array(y_true), np.array(predictions)
    return np.mean(np.abs(y_true - predictions))
```

รูปที่ 3.15 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนรวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์ คืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน และฟังก์ชันคำนวณค่า Mean Squared Error (MSE) และฟังก์ชันคำนวณค่า Mean Absolute Deviation (MAD)

จากรูปที่ 3.15 โค้ดส่วนนี้ทำหน้าที่สรุปผลการพยากรณ์ โดยนำค่าที่คาดการณ์ได้จาก PredictList มาต่อเข้ากับรายการของค่าผลลัพธ์จริง Yt เพื่อให้ข้อมูลสมบูรณ์สำหรับการประเมินผลลัพธ์ในภายหลัง จากนั้นฟังก์ชันจะส่งคืนค่าพยากรณ์ (PredictList) พร้อมกับค่าความคลาดเคลื่อนที่คำนวณได้ ได้แก่ MSE MAD และ MAPE รวมถึงรายการค่าผลลัพธ์จริงที่อัปเดตแล้ว (Yt) และค่าจริงทั้งหมด (Y) เพื่อใช้ในส่วนอื่นของโปรแกรม ฟังก์ชัน Mse() และ Mad() ที่ตามมาทำหน้าที่คำนวณค่าความคลาดเคลื่อนในรูปแบบต่าง ๆ โดย Mse() จะหาค่าเฉลี่ยของผลต่างกำลังสองระหว่างค่าจริงและค่าพยากรณ์ เพื่อเน้นข้อผิดพลาดขนาดใหญ่ ส่วน Mad() จะหาค่าเฉลี่ยของค่าความคลาดเคลื่อนแบบสัมบูรณ์ ซึ่งให้ภาพรวมของขนาดข้อผิดพลาดโดยไม่สนใจทิศทาง ทำให้ทั้งสองฟังก์ชันนี้สนับสนุนในการประเมินความแม่นยำของการพยากรณ์จากมุมมองที่แตกต่างกัน

```
# ฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE)
def mape(y_test, pred):
    y_test, pred = np.array(y_test), np.array(pred)
    mape = np.mean(np.abs((y_test - pred) / y_test))
    return mape * 100

# ฟังก์ชัน Double Exponential Smoothing (DES) สำหรับพยากรณ์ข้อมูลอนุกรมเวลา
def DES(data_input, alpha=0.2, beta=0.1, predict_period=1):
    Y = data_input.tolist() # แปลงข้อมูลอินพุตเป็นลิสต์
    Lt = [] # ลิสต์สำหรับเก็บค่า Level
    Tt = [] # ลิสต์สำหรับเก็บค่า Trend
    Yt_1 = [] # ลิสต์สำหรับเก็บค่าที่พยากรณ์

# ลูปผ่านข้อมูล
for idx, value in enumerate(Y):
    if idx == 0: # กรณีข้อมูลจุดแรก
        Lt.append(None)
```

รูปที่ 3.16 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE) ฟังก์ชัน Double Exponential Smoothing (DES) สำหรับพยากรณ์ข้อมูลอนุกรมเวลา และลูป

```

Tt = [] # ลิสต์สำหรับเก็บค่า Trend
Yt_1 = [] # ลิสต์สำหรับเก็บค่าที่พยากรณ์

# ลูปผ่านข้อมูล
for idx, value in enumerate(Y):
    if idx == 0: # กรณีข้อมูลจุดแรก
        Lt.append(None)
        Tt.append(None)
        Yt_1.append(None)
    elif idx == 1: # กรณีข้อมูลจุดที่สอง กำหนดค่าเริ่มต้นสำหรับ Level และ Trend
        Lt.append(round(Y[idx], 2))
        Tt.append(round(Y[idx] - Y[idx - 1], 2))
        Yt_1.append(None)

```

รูปที่ 3.16 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนฟังก์ชันคำนวณค่า Mean Absolute Percentage Error (MAPE) ฟังก์ชัน Double Exponential Smoothing (DES) สำหรับพยากรณ์ข้อมูลอนุกรมเวลา และลูป (ต่อ)

จากรูปที่ 3.16 โค้ดนี้เริ่มต้นด้วยฟังก์ชัน `Mape()` ซึ่งใช้ในการคำนวณค่า Mean Absolute Percentage Error (MAPE) โดยเป็นการวัดความแม่นยำของค่าที่พยากรณ์ไว้ เปรียบเทียบกับค่าจริงในเชิงเปอร์เซ็นต์ ซึ่งมีประโยชน์มากเมื่อเราต้องการเข้าใจว่าข้อผิดพลาดมีขนาดเท่าใดเมื่อเทียบกับค่าจริง ถัดมาคือการประกาศฟังก์ชัน `DES()` หรือ Double Exponential Smoothing ซึ่งเป็นเทคนิคสำหรับการพยากรณ์ข้อมูลแบบอนุกรมเวลา (Time Series) ที่สามารถจับทั้งค่าระดับ (Level) และแนวโน้ม (Trend) ได้ โดยเริ่มจากการแปลงข้อมูลอินพุตให้อยู่ในรูปแบบลิสต์ แล้วสร้างลิสต์ว่างสำหรับเก็บค่า Level (Lt) Trend (Tt) และค่าที่พยากรณ์ไว้ (Yt_1)

จากนั้นในขั้นตอนการวนลูปเพื่อเตรียมข้อมูล ระบบจะกำหนดให้ข้อมูลจุดแรก (Index 0) มีค่าเป็น None เพราะยังไม่มีข้อมูลก่อนหน้าเพียงพอที่จะใช้คำนวณค่าพยากรณ์ ในขณะที่จุดที่สอง (Index 1) เริ่มมีการคำนวณค่า Level และ Trend เริ่มต้นจากความแตกต่างของสองจุดข้อมูลแรก ซึ่งเป็นพื้นฐานสำคัญในการใช้สูตรของ DES ต่อไปในการพยากรณ์ข้อมูลในอนาคต ค่าที่ได้จะถูกนำมาใช้ในการคำนวณค่าพยากรณ์ในรอบถัดไปอย่างต่อเนื่อง เพื่อให้ผลลัพธ์สอดคล้องกับแนวโน้มที่แท้จริงของชุดข้อมูลที่น่าเข้ามา

```

elif idx == 1: # กรณีข้อมูลจุดที่สอง กำหนดค่าเริ่มต้นสำหรับ Level และ Trend
    Lt.append(round(Y[idx], 2))
    Tt.append(round(Y[idx] - Y[idx - 1], 2))
    Yt_1.append(None)
else:
    # คำนวณค่า Level
    cal_Lt = round((alpha * Y[idx]) + ((1 - alpha) * (Lt[idx - 1] + Tt[idx - 1])), 2)
    Lt.append(cal_Lt)
    # คำนวณค่า Trend
    cal_Tt = round((beta * (Lt[idx] - Lt[idx - 1])) + ((1 - beta) * Tt[idx - 1]), 2)
    Tt.append(cal_Tt)

    # คำนวณค่าที่พยากรณ์
    cal_Yt_1 = round(Lt[idx - 1] + Tt[idx - 1], 2)
    Yt_1.append(cal_Yt_1)

# คำนวณค่าพยากรณ์ล่วงหน้า
lastValueY = Y[len(Y)-1] # ดึงค่าข้อมูล Y สุดท้ายในลิสต์ (ค่าจริงของข้อมูลอนุกรมเวลา)
lastValueLt = Lt[len(Lt)-1] # ดึงค่า Level สุดท้ายในลิสต์ Lt (ค่าพยากรณ์ระดับสุดท้าย)
lastValueTt = Tt[len(Tt)-1] # ดึงค่า Trend สุดท้ายในลิสต์ Tt (ค่าพยากรณ์แนวโน้มสุดท้าย)
lastValueYt_1 = Yt_1[len(Yt_1)-1] # ดึงค่าพยากรณ์สุดท้ายในลิสต์ Yt_1 (ค่าที่พยากรณ์
สำหรับ อนุกรมเวลาถัดไป)

predictList = [] # สร้างลิสต์ว่างสำหรับเก็บค่าพยากรณ์ในอนาคต
for i in range(predict_period):
    Y.append(None)
    Lt.append(None)

```

รูปที่ 3.17 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนกรณีข้อมูลจุดที่สอง กำหนดค่าเริ่มต้นสำหรับ Level Trend คำนวณค่า Level คำนวณค่าที่พยากรณ์ คำนวณค่าพยากรณ์ล่วงหน้า และ สร้างลิสต์ว่างสำหรับเก็บค่าพยากรณ์ในอนาคต

จากรูปที่ 3.17 โค้ดส่วนนี้เป็นการดำเนินการคำนวณการหลักของเทคนิค Double Exponential Smoothing (DES) โดยหลังจากกำหนดค่าเริ่มต้นในช่วงสองจุดแรกของข้อมูลแล้ว สำหรับจุดถัดไป (Index ตั้งแต่ 2 เป็นต้นไป) จะเริ่มมีการคำนวณค่าพยากรณ์โดยอิงจากสมการของ DES โดยเริ่มจากการคำนวณค่า Level (L_t) ซึ่งเป็นค่าที่สะท้อนระดับของข้อมูลในแต่ละช่วงเวลา โดยผสมผสานระหว่างค่าจริง (Y) และค่าที่ได้จากการพยากรณ์ในรอบก่อน ($L_t + T_t$) โดยมีพารามิเตอร์ Alpha ควบคุมน้ำหนัก จากนั้นจะคำนวณค่า Trend (T_t) หรือแนวโน้มของข้อมูล ซึ่งคำนวณจากความแตกต่างของค่า Level ระหว่างรอบก่อนหน้าและรอบปัจจุบัน โดยมี Beta เป็นตัวควบคุมน้ำหนัก แนวโน้ม สุดท้ายจะคำนวณค่าพยากรณ์ (Y_{t+1}) สำหรับรอบนั้น โดยใช้ค่า Level และ Trend ของรอบก่อนหน้า ผลลัพธ์ทั้งหมดจะถูกเก็บไว้ในแต่ละลิสต์สำหรับใช้งานในรอบถัดไป

หลังจากสิ้นสุดการคำนวณบนข้อมูลจริงแล้ว โค้ดจะเตรียมพร้อมสำหรับการพยากรณ์ล่วงหน้า โดยดึงค่าล่าสุดจาก Y L_t T_t และ Y_{t-1} เพื่อใช้เป็นฐานในการสร้างค่าพยากรณ์อนาคต และสร้างลิสต์เปล่า PredictList สำหรับเก็บค่าที่จะพยากรณ์ออกมาในอนาคตตามจำนวนช่วงเวลาที่กำหนด (Predict_Period) พร้อมทั้งเพิ่มค่า None ลงในลิสต์ Y และ L_t เพื่อเตรียมความยาวของข้อมูลให้ตรงกับจำนวนรอบการพยากรณ์ล่วงหน้า ซึ่งเป็นการปูพื้นฐานก่อนเข้าสู่การพยากรณ์ข้อมูลในอนาคต

```
Tt.append(i + 1) # เก็บค่าของช่วงการพยากรณ์
if i == 0:
    cal_predict = round(lastValueLt + lastValueTt, 2)
    predictList.append(cal_predict)
else:
    cal_predict = round(lastValueLt + (Tt[-1] * lastValueTt), 2)
    predictList.append(cal_predict)

# คำนวณค่าความคลาดเคลื่อน (Errors) เริ่มจากข้อมูลจุดที่ 9
MSE = round(mse(data_input[8:], Yt_1[8:len(Yt_1)]), 2)
MAD = round(mad(data_input[8:], Yt_1[8:len(Yt_1)]), 2)
MAPE = round(mape(data_input[8:], Yt_1[8:len(Yt_1)]), 2)
```

รูปที่ 3.18 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนเก็บค่าของช่วงการพยากรณ์ คำนวณค่าความคลาดเคลื่อน (Errors) เริ่มจากข้อมูลจุดที่ 9

```
Yt_1.extend(predictList) # รวมค่าพยากรณ์ล่วงหน้ากับผลลัพธ์
return predictList, MSE, MAD, MAPE, Yt_1, Y # คืนค่าที่พยากรณ์และค่าความคลาดเคลื่อน
```

รูปที่ 3.18 โค้ดการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ส่วนเก็บค่าของช่วงการพยากรณ์ คำนวณค่าความคลาดเคลื่อน (Errors) เริ่มจากข้อมูลจุดที่ 9 (ต่อ)

จากรูปที่ 3.18 โค้ดส่วนนี้เป็นการดำเนินการกระบวนการหลักของเทคนิค Double Exponential Smoothing (DES) โดยหลังจากกำหนดค่าเริ่มต้นในช่วงสองจุดแรกของข้อมูลแล้ว สำหรับจุดถัดไป (Index ตั้งแต่ 2 เป็นต้นไป) จะเริ่มมีการคำนวณค่าพยากรณ์โดยอิงจากสมการของ DES โดยเริ่มจากการคำนวณค่า Level (L_t) ซึ่งเป็นค่าที่สะท้อนระดับของข้อมูลในแต่ละช่วงเวลา โดยผสมผสานระหว่างค่าจริง (Y) และค่าที่ได้จากการพยากรณ์ในรอบก่อน ($L_t + T_t$) โดยมีพารามิเตอร์ Alpha ควบคุมน้ำหนัก จากนั้นจะคำนวณค่า Trend (T_t) หรือแนวโน้มของข้อมูล ซึ่งคำนวณจากความแตกต่างของค่า Level ระหว่างรอบก่อนหน้าและรอบปัจจุบัน โดยมี Beta เป็นตัวควบคุมน้ำหนัก แนวโน้ม สุดท้ายจะคำนวณค่าพยากรณ์ (Yt_1) สำหรับรอบนั้น โดยใช้ค่า Level และ Trend ของรอบก่อนหน้า ผลลัพธ์ทั้งหมดจะถูกเก็บไว้ในแต่ละลิสต์สำหรับใช้งานในรอบถัดไป

หลังจากสิ้นสุดการคำนวณบนข้อมูลจริงแล้ว โค้ดจะเตรียมพร้อมสำหรับการพยากรณ์ล่วงหน้า โดยดึงค่าล่าสุดจาก Y L_t T_t และ Yt_1 เพื่อใช้เป็นฐานในการสร้างค่าพยากรณ์อนาคต และสร้างลิสต์เปล่า PredictList สำหรับเก็บค่าที่จะพยากรณ์ออกมาในอนาคตตามจำนวนช่วงเวลาที่กำหนด (Predict_Period) พร้อมทั้งเพิ่มค่า None ลงในลิสต์ Y และ L_t เพื่อเตรียมความยาวของข้อมูลให้ตรงกับจำนวนรอบการพยากรณ์ล่วงหน้า ซึ่งเป็นการปูพื้นฐานก่อนเข้าสู่การพยากรณ์ข้อมูลในอนาคต

3.9 โค้ดคำนวณความแม่นยำ

จากขั้นตอนที่ 3.8 คำนวณความแม่นยำ สรุปผลและนำเสนอ จะทำการสรุปวิธีที่ดีที่สุดในการพยากรณ์จาก 4 วิธีการที่ใช้ และแสดงผลพยากรณ์ด้วยกราฟแท่งเพื่อการนำเสนอ โดยโค้ดสำหรับคำนวณความแม่นยำ มีรายละเอียดดังนี้

```
import seaborn as sns # นำเข้าไลบรารี seaborn สำหรับการวาดกราฟที่มีความสวยงาม
```

รูปที่ 3.19 นำเข้าไลบรารี Seaborn สำหรับการวาดกราฟ และ นำเข้าไลบรารี Matplotlib สำหรับการวาดกราฟ และสร้างลิสต์ของชื่อวิธีการพยากรณ์

```
import matplotlib.pyplot as plt # นำเข้าไลบรารี matplotlib สำหรับการวาดกราฟ

# สร้างลิสต์ของชื่อวิธีการพยากรณ์
labels = ['Simple Exponential','Double Exponential','Holt Winter Additive','Holt Winter
Multiple']
```

รูปที่ 3.19 นำเข้าไลบรารี Seaborn สำหรับการวาดกราฟ และ นำเข้าไลบรารี Matplotlib สำหรับการวาดกราฟ และสร้างลิสต์ของชื่อวิธีการพยากรณ์ (ต่อ)

จากรูปที่ 3.19 โค้ดนี้เริ่มจากการนำเข้าไลบรารี Seaborn และ Matplotlib.Pyplot ซึ่งเป็นเครื่องมือสำหรับการสร้างกราฟและการแสดงผลข้อมูลในรูปแบบภาพ โดย Seaborn มักถูกใช้เมื่อต้องการสร้างกราฟที่มีความสวยงามและปรับแต่งสะดวก ส่วน Matplotlib.Pyplot เป็นไลบรารีพื้นฐานที่ใช้กันอย่างแพร่หลายสำหรับการวาดกราฟทุกประเภท เช่น เส้น แท่ง จุดกระจาย เป็นต้น จากนั้นมีการสร้างลิสต์ชื่อ Labels ซึ่งเก็บชื่อของเทคนิคการพยากรณ์ข้อมูลแบบอนุกรมเวลาไว้ 4 แบบ ได้แก่ Simple Exponential Smoothing Double Exponential Smoothing Holt-Winter แบบบวก (Additive) และ Holt-Winter แบบคูณ (Multiplicative) โดยลิสต์นี้จะถูกนำไปใช้ภายหลังในการระบุชื่อกราฟหรือใช้เป็น Label บนกราฟเปรียบเทียบประสิทธิภาพของแต่ละวิธีการพยากรณ์ กระบวนการนี้ทำให้การนำเสนอผลการวิเคราะห์มีความชัดเจนและสามารถเปรียบเทียบความแม่นยำของแต่ละเทคนิคได้อย่างมีประสิทธิภาพ

```
# สร้างลิสต์ของค่า Accuracy โดยคำนวณจาก 100 - ค่าความคลาดเคลื่อน (MAPE) ของแต่ละ
วิธีการพยากรณ์
values = [100-SES_MAPE, 100-DES_MAPE, 100-HWA_MAPE, 100-HWM_MAPE]

# สร้างกราฟแท่ง (Bar Plot)
plt.figure(figsize=(9, 6)) # กำหนดขนาดของกราฟ
ax = sns.barplot(x=labels, y=values) # สร้างกราฟแท่งโดยใช้ชื่อวิธีการพยากรณ์เป็นแกน X
และค่า Accuracy เป็นแกน Y
ax.set(title='Accuracy Percentage for Forecasting Methods') # กำหนดชื่อเรื่องของกราฟ
```

รูปที่ 3.20 สร้างลิสต์ของค่า Accuracy สร้างกราฟแท่ง (Bar Plot) เพิ่มข้อความแสดงค่าบนกราฟแท่ง และ แสดงกราฟ

```
# เพิ่มข้อความแสดงค่าบนกราฟแท่ง
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}%', # แสดงค่าความสูงของแต่ละแท่งในรูปแบบเปอร์เซ็นต์
                (p.get_x() + p.get_width() / 2., p.get_height()), # กำหนดตำแหน่งข้อความตรง
                กลางแท่ง
                ha = 'center', va = 'center', # จัดตำแหน่งข้อความให้อยู่กลางแท่ง
                xytext = (0, 9), # กำหนดระยะห่างของข้อความจากจุดที่กำหนดไว้
                textcoords = 'offset points') # ใช้การวัดระยะห่างจากจุดกำหนด
plt.ylabel('Accuracy Value') # กำหนดชื่อแกน Y เป็น 'Accuracy Value'
plt.show() # แสดงกราฟ
```

รูปที่ 3.20 สร้างลิสต์ของค่า Accuracy สร้างกราฟแท่ง (Bar Plot) เพิ่มข้อความแสดงค่าบนกราฟแท่ง และ แสดงกราฟ (ต่อ)

จากรูปที่ 3.20 โค้ดส่วนนี้มีหน้าที่สร้างกราฟแท่งเพื่อเปรียบเทียบประสิทธิภาพของวิธีการพยากรณ์ต่าง ๆ โดยเริ่มจากการสร้างลิสต์ Values ซึ่งเก็บค่าความแม่นยำ (Accuracy) ของแต่ละวิธี โดยคำนวณจากสูตร $100 - \text{MAPE}$ เนื่องจาก MAPE เป็นค่าความคลาดเคลื่อนที่ยิ่งน้อยยิ่งดี จึงสามารถแปลงเป็นค่าความแม่นยำได้ด้วยการลบออกจาก 100 เพื่อให้เห็นภาพได้ชัดเจนมากขึ้น จากนั้นใช้ Seaborn ในการสร้างกราฟแท่ง โดยกำหนดให้แกน X แสดงชื่อของวิธีการพยากรณ์ และ แกน Y แสดงค่าความแม่นยำที่ได้ พร้อมกำหนดชื่อหัวข้อของกราฟเป็น “Accuracy Percentage for Forecasting Methods” เพื่อให้ผู้ใช้เข้าใจได้ชัดเจนขึ้นว่ากำลังเปรียบเทียบอะไร สำหรับการเพิ่มความชัดเจนในกราฟ มีการใส่ข้อความแสดงค่าความแม่นยำ (ในรูปแบบเปอร์เซ็นต์) บนแต่ละแท่งของกราฟ เพื่อให้สามารถอ่านค่าได้อย่างรวดเร็วโดยไม่ต้องกะด้วยสายตา สุดท้ายสั่งให้แสดงกราฟด้วย `Plt.Show()` ซึ่งเป็นคำสั่งแสดงผลภาพที่สร้างขึ้นจริงในหน้าต่างหรือเอด์พุตของโปรแกรม

3.10 โค้ดการ Optimize ค่าพารามิเตอร์

จากขั้นตอนที่ 3.9 วิธีในการพยากรณ์ด้วยวิธีการที่ต่างกันสามารถทำได้โดยการหาค่าที่ดีที่สุดของพารามิเตอร์ที่เกี่ยวข้อง โดยใช้ Grid Search หรือวิธีอื่น ๆ ที่ทำให้สามารถหาค่าที่เหมาะสมเพื่อให้ได้ค่าพยากรณ์ที่แม่นยำที่สุด ในที่นี้จะใช้ Nested Loop เพื่อวนลูปทดลองค่าต่าง ๆ ของ α β และ γ ที่เป็นไปได้ จากนั้นเก็บค่าของพารามิเตอร์แต่ละชุดไว้ใน Dictionary (DicMSE DicMAD

และ DicMAPE) ต่อไปเลือกค่าที่ดีที่สุดพร้อมกับค่าความผิดพลาดที่คำนวณได้ และเลือกค่าที่ดีที่สุดโดยหาค่า Minimum ของ MSE MAD MAPE และบันทึกพารามิเตอร์ที่ให้ค่าต่ำสุด

การแสดงผลใช้กราฟ 3D Scatter Plot เพื่อแสดงความสัมพันธ์ระหว่างพารามิเตอร์ (α β γ) และค่าความผิดพลาด (MSE) สำหรับการพยากรณ์โดยวิธีต่าง ๆ ดังนี้ ทั้งนี้ การวิเคราะห์ลักษณะของกราฟยังทำให้เห็นแนวโน้มของผลลัพธ์เมื่อมีการเปลี่ยนแปลงค่าพารามิเตอร์ ทำให้สะดวกต่อการตัดสินใจเลือกช่วงค่าที่เหมาะสมในการพยากรณ์ในอนาคต และสามารถปรับปรุงแบบจำลองให้มีความเสถียรและแม่นยำมากยิ่งขึ้น

3.10.1 โค้ดการ Optimize ค่าพารามิเตอร์วิธีการบวกลบ-วินเทอร์

ในวิธีนี้ จะใช้การพยากรณ์ที่มีแนวโน้มและฤดูกาลแบบ Additive โดยใช้พารามิเตอร์

- 1) α (Smoothing Level): ค่าการทำให้ราบเรียบของระดับข้อมูล
- 2) β (Smoothing Slope): ค่าการทำให้ราบเรียบของแนวโน้ม
- 3) γ (Smoothing Seasonal): ค่าการทำให้ราบเรียบของฤดูกาล

โค้ดในการ Optimize ดังนี้

```
minMSE, minMAD, minMAPE, dicMSE, dicMAD, dicMAPE =
errOptHW_nastedLoop(data['Sales'], HWA, 4) # เรียกใช้ฟังก์ชัน 'ErrOptHW_nastedLoop'
เพื่อหาค่า MSE, MAD, MAPE ที่น้อยที่สุด และเก็บค่าที่ได้ไว้ในตัวแปรต่าง ๆ
print("Smallest MSE: ", minMSE) # แสดงผลลัพธ์ของค่า MSE ที่น้อยที่สุด
print("Smallest MAD: ", minMAD) # แสดงผลลัพธ์ของค่า MAD ที่น้อยที่สุด
print("Smallest MAPE: ", minMAPE) # แสดงผลลัพธ์ของค่า MAPE ที่น้อยที่สุด

fig_MSE = plt.figure(figsize=(10, 10), dpi=100) # สร้างกราฟ 3 มิติ โดยกำหนดขนาด 10x10
นิ้วและความละเอียด 100 DPI
ax_MSE = fig_MSE.add_subplot(111, projection='3d') # เพิ่ม Subplot สำหรับกราฟ 3 มิติ
ax_MSE.set_xlabel('Alpha') # กำหนดชื่อแกน X เป็น 'Alpha'
ax_MSE.set_ylabel('Beta') # กำหนดชื่อแกน Y เป็น 'Beta'
ax_MSE.set_zlabel('gamma') # กำหนดชื่อแกน Z เป็น 'Gamma'
```

รูปที่ 3.21 เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับ Optimize ค่าพารามิเตอร์วิธีการบวกลบ-วินเทอร์ และสร้างกราฟ 3 มิติ

จากรูปที่ 3.21 โค้ดนี้เริ่มจากการเรียกใช้ฟังก์ชัน `ErrOptHW_NastedLoop()` เพื่อค้นหาค่าความคลาดเคลื่อนที่น้อยที่สุด ได้แก่ MSE (Mean Squared Error) MAD (Mean Absolute Deviation) และ MAPE (Mean Absolute Percentage Error) สำหรับการพยากรณ์ด้วยวิธี Holt-Winter แบบ Additive โดยใช้ข้อมูลยอดขายจาก `Data['Sales']` และกำหนดพารามิเตอร์การพยากรณ์ไว้ล่วงหน้า 4 ช่วงเวลา ค่าที่น้อยที่สุดแต่ละประเภทจะถูกเก็บไว้ในตัวแปร `MinMSE` `MinMAD` `MinMAPE` และแสดงออกทางหน้าจอด้วยคำสั่ง `Print()` ซึ่งประเมินว่าพารามิเตอร์แบบใดให้ผลพยากรณ์แม่นยำ

จากนั้นโค้ดเตรียมสร้างกราฟแบบ 3 มิติ โดยใช้ `Matplotlib` เพื่อแสดงความสัมพันธ์ระหว่างพารามิเตอร์ที่ใช้ในโมเดล Holt-Winter (เช่น Alpha Beta และ Gamma) กับค่าความคลาดเคลื่อน ซึ่งทำให้วิเคราะห์เชิงภาพได้ว่าค่าพารามิเตอร์ใดส่งผลกระทบต่อประสิทธิภาพของโมเดลมากที่สุด โดยกำหนดขนาดกราฟเป็น 10x10 นิ้ว และความละเอียด 100 DPI พร้อมตั้งชื่อแกน X Y และ Z ตามพารามิเตอร์ทั้งสาม เพื่อให้กราฟแสดงผลได้เข้าใจชัดเจนและเป็นประโยชน์ต่อการวิเคราะห์ค่าที่เหมาะสมในกระบวนการปรับแต่งโมเดลพยากรณ์ นอกจากนี้ ยังสามารถใช้กราฟดังกล่าวเปรียบเทียบลักษณะของพื้นผิวความผิดพลาด เพื่อค้นหาพื้นที่ที่ค่าพยากรณ์มีเสถียรภาพสูง ซึ่งส่งเสริมความมั่นใจในการเลือกพารามิเตอร์เชิงปฏิบัติ

```
# แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ
x_MSE = [t[0] for t in dicMSE.keys()] # แยกค่าของ Alpha
y_MSE = [t[1] for t in dicMSE.keys()] # แยกค่าของ Beta
z_MSE = [t[2] for t in dicMSE.keys()] # แยกค่าของ Gamma
c_MSE = list(dicMSE.values()) # แยกค่าของ MSE

img_MSE = ax_MSE.scatter(x_MSE, y_MSE, z_MSE, c=c_MSE, cmap=plt.hot()) # สร้าง
กราฟแบบ Scatter Plot ในรูปแบบ 3 มิติ แสดงค่าของ MSE โดยใช้สีเป็นตัวบ่งชี้ (Cmap ใช้โทน
สี 'hot')

fig_MSE.colorbar(img_MSE, fraction=0.025, pad=0.2) # เพิ่มแถบสี (Colorbar) เพื่อแสดง
ระดับของค่า MSE ในกราฟ

plt.show() # แสดงกราฟ
```

รูปที่ 3.22 แยกค่าของ Alpha Beta Gamma ค่า MSE สำหรับใช้ในการพล็อตกราฟ สร้างกราฟแบบ Scatter Plot ในรูปแบบ 3 มิติ และเพิ่มแถบสี (Colorbar)

จากรูปที่ 3.22 โค้ดส่วนนี้เป็นการเตรียมข้อมูลและแสดงผลกราฟ 3 มิติแบบกระจาย (3D Scatter Plot) เพื่อวิเคราะห์ผลลัพธ์ของค่า MSE จากการปรับค่าพารามิเตอร์ในโมเดล Holt-Winter โดยเริ่มจากการแยกข้อมูลพารามิเตอร์ Alpha Beta และ Gamma ออกจาก Key ของ DicMSE ซึ่งเป็น Dictionary ที่เก็บคู่พารามิเตอร์และค่าความคลาดเคลื่อน (MSE) ไว้ โดยค่าทั้งสามถูกจัดเก็บในลิสต์ X_MSE Y_MSE และ Z_MSE ตามลำดับ ส่วนค่าของ MSE จะถูกแยกเก็บในลิสต์ C_MSE เพื่อใช้กำหนดสีของจุดในกราฟ โดยการพล็อตใช้ฟังก์ชัน Scatter() ของกราฟ 3 มิติที่สร้างไว้ก่อนหน้านี้ พร้อมกำหนดแผนที่สี (Colormap) เป็น 'Hot' เพื่อแสดงความเข้มของสีตามระดับของ MSE (ยิ่งค่ามาก สีจะยิ่งร้อนแรงขึ้น) ทำให้สามารถมองเห็นได้อย่างชัดเจนว่าค่า MSE จะน้อยที่สุดที่จุดพารามิเตอร์ใด สุดท้ายมีการเพิ่มแถบแสดงระดับสี (Color Bar) เพื่อให้ผู้ใช้สามารถอ่านค่าระดับความคลาดเคลื่อนจากสีในกราฟได้ชัดเจนขึ้น และปิดท้ายด้วย Plt.Show() เพื่อแสดงผลกราฟออกมาให้ผู้ใช้งาน

3.10.2 โค้ดการ Optimize ค่าพารามิเตอร์วิธีการคูณโฮลท์-วินเทอร์

โมเดลนี้คล้ายกับ Additive แต่ต่างกันที่ฤดูกาลจะใช้แบบ Multiplicative

- 1) α (Smoothing Level)
- 2) β (Smoothing Slope)
- 3) γ (Smoothing Seasonal)

โค้ดในการ Optimize ดังนี้

```
minMSE, minMAD, minMAPE, dicMSE, dicMAD, dicMAPE =
errOptHW_nastedLoop(data['Sales'], HWM, 4) # เรียกใช้ฟังก์ชัน 'ErrOptHW_nastedLoop'
เพื่อหาค่า MSE, MAD, MAPE ที่น้อยที่สุดสำหรับ โมเดล HWM และเก็บค่าที่ได้ไว้ในตัวแปร
ต่าง ๆ
```

รูปที่ 3.23 เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับ Optimize ค่าพารามิเตอร์วิธีการคูณโฮลท์วินเทอร์

จากรูปที่ 3.23 โค้ดบรรทัดนี้เป็นการเรียกใช้ฟังก์ชันชื่อ ErrOptHW_NastedLoop ซึ่งมีหน้าที่ในการคำนวณค่าความคลาดเคลื่อนของการพยากรณ์จากโมเดล Holt-Winters Multiplicative (HWM) โดยจะประเมินค่าความผิดพลาดด้วยเกณฑ์สามแบบ คือ MSE (Mean Squared Error) MAD (Mean Absolute Deviation) และ MAPE (Mean Absolute Percentage Error) เพื่อหาค่าที่น้อยที่สุด

จากการวนลูปทดสอบค่าพารามิเตอร์ต่าง ๆ ของโมเดล โดยในที่นี้ฟังก์ชันจะรับข้อมูลยอดขายจาก Data['Sales'] และค่าที่ระบุจะให้พยากรณ์ล่วงหน้า 4 ช่วงเวลา (เช่น 4 เดือน หรือ 4 ไตรมาส ขึ้นอยู่กับบริบทของข้อมูล) จากนั้นฟังก์ชันจะคืนค่ากลับมาเป็นค่า MSE MAD และ MAPE ที่น้อยที่สุด (เก็บในตัวแปร MinMSE MinMAD MinMAPE) พร้อมกับข้อมูลรายละเอียดของการคำนวณที่เป็นลักษณะของพจนานุกรม (Dictionary) ซึ่งเก็บไว้ในตัวแปร DicMSE DicMAD และ DicMAPE ตามลำดับ เพื่อใช้ในการวิเคราะห์เพิ่มเติมหรือแสดงผลต่อไป

```
print("Smallest MSE: ", minMSE) # แสดงผลลัพธ์ของค่า MSE ที่น้อยที่สุด
print("Smallest MAD: ", minMAD) # แสดงผลลัพธ์ของค่า MAD ที่น้อยที่สุด
print("Smallest MAPE: ", minMAPE) # แสดงผลลัพธ์ของค่า MAPE ที่น้อยที่สุด

fig_MSE = plt.figure(figsize=(10, 10), dpi=100) # สร้างกราฟ 3 มิติ โดยกำหนดขนาด 10x10 นิ้วและความละเอียด 100 DPI
ax_MSE = fig_MSE.add_subplot(111, projection='3d') # เพิ่ม Subplot สำหรับกราฟ 3 มิติ
ax_MSE.set_xlabel('Alpha') # กำหนดชื่อแกน X เป็น 'Alpha'
ax_MSE.set_ylabel('Beta') # กำหนดชื่อแกน Y เป็น 'Beta'
ax_MSE.set_zlabel('gamma') # กำหนดชื่อแกน Z เป็น 'Gamma'

# แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ
x_MSE = [t[0] for t in dicMSE.keys()] # แยกค่าของ Alpha
y_MSE = [t[1] for t in dicMSE.keys()] # แยกค่าของ Beta
z_MSE = [t[2] for t in dicMSE.keys()] # แยกค่าของ Gamma
c_MSE = list(dicMSE.values()) # แยกค่าของ MSE

img_MSE = ax_MSE.scatter(x_MSE, y_MSE, z_MSE, c=c_MSE, cmap=plt.hot()) # สร้าง
กราฟแบบ Scatter Plot ในรูปแบบ 3 มิติ แสดงค่าของ MSE โดยใช้สีเป็นตัวบ่งชี้ (Cmap ใช้โทน
สี 'Hot')
```

รูปที่ 3.24 แสดงผลลัพธ์ของค่า MSE MAD และ MAPE ที่น้อยที่สุด สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ และเพิ่มแถบสี (Colorbar)

```
fig_MSE.colorbar(img_MSE, fraction=0.025, pad=0.2) # เพิ่มแถบสี (Colorbar) เพื่อแสดง
ระดับของค่า MSE ในกราฟ
plt.show() # แสดงกราฟ
```

รูปที่ 3.24 แสดงผลลัพธ์ของค่า MSE MAD และ MAPE ที่น้อยที่สุด สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ และเพิ่มแถบสี (Colorbar) (ต่อ)

จากรูปที่ 3.24 โค้ดนี้เริ่มต้นด้วยการแสดงค่าความคลาดเคลื่อนที่น้อยที่สุดจากผลการวิเคราะห์ของโมเดล Holt-Winters โดยพิมพ์ค่า MSE MAD และ MAPE ที่ได้จากการปรับพารามิเตอร์ในฟังก์ชันก่อนหน้านี้ จากนั้นจึงสร้างกราฟ 3 มิติเพื่อแสดงความสัมพันธ์ระหว่างพารามิเตอร์ทั้งสามของโมเดล คือ Alpha Beta และ Gamma กับค่า MSE ที่ได้ โดยการใช้การกระจายจุด (Scatter Plot) ในรูปแบบสามมิติ ซึ่งสีของจุดจะแสดงระดับค่าของ MSE ด้วยแถบสีในโทนร้อน (Hot Colormap) เพื่อให้สามารถมองเห็นภาพรวมได้อย่างชัดเจนว่าชุดของพารามิเตอร์ใดให้ค่า MSE ต่ำสุดหรือสูงขึ้น การแสดงผลนี้เอื้อต่อการวิเคราะห์เชิงลึกและการตัดสินใจเลือกพารามิเตอร์ที่เหมาะสมสำหรับการพยากรณ์ในระยะถัดไป โดยกราฟจะแสดงในหน้าต่างใหม่ผ่าน Plt.Show() พร้อมทั้งมีแถบสีแสดงค่าความเข้มของ MSE เพื่อเสริมความเข้าใจเชิงภาพมากยิ่งขึ้น

3.10.3 โค้ดการ Optimize ค่าวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

การปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย วิธีนี้จะใช้เฉพาะพารามิเตอร์ α (Smoothing Level) สำหรับการทำให้ข้อมูลราบเรียบ โดยโค้ดในการ Optimize ดังนี้

```
minMSE, minMAD, minMAPE, dicMSE, dicMAD, dicMAPE =
errOptHW_nastedLoop(data['Sales'], SES, 4) # เพื่อหาค่า MSE, MAD และ MAPE ที่น้อยที่สุด
MinMSE, MinMAD, MinMAPE, DicMSE, DicMAD, DicMAPE =
errOptHW_nastedLoop(data['Sales'], SES, 4) # เพื่อหาค่า MSE MAD และ MAPE ที่น้อยที่สุด
สำหรับโมเดล SES และเก็บค่าที่ได้ไว้ในตัวแปรต่าง ๆ
print("Smallest MSE: ", minMSE) # แสดงผลลัพธ์ของค่า MSE ที่น้อยที่สุด
```

รูปที่ 3.25 เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับ Optimize ค่าพารามิเตอร์วิธีการคูณไฮลท์-วินเทอร์ สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ

```

print("Smallest MAD: ", minMAD) # แสดงผลลัพธ์ของค่า MAD ที่น้อยที่สุด
print("Smallest MAPE: ", minMAPE) # แสดงผลลัพธ์ของค่า MAPE ที่น้อยที่สุด

fig_MSE = plt.figure(figsize=(10, 10), dpi=100) # สร้างกราฟ 3 มิติ โดยกำหนดขนาด 10x10
นิ้วและความละเอียด 100 DPI

fig_MSE = plt.figure(figsize=(10, 10), dpi=100) # สร้างกราฟ 3 มิติ โดยกำหนดขนาด 10x10
นิ้วและความละเอียด 100 DPI

ax_MSE.set_xlabel('Alpha') # กำหนดชื่อแกน X เป็น 'Alpha'
ax_MSE = fig_MSE.add_subplot(111, projection='3d') # เพิ่ม Subplot สำหรับกราฟ 3 มิติ
ax_MSE.set_ylabel('Beta') # กำหนดชื่อแกน Y เป็น 'Beta'
ax_MSE.set_zlabel('gamma') # กำหนดชื่อแกน Z เป็น 'Gamma'

# แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ
x_MSE = [t[0] for t in dicMSE.keys()] # แยกค่าของ Alpha
y_MSE = [t[1] for t in dicMSE.keys()] # แยกค่าของ Beta
z_MSE = [t[2] for t in dicMSE.keys()] # แยกค่าของ Gamma
c_MSE = list(dicMSE.values()) # แยกค่าของ MSE

img_MSE = ax_MSE.scatter(x_MSE, y_MSE, z_MSE, c=c_MSE, cmap=plt.hot()) # สร้าง
กราฟแบบ Scatter Plot ในรูปแบบ 3 มิติ แสดงค่าของ MSE โดยใช้สีเป็นตัวบ่งชี้ (Cmap ใช้โทน
สี 'Hot')

```

รูปที่ 3.25 เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับ Optimize ค่าพารามิเตอร์วิธีการคูณ
 โฮลท์-วินเทอร์ สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อ
 ใช้ในการพล็อตกราฟ (ต่อ)

จากรูปที่ 3.25 โค้ดนี้มีจุดประสงค์เพื่อหาค่าความผิดพลาดต่ำสุดของโมเดล Simple
 Exponential Smoothing (SES) โดยใช้ฟังก์ชัน ErrOptHW_NastedLoop ซึ่งจะประเมินค่าความ
 คลาดเคลื่อนของการพยากรณ์ในรูปแบบ MSE MAD และ MAPE โดยค่าที่ได้จะถูกจัดเก็บไว้ทั้งใน
 ตัวแปรเดี่ยว (เช่น MinMSE) สำหรับค่าต่ำสุด และในพจนานุกรม (เช่น DicMSE) สำหรับค่าทั้งหมด
 ที่ได้จากการทดสอบค่าพารามิเตอร์ต่าง ๆ แม้จะมีการเรียกฟังก์ชันซ้ำกันอยู่สองครั้ง

โดยใช้ชื่อชุดตัวแปรต่างกันสำหรับ MinMSE ซึ่งอาจเป็นความซ้ำซ้อนที่ไม่จำเป็นก็ตาม จากนั้นจะแสดงผลลัพธ์ค่าที่ต่ำที่สุดของ MSE MAD และ MAPE ออกทางหน้าจอ เพื่อให้ผู้ใช้ทราบผลลัพธ์เชิงตัวเลข

ต่อมาเป็นการสร้างกราฟแบบ 3 มิติเพื่อแสดงความสัมพันธ์ระหว่างค่าพารามิเตอร์ Alpha Beta และ Gamma กับค่า MSE ที่ได้จากการทดสอบ โดยใช้กราฟแบบ Scatter Plot ซึ่งจุดแต่ละจุดในกราฟจะแสดงตำแหน่งของชุดพารามิเตอร์ และใช้สีเพื่อแสดงระดับของค่า MSE ที่สัมพันธ์กับจุดนั้น การใช้ Colormap แบบ “Hot” ทำให้สามารถมองเห็นระดับความผิดพลาดจากสีได้อย่างชัดเจน อย่างไรก็ตาม โค้ดมีความซ้ำซ้อนบางส่วน เช่น การสร้าง Fig_MSE ซ้ำโดยไม่จำเป็น และการตั้งชื่อแกนก่อนการสร้าง Subplot ซึ่งอาจทำให้เกิด Error ได้หากยังไม่มี Ax_MSE ถูกสร้างไว้ก่อน การจัดระเบียบโค้ดในส่วนนี้จึงสามารถปรับปรุงให้กระชับและถูกต้องมากยิ่งขึ้น

```
fig_MSE.colorbar(img_MSE, fraction=0.025, pad=0.2) # เพิ่มแถบสี (Colorbar) เพื่อแสดง
ระดับของค่า MSE ในกราฟ
plt.show() # แสดงกราฟ
```

รูปที่ 3.26 เพิ่มแถบสี (Colorbar) เพื่อแสดงระดับของค่า MSE ในกราฟ และแสดงกราฟ

จากรูปที่ 3.26 โค้ดส่วนนี้ทำหน้าที่จัดการการแสดงผลกราฟ 3 มิติที่สร้างขึ้นก่อนหน้านี้ โดยเพิ่มแถบสี (Colorbar) ลงในกราฟเพื่อแสดงระดับค่าของ MSE ในการไล่โทนสีของจุดในกราฟ Scatter Plot ซึ่งทำให้ผู้ใช้สามารถแยกแยะค่าความคลาดเคลื่อนจากสีที่ปรากฏ การตั้งค่า Fraction และ Pad จะปรับขนาดและระยะห่างของแถบสีจากตัวกราฟให้เหมาะสม สุดท้าย Plt.Show() จะทำการแสดงกราฟทั้งหมดในหน้าต่างกราฟิก เพื่อให้ผู้ใช้สามารถตรวจสอบและวิเคราะห์ผลลัพธ์ได้อย่างชัดเจนและมีประสิทธิภาพ

3.10.4 โค้ดการ Optimize ค่าวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

การปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า โมเดลนี้เหมาะสำหรับข้อมูลอนุกรมเวลาที่มีแนวโน้มชัดเจน โดยสามารถติดตามการเปลี่ยนแปลงของข้อมูลทั้งระดับค่าและทิศทางของแนวโน้ม โดยใช้พารามิเตอร์ α (Smoothing Level) และ β (Smoothing Slope) เพื่อค้นหาค่าที่ให้ค่าความคลาดเคลื่อนต่ำที่สุด โดยพิจารณาค่าความคลาดเคลื่อน เช่น Mean Squared Error (MSE) ในการประเมินประสิทธิภาพ ผลลัพธ์ที่ได้ทำให้เพิ่มความแม่นยำในการพยากรณ์ของโมเดล Double Exponential Smoothing (DES) สำหรับช่วงข้อมูลในระยะถัดไป โค้ดในการ Optimize ดังนี้

```

minMSE, minMAD, minMAPE, dicMSE, dicMAD, dicMAPE =
errOptHW_nastedLoop(data['Sales'], DES, 4) # เรียกใช้ฟังก์ชัน 'ErrOptHW_nastedLoop'
เพื่อหาค่า MSE, MAD และ MAPE ที่น้อยที่สุดสำหรับโมเดล DES และเก็บค่าที่ได้ไว้ในตัวแปร
ต่าง ๆ

print("Smallest MSE: ", minMSE) # แสดงผลลัพธ์ของค่า MSE ที่น้อยที่สุด
print("Smallest MAD: ", minMAD) # แสดงผลลัพธ์ของค่า MAD ที่น้อยที่สุด
print("Smallest MAPE: ", minMAPE) # แสดงผลลัพธ์ของค่า MAPE ที่น้อยที่สุด

fig_MSE = plt.figure(figsize=(10, 10), dpi=100) # สร้างกราฟ 3 มิติ โดยกำหนดขนาด 10x10
นิ้วและความละเอียด 100 DPI

ax_MSE = fig_MSE.add_subplot(111, projection='3d') # เพิ่ม Subplot สำหรับกราฟ 3 มิติ
ax_MSE.set_xlabel('Alpha') # กำหนดชื่อแกน X เป็น 'Alpha'
ax_MSE.set_ylabel('Beta') # กำหนดชื่อแกน Y เป็น 'Beta'
ax_MSE.set_zlabel('gamma') # กำหนดชื่อแกน Z เป็น 'Gamma'

# แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ
x_MSE = [t[0] for t in dicMSE.keys()] # แยกค่าของ Alpha จาก Keys ของ DicMSE
y_MSE = [t[1] for t in dicMSE.keys()] # แยกค่าของ Beta จาก Keys ของ DicMSE
z_MSE = [t[2] for t in dicMSE.keys()] # แยกค่าของ Gamma จาก Keys ของ DicMSE
c_MSE = list(dicMSE.values()) # แยกค่าของ MSE จาก Values ของ DicMSE

```

รูปที่ 3.27 เรียกใช้ฟังก์ชัน 'ErrOptHW_NastedLoop' สำหรับการ Optimize ค่าวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า สร้างกราฟ 3 มิติ แยกค่าของ Alpha Beta Gamma และค่า MSE จาก Dictionary เพื่อใช้ในการพล็อตกราฟ

จากรูปที่ 3.27 โค้ดนี้เป็นกระบวนการวิเคราะห์โมเดล Double Exponential Smoothing (DES) โดยใช้ฟังก์ชัน ErrOptHW_NastedLoop เพื่อค้นหาค่าพารามิเตอร์ที่ให้ผลการพยากรณ์ดีที่สุด ในแง่ของความผิดพลาดที่ต่ำที่สุด โดยพิจารณาจากเกณฑ์ MSE MAD และ MAPE ซึ่งค่าที่ได้จะถูกเก็บไว้ทั้งในรูปแบบของค่าต่ำสุด (MinMSE MinMAD MinMAPE) และพหุนามที่เก็บค่าทั้งหมดจากการลองพารามิเตอร์หลายชุด (DicMSE เป็นต้น) ซึ่งสนับสนุนการวิเคราะห์เชิงเปรียบเทียบเพิ่มเติม หลังจากแสดงผลค่าความผิดพลาดที่ต่ำที่สุดทั้งสามออกมาทางหน้าจอ

แล้วโค้ดจะสร้างกราฟ 3 มิติที่ใช้แสดงความสัมพันธ์ระหว่างค่าพารามิเตอร์ Alpha Beta และ Gamma กับค่าความผิดพลาด MSE ที่ได้จากการทดลอง จุดในกราฟจะถูกพล็อตตามตำแหน่งของชุดพารามิเตอร์ โดยมีสีที่แสดงระดับของ MSE ซึ่งทำให้สามารถวิเคราะห์ได้อย่างชัดเจนว่าชุดของพารามิเตอร์ใดให้ผลการพยากรณ์ที่แม่นยำที่สุด ทั้งนี้ แม้ว่าโมเดล DES โดยทั่วไปจะใช้เพียง Alpha และ Beta แต่ในที่นี้ยังมีการรวม Gamma เข้ามาในกระบวนการวิเคราะห์ด้วย อาจเพื่อให้ฟังก์ชันสามารถใช้งานร่วมกับโมเดลอื่นที่ใช้พารามิเตอร์ครบสามตัวได้อย่างยืดหยุ่นมากขึ้น

```
img_MSE = ax_MSE.scatter(x_MSE, y_MSE, z_MSE, c=c_MSE, cmap=plt.hot()) # สร้าง
กราฟแบบ Scatter Plot ในรูปแบบ 3 มิติ แสดงค่าของ MSE โดยใช้สีเป็นตัวบ่งชี้ (Cmap ใช้โทน
สี 'Hot')
fig_MSE.colorbar(img_MSE, fraction=0.025, pad=0.2) # เพิ่มแถบสี (Colorbar) เพื่อแสดง
ระดับของค่า MSE ในกราฟ
plt.show() # แสดงกราฟ
```

รูปที่ 3.28 สร้างกราฟแบบ Scatter Plot ในรูปแบบ 3 มิติ แสดงค่าของ MSE และเพิ่มแถบสี (Colorbar) เพื่อแสดงระดับของค่า MSE ในกราฟ

จากรูปที่ 3.28 โค้ดส่วนนี้เป็นการสร้างกราฟ 3 มิติแบบ Scatter Plot เพื่อแสดงค่าความผิดพลาด MSE ที่ได้จากการทดลองในแต่ละชุดพารามิเตอร์ Alpha Beta และ Gamma สีของจุดในกราฟแสดงระดับของค่า MSE โดยใช้ Colormap แบบ “Hot” เพื่อให้เห็นค่าที่สูงหรือต่ำได้ชัดเจน การใช้สีทำให้สามารถเปรียบเทียบประสิทธิภาพของแต่ละพารามิเตอร์ได้สะดวกยิ่งขึ้น มีการเพิ่ม Colorbar เพื่อแสดงความหมายของระดับสีที่ใช้ในกราฟ สุดท้ายแสดงกราฟทั้งหมดด้วย Plt.Show() เพื่อให้ผู้ใช้เห็นผลลัพธ์ทางภาพอย่างสมบูรณ์

การ Optimize สำหรับการพยากรณ์แต่ละวิธี จะใช้ค่าพารามิเตอร์ต่างกัน ดังนี้

- 1) Holt-Winters Additive และ Multiplicative ใช้ α β และ γ
- 2) Simple Exponential Smoothing ใช้ α
- 3) Double Exponential Smoothing ใช้ α และ β

การเลือกค่าพารามิเตอร์ที่ดีที่สุดจะขึ้นอยู่กับการทดสอบหลาย ๆ ชุดและเปรียบเทียบค่า MAPE เพื่อเลือกค่าที่ให้ผลลัพธ์ที่แม่นยำที่สุด

บทที่ 4

ผลการวิจัย

จากการพยากรณ์ยอดขายประกันชีวิตด้วย 4 วิธี ซึ่งประกอบด้วย วิธีการบวกโฮลท์-วินเทอร์ (Holt Winters' Additive) วิธีการคูณโฮลท์-วินเทอร์ (Holt Winters' Multiplicative) วิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (Simple Exponential Smoothing) และวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า (Double Exponential Smoothing) จะต้องนำเซตของข้อมูลนำไปอัปโหลดที่โปรแกรมภาษา Python โดยในบทนี้จะได้นำเสนอผลการพยากรณ์ยอดขายประกันชีวิตแบบต่าง ๆ ที่ประกอบด้วย

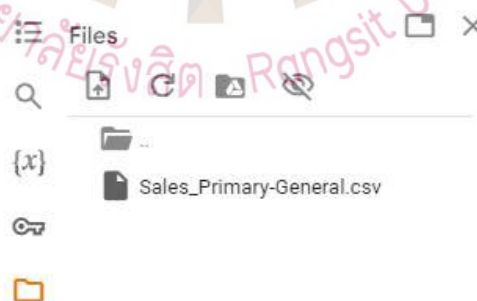
4.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรรมสิทธิ์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป

4.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรรมสิทธิ์หลัก

4.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรรมสิทธิ์สัญญาเพิ่มเติม

4.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรรมสิทธิ์

ตัวอย่างการอัปโหลดไฟล์ Sales_Primary-General.csv ดังรูปที่ 4.1



รูปที่ 4.1 ตัวอย่างการอัปโหลดไฟล์ Sales_Primary-General.csv ที่ Colab

จากรูปที่ 4.1 ในการอัปโหลดไฟล์เข้าสู่ Google Colab เพื่อใช้งานในโปรแกรม ผู้ใช้สามารถใช้เมนู "Files" ทางด้านซ้ายของหน้าจอ โดยคลิกไอคอน "Upload" (รูปกระดาษพร้อมลูกศรขึ้น) แล้วเลือกไฟล์จากเครื่องคอมพิวเตอร์ เช่น ไฟล์ Sales_Primary-General.csv จากนั้นไฟล์จะปรากฏในรายการไฟล์ภายในพื้นที่การทำงานของ Colab ซึ่งสามารถเรียกใช้งานในโค้ดได้ทันที

ผ่าน Path ของไฟล์นั้น ตัวอย่างเช่น `Pd.Read_Csv('Sales_Primary-General.csv')` หากใช้ร่วมกับไลบรารี Pandas เพื่ออ่านข้อมูลจากไฟล์ CSV ดังกล่าว

ตัวอย่างคำสั่งการโหลด Sales_Primary-General.csv ดังรูปที่ 4.2

```
# Load the data into a Pandas DataFrame
path = 'Sales_Primary-General.csv' #change data path accordingly
data = pd.read_csv(path)
```

รูปที่ 4.2 คำสั่งการโหลด

จากรูปที่ 4.2 หลังจากอัปโหลดไฟล์ Sales_Primary-General.csv เข้าสู่ Google Colab ขั้นตอนถัดไปคือการโหลดข้อมูลเข้าสู่โปรแกรมโดยใช้ไลบรารี Pandas ซึ่งเป็นไลบรารียอดนิยมในการจัดการข้อมูลในรูปแบบตาราง (DataFrame) โดยกำหนด Path ของไฟล์ไว้ในตัวแปร เช่น `Path = 'Sales_Primary-General.csv'` จากนั้นใช้คำสั่ง `Pd.Read_Csv(Path)` เพื่ออ่านข้อมูลจากไฟล์ CSV และเก็บไว้ในตัวแปร `Data` ซึ่งพร้อมสำหรับการประมวลผล การกรองข้อมูล การวิเคราะห์ หรือการแสดงผลข้อมูลในลำดับถัดไป

4.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป

การพยากรณ์ยอดขายประกันชีวิตจากการประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป โดยการอัปโหลด Sales_Primary-General.csv ที่โปรแกรมภาษา Python ผลลัพธ์ที่ได้โดยใช้ 4 วิธี แสดงดังตารางที่ 4.1 คือ

- 1) ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีการบวกโสมท์-วินเทอร์
- 2) ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีการคูณโสมท์-วินเทอร์
- 3) ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย
- 4) ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

ตารางที่ 4.1 ผลการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีต่าง ๆ

Period	Year	Quarter	Sales	ผลจาก HWA	ผลจาก HWM	ผลจาก SES	ผลจาก DES
1	2018	1	120957430.86	None	None	120957430.86	None
2	2018	2	237176683.26	None	None	120957430.86	None
3	2018	3	348315999.34	None	None	120957430.86	353395935.66
4	2018	4	473640932.31	None	None	120957430.86	467583213.54
5	2019	1	111696754.49	120957430.86	120957430.86	120957430.86	585209566.18
6	2019	2	223261561.44	237084076.50	234963767.54	120957430.86	512219250.39
7	2019	3	334183042.80	348085167.43	342899853.66	120957430.86	418348421.36
8	2019	4	456844390.25	473271079.15	464669258.09	120957430.86	348602978.70
9	2020	1	107824503.86	113555357.60	105392240.37	120957430.86	338987176.37
10	2020	2	207898856.27	229125598.20	209660465.93	120957430.86	215258022.73
11	2020	3	311040107.44	338806237.49	310742947.90	120957430.86	134817737.01
12	2020	4	432801114.79	461667033.74	424331497.62	120957430.86	126338232.76
13	2021	1	103863695.65	100834117.90	99097365.37	120957430.86	205199407.24
14	2021	2	00041391.53	215137479.58	196332187.24	120957430.86	182233720.67
15	2021	3	297642101.18	324342653.73	297549784.06	120957430.86	187338244.76
16	2021	4	413999403.31	446551696.91	414429643.13	120957430.86	232662777.24
17	2022	1	98786804.70	87524544.75	98491670.15	120957430.86	328461188.86
18	2022	2	191643743.94	85458334.69	97470499.67	120957430.86	387992275.27
19	2022	3	292947075.59	83392124.63	96449329.19	120957430.86	447523361.68
20	2022	4	408846655.82	81325914.57	95428158.72	120957430.86	507054448.09

4.1.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีการบวกโสมท์-วินเทอร์ จากตารางที่ 4.1 พบว่าการพยากรณ์ยอดขายประกันชีวิตจากการประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไปในปี ค.ศ. 2018 จะไม่มีค่าการพยากรณ์เนื่องจากสมการของการพยากรณ์โดยวิธีการบวกโสมท์-วินเทอร์ จะนำข้อมูลในปีแรกเป็นค่าตั้งต้นในการคำนวณค่าพยากรณ์ในปีถัดไป ค่าพยากรณ์ยอดขายในช่วงปี ค.ศ. 2019 - 2021 มีความใกล้เคียงกับยอดขายจริงของยอดขายประกันชีวิต และในปี ค.ศ. 2022 ค่าการพยากรณ์เป็นจำนวนตัวเลขที่น้อยกว่าค่าจริงของยอดขายประกันชีวิต แสดงให้เห็นว่าแบบจำลองสามารถติดตามแนวโน้มของข้อมูลได้ดีในช่วงปีที่มีรูปแบบต่อเนื่อง อย่างไรก็ตาม โมเดลอาจประเมินค่าต่ำเมื่อยอดขายเปลี่ยนแปลงอย่างรวดเร็วในช่วงปลายระยะเวลา

การพยากรณ์ด้วยวิธีการบวกโสมท์-วินเทอร์ ใช้เพื่อคาดการณ์ค่าการขายโดยพิจารณา ค่าเฉลี่ยเคลื่อนที่ การเปลี่ยนแปลงตามฤดูกาล และแนวโน้ม (Trend) ของข้อมูลที่มีฤดูกาล ซึ่งโมเดลนี้เหมาะสมสำหรับข้อมูลที่มีลักษณะการเปลี่ยนแปลงทั้งในเชิงฤดูกาลและแนวโน้ม

โดยคำนึงถึงระดับ (Level) แนวโน้ม (Trend) และฤดูกาล (Seasonality) เพื่อให้ผลลัพธ์ที่แม่นยำสูงสุด โมเดลนี้ส่งเสริมให้สามารถคาดการณ์ยอดขายได้อย่างมีประสิทธิภาพ โดยพิจารณาจากข้อมูลในปัจจุบันและลักษณะการเปลี่ยนแปลงในอดีต ซึ่งจากตารางที่ 4.1 มีข้อมูลดังนี้

4.1.1.1 แนวโน้มโดยรวม

- 1) ในช่วงปี ค.ศ. 2019 - 2022 การพยากรณ์มีการพยากรณ์ที่เป็นไปตามแนวโน้มของยอดขายที่สูงขึ้นเรื่อย ๆ
- 2) ข้อมูลยอดขายจริงและการพยากรณ์ในปี ค.ศ. 2019 - 2021 ดูเหมือนจะมีแนวโน้มที่ค่อนข้างดีเมื่อเปรียบเทียบกัน

4.1.1.2 ความแม่นยำของการพยากรณ์

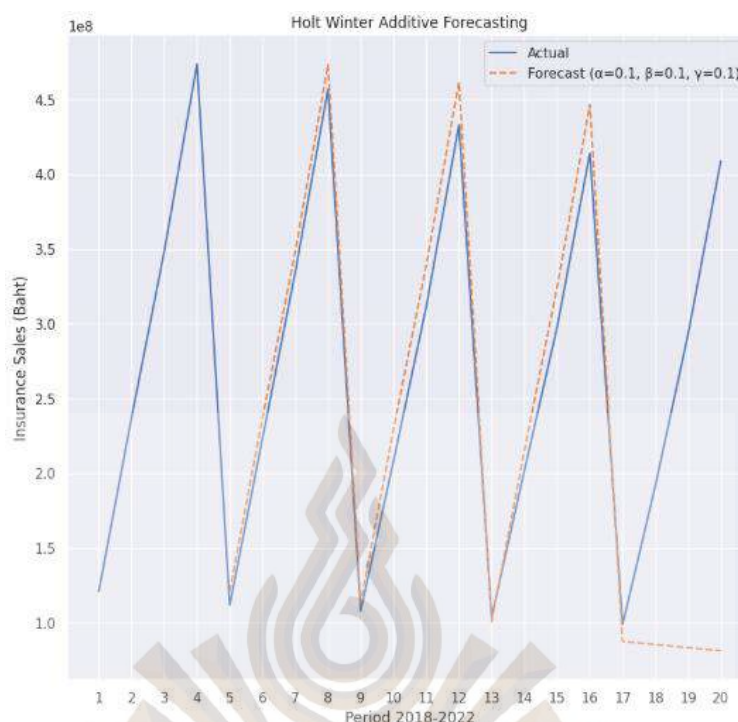
- 1) สำหรับช่วงปี ค.ศ. 2019 การพยากรณ์มีความใกล้เคียงกับยอดขายจริง
- 2) ในปี ค.ศ. 2020 การพยากรณ์เริ่มมีความคลาดเคลื่อนเพิ่มขึ้น แต่ยังคงมีทิศทางใกล้เคียงกับข้อมูลจริง
- 3) ในปี ค.ศ. 2021 - 2022 การพยากรณ์ดูเหมือนจะต่ำกว่าค่าจริง ซึ่งอาจบ่งบอกถึงการเปลี่ยนแปลงในแนวโน้มหรือฤดูกาลที่ไม่สามารถจับได้อย่างแม่นยำโดยวิธีการบวกโฮลท์-วินเทอร์

4.1.1.3 ความผิดพลาดในฤดูกาล

- 1) การพยากรณ์สำหรับปี ค.ศ. 2022 มีค่าที่ต่ำกว่าค่าจริงโดยรวม ซึ่งแสดงถึงการพยากรณ์ไม่ถูกต้องในฤดูกาลนั้น

การพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ สามารถให้การพยากรณ์ที่ดีในช่วงเวลาหนึ่ง แต่การเปลี่ยนแปลงที่สำคัญในแนวโน้มหรือฤดูกาลอาจทำให้ผลลัพธ์ไม่ตรงตามจริงเสมอไป ในกรณีนี้ การพยากรณ์ในปี ค.ศ. 2022 ดูเหมือนจะมีปัญหาในการจับแนวโน้มและฤดูกาลอย่างมีประสิทธิภาพ อาจต้องพิจารณาวิธีพยากรณ์ที่แตกต่าง เช่น การคูณโฮลท์-วินเทอร์ หรือการใช้โมเดลที่พิจารณาแนวโน้มและฤดูกาลในระดับที่สูงขึ้นเพื่อปรับปรุงความแม่นยำของการพยากรณ์

โมเดลพยากรณ์นี้ทำงานได้ดีในบางช่วงที่มีฤดูกาลชัดเจน ($\gamma = 0.8$) แต่ยังไม่สามารถสะท้อนแนวโน้มระยะยาวได้ดี เพราะค่าพารามิเตอร์ α และ β เท่ากับ 0 ซึ่งทำให้ไม่มีการพิจารณาค่าการเปลี่ยนแปลงในระดับปัจจุบันและแนวโน้มที่เกิดขึ้น



รูปที่ 4.3 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีการบวกลด-วินเทอร์ระหว่างปี ค.ศ. 2018 - 2022

รูปที่ 4.3 คือแผนภูมิเส้นนี้แสดงการพยากรณ์อนุกรมเวลาด้วยวิธีการบวกลด-วินเทอร์ โดยแผนภูมิเส้นเปรียบเทียบค่าจริง (เส้นสีน้ำเงินทึบ) กับค่าที่พยากรณ์ (เส้นประสีส้ม) ในช่วง 20 ช่วงเวลาตั้งแต่ปี ค.ศ. 2018 - 2022 โดยพารามิเตอร์ที่ใช้สำหรับแบบจำลองการพยากรณ์คือ α (Alpha) = 0.1 β (Beta) = 0.1 และ γ (Gamma) = 0.1 ซึ่งสอดคล้องกับส่วนระดับ (Level) แนวโน้ม (Trend) และฤดูกาล (Seasonal) ตามลำดับ

4.1.1.4 รูปแบบและความเป็นฤดูกาล

ข้อมูลแสดงให้เห็นรูปแบบความเป็นฤดูกาลอย่างชัดเจน โดยมีจุดสูงสุดและต่ำสุดที่เกิดขึ้นซ้ำประมาณทุก ๆ 5 ช่วงเวลา ค่าที่พยากรณ์ตามเส้นประสีส้มแสดงให้เห็นว่ารูปแบบการพยากรณ์สามารถจับความเป็นฤดูกาลของข้อมูลได้อย่างมีประสิทธิภาพ

4.1.1.5 ประสิทธิภาพของแบบจำลอง

เส้นพยากรณ์อยู่ใกล้เคียงกับเส้นค่าจริงโดยทั่วไป ซึ่งแสดงว่าแบบจำลองมีประสิทธิภาพดีในการพยากรณ์ค่าในอนาคต อย่างไรก็ตาม มีความคลาดเคลื่อนเล็กน้อยในช่วงเวลาที่ 17 - 20 ที่แบบจำลองพยากรณ์ค่าต่ำกว่าค่าจริงที่มีการลดลงอย่างรวดเร็ว

4.1.1.6 ขอบเขตการพยากรณ์

การพยากรณ์ดำเนินต่อไปหลังจากจุดข้อมูลสุดท้ายที่สังเกตเห็น โดยให้การประมาณค่าช่วงเวลาถัดไป การพยากรณ์ในช่วงเวลาที่ 19 และ 20 แสดงให้เห็นการคลาดเคลื่อนอย่างมีนัยสำคัญจากค่าจริง ซึ่งบ่งบอกถึงข้อจำกัดของแบบจำลองในการจัดการกับการเปลี่ยนแปลงอย่างกะทันหันหรือข้อมูลที่ผิดปกติ

แบบจำลองการบวกโซลท์-วินเทอร์ สามารถพยากรณ์ได้อย่างเหมาะสม โดยสามารถจับความเป็นเปอร์เซ็นต์ฤดูกาลและแนวโน้มของข้อมูลได้อย่างดี แต่แบบจำลองอาจมีข้อบกพร่องเมื่อต้องเจอกับการเปลี่ยนแปลงที่ไม่คาดคิดในอนุกรมเวลา ดังที่เห็นได้ในช่วงท้ายของขอบเขตการพยากรณ์ อาจจำเป็นต้องปรับพารามิเตอร์ของแบบจำลองหรือใช้วิธีการพยากรณ์อื่นเพื่อเพิ่มความแม่นยำในกรณีดังกล่าว

ค่าพยากรณ์ที่เบี่ยงเบนในช่วงปลายบ่งชี้ว่าแบบจำลองอาจไม่สามารถปรับตัวต่อเหตุการณ์ได้ การใช้พารามิเตอร์ค่าคงที่ตลอดเวลาอาจไม่เหมาะสมเมื่อเกิดความเปลี่ยนแปลงในข้อมูล จึงควรพิจารณาปรับพารามิเตอร์หรือใช้วิธีพยากรณ์อื่นร่วมด้วยเพื่อความแม่นยำมากขึ้น

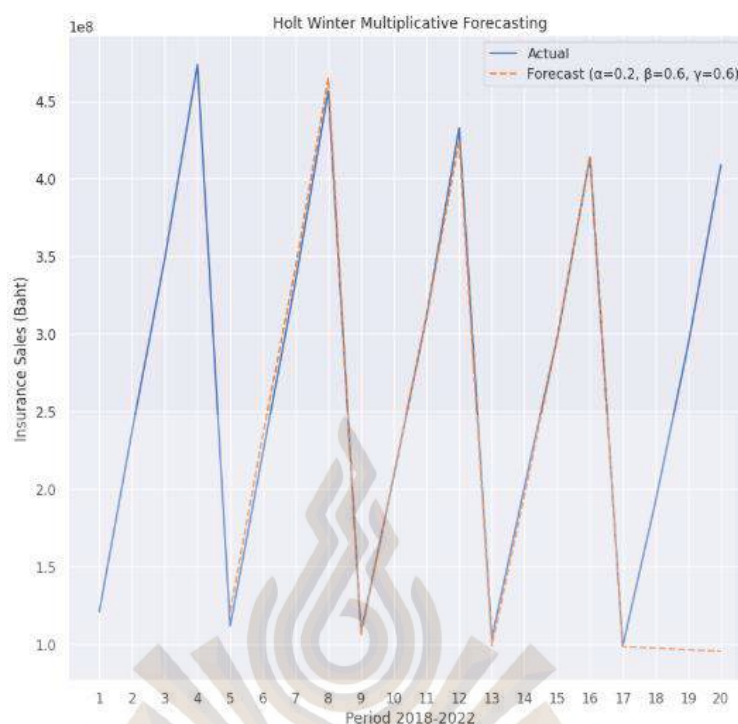
4.1.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีการคูณโซลท์-วินเทอร์

จากตาราง 4.1 พบว่าในปี ค.ศ. 2018 ไม่มีค่าการพยากรณ์สำหรับยอดขายประกันชีวิตจากการประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป เนื่องจากการพยากรณ์ด้วยวิธีการคูณโซลท์-วินเทอร์ จะนำข้อมูลที่บันทึกมาเพื่อทำการพยากรณ์ในปีแรก เป็นค่าตั้งต้นเพื่อคำนวณค่าพยากรณ์ในปีถัดไป ค่าที่ได้จากการพยากรณ์ในปี ค.ศ. 2019 - 2021 มีความใกล้เคียงกับยอดขายจริงสำหรับยอดขายประกันชีวิต อย่างไรก็ตามค่าการพยากรณ์ในปี ค.ศ. 2022 มีค่าน้อยกว่าค่าจริงสำหรับยอดขายประกันชีวิต ซึ่งจากตาราง 4.1 แสดงข้อมูลยอดขายจริงและยอดขายที่พยากรณ์ไว้โดยใช้วิธีการคูณโซลท์-วินเทอร์ในช่วงปี ค.ศ. 2018 - 2022 แยกตามไตรมาส

รูปที่ 4.4 คือแผนภูมิเส้นแสดงผลการพยากรณ์โดยใช้วิธีการคูณโซลท์-วินเทอร์ สำหรับช่วงเวลา ค.ศ. 2018 - 2022 โดยมีการเปรียบเทียบค่าจริง (Actual) กับค่าที่พยากรณ์ (Forecast) ดังนี้

4.1.2.1 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

1) ในปี ค.ศ. 2018 ไม่มีการพยากรณ์ เนื่องจากต้องใช้ข้อมูลในปีแรกในการสร้างแบบจำลองและฝึกฝน โมเดลการพยากรณ์



รูปที่ 4.4 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีการคูณฮอลท์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022

- 2) ตั้งแต่ปี ค.ศ. 2019 เป็นต้นไป สามารถเห็นความแตกต่างระหว่างยอดขายจริง (Sales) และยอดขายที่พยากรณ์ (Forecast)
- 3) ในช่วงแรก (ปี ค.ศ. 2019) การพยากรณ์ค่อนข้างใกล้เคียงกับยอดขายจริง แต่ยังมีความแตกต่างเล็กน้อย
- 4) ตั้งแต่ปี ค.ศ. 2020 เป็นต้นมา การพยากรณ์ยังคงมีความใกล้เคียงกับยอดขายจริง แต่เริ่มมีความคลาดเคลื่อนในบางจุด โดยเฉพาะในไตรมาสที่ 2 ของปี ค.ศ. 2021 และต่อเนื่องถึงปี ค.ศ. 2022 ที่การพยากรณ์มีความแตกต่างอย่างมีนัยสำคัญจากยอดขายจริง

4.1.2.2 แนวโน้มการพยากรณ์และยอดขายจริง

- 1) ยอดขายจริงมีความผันผวนตามไตรมาสและมีแนวโน้มเพิ่มขึ้นในบางช่วงและลดลงในบางช่วง ซึ่งสะท้อนถึงปัจจัยต่าง ๆ เช่น ความต้องการของตลาด หรือเหตุการณ์ที่ส่งผลกระทบต่อการค้าสินค้าของผู้บริโภค
- 2) โมเดล Holt-Winters Multiplicative ดูเหมือนจะสามารถติดตามแนวโน้มการเปลี่ยนแปลงตามฤดูกาลได้ดีในช่วงแรก แต่ในระยะยาว การพยากรณ์อาจไม่สามารถตามการเปลี่ยนแปลงจริงได้

3) วิธีการคูณโฮลท์-วินเทอร์สามารถจับแนวโน้มและฤดูกาลของข้อมูลได้ดีในระยะสั้น แต่ในระยะยาวความแม่นยำของการพยากรณ์จะลดลงเนื่องจากความไม่เสถียรของข้อมูลที่มีการเปลี่ยนแปลงอย่างต่อเนื่อง

4) การปรับพารามิเตอร์หรือการเลือกใช้วิธีการพยากรณ์ที่เหมาะสมสามารถเสริมความแม่นยำในการพยากรณ์ โดยเฉพาะเมื่อข้อมูลมีลักษณะการเปลี่ยนแปลงที่ไม่คงที่หรือมีความผันผวนสูง การเลือกใช้วิธีที่เหมาะสมจะทำให้การพยากรณ์สะท้อนถึงแนวโน้มและฤดูกาลได้ดียิ่งขึ้น

4.1.2.3 รูปแบบและฤดูกาล (Seasonality)

แผนภูมิเส้นนี้แสดงให้เห็นถึงลักษณะการเปลี่ยนแปลงที่เป็นรูปแบบซ้ำ ๆ ซึ่งมีการเพิ่มขึ้นและลดลงตามลำดับในแต่ละช่วงเวลาที่กำหนด โดยสามารถสังเกตเห็นการเปลี่ยนแปลงที่เกิดขึ้นในช่วงเวลาคล้ายกันทุกปี การเปลี่ยนแปลงนี้สะท้อนถึงอิทธิพลของปัจจัยฤดูกาลที่มีผลต่อข้อมูลและทำให้การวิเคราะห์และการพยากรณ์มีความแม่นยำยิ่งขึ้น

1) ความแม่นยำของการพยากรณ์

ค่าที่พยากรณ์ (Forecast) แสดงด้วยเส้นประสีส้มใกล้เคียงกับค่าจริง (Actual) ซึ่งหมายความว่า การพยากรณ์ทำได้แม่นยำดีในช่วงก่อนหน้านี้ อย่างไรก็ตามในช่วงท้าย ๆ ของแผนภูมิเส้น การพยากรณ์เริ่มมีความคลาดเคลื่อนมากขึ้น (เช่น ที่จุดที่ 17 เป็นต้นไป)

2) ช่วงสุดท้ายของการพยากรณ์

หลังจากจุดที่ 17 ค่าพยากรณ์มีแนวโน้มคงที่ ไม่สะท้อนถึงการเพิ่มขึ้นของค่าจริงในช่วงท้าย ซึ่งอาจแสดงถึงข้อจำกัดของแบบจำลองในการจับรูปแบบที่ไม่ปกติหรือแนวโน้มที่เกิดขึ้นในอนาคต

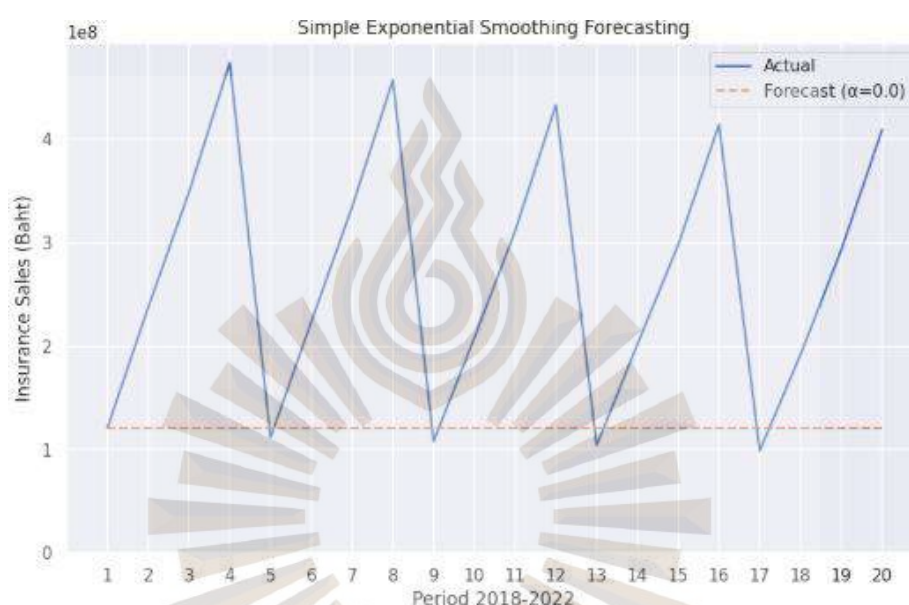
3) การตั้งค่าพารามิเตอร์

พารามิเตอร์ของโมเดลที่ใช้ในการพยากรณ์นี้คือ $\alpha = 0.2$ (สำหรับการทำให้น้ำหนักกับค่าปัจจุบัน) $\beta = 0.6$ (สำหรับแนวโน้ม) และ $\gamma = 0.6$ (สำหรับฤดูกาล) ซึ่งพารามิเตอร์เหล่านี้สามารถปรับเปลี่ยนได้เพื่อเพิ่มความแม่นยำของการพยากรณ์

การพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ ในแผนภูมิเส้นนี้สามารถจับแนวโน้มและฤดูกาลได้ค่อนข้างดี แต่มีข้อจำกัดในการพยากรณ์ค่าที่มีความผันผวนหรือไม่เป็นไปตามแนวโน้มเดิมในอนาคต การปรับพารามิเตอร์หรือการใช้โมเดลที่ซับซ้อนขึ้นอาจส่งเสริมความแม่นยำได้ ในกรณีที่ข้อมูลมีการเปลี่ยนแปลงอย่างฉับพลันหรือผิดปกติ

4.1.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

จากตาราง 4.1 พบว่าค่าการพยากรณ์สำหรับยอดขายประกันชีวิตจากกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไปมีค่าเท่ากันทุกไตรมาสทั้ง 5 ปี เนื่องจากตัวแปรการปรับให้เรียบ α มีค่าเป็น 0 ซึ่งเหมาะสมที่สุดสำหรับการพยากรณ์และให้ความแม่นยำสูงสุด



รูปที่ 4.5 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายระหว่างปี ค.ศ. 2018 - 2022

รูปที่ 4.5 แผนภูมิแสดงยอดขายประกันชีวิตจากการประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป จากการพยากรณ์แบบปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ซึ่งเป็นเทคนิคการพยากรณ์ที่ใช้ในการพยากรณ์ข้อมูลตามลำดับเวลา โดยใช้ค่าเฉลี่ยถ่วงน้ำหนักของข้อมูลในอดีตมีรายละเอียดดังนี้

4.1.3.1 การพยากรณ์โดยวิธีซิมเปิลเอ็กซ์โพเนนเชียล

วิธีการปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (SES) จะใช้ค่าเฉลี่ยถ่วงน้ำหนักของข้อมูลในอดีตเพื่อพยากรณ์ค่าในอนาคต โดยให้ค่าน้ำหนักมากขึ้นกับข้อมูลที่ใกล้ปัจจุบัน แต่ในกรณีนี้ค่าพยากรณ์ทุกช่วงเวลาหลังจากช่วงแรกเท่ากับยอดขายจริงของไตรมาสแรกในปี ค.ศ. 2018 (120,957,430.86) ซึ่งโมเดล SES ไม่สามารถสะท้อนการเปลี่ยนแปลงของข้อมูลในช่วงเวลาถัดไปได้อย่างมีประสิทธิภาพ

4.1.3.2 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

1) ยอดขายจริง (Sales) แสดงถึงความผันผวนตามแต่ละไตรมาสและปี โดยมีการเปลี่ยนแปลงที่ชัดเจนจากช่วงเวลาไปยังช่วงเวลา ขณะที่การพยากรณ์ (Forecast) กลับคงที่เท่ากับยอดขายจริงในไตรมาสแรกของปี ค.ศ. 2018 ตลอดช่วงเวลาทั้งหมด ซึ่งวิธีการพยากรณ์นี้ไม่สามารถสะท้อนการเปลี่ยนแปลงหรือแนวโน้มที่เกิดขึ้นในข้อมูลจริงได้อย่างแม่นยำ

2) ความคลาดเคลื่อนระหว่างยอดขายจริงและยอดขายที่พยากรณ์มีค่าสูง เนื่องจากโมเดลพยากรณ์ไม่สามารถสะท้อนถึงการเปลี่ยนแปลงในแนวโน้มของข้อมูลได้อย่างมีประสิทธิภาพ และไม่พิจารณาปัจจัยภายนอกที่มีผลกระทบต่อข้อมูลในช่วงเวลาต่าง ๆ

4.1.3.3 แนวโน้มของการพยากรณ์

การพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายในกรณีนี้ไม่ได้ปรับตัวตามยอดขายที่เปลี่ยนแปลงจริง เนื่องจากโมเดลนี้ใช้ค่าเริ่มต้นคงที่ และไม่มีการปรับตัวตามข้อมูลในอดีตที่เกิดขึ้นในช่วงเวลาอื่น

1) การพยากรณ์โดยใช้วิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (SES) ในกรณีนี้ไม่ได้มีการอัปเดตค่าพยากรณ์ตามข้อมูลใหม่ ทำให้การพยากรณ์มีค่าเท่ากับยอดขายในไตรมาสแรกของปี ค.ศ. 2018 ตลอดช่วงเวลา ซึ่งส่งผลให้การพยากรณ์นี้ไม่สอดคล้องกับแนวโน้มของข้อมูลจริงในช่วงหลังที่มีการเปลี่ยนแปลงอย่างมาก

2) วิธีการนี้อาจเหมาะสมกับข้อมูลที่ไม่มีการเปลี่ยนแปลงหรือมีแนวโน้มคงที่ แต่สำหรับข้อมูลที่มีความผันผวนหรือแนวโน้มที่ไม่คงที่ อาจจำเป็นต้องปรับพารามิเตอร์เริ่มต้นหรือน้ำหนัก (Alpha) เพื่อให้การพยากรณ์สะท้อนถึงความแปรผันของข้อมูลได้อย่างถูกต้องและแม่นยำมากยิ่งขึ้น

4.1.3.4 Actual (ข้อมูลจริง) เส้นสีน้ำเงินแสดงข้อมูลจริงในช่วงปี ค.ศ. 2018 - 2022 ซึ่งเห็นได้ว่ามีรูปแบบการขึ้นลงเป็นลักษณะคลื่นชัดเจน โดยมีค่าเพิ่มขึ้นไปสูงสุดที่ประมาณ $4.5e8$ และลดลงต่ำสุดใกล้กับ 0 ซึ่งบ่งบอกถึงความแปรปรวนของข้อมูลอย่างมาก

4.1.3.5 Forecast (การพยากรณ์) เส้นประสีส้มแสดงค่าพยากรณ์จากการซึมเปิดเอ็กซ์โพเนนเชียลโดยตั้งค่า $\alpha = 0.0$ ซึ่งหมายความว่าไม่มีการถ่วงน้ำหนักให้กับข้อมูลใหม่ ๆ ค่าพยากรณ์จึงเป็นค่าคงที่ตลอดช่วงเวลาที่แสดงและสะท้อนค่าเฉลี่ยของข้อมูลจริงในอดีต ซึ่งไม่สามารถจับการเปลี่ยนแปลงหรือแนวโน้มใหม่ ๆ ในข้อมูลได้

4.1.3.6 การไม่ตอบสนองต่อความแปรปรวนของข้อมูลจริง การตั้งค่า $\alpha = 0.0$ นั้นเหมาะสมสำหรับข้อมูลที่มีความผันผวนต่ำ ซึ่งไม่เหมาะกับข้อมูลจริงที่แสดงในแผนภูมิเส้นนี้ เพราะข้อมูลมีการขึ้นลงชัดเจนและต่อเนื่อง

4.1.3.7 ความเหมาะสมของโมเดล แผนภูมิเส้นนี้แสดงให้เห็นว่าการใช้ซิมเปิลเอ็กซ์โพเนนเชียล ด้วยค่า $\alpha = 0.0$ ไม่เหมาะสมสำหรับข้อมูลที่มีความแปรปรวนหรือมีแนวโน้มเปลี่ยนแปลงอย่างมาก เนื่องจากโมเดลไม่สามารถจับความเปลี่ยนแปลงหรือแนวโน้มได้เลย

หากต้องการให้โมเดลการพยากรณ์จับแนวโน้มการเปลี่ยนแปลงได้ดียิ่งขึ้น ควรเลือกค่าของ α ที่สูงขึ้น หรือพิจารณาใช้เทคนิคการพยากรณ์ที่ซับซ้อนกว่าเช่น Holt-Winters Exponential Smoothing หรือ ARIMA ที่สามารถจัดการกับข้อมูลที่มีแนวโน้มและความแปรปรวนสูงได้ดีกว่า

4.1.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

จากตาราง 4.1 ผลการพยากรณ์โดยวิธีการปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าสามารถทำได้ โดยพิจารณาความแม่นยำของการพยากรณ์และการเปรียบเทียบกับยอดขายจริงในแต่ละช่วงเวลา

4.1.4.1 ความแม่นยำของการพยากรณ์

1) ดูว่าในช่วงเวลาไหนการพยากรณ์ตรงกับยอดขายจริงมากที่สุดและช่วงเวลาไหนพยากรณ์ผิดพลาดมากที่สุด โดยทั่วไปการพยากรณ์ที่ดีควรมีค่า Forecast ที่ใกล้เคียงกับยอดขายจริงมากที่สุด

2) คำนวณความผิดพลาดของการพยากรณ์ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย (Mean Absolute Percentage Error, MAPE) หรือค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) เพื่อดูว่าโมเดลมีความแม่นยำแค่ไหน

4.1.4.2 ความสอดคล้องของแนวโน้ม

ตรวจสอบว่าโมเดลสามารถจับแนวโน้มของยอดขายได้ดีแค่ไหน ในกรณีนี้ยอดขายมีแนวโน้มเพิ่มขึ้นเรื่อย ๆ โมเดลปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ถูกออกแบบมาเพื่อจับแนวโน้มนี้ได้ แต่ความแม่นยำจะขึ้นอยู่กับค่าพารามิเตอร์ที่ใช้ในโมเดล

4.1.4.3 การเปรียบเทียบกับข้อมูลจริง

1) ค่าพยากรณ์ในแต่ละช่วงเวลาถูกนำมาเปรียบเทียบกับยอดขายจริงเพื่อประเมินความแม่นยำของโมเดล ตัวอย่างเช่น

1.1) Q3 2018: (คาดการณ์ใกล้เคียง) Sales = 348,315,999.34

Forecast = 353,395,935.66

1.2) Q1 2019: (คาดการณ์ผิดพลาดมาก) Sales = 111,696,754.49

Forecast = 585,209,566.18

2) ช่วงที่มีความผิดพลาดมากอาจบ่งบอกว่าโมเดลไม่สามารถจับแนวโน้มหรือฤดูกาลได้ดีในช่วงเวลานั้น

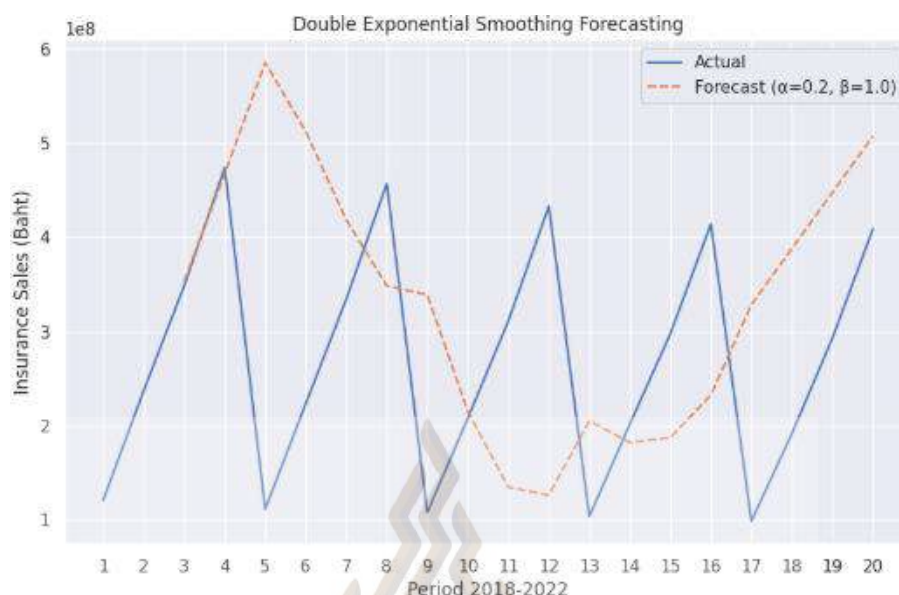
4.1.4.4 การพิจารณาปัจจัยภายนอก

อาจมีปัจจัยภายนอกที่ทำให้ยอดขายเปลี่ยนแปลงอย่างไม่ปกติ เช่น เหตุการณ์สำคัญที่ส่งผลกระทบต่อยอดขาย การพิจารณาความผิดพลาดที่เกิดจากปัจจัยเหล่านี้อาจส่งเสริมการปรับปรุงโมเดล

จากตารางที่นำมาเห็นได้ว่ามีความผิดพลาดในการพยากรณ์ที่เปลี่ยนแปลงไปในแต่ละช่วงเวลา โดยเฉพาะในปี ค.ศ. 2019 ที่มีการพยากรณ์ที่ผิดพลาดสูง อาจจะต้องพิจารณาการปรับพารามิเตอร์ของโมเดลหรือใช้เทคนิคการพยากรณ์อื่น ๆ ที่อาจจะเหมาะสมมากกว่าเพื่อปรับปรุงความแม่นยำของการพยากรณ์ในอนาคต

รูปที่ 4.6 แผนภูมิเส้นการพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า สำหรับข้อมูลในช่วงปี ค.ศ. 2018 - 2022 โดยมีการตั้งค่า $\alpha = 0.2$ และ เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในแต่ละช่วงเวลา โดยมีการผันผวนสูงและแสดงให้เห็นแนวโน้มที่ไม่คงที่และมีลักษณะของ Seasonality หรือแนวโน้มที่เป็นวัฏจักร

4.1.4.5 เส้นประสีส้ม (Forecast) แสดงค่าพยากรณ์ที่ได้จากการใช้ Double Exponential Smoothing ซึ่งการพยากรณ์นี้ใช้ค่า $\alpha = 0.2$ และ $\beta = 0.1$ ทำให้การพยากรณ์มีความนุ่มนวลกว่าค่าจริงที่มีความผันผวนสูง เส้นพยากรณ์สามารถติดตามแนวโน้มของข้อมูลได้ในภาพรวม แม้จะไม่สามารถสะท้อนการเปลี่ยนแปลงเฉียบพลันได้อย่างแม่นยำ การเลือกใช้ค่าสัมประสิทธิ์ที่ต่ำลดผลกระทบจากความผันผวนระยะสั้น ทำให้ผลลัพธ์เสถียรมากขึ้น



รูปที่ 4.6 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าระหว่างปี ค.ศ. 2018 - 2022

4.1.4.6 ประสิทธิภาพของการพยากรณ์

1) เส้นพยากรณ์ (Forecast) มีลักษณะการตามแนวโน้มของข้อมูลจริง แต่จะมีการทำให้เส้นแผนภูมินุ่มนวลขึ้น

2) ค่า $\alpha = 0.2$ หมายถึงการให้ความสำคัญกับข้อมูลล่าสุดในระดับปานกลาง ดังนั้นการพยากรณ์จะไม่ตอบสนองมากนักต่อการเปลี่ยนแปลงระยะสั้น

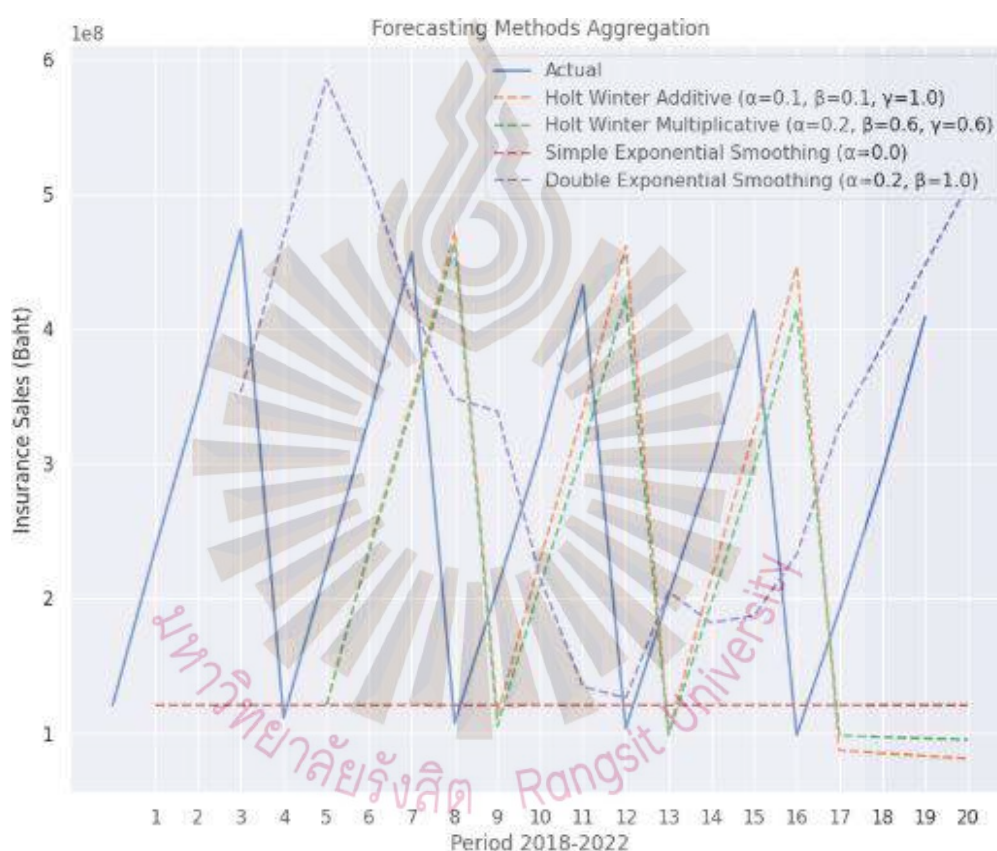
4.1.4.7 ค่า $\beta = 0.1$ แสดงถึงการตอบสนองต่อแนวโน้มที่เปลี่ยนแปลงอย่างช้า ๆ ทำให้แนวโน้มมีความนุ่มนวลกว่าเส้นแผนภูมิจริงที่มีความผันผวนสูง

การพยากรณ์ด้วยวิธีนี้เหมาะสมกับข้อมูลที่มีแนวโน้มและวัฏจักร แต่ไม่เหมาะสมกับข้อมูลที่มีความผันผวนสูงมากนัก เนื่องจากเส้นพยากรณ์อาจไม่สามารถตอบสนองกับการเปลี่ยนแปลงของข้อมูลจริงได้ทัน การใช้ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่ากับข้อมูลที่มีแนวโน้มและความผันผวนนี้อาจจะมีข้อจำกัดในการพยากรณ์ผลลัพธ์ในระยะยาวที่เที่ยงตรง เนื่องจากการตั้งค่า α และ β ที่ทำให้แผนภูมิเส้นมีความนุ่มนวลเกินไปเมื่อเทียบกับค่าจริงที่เปลี่ยนแปลงบ่อยครั้ง อย่างไรก็ตาม วิธีนี้ยังมีประโยชน์ในการทำให้เห็นภาพรวมของแนวโน้มข้อมูลได้ชัดเจนขึ้น การเลือกค่าพารามิเตอร์ที่เหมาะสมจึงเป็นสิ่งสำคัญในการเพิ่มประสิทธิภาพของแบบจำลองพยากรณ์

4.1.5 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปด้วยวิธีต่าง ๆ

เมื่อนำผลทั้ง 4 วิธี มารวมกัน จะได้ดังรูปที่ 4.7 แผนภูมิเส้นที่แสดงการพยากรณ์วิธีต่าง ๆ เปรียบเทียบกับข้อมูลยอดขายจริงในช่วงปี ค.ศ. 2018 - 2022

4.1.5.1 ยอดขายในบางช่วงสูงขึ้นอย่างมาก เช่น ในช่วงที่ 3 9 และ 15 แสดงว่าการเปลี่ยนแปลงของพฤติกรรมผู้บริโภคของผู้บริโภคที่สำคัญ



รูปที่ 4.7 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไปโดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022

4.1.5.2 การระบาดของโควิด-19 ส่งผลกระทบต่อพฤติกรรมผู้บริโภค โดยเฉพาะในช่วงที่เกิดความไม่แน่นอนด้านสุขภาพและการเงิน ทำให้ผู้คนให้ความสำคัญกับการซื้อประกันตั้งแต่ปี ค.ศ. 2020 เป็นต้นมา

4.1.5.3 ยอดขายที่ลดลงอาจเกิดจากสถานการณ์หลังการระบาด เมื่อผู้บริโภคผ่อนคลายและลดการซื้อประกันหลังโควิดดีขึ้น หรือมาตรการบรรเทาผลกระทบทางการเงินจากภาครัฐ

การเปลี่ยนแปลงนี้สะท้อนให้เห็นในแผนภูมิที่แสดงว่าบางวิธีการพยากรณ์ เช่น Double Exponential Smoothing มีแนวโน้มขึ้นและลงอย่างชัดเจน ขณะที่ Simple Exponential Smoothing แสดงผลลัพธ์ที่มีความเสถียร แต่ไม่ตรงกับยอดขายจริงในช่วงที่เกิดการระบาดของโควิด-19

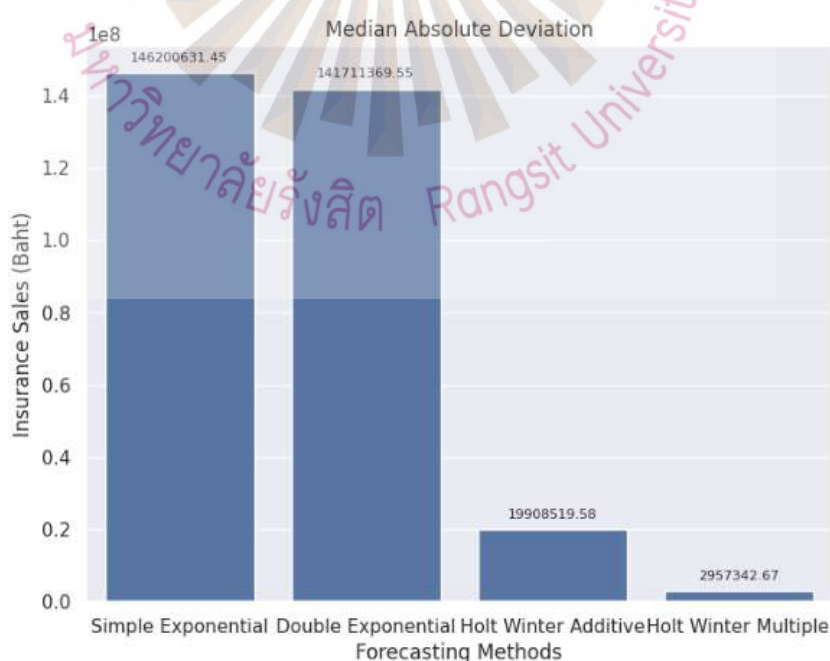
4.1.6 ผลลัพธ์การวัดข้อผิดพลาดการพยากรณ์กรรมวิธีแบบทั่วไป

ในการประเมินประสิทธิภาพของโมเดลพยากรณ์ที่ใช้กับข้อมูลของกรรมวิธีแบบทั่วไป ได้มีการวัดค่าความผิดพลาดของการพยากรณ์ด้วยตัวชี้วัดมาตรฐาน ได้แก่ MAD (Mean Absolute Deviation) MAPE (Mean Absolute Percentage Error) และ MSE (Mean Squared Error) เพื่อวิเคราะห์ความแม่นยำของโมเดล ดังนี้

4.1.6.1 ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD)

จากแผนภูมิแท่งรูปที่ 4.8 ซึ่งแสดงค่า "Median Absolute Deviation" ของวิธีการพยากรณ์ต่าง ๆ ดังนี้

- 1) Simple Exponential มีค่า MAD ประมาณ 146,200,631.45
- 2) Double Exponential มีค่า MAD ประมาณ 141,711,369.55
- 3) Holt Winter Additive มีค่า MAD ประมาณ 19,908,519.58



รูปที่ 4.8 แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) สำหรับการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป

4) Holt Winter Multiple มีค่า MAD ประมาณ 2,957,342.67

5) วิธีซิมเปิลเอ็กซ์โพเนนเชียล และ ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า มีค่า MAD สูงมาก (ประมาณ $1.4e8$) ซึ่งบ่งบอกถึงความไม่แม่นยำของวิธีการเหล่านี้ในการพยากรณ์ข้อมูล

หมายเหตุ

การแปลง $1.4e8$ ให้เป็นตัวเลขปกติทำได้โดยการคูณ 1.4 ด้วย 10 ยกกำลัง 8

$$1.4e8 = 1.4 \times 10^8$$

ซึ่งหมายถึงการคูณ 1.4 ด้วย 10^8 เมื่อคำนวณค่า 10^8 จะได้ 100,000,000 (หนึ่งร้อยล้าน) ดังนั้น

$$1.4 \times 100,000,000 = 140,000,000$$

ดังนั้น $1.4e8$ จึงเท่ากับ 140,000,000

6) วิธีการบวกโฮลท์-วินเทอร์ มีค่า MAD น้อยลงมาก (ประมาณ $2e7$) แสดงถึงความแม่นยำที่เพิ่มขึ้น

7) วิธีการคูณโฮลท์-วินเทอร์ มีค่า MAD ต่ำที่สุด (ประมาณ $2.9e6$) แสดงถึงความแม่นยำในการพยากรณ์ที่ดีที่สุดเมื่อเทียบกับวิธีการอื่น ๆ ที่แสดงในแผนภูมิ

วิธีการคูณโฮลท์-วินเทอร์เป็นวิธีที่มีประสิทธิภาพสูงสุดในการลดค่าความผิดพลาด (MAD) ในการพยากรณ์ข้อมูล โดยสามารถสะท้อนพฤติกรรมของข้อมูลได้อย่างแม่นยำและน่าเชื่อถือเนื่องจากโมเดลสามารถจับแนวโน้มและฤดูกาลของข้อมูลได้ผ่านการปรับค่าพารามิเตอร์ที่เหมาะสม จึงเหมาะสำหรับการพยากรณ์ข้อมูลที่มีลักษณะเปลี่ยนแปลงตามเวลาอย่างต่อเนื่องและเป็นคาบ นอกจากนี้ยังสามารถประยุกต์ใช้ได้อย่างมีประสิทธิภาพในบริบทของข้อมูลที่มีความซับซ้อน เช่น ภาครัฐกิจหรือการวิเคราะห์ข้อมูลเชิงลึก

4.1.6.2 ค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE)

จากแผนภูมิแท่งรูปที่ 4.9 ซึ่งแสดงค่า "Mean Absolute Percentage Error" (MAPE) ของวิธีการต่าง ๆ ดังนี้

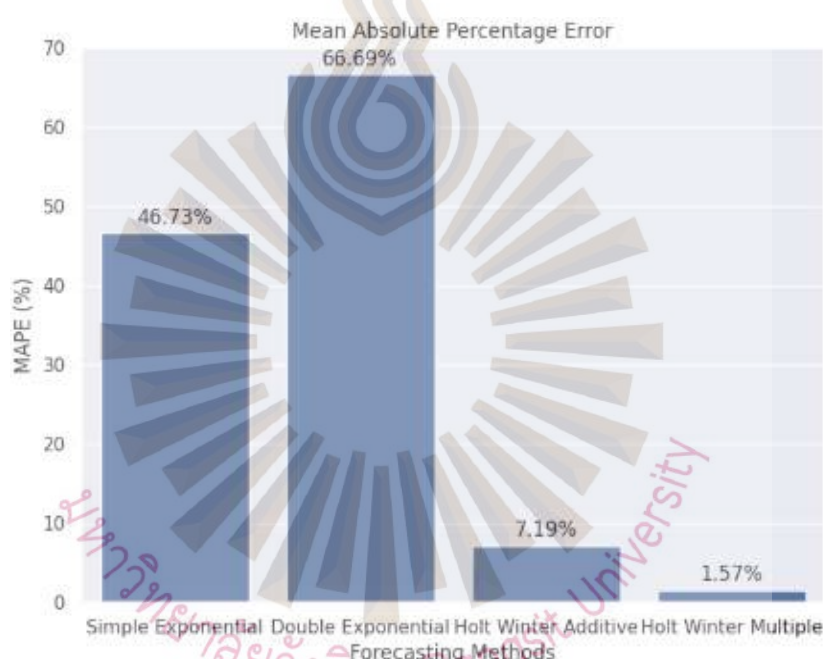
- 1) Simple Exponential มีค่า MAPE ประมาณ 46.73%
- 2) Double Exponential มีค่า MAPE ประมาณ 66.69%
- 3) Holt Winter Additive มีค่า MAPE ประมาณ 7.19%
- 4) Holt Winter Multiple มีค่า MAPE ประมาณ 1.57%

5) วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า มีค่า MAPE สูงสุดที่ 66.69% ซึ่งบ่งบอกว่ามีความคลาดเคลื่อนสูงสุดในกระบวนการพยากรณ์

6) วิธีซึมเปิลเอ็กซ์โพเนนเชียล มีค่า MAPE สูงที่ 46.73% ซึ่งยังคงมีความคลาดเคลื่อนสูงเช่นกัน

7) วิธีการบวกโฮลท์-วินเทอร์ มีค่า MAPE ลดลงเหลือ 7.19% ซึ่งแสดงถึงความแม่นยำที่เพิ่มขึ้นเมื่อเทียบกับสองวิธีแรก

8) วิธีการคูณโฮลท์-วินเทอร์ มีค่า MAPE ต่ำที่สุดที่ 1.57% ซึ่งบ่งบอกถึงความแม่นยำในการพยากรณ์สูงสุดเมื่อเทียบกับวิธีการอื่น ๆ ในแผนภูมิ



รูปที่ 4.9 แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) สำหรับการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป

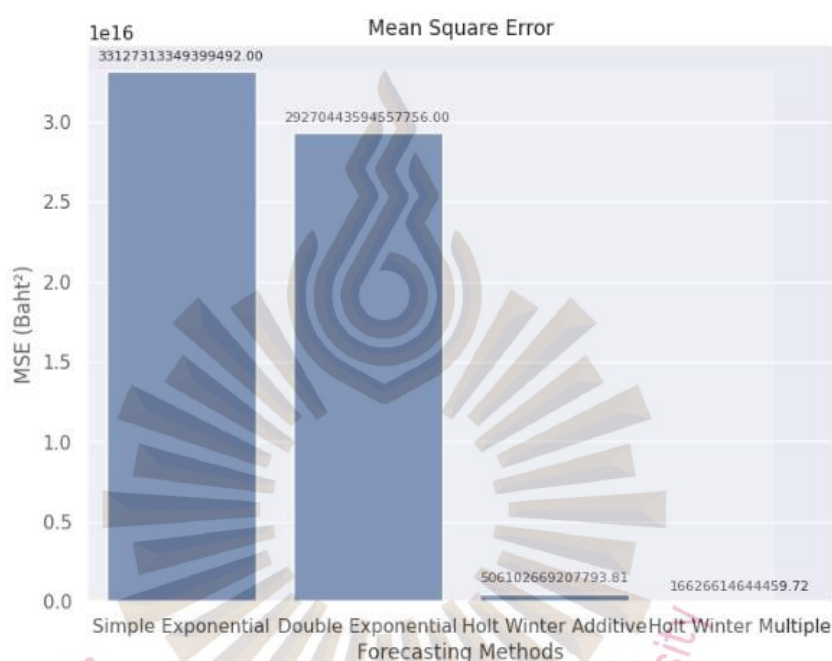
วิธีการคูณโฮลท์-วินเทอร์เป็นวิธีที่มีประสิทธิภาพดีที่สุดสำหรับการพยากรณ์ เนื่องจากลดค่าเปอร์เซ็นต์ความผิดพลาดสัมบูรณ์ (MAPE) และให้ค่า MAD ต่ำที่สุดที่ประมาณ 2.96 ล้านบาท แสดงถึงความคลาดเคลื่อนน้อยที่สุด วิธีนี้สามารถจับแนวโน้มและฤดูกาลได้ดี เหมาะสมกับข้อมูลที่มีความแปรผันตามช่วงเวลาและลักษณะฤดูกาลอย่างชัดเจน

นอกจากนี้ วิธีการคูณโฮลท์-วินเทอร์ทำให้การคาดการณ์ยอดขายมีความแม่นยำมากยิ่งขึ้น แม้ในช่วงที่ข้อมูลมีความผันผวนหรือมีการเปลี่ยนแปลงอย่างรวดเร็ว ซึ่งทำให้ธุรกิจสามารถวางแผน

และตัดสินใจได้อย่างมีประสิทธิภาพและทันสถานการณ์ โดยการปรับแต่งแบบจำลองให้เหมาะสมกับข้อมูลที่มีลักษณะเฉพาะและผลลัพธ์ที่ต้องการ

4.1.6.3 ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE)

จากแผนภูมิแท่งรูปที่ 4.10 ซึ่งแสดงค่า "Mean Square Error" (MSE) ของวิธีการพยากรณ์ต่าง ๆ ดังนี้



รูปที่ 4.10 แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) สำหรับการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป

- 1) Simple Exponential มีค่า MSE ประมาณ 3.3×10^{16}
- 2) Double Exponential มีค่า MSE ประมาณ 2.9×10^{16}
- 3) Holt Winter Additive มีค่า MSE ประมาณ 5.1×10^{12}
- 4) Holt Winter Multiple มีค่า MSE ประมาณ 1.7×10^{12}
- 5) วิธีซิมเปิลเอ็กซ์โพเนนเชียล มีค่า MSE สูงที่สุด (ประมาณ 3.3×10^{16}) ซึ่งบ่งบอกถึงความคลาดเคลื่อนมากที่สุดในกระบวนการพยากรณ์
- 6) วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า มีค่า MSE สูงรองลงมา (ประมาณ 2.9×10^{16}) ซึ่งยังคงมีความคลาดเคลื่อนสูงมาก
- 7) วิธีการบวกโฮลท์-วินเทอร์ มีค่า MSE ต่ำกว่าสองวิธีแรกอย่างมาก (ประมาณ 5.1×10^{12}) ซึ่งแสดงถึงความแม่นยำที่เพิ่มขึ้น

8) วิธีการคูณโฮลท์-วินเทอร์ มีค่า MSE ต่ำที่สุด (ประมาณ 1.7×10^{12}) ซึ่งบ่งบอกถึงความแม่นยำในการพยากรณ์สูงสุดเมื่อเทียบกับวิธีการอื่น ๆ ในแผนภูมิ

วิธีการคูณโฮลท์-วินเทอร์ เป็นวิธีที่มีประสิทธิภาพดีที่สุดสำหรับการพยากรณ์ในแง่ของการลดค่าความคลาดเคลื่อน (MSE) ทำให้การพยากรณ์มีความแม่นยำมากขึ้น ในขณะที่วิธีหิมเปิดเอ็กซ์โพเนนเชียล และ ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า มีค่า MSE สูงมากและมีความคลาดเคลื่อนในการพยากรณ์มากที่สุด การมีค่า MSE ที่สูงในโมเดล Double Exponential แสดงให้เห็นว่าโมเดลไม่สามารถปรับตัวให้สอดคล้องกับลักษณะการเปลี่ยนแปลงของข้อมูลได้และไม่สามารถจับความสัมพันธ์ของแนวโน้มและฤดูกาลที่มีอยู่ได้ วิธีการคูณโฮลท์-วินเทอร์จึงเหมาะสมกับข้อมูลที่มีความแปรผันตามฤดูกาลและแนวโน้มอย่างชัดเจน ในขณะที่วิธีอื่นไม่สามารถรองรับความซับซ้อนของข้อมูลลักษณะนี้ได้

4.1.7 ผลลัพธ์ความแม่นยำของการพยากรณ์กรรมวิธีแบบทั่วไป

แผนภูมิแท่งรูปที่ 4.11 ซึ่งแสดงเปอร์เซ็นต์ความแม่นยำของวิธีการพยากรณ์ทั้ง 4 วิธี จากสูตร

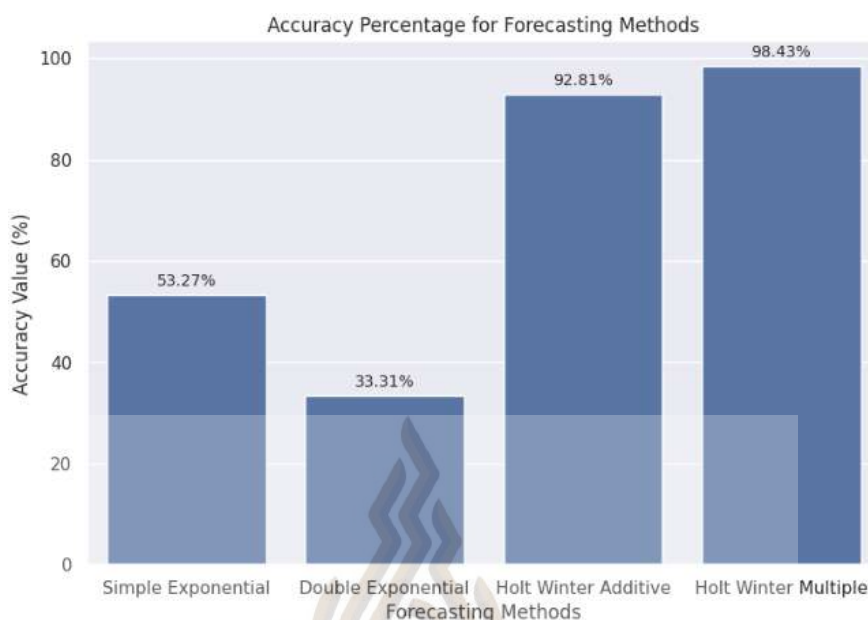
$$\text{Accuracy} = 100\% - \text{MAPE}$$

$$\text{Accuracy} = 100\% - \frac{\sum \left[\frac{|A_t - F_t|}{A_t} \right]}{n} \times 100\% \quad (4-1)$$

มีรายละเอียดดังนี้

- 1) Holt Winter Multiple มีความแม่นยำสูงที่สุดที่ 98.43% ซึ่งเป็นวิธีที่แม่นยำที่สุด
- 2) Holt Winter Additive มีความแม่นยำ 92.81% รองลงมา
- 3) Simple Exponential มีความแม่นยำ 53.27%
- 4) Double Exponential มีความแม่นยำต่ำสุดที่ 33.31%

วิธีที่แม่นยำที่สุดในการพยากรณ์คือ Holt-Winter Multiple โดยมีความแม่นยำ 98.43% ซึ่งเป็นวิธีที่ให้ผลลัพธ์ที่ดีที่สุดในแผนภูมินี้ เนื่องจากสามารถจับแนวโน้มและฤดูกาลของข้อมูลได้อย่างมีประสิทธิภาพ ส่งผลให้ค่าพยากรณ์ที่ได้มีความใกล้เคียงกับค่าจริงมากที่สุดเมื่อเปรียบเทียบกับวิธีการพยากรณ์อื่น ๆ การที่วิธีนี้สามารถผสานองค์ประกอบของแนวโน้มและฤดูกาลได้อย่างลงตัว ทำให้การพยากรณ์มีความแม่นยำและสอดคล้องกับพฤติกรรมของข้อมูลในอดีต การเลือกใช้ Holt-Winter Multiple จึงเป็นแนวทางที่เหมาะสมสำหรับข้อมูล Time Series ที่มีความซับซ้อน



รูปที่ 4.11 แผนภูมิแท่งแสดงผลลัพธ์ความแม่นยำของการพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป

4.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก

การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยใช้ 4 วิธี ผลลัพธ์แสดงดังตารางที่ 4.2 โดยการอัปโหลดคลดจะใช้ไฟล์ชื่อ Sales_Primary.csv ส่งไปที่โปรแกรมภาษา Python ซึ่งแสดงผลพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีต่าง ๆ โดยจำแนกตามประเภทของวิธีการพยากรณ์และยอดขายจากช่วงเวลา 4 ปี เพื่อพยากรณ์ยอดขายปีที่ 5 และใช้ในการวิเคราะห์ผลลัพธ์ ข้อมูลเหล่านี้จะถูกนำไปใช้ในการประเมินประสิทธิภาพของการดำเนินงานผ่านโปรแกรม Python

ตารางที่ 4.2 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีต่าง ๆ

Period	Year	Quarter	Sales	ผลจาก HWA	ผลจาก HWM	ผลจาก SES	ผลจาก DES
1	2018	1	12064969.75	None	None	12064969.75	None
2	2018	2	22415287.00	None	None	12064969.75	None
3	2018	3	38032011.99	None	None	13100001.48	32765604.25
4	2018	4	53146719.25	None	None	15593202.53	45801789.44
5	2019	1	7409569.12	12064969.75	12064969.75	19348554.20	61003966.51
6	2019	2	17086034.17	22415287.00	22415287.00	18154655.69	46669521.97
7	2019	3	27396807.51	38032011.99	38032011.99	18047793.54	33325817.87

ตารางที่ 4.2 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีต่าง ๆ (ต่อ)

Period	Year	Quarter	Sales	ผลจาก HWA	ผลจาก HWM	ผลจาก SES	ผลจาก DES
8	2019	4	42522874.74	53146719.25	53146719.25	18982694.94	25833364.82
9	2020	1	8213632.78	8340649.25	7228883.00	21336712.92	28631264.95
10	2020	2	17182364.88	18151884.74	16987241.78	20024404.91	16009319.69
11	2020	3	27695833.73	29523848.41	30553359.67	19740200.91	10110917.13
12	2020	4	45026611.16	44647643.64	46941731.16	20535764.19	12828908.48
13	2021	1	14702125.45	8239036.07	7580608.75	22984848.89	26692253.21
14	2021	2	27753519.57	17376268.85	21060880.07	22156576.55	24781321.98
15	2021	3	43886109.98	28061436.67	39796936.99	22716270.85	27983249.86
16	2021	4	67116178.95	44950817.66	66137668.23	24833254.76	38403977.13
17	2022	1	14242461.68	13409507.57	13847001.62	29061547.18	58697069.30
18	2022	2	26229579.05	13409507.57	13847001.62	32867010.36	70376500.92
19	2022	3	38337511.45	13409507.57	13847001.62	36291927.22	82055932.54
20	2022	4	57804859.63	13409507.57	13847001.62	39374352.39	93735364.16

4.2.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีการบวกโซลท์-วินเทอร์

การพยากรณ์ด้วยวิธีการบวกโซลท์-วินเทอร์ เป็นเทคนิคการพยากรณ์เชิงปริมาณที่ใช้สำหรับข้อมูลที่มีแนวโน้มและฤดูกาล (Trend and Seasonality) ที่เป็นแบบเส้นตรง (Additive) โดยการคำนวณจะรวมทั้งแนวโน้มและผลกระทบจากฤดูกาล การพยากรณ์ด้วยวิธีนี้สามารถประเมินค่าที่แม่นยำได้มากขึ้น เหมาะสำหรับข้อมูลที่มีการเปลี่ยนแปลงสม่ำเสมอตามช่วงเวลาและฤดูกาล

จากตารางที่ 4.2 แสดงถึงการพยากรณ์ยอดขายรายไตรมาสในช่วงปี ค.ศ. 2018 - 2022 โดยใช้วิธีการบวกโซลท์-วินเทอร์ ซึ่งจะมีค่า Sales (ยอดขายจริง) และ Forecast (ค่าพยากรณ์) ในช่วงปี ค.ศ. 2018 - 2022 แยกตามไตรมาส ข้อมูลในตารางสะท้อนให้เห็นถึงแนวโน้มและรูปแบบตามฤดูกาลของยอดขายในแต่ละไตรมาสอย่างชัดเจน ดังนี้

4.2.1.1 การพยากรณ์ด้วยวิธีการบวกโซลท์-วินเทอร์

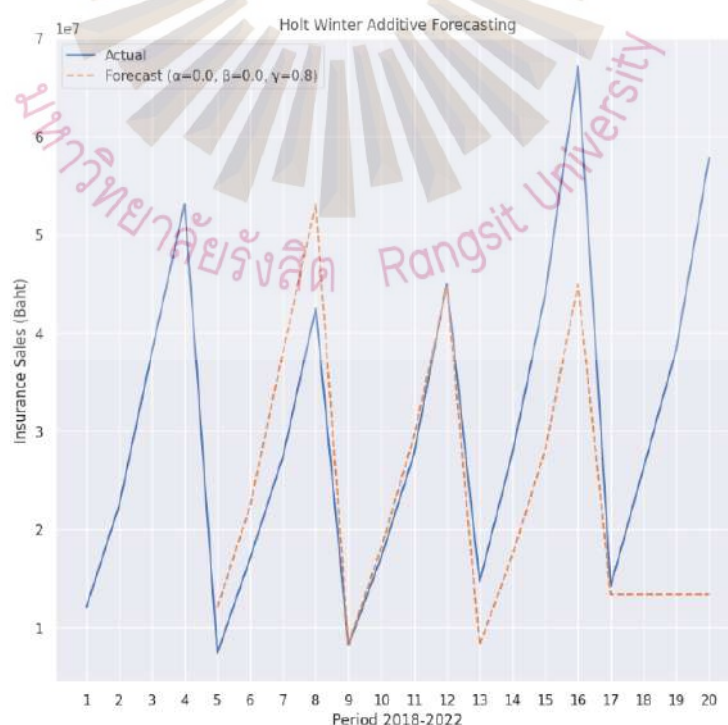
- 1) วิธีการบวกโซลท์-วินเทอร์ เหมาะสำหรับข้อมูลที่มีแนวโน้มตามฤดูกาลและมีการเพิ่มขึ้นหรือลดลงอย่างต่อเนื่อง โดยการเพิ่มหรือลดนั้นมีลักษณะเชิงเส้น
- 2) ในกรณีนี้ ปีแรก (ปี ค.ศ. 2018) ไม่มีข้อมูลการพยากรณ์ เนื่องจากต้องใช้ข้อมูลจริงเพื่อสร้างแบบจำลองพยากรณ์สำหรับปีถัดไป

4.2.1.2 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

- 1) ปี ค.ศ. 2019 การพยากรณ์ค่อนข้างใกล้เคียงกับยอดขายจริงในช่วงแรก ๆ แต่ยังมีคลาดเคลื่อน โดยเฉพาะในไตรมาสที่ 1 (ยอดขายจริงต่ำกว่าค่าพยากรณ์มาก)
- 2) ปี ค.ศ. 2020 การพยากรณ์ยังคงตามยอดขายจริงได้ดีในบางช่วง แต่ยังมี ความแตกต่างพอสมควรในไตรมาสที่ 3 และ 4 ซึ่งคาดการณ์ได้ดีขึ้นในไตรมาสที่ 4 การพยากรณ์ใน ไตรมาสที่ 4 สามารถสะท้อนแนวโน้มได้แม่นยำขึ้น หลังจากปรับปรุงการคำนวณ
- 3) ปี ค.ศ. 2021 ความแตกต่างระหว่างยอดขายจริงกับการพยากรณ์เพิ่มขึ้น อย่างเห็นได้ชัด โดยเฉพาะในไตรมาสที่ 2 และ 3 ที่การพยากรณ์ต่ำกว่ายอดขายจริงมาก
- 4) ปี ค.ศ. 2022 การพยากรณ์คงที่ (Forecast) โดยมีค่าเท่ากับประมาณ 13.4 ล้านสำหรับทุกไตรมาส ซึ่งแตกต่างจากยอดขายจริงที่มีความผันผวนสูงขึ้นอย่างมาก

4.2.1.3 แนวโน้มของการพยากรณ์และยอดขายจริง

ในช่วงปีแรก ๆ (ค.ศ. 2019 - 2020) การพยากรณ์ยังตามยอดขายจริงได้ดีในระดับหนึ่ง แต่เมื่อถึงปี 2021 - 2022 การพยากรณ์เริ่มคลาดเคลื่อนมากขึ้น โดยเฉพาะเมื่อยอดขายจริงมีการเพิ่มขึ้นสูงอย่าง มากในบางไตรมาส ขณะที่การพยากรณ์ยังคงอยู่ในระดับต่ำ



รูปที่ 4.12 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีการบวกโฮลท์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022

การพยากรณ์โดยวิธีการบวกโฮสต์-วินเทอร์สามารถติดตามแนวโน้มได้ดีในระยะสั้นหรือกลาง แต่ไม่สามารถตามการเปลี่ยนแปลงที่มีผันผวนได้ การพยากรณ์ในปี ค.ศ. 2022 ไม่สะท้อนแนวโน้มที่เปลี่ยนแปลงของข้อมูลจริง อาจต้องปรับพารามิเตอร์หรือใช้วิธีการพยากรณ์ที่เหมาะสม

แผนภูมิเส้นจากรูปภาพ 4.12 ที่ได้อัปโหลดแสดงผลการพยากรณ์โดยวิธีการบวกโฮสต์-วินเทอร์ โดยมีเส้นสีฟ้าแสดงถึงค่าจริง (Actual) และเส้นประสีส้มแสดงถึงค่าพยากรณ์ (Forecast) ที่ใช้พารามิเตอร์ $\alpha = 0.0$ $\beta = 0.0$ และ $\gamma = 0.8$

แนวโน้มและฤดูกาล แผนภูมิเส้นแสดงให้เห็นแนวโน้มและรูปแบบฤดูกาลที่ชัดเจนในข้อมูลยอดขายจริง (Actual) โดยมีการขึ้นและลงเป็นระยะ ๆ ในทุกปี โดยมีรายละเอียดดังนี้

1) ข้อมูลจริง (Actual) แสดงแนวโน้มที่ผันผวนเป็นอย่างมาก ขึ้นและลงอย่างรวดเร็ว โดยมีจุดสูงสุดอยู่ที่ช่วงเวลาที่ 3 8 13 และ 18

2) ข้อมูลพยากรณ์ (Forecast) เส้นแผนภูมิสีส้มที่แสดงข้อมูลพยากรณ์จะมีรูปแบบการเคลื่อนไหวคล้ายกับเส้นสีฟ้า แต่ความแตกต่างจะอยู่ที่ค่าที่พยากรณ์มีความชันที่น้อยกว่า และมีการปรับเรียบค่าเพื่อให้สอดคล้องกับแนวโน้มในข้อมูลจริง

3) ผลลัพธ์ของโมเดลการบวกโฮสต์-วินเทอร์ การตั้งค่าพารามิเตอร์ ($\alpha = 0.0$ $\beta = 0.0$ $\gamma = 0.8$) บ่งบอกว่าโมเดลนี้มีการใช้น้ำหนักการปรับระดับตามฤดูกาลค่อนข้างมาก แต่ไม่ได้ปรับค่าของระดับหรือเทรนด์ ส่งผลให้การพยากรณ์คาดการณ์เฉพาะลักษณะตามฤดูกาลได้ดี แต่ไม่สามารถตามแนวโน้มโดยรวมได้

4) ความแตกต่างระหว่าง Actual และ Forecast ในช่วงแรกแผนภูมิเส้นจะสอดคล้องกัน แต่ในช่วงเวลาหลัง (16 - 20) การพยากรณ์แสดงเส้นที่ค่อนข้างคงที่ ซึ่งไม่ตรงกับข้อมูลจริงที่มีความผันผวนมาก

โมเดลพยากรณ์นี้ทำงานได้ดีในบางส่วนที่มีลักษณะตามฤดูกาล (ตามการตั้งค่าของ $\gamma = 0.8$) แต่ยังไม่สามารถจับแนวโน้มในระยะยาวได้ดี เนื่องจากค่าพารามิเตอร์ของ α และ β เท่ากับ 0 และจำกัดความสามารถโมเดลในการเรียนรู้เทรนด์ระยะยาวจากข้อมูลที่เปลี่ยนแปลงต่อเนื่อง

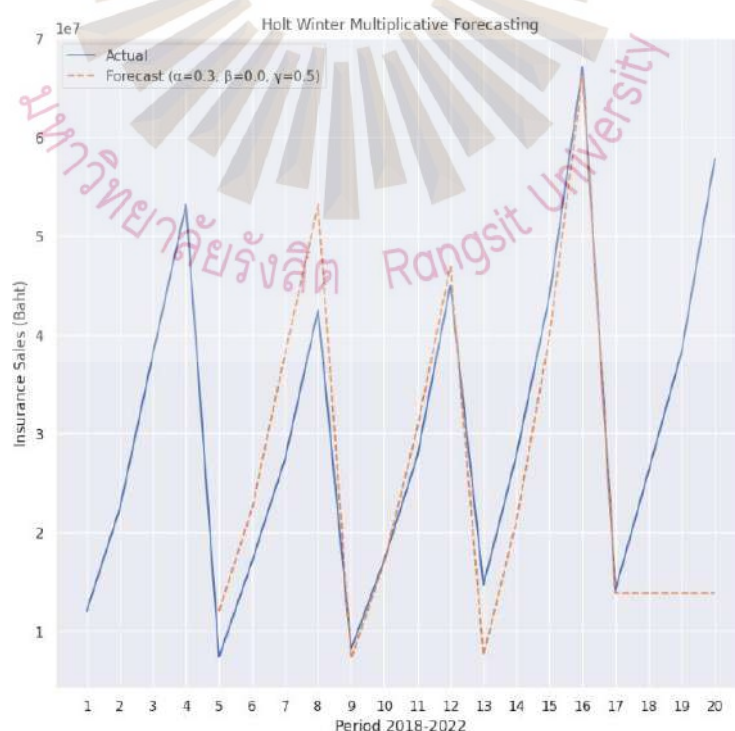
4.2.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีการคูณ โฮสต์-วินเทอร์

จากตารางที่ 4.2 แสดงข้อมูลยอดขายจริงและค่าพยากรณ์รายไตรมาสในช่วงปี ค.ศ. 2018 - 2022 ซึ่งได้จากการใช้วิธีคูณโฮสต์-วินเทอร์ โดยมุ่งเน้นการประเมินประสิทธิภาพของการพยากรณ์

ผ่านการเปรียบเทียบระหว่างยอดขายจริงกับค่าพยากรณ์ รวมถึงการตรวจสอบข้อผิดพลาดหรือความคลาดเคลื่อนที่อาจเกิดขึ้น เพื่อสรุปข้อดี ข้อจำกัดของโมเดล และเสนอแนวทางปรับปรุงให้มีความแม่นยำมากยิ่งขึ้น ข้อมูลจากตารางดังกล่าวเปิดโอกาสให้สามารถระบุช่วงเวลาที่โมเดลมีความคลาดเคลื่อนสูงได้อย่างชัดเจน ซึ่งเป็นประโยชน์ต่อการประเมินความเสถียรของการพยากรณ์ และยังเป็นแนวทางในการพัฒนาการจัดการข้อมูลเบื้องต้นให้มีความถูกต้อง ครบถ้วน และเหมาะสมต่อการวิเคราะห์เชิงลึกอย่างมีประสิทธิภาพ

แผนภูมิเส้นจากรูปที่ 4.13 แสดงผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก โดยใช้วิธีการคูณโฮลต์-วินเทอร์ ซึ่งเป็นเทคนิคการพยากรณ์ที่เหมาะสมสำหรับข้อมูลที่มีแนวโน้มและฤดูกาลชัดเจน การพยากรณ์นี้ใช้ค่า $\alpha = 0.3$ $\beta = 0.0$ และ $\gamma = 0.5$ เพื่อสะท้อนรูปแบบความผันผวนตามฤดูกาลของข้อมูลจริงและเปรียบเทียบค่าจริงกับค่าที่คาดการณ์ในช่วงปี ค.ศ. 2018 - 2022 จากกราฟจะเห็นได้ว่าค่าพยากรณ์ใกล้เคียงข้อมูลจริง โดยเฉพาะช่วงที่มีฤดูกาลเด่นชัด แสดงถึงความเหมาะสมของโมเดลในการประยุกต์ใช้กับข้อมูลดังกล่าว โดยมีรายละเอียดดังนี้

4.2.2.1 การตรวจสอบแนวโน้ม (Trend Analysis)



รูปที่ 4.13 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยใช้วิธีการคูณโฮลต์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022

1) ยอดขายจริง (Sales) มีแนวโน้มเพิ่มขึ้นในช่วงปี ค.ศ. 2018 - 2019 แต่ในปี ค.ศ. 2020 ยอดขายลดลง โดยเฉพาะในไตรมาสที่ 1 และ 2 ซึ่งอาจบ่งบอกถึงเหตุการณ์ภายนอก (เช่น ภาวะเศรษฐกิจ หรือการแพร่ระบาดของ COVID-19) ที่ส่งผลกระทบต่อยอดขาย

2) ในปี ค.ศ. 2021 ยอดขายกลับมาเพิ่มขึ้นอีกครั้ง โดยเฉพาะในไตรมาสที่ 4 ซึ่งมียอดขายสูงที่สุดของตารางที่ 67,116,178.95

3) ปี ค.ศ. 2022 ยอดขายในไตรมาสแรกค่อนข้างต่ำเมื่อเทียบกับปีที่แล้ว แต่ไตรมาสที่ 4 กลับเพิ่มขึ้นเป็น 57,804,859.63 ซึ่งเป็นระดับสูงเมื่อเทียบกับค่าเฉลี่ยโดยรวม และสะท้อนถึงการฟื้นตัวของยอดขายในช่วงปลายปี

4) การตรวจสอบค่าพยากรณ์ (Forecast Analysis)

4.1) ค่าพยากรณ์ที่แสดงในตารางเริ่มจากปี ค.ศ. 2019 โดยค่าพยากรณ์ของปี ค.ศ. 2019 นั้นสอดคล้องกับยอดขายจริงในปี ค.ศ. 2018 (เช่น ไตรมาส 1 ปี ค.ศ. 2019 มีค่าพยากรณ์ 12,064,969.75 ซึ่งเท่ากับยอดขายจริงในไตรมาส 1 ปี ค.ศ. 2018) นั้นหมายความว่าค่าพยากรณ์อาจใช้ข้อมูลยอดขายปีที่แล้วมาเป็นเกณฑ์

4.2) ปี ค.ศ. 2020 การพยากรณ์ยอดขายเริ่มมีความคลาดเคลื่อนจากยอดขายจริง โดยเฉพาะในช่วงไตรมาสที่ 3 และ 4 ที่ยอดขายจริงต่ำกว่าค่าพยากรณ์ เนื่องจากผลกระทบจากสถานการณ์โควิด-19 ทำให้ความต้องการสินค้าเปลี่ยนแปลงอย่างรวดเร็ว

4.3) ปี ค.ศ. 2021 และ ค.ศ. 2022 ค่าพยากรณ์เริ่มมีความแม่นยำน้อยลง โดยค่าพยากรณ์ในปี 2022 มีค่าน้อยมาก (ประมาณ 13,847,001.62) ซึ่งต่ำกว่ายอดขายจริงในทุกไตรมาส การพยากรณ์ในปี 2022 ผิดพลาดเนื่องจากผลกระทบจากภาวะเศรษฐกิจและสถานการณ์ไม่แน่นอน

5) ความคลาดเคลื่อนจากค่าพยากรณ์ (Error Analysis)

5.1) ความคลาดเคลื่อนระหว่างยอดขายจริงและค่าพยากรณ์ (Forecast Error) ในหลายช่วงเวลาแสดงให้เห็นว่าค่าพยากรณ์ไม่สอดคล้องกับแนวโน้มที่เกิดขึ้น โดยเฉพาะในปี ค.ศ. 2021 - 2022 ซึ่งแสดงให้เห็นว่าการคำนวณค่าฤดูกาลและแนวโน้มอาจไม่ถูกต้อง หรือมีปัจจัยภายนอกที่ไม่ถูกนำมาพิจารณา

5.2) วิธีการพยากรณ์แบบการคูณ Holt-Winters จะเหมาะสมกับข้อมูลที่มีความเป็นฤดูกาลและแนวโน้มที่ชัดเจน ซึ่งข้อมูลในช่วงปี ค.ศ. 2019 - 2022 อาจไม่เป็นไปตามลักษณะดังกล่าว ทำให้เกิดความคลาดเคลื่อน การพยากรณ์ในตารางนี้มีความแม่นยำในช่วงปีแรก ๆ แต่ความแม่นยำลดลงในช่วงปีหลังซึ่งอาจเกิดจากการเปลี่ยนแปลงของแนวโน้มที่ไม่สามารถคาดการณ์ได้ด้วยวิธีการคูณ Holt-Winters ที่ใช้อยู่

6) เส้นจริง (Actual) เส้นสีฟ้าแสดงค่าจริงของข้อมูลในช่วงเวลาที่กำหนด (ค.ศ. 2018 - 2022) โดยมีค่าตั้งแต่ประมาณ 1 ถึง 7 (1e7 หมายถึง เลขศูนย์จำนวน 7 ตัว) ซึ่งแสดงให้เห็นว่ามีความผันผวนสูงขึ้นและลงอย่างมากตลอดช่วงเวลา

7) เส้นพยากรณ์ (Forecast) เส้นประสีส้มแสดงผลการพยากรณ์ที่ได้จากวิธีการคูณโฮลท์-วินเทอร์ ด้วยค่าพารามิเตอร์ $\text{Alpha} = 0.3$ $\text{Beta} = 0.0$ และ $\text{Gamma} = 0.5$ โดยเส้นพยากรณ์จะมีรูปแบบคล้ายกับเส้นจริง แต่จะคาดการณ์ค่าที่สูงหรือต่ำกว่าในบางช่วงเวลา

8) ความแม่นยำของการพยากรณ์ ผลการพยากรณ์ใกล้เคียงกับค่าจริงในบางช่วงเวลา (เช่น ช่วงที่ 11 - 14) แต่จะมีความคลาดเคลื่อนในช่วงอื่น ๆ เช่น ช่วงที่ 5 - 8 และ 15 - 17 ซึ่งอาจเกิดจากค่าพารามิเตอร์ที่ใช้ในการพยากรณ์ หรืออาจเป็นเพราะข้อมูลมีลักษณะไม่เสถียร หรือมีความผันผวนที่สูง

วิธีการคูณโฮลท์-วินเทอร์สามารถจับรูปแบบของข้อมูลได้บ้าง แต่ค่าพารามิเตอร์ที่ใช้ อาจไม่เหมาะสมทั้งหมด ทำให้เกิดความคลาดเคลื่อนในบางช่วง ควรปรับค่าพารามิเตอร์ (Alpha Beta และ Gamma) ให้เหมาะสมมากขึ้นเพื่อเพิ่มความแม่นยำของการพยากรณ์ โดยอาจใช้เทคนิคการค้นหาค่าพารามิเตอร์ที่เหมาะสมที่สุดผ่านการทดสอบค่าต่าง ๆ หรือการใช้วิธีการปรับค่าพารามิเตอร์อัตโนมัติเพื่อให้การพยากรณ์มีความแม่นยำสูงสุดในทุกช่วงเวลา การปรับค่าพารามิเตอร์จะเพิ่มประสิทธิภาพการพยากรณ์และลดความคลาดเคลื่อน

4.2.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

จากตารางที่ 4.2 ข้อมูลการพยากรณ์ด้วยวิธีเอ็กซ์โพเนนเชียลซิมเปิลเอ็กซ์โพเนนเชียลสามารถพิจารณาได้จากยอดขาย (Sales) และค่าพยากรณ์ (Forecast) ในแต่ละไตรมาสของแต่ละปี โดยจะพิจารณาถึงการเปลี่ยนแปลงของยอดขายจริงและความแม่นยำของการพยากรณ์ โดยใช้ค่าความต่างระหว่างยอดขายจริงและค่าพยากรณ์ ในช่วงปี ค.ศ. 2018 - 2022 แยกตามไตรมาสดังนี้

4.2.3.1 วิธีการปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (SES) ใช้ค่าเฉลี่ยถ่วงน้ำหนักของข้อมูลในอดีตเพื่อพยากรณ์ค่าในอนาคต โดยข้อมูลที่ใกล้เคียงปัจจุบันจะมีน้ำหนักมากกว่า

1) ในกรณีนี้ การพยากรณ์สำหรับไตรมาสแรก (ค.ศ. 2018 Q1) ใช้ยอดขายจริงเป็นค่าพื้นฐานในการเริ่มต้นคำนวณ กำหนดทิศทางแนวโน้มและลดข้อผิดพลาดของโมเดลในระยะถัดไปอย่างมีประสิทธิภาพและยั่งยืน

4.2.3.2 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

1) ปี ค.ศ. 2018 การพยากรณ์เริ่มต้นจากยอดขายจริงในไตรมาสแรก และปรับเปลี่ยนเล็กน้อยในไตรมาสต่อมา การพยากรณ์ยังไม่สามารถติดตามแนวโน้มการเพิ่มขึ้นที่ชัดเจนในยอดขายจริงได้

2) ปี ค.ศ. 2019 การพยากรณ์ค่อย ๆ ปรับตัวตามยอดขายจริง แต่ยังคงมีความแตกต่างระหว่างค่าพยากรณ์และยอดขายจริงอยู่ โดยเฉพาะในไตรมาสที่ 3 และ 4 ที่การพยากรณ์ต่ำกว่ายอดขายจริง

3) ปี ค.ศ. 2020 การพยากรณ์เริ่มใกล้เคียงกับยอดขายจริงมากขึ้นในบางไตรมาส แต่ยังคงมีความแตกต่างในไตรมาสที่ 4 ซึ่งยอดขายจริงสูงกว่าที่พยากรณ์ อาจเกิดจากปัจจัยภายนอกที่มีผลกระทบในช่วงนั้น

4) ปี ค.ศ. 2021 - 2022 การพยากรณ์เริ่มตามทันยอดขายจริงมากขึ้น โดยในปี 2022 ไตรมาส 2 และ 3 การพยากรณ์ค่อนข้างใกล้เคียงกับยอดขายจริง แต่ยังคงมีความแตกต่างเล็กน้อยในไตรมาสที่ 4 ซึ่งยอดขายจริงเพิ่มขึ้นมากกว่าที่พยากรณ์ไว้

4.2.3.3 แนวโน้มของการพยากรณ์และยอดขายจริง

ในช่วงปีแรก ๆ การพยากรณ์ยังไม่สามารถตามยอดขายจริงที่มีแนวโน้มเพิ่มขึ้นได้ดี แต่เมื่อผ่านไปแล้ว การพยากรณ์เริ่มปรับตัวใกล้เคียงกับแนวโน้มของยอดขายจริงมากขึ้น โดยเฉพาะในปี ค.ศ. 2021 - 2022

จากการพยากรณ์ยอดขายประกันชีวิตด้วยวิธีเอ็กซ์โพเนนเชียลอย่างง่าย พบว่าแบบจำลองสามารถตามแนวโน้มของข้อมูลได้ดี แม้จะมีความแตกต่างในช่วงแรก การพยากรณ์จะดีขึ้นเมื่อปรับค่าถ่วงน้ำหนักตามข้อมูลในอดีต แต่มีข้อจำกัดหากข้อมูลผันผวนสูง วิธีนี้เหมาะกับข้อมูลที่มีแนวโน้มไม่ซับซ้อนหรือการเปลี่ยนแปลงเชิงเส้น หากข้อมูลมีการเปลี่ยนแปลงซับซ้อน ควรพิจารณาวิธีพยากรณ์ที่ซับซ้อนกว่า เช่น Holt-Winters Additive หรือ Multiplicative ซึ่งสามารถจับลักษณะฤดูกาลและแนวโน้มได้ดีกว่า

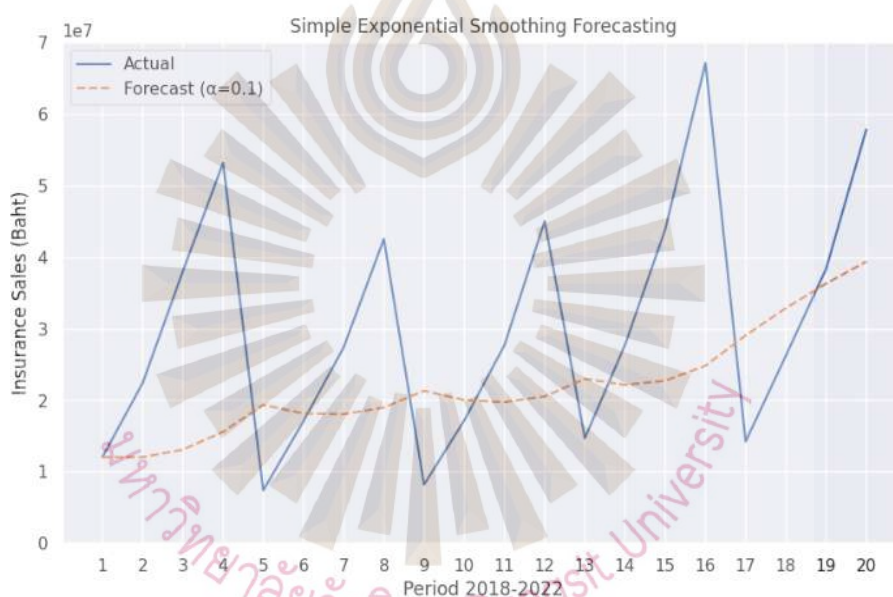
แผนภูมิเส้นจากรูปที่ 4.14 แสดงการเปรียบเทียบระหว่างยอดขายจริง (Actual) และค่าพยากรณ์ (Forecast) ที่ได้จากการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย โดยใช้ค่าความราบเรียบ ($\alpha = 0.1$) สำหรับช่วงเวลาตั้งแต่ปี ค.ศ. 2018 - 2022 เพื่อประเมินแนวโน้มและความแม่นยำของการพยากรณ์

4.2.3.4 Actual (เส้นสีน้ำเงิน)

- 1) แสดงค่าจริงผันผวนค่อนข้างมาก โดยมีจุดพีคสูงต่ำชัดเจน
- 2) ลักษณะของข้อมูลบ่งบอกถึงการเกิดความผันผวนสูงในแต่ละช่วงเวลา

4.2.3.5 Forecast (เส้นประสีส้ม)

- 1) แสดงค่าพยากรณ์ที่ค่อนข้างราบเรียบและไม่สามารถติดตามแนวโน้มของค่าจริงได้ดีมากนัก
- 2) การใช้ค่า $\alpha = 0.1$ ทำให้ค่าพยากรณ์ตอบสนองต่อการเปลี่ยนแปลงของข้อมูลจริงช้า



รูปที่ 4.14 แผนภูมิเส้นแสดงการพยากรณ์ประกันชีวิตกรมธรรม์หลัก โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ระหว่างปี ค.ศ. 2018 - 2022

4.6.3.6 ไตรมาสที่ 1 (ช่วงที่ 1 5 9 13 17)

- 1) ค่าจริง (Actual) เริ่มต้นต่ำและมีแนวโน้มเพิ่มขึ้นเรื่อย ๆ
- 2) มักจะเป็นจุดเริ่มต้นที่ค่าต่ำสุดในรอบรอบถัดไป

4.6.3.7 ไตรมาสที่ 2 (ช่วงที่ 2 6 10 14 18)

- 1) ค่าจริงเพิ่มขึ้นอย่างรวดเร็วและสูงกว่าค่าเฉลี่ยของช่วงอื่น ๆ
- 2) มักจะเป็นจุดที่มีค่าพีคสูงที่สุดหรือใกล้เคียงในแผนภูมิเส้น

4.6.3.8 ไตรมาสที่ 3 (ช่วงที่ 3 7 11 15 19)

- 1) ค่าจริงเริ่มลดลงหลังจากพีคของไตรมาสที่ 2
- 2) เป็นช่วงที่มักจะเห็นการลดลงอย่างมากหลังจากค่าพีค

4.6.3.9 ไตรมาสที่ 4 (ช่วงที่ 4 8 12 16 20)

- 1) ค่าจริงยังคงลดลงและมักจะถึงจุดต่ำสุด
- 2) ช่วงนี้มักเป็นจุดที่ค่ากลับมาเริ่มต้นใหม่ในรอบถัดไป

ข้อมูลแสดงให้เห็นรูปแบบตามฤดูกาลที่ชัดเจน โดยมีการเพิ่มขึ้นอย่างรวดเร็วในไตรมาสที่ 2 ของทุกปี และการลดลงอย่างรวดเร็วในไตรมาสที่ 4 การพยากรณ์ด้วยวิธีซิมเปลเอ็กซ์โพเนนเชียลไม่สามารถจับการผันผวนเหล่านี้ได้ด้นัก เนื่องจากเป็นวิธีการที่ราบเรียบ ซึ่งอาจเหมาะกับข้อมูลที่มีแนวโน้มคงที่มากกว่า แนะนำให้ใช้วิธีการพยากรณ์ที่สามารถจับการผันผวนตามฤดูกาลได้ดีขึ้น เช่น วิธีการ Holt-Winters หรือ SARIMA เพื่อการพยากรณ์ที่แม่นยำมากขึ้น ทั้งนี้วิธีเหล่านี้สามารถจัดการกับฤดูกาลและแนวโน้มได้อย่างมีประสิทธิภาพ

4.2.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

จากตารางที่ 4.2 เป็นการพยากรณ์ยอดขายโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าระหว่างปี ค.ศ. 2018 - 2022 แยกตามไตรมาส

4.2.4.1 การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

1) วิธีการนี้เหมาะกับข้อมูลที่มีแนวโน้มที่ชัดเจน (Trend) โดยจะใช้การปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าเพื่อให้สามารถติดตามแนวโน้มการเพิ่มขึ้นหรือลดลงของข้อมูลได้ดีขึ้น เนื่องจากสามารถลดผลกระทบจากความผันผวนและสะท้อนทิศทางของข้อมูลได้อย่างแม่นยำมากยิ่งขึ้น ทำให้การพยากรณ์มีความแม่นยำและสามารถปรับตัวตามการเปลี่ยนแปลงได้ดี รวมถึงสนับสนุนการตัดสินใจในระดับที่ดีกว่า

2) การพยากรณ์ยอดขายที่คำนวณได้จะมีการปรับค่าตามทั้งระดับ (Level) และแนวโน้ม (Trend) ของข้อมูลในอดีต การปรับค่าดังกล่าวส่งผลให้การพยากรณ์สะท้อนถึงพฤติกรรมในอดีตและทิศทางในอนาคตได้แม่นยำยิ่งขึ้น โดยสามารถคาดการณ์การเปลี่ยนแปลงและความผันผวนในอนาคตได้อย่างมีประสิทธิภาพ

4.2.4.2 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

1) ปี ค.ศ. 2018 การพยากรณ์เริ่มจากช่วงไตรมาสที่ 3 ซึ่งการพยากรณ์ต่ำกว่ายอดขายจริงเล็กน้อย การพยากรณ์ในไตรมาสที่ 4 ก็ดูเหมือนจะติดตามแนวโน้มของยอดขายจริงได้ใกล้เคียง

2) ปี ค.ศ. 2019 ในไตรมาสที่ 1 การพยากรณ์สูงกว่ายอดขายจริงมาก และมีความผันผวนสูงในช่วงไตรมาสถัดไป การพยากรณ์ไตรมาสที่ 2 ถึง 4 ต่ำกว่ายอดขายจริงทั้งหมด โดยเฉพาะในไตรมาสที่ 4 ซึ่งยอดขายจริงสูงกว่ามาก

3) ในปี ค.ศ. 2020 การพยากรณ์ในช่วงปีที่ไม่สามารถติดตามแนวโน้มที่เกิดขึ้นจริงได้อย่างแม่นยำ อาจเกิดจากปัจจัยภายนอกที่มีผลกระทบต่อการเปลี่ยนแปลงในตลาด

4) ปี ค.ศ. 2021 - 2022 การพยากรณ์เริ่มปรับเพิ่มขึ้นอย่างต่อเนื่องในช่วงปี ค.ศ. 2021 - 2022 อย่างไรก็ตาม ยอดขายจริงมีความผันผวนค่อนข้างมากและไม่เป็นแนวโน้มเดียวกับการพยากรณ์ที่เกิดขึ้น

4.2.4.3 แนวโน้มของการพยากรณ์และยอดขายจริง

1) การพยากรณ์โดยใช้วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าจะมีแนวโน้มเพิ่มขึ้นอย่างต่อเนื่องตั้งแต่ปี ค.ศ. 2020 เป็นต้นไป แต่ยอดขายจริงมีแนวโน้มที่ผันผวนสูงและไม่ตรงกับการพยากรณ์ในหลายช่วงเวลา

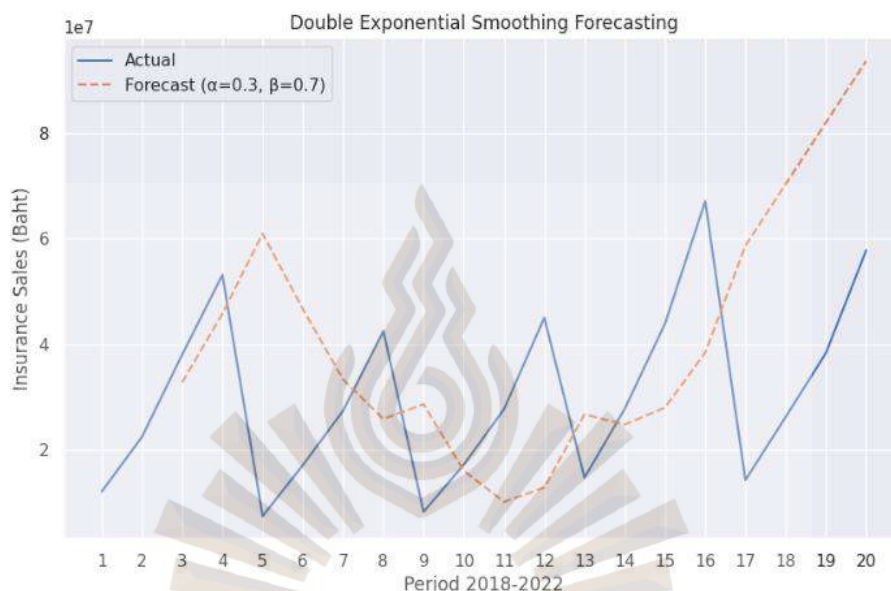
2) มีแนวโน้มว่ายอดขายจริงในปี ค.ศ. 2021 - 2022 นั้นผันผวนมากขึ้น อาจเป็นผลมาจากปัจจัยภายนอกหรือปัจจัยที่ไม่คงที่

4.2.4.4 ความแม่นยำของการพยากรณ์ วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าแสดงให้เห็นว่าสามารถติดตามแนวโน้มของยอดขายได้ดีในช่วงปี ค.ศ. 2018 - 2019 แต่มีความแตกต่างมากในช่วงปีต่อ ๆ มา เนื่องจากยอดขายจริงมีความผันผวนสูง

1) ข้อดี วิธีนี้มีความเหมาะสมในการพยากรณ์ข้อมูลที่มีแนวโน้มเชิงเส้นหรือการเปลี่ยนแปลงที่ค่อนข้างคงที่

2) ข้อจำกัด หากข้อมูลมีความผันผวนสูงหรือแนวโน้มไม่ชัดเจน เช่นในกรณีของปี ค.ศ. 2020 - 2022 วิธีนี้อาจให้ผลลัพธ์ที่ไม่แม่นยำ จำเป็นต้องใช้วิธีการที่ซับซ้อนมากขึ้น เช่น Triple Exponential Smoothing หรือวิธีการพยากรณ์ที่รองรับการเปลี่ยนแปลงเชิงซับซ้อนกว่า การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าสามารถให้แนวโน้มเบื้องต้นของยอดขายในช่วงที่แนวโน้มค่อนข้างชัดเจน แต่ไม่เหมาะสมกับข้อมูลที่มีการผันผวนหรือแนวโน้มที่ซับซ้อน

รูปที่ 4.15 แผนภูมิเส้นนี้แสดงผลการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า โดยมีค่าพารามิเตอร์ $\alpha = 0.3$ และ $\beta = 0.7$ ซึ่งถูกใช้ในการพยากรณ์ข้อมูลในช่วงปี ค.ศ. 2018 - 2022



รูปที่ 4.15 แผนภูมิเส้นแสดงการพยากรณ์ประกันชีวิตกรมธรรม์หลักโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ระหว่างปี ค.ศ. 2018 - 2022

4.2.4.5 เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในแต่ละช่วงเวลา โดยค่าจริงมีการเปลี่ยนแปลงขึ้นลงไม่สม่ำเสมอ โดยมีทั้งช่วงเพิ่มขึ้นที่ชัดเจนและช่วงลดลงอย่างรวดเร็ว

4.2.4.6 เส้นประสีส้ม (Forecast) เป็นการพยากรณ์จากปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ซึ่งสนับสนุนการคาดการณ์แนวโน้มข้อมูลในอนาคต โดยสังเกตเห็นว่าในช่วงต้นแผนภูมิเส้น การพยากรณ์ใกล้เคียงกับค่าจริง แต่เมื่อถึงช่วงท้ายการพยากรณ์ ค่าที่พยากรณ์มีแนวโน้มเพิ่มขึ้นอย่างต่อเนื่องและไม่ตรงกับค่าจริงในหลายจุด โดยเฉพาะช่วงหลังจาก Period 16 เป็นต้นไป

4.2.4.7 แนวโน้มการพยากรณ์ ค่าการพยากรณ์แสดงแนวโน้มที่เพิ่มขึ้นอย่างชัดเจน หลังจากช่วงที่ 14 ถึงช่วงที่ 20 ซึ่งอาจแสดงถึงการพยากรณ์ว่าแนวโน้มของข้อมูลจะเพิ่มขึ้นเรื่อยๆ ในอนาคต การเพิ่มขึ้นของค่าพยากรณ์แสดงการคาดการณ์การเติบโตในอนาคตตามแนวโน้มที่ชัดเจนและต่อเนื่อง ซึ่งสามารถสนับสนุนการวางแผนกลยุทธ์และการตัดสินใจทางธุรกิจที่มีประสิทธิภาพ

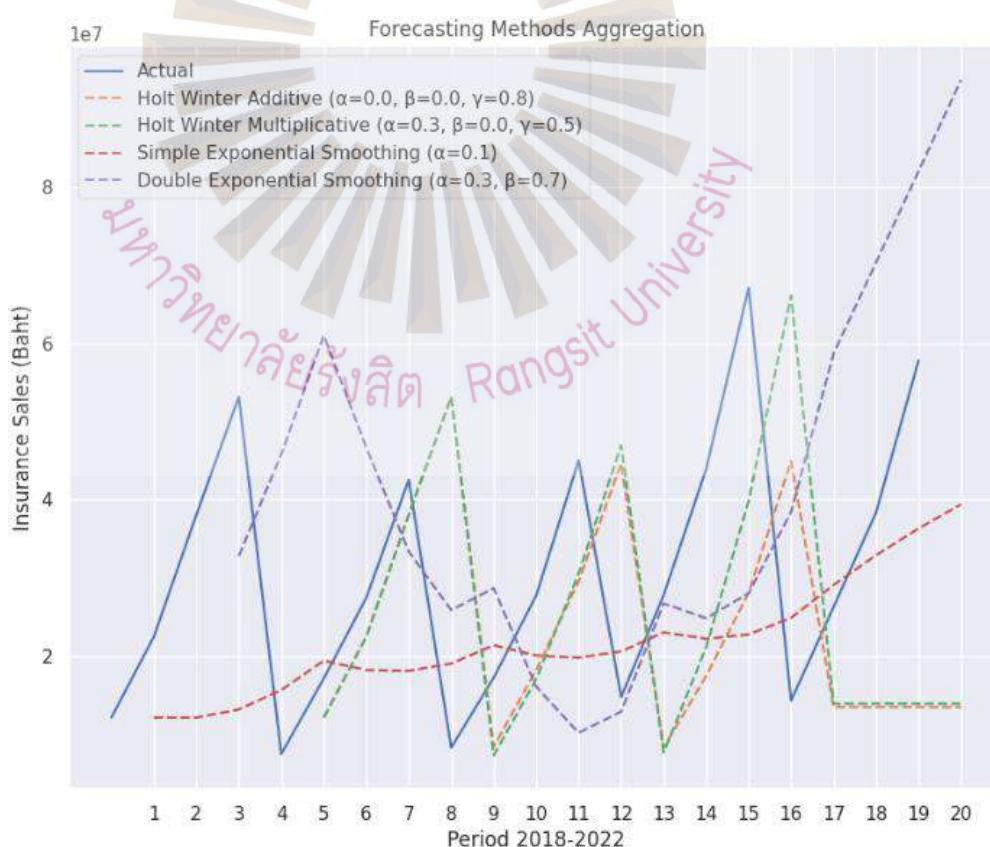
การพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ในแผนภูมิเส้นนี้ทำงานได้ดีในช่วงต้นของข้อมูล แต่ในช่วงหลังค่าพยากรณ์มีแนวโน้มเบี่ยงเบนจากค่าจริงและแสดงการเพิ่มขึ้นอย่างรวดเร็วเกินไป

4.2.5 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักด้วยวิธีต่าง ๆ

เมื่อนำผลทั้ง 4 วิธี มารวมกัน จะได้ดังรูปที่ 4.16 แผนภูมิเส้นที่แสดงการต่าง ๆ เปรียบเทียบกับข้อมูลยอดขายจริงในช่วงปี ค.ศ. 2018 - 2022

4.2.5.1 ยอดขายเพิ่มขึ้นในช่วงปีหลังจาก ค.ศ. 2020

การระบาดของโควิด-19 ในปี 2020 ทำให้ผู้บริโภคตระหนักถึงความสำคัญของการประกันชีวิตมากขึ้น โดยเฉพาะในด้านการป้องกันความเสี่ยงจากสุขภาพและการดูแลครอบครัวในกรณีฉุกเฉิน นอกจากนี้ ผู้บริโภคยังมีแนวโน้มที่จะมองหาความมั่นคงทางการเงินในช่วงที่มีวิกฤตการณ์ต่าง ๆ ส่งผลให้ความต้องการในผลิตภัณฑ์ประกันชีวิตเพิ่มสูงขึ้น



รูปที่ 4.16 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลักโดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022

4.2.5.2 ยอดขายลดลงในช่วงปี ค.ศ. 2020 - 2021

การระบาดของโควิด-19 ส่งผลกระทบต่อเศรษฐกิจทั่วโลก การเลิกจ้างงานและความไม่มั่นคงทางเศรษฐกิจทำให้ผู้บริโภคลดการใช้จ่ายในผลิตภัณฑ์ที่ไม่จำเป็น นอกจากนี้ ความผันผวนทางเศรษฐกิจอาจทำให้บางคนยกเลิกหรือไม่ต่ออายุกรมธรรม์ประกันชีวิต

4.2.5.3 พฤติกรรมการซื้อประกันที่เปลี่ยนแปลงไป

ผู้บริโภคให้ความสนใจผลิตภัณฑ์ประกันที่ครอบคลุมสุขภาพและมีผลตอบแทนการลงทุนมากขึ้น ส่งผลให้ยอดขายประกันชีวิตมีความผันผวนตามสถานการณ์เศรษฐกิจและความต้องการของตลาด

แผนภูมิเส้นแสดงความผันผวนของยอดขายกรมธรรม์ประกันชีวิตในช่วงปี ค.ศ. 2018 - 2022 ซึ่งได้รับผลกระทบจากโควิด-19 ช่วงที่โควิดระบาดหนักในปี ค.ศ. 2020 - 2021 ยอดขายลดลง แต่หลังจากนั้นยอดขายฟื้นตัวขึ้นเนื่องจากผู้บริโภคตระหนักถึงความสำคัญของประกันชีวิต บางบริษัทปรับกลยุทธ์การตลาดและผลิตภัณฑ์ให้ตอบโจทย์สถานการณ์

4.2.6 ผลลัพธ์การวัดข้อผิดพลาดการพยากรณ์กรมธรรม์หลัก

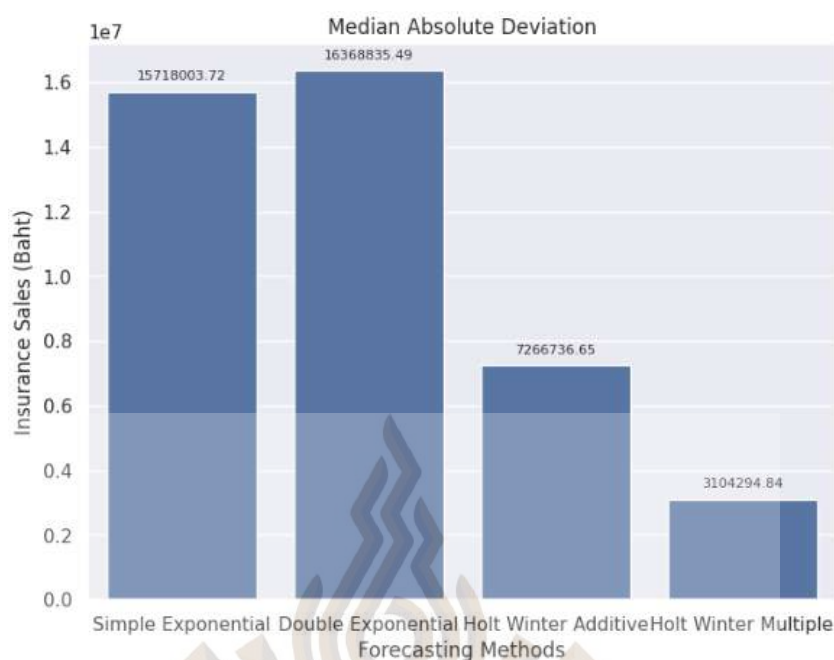
ในการประเมินประสิทธิภาพของโมเดลพยากรณ์ที่ใช้กับข้อมูลของกรมธรรม์หลัก ได้มีการวัดค่าความผิดพลาดของการพยากรณ์ด้วยตัวชี้วัดมาตรฐาน ได้แก่ MAD (Mean Absolute Deviation), MAPE (Mean Absolute Percentage Error) และ MSE (Mean Squared Error) เพื่อวิเคราะห์ความแม่นยำของโมเดล ดังนี้

4.2.6.1 ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD)

แผนภูมิแท่งรูปที่ 4.17 นำเสนอผลการเปรียบเทียบค่าความคลาดเคลื่อน Median Absolute Deviation (MAD) ของวิธีการพยากรณ์จำนวน 4 วิธี

1) ซิมเปิลเอ็กซ์โพเนนเชียล

ค่า MAD ที่ได้คือ 15,718,003.72 ซึ่งค่อนข้างสูง แสดงว่าการพยากรณ์โดยวิธีนี้มีค่าความคลาดเคลื่อนสูง หมายความว่าความแม่นยำในการพยากรณ์ไม่ดีนัก การทำให้เรียบแบบเอกซ์โพเนนเชียลอย่างง่ายไม่ได้คำนึงถึงแนวโน้มหรือฤดูกาลในข้อมูลอนุกรมเวลา จึงอาจไม่เหมาะสมกับข้อมูลที่มีรูปแบบซับซ้อนหรือข้อมูลที่มีความผันผวนสูง



รูปที่ 4.17 แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก

2) ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าค่า MAD คือ 16,368,835.49 ซึ่งสูงที่สุดในทั้ง 4 วิธี แสดงว่าการใช้วิธีการทำให้เรียบแบบเอกซโพเนนเชียลสองชั้นให้ผลลัพธ์ที่มีความคลาดเคลื่อนสูงที่สุดในที่นี้ ถึงแม้ว่าวิธีนี้จะคำนึงถึงแนวโน้มของข้อมูล แต่ไม่สามารถจับรูปแบบฤดูกาลได้ ทำให้ความแม่นยำในการพยากรณ์ยังคงต่ำ โดยเฉพาะในข้อมูลที่มีความแปรผันตามฤดูกาลอย่างชัดเจน วิธีนี้จึงอาจไม่เหมาะสมในบริบทดังกล่าว

3) การบวกโฮสต์-วินเทอร์

ค่า MAD ลดลงมาที่ 7,266,736.65 ซึ่งดีกว่าวิธีการก่อนหน้านี้ วิธีนี้สามารถจับแนวโน้มและฤดูกาลของข้อมูลได้ แสดงว่าการเพิ่มองค์ประกอบฤดูกาลในการพยากรณ์ทำให้ค่าความคลาดเคลื่อนลดลง และสนับสนุนให้การพยากรณ์แม่นยำขึ้นสำหรับข้อมูลที่มีฤดูกาลคงที่ อีกทั้งยังสามารถตอบสนองต่อการเปลี่ยนแปลงตามรอบเวลาที่เกิดขึ้นอย่างสม่ำเสมอได้ดีกว่า

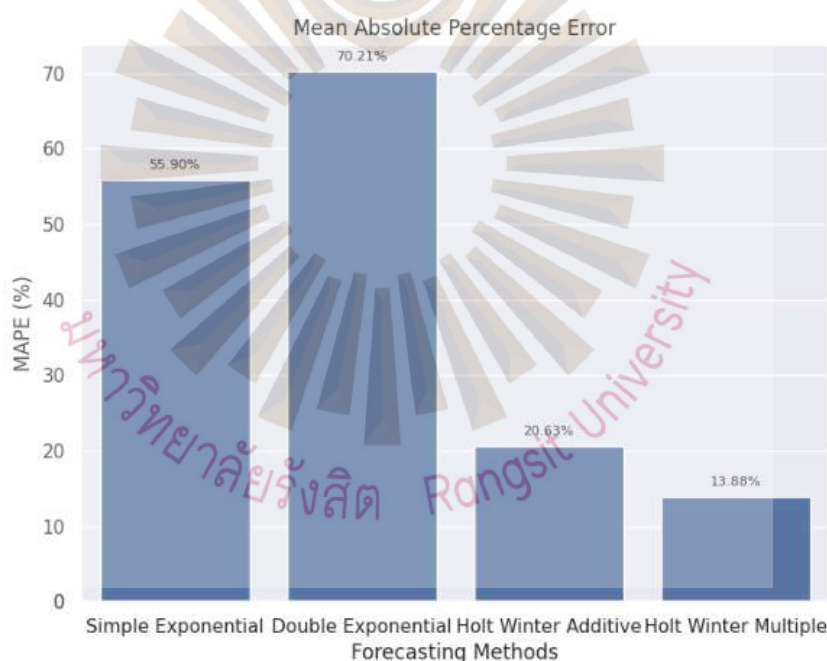
4) การคูณโฮสต์-วินเทอร์

วิธีนี้ให้ค่า MAD ต่ำที่สุดที่ 3,104,294.84 ซึ่งแสดงว่ามีค่าความคลาดเคลื่อนต่ำที่สุดเมื่อเทียบกับวิธีอื่น ๆ Holt-Winters แบบคูณจะเหมาะกับข้อมูลที่มีรูปแบบฤดูกาลที่มีการเปลี่ยนแปลงตามระดับของข้อมูล โดยการพยากรณ์ด้วยวิธีนี้สามารถจับความผันผวนตามฤดูกาลได้อย่างมีประสิทธิภาพ ซึ่งส่งผลให้ค่าความคลาดเคลื่อนลดลงมากที่สุด

จากค่า MAD แสดงถึงความคลาดเคลื่อนของการพยากรณ์แต่ละวิธี โดยค่า MAD ที่น้อยที่สุดหมายถึงการพยากรณ์ที่แม่นยำที่สุด สำหรับข้อมูลในแผนภูมิ วิธีการการคูณโฮลท์-วินเทอร์ ให้ค่าความคลาดเคลื่อนต่ำที่สุด แสดงว่ามีความแม่นยำในการพยากรณ์มากที่สุด ขณะที่วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ให้ค่าความคลาดเคลื่อนสูงสุด ซึ่งแสดงให้เห็นถึงความแม่นยำที่น้อยที่สุด

4.2.6.2 ค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE)

แผนภูมิแท่งรูปที่ 4.18 แสดงค่า Mean Absolute Percentage Error (MAPE) ซึ่งเป็นตัวชี้วัดที่ใช้ประเมินความแม่นยำของการพยากรณ์อนุกรมเวลา (Time Series Forecasting) โดยมีการเปรียบเทียบความคลาดเคลื่อนในรูปแบบเปอร์เซ็นต์ของการพยากรณ์ในแต่ละวิธีการ 4 แบบ ดังนี้



รูปที่ 4.18 แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก

1) ซิมเปิลเอ็กซ์โพเนนเชียล

ค่า MAPE อยู่ที่ 55.90% ซึ่งค่อนข้างสูง แสดงให้เห็นว่าการพยากรณ์ด้วยวิธีนี้มีความคลาดเคลื่อนสูง วิธีนี้มักจะถูกใช้กับข้อมูลที่ไม่มีแนวโน้มหรือฤดูกาลชัดเจน ดังนั้นจึงอาจไม่เหมาะกับข้อมูลที่มีลักษณะผันผวนหรือมีความซับซ้อนสูง

2) ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

ค่า MAPE สูงที่สุดในทั้ง 4 วิธีอยู่ที่ 70.21% แสดงว่าการพยากรณ์ด้วยวิธีนี้มีความคลาดเคลื่อนสูงที่สุด แม้ว่าวิธีนี้จะเหมาะกับข้อมูลที่มีแนวโน้มชัดเจน แต่ไม่สามารถรับมือกับข้อมูลที่มีฤดูกาลหรือความผันผวนในช่วงต่าง ๆ ได้อย่างดี

3) การบวกโฮลท์-วินเทอร์

ค่า MAPE อยู่ที่ 20.63% ซึ่งลดลงอย่างมีนัยสำคัญ แสดงว่าการพยากรณ์ด้วยวิธีนี้มีความแม่นยำมากขึ้นเมื่อเทียบกับสองวิธีก่อนหน้านี้ การทำให้เรียบแบบเสริมฤดูกาลเหมาะสมสำหรับข้อมูลที่มีฤดูกาลคงที่ ทำให้ลดค่าความคลาดเคลื่อนได้ อีกทั้งยังสนับสนุนการสะท้อนรูปแบบการเปลี่ยนแปลงตามรอบฤดูกาลได้อย่างมีประสิทธิภาพ

4) การคูณโฮลท์-วินเทอร์

วิธีนี้ให้ค่า MAPE ต่ำที่สุดที่ 13.88% แสดงให้เห็นว่ามีความคลาดเคลื่อนต่ำที่สุดเมื่อเทียบกับวิธีอื่น ๆ Holt-Winters Multiplicative เหมาะกับข้อมูลที่มีฤดูกาลที่เปลี่ยนแปลงตามระดับของข้อมูล ซึ่งวิธีนี้สามารถจับความผันผวนตามฤดูกาลได้ดีและให้ผลการพยากรณ์ที่แม่นยำที่สุด

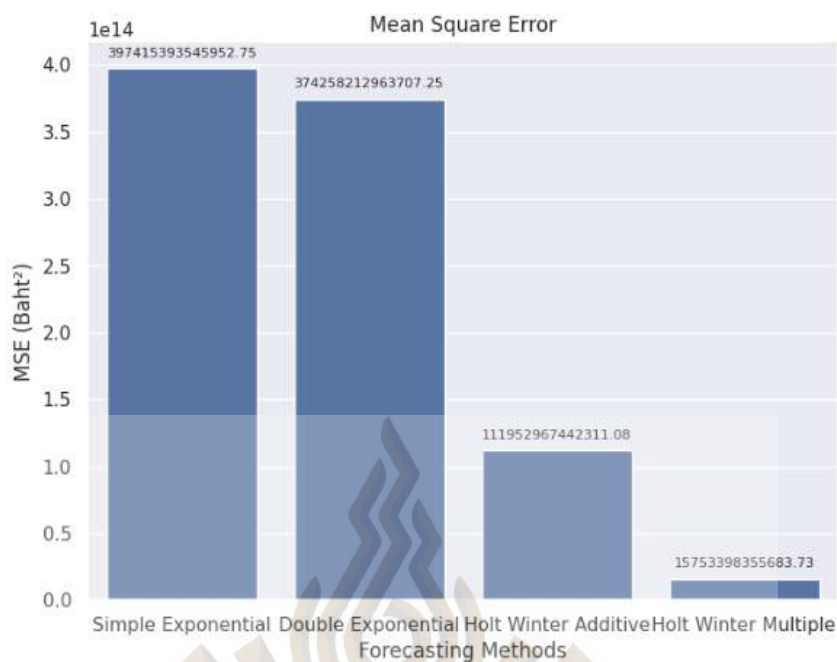
จากค่า MAPE ที่แสดงในแผนภูมิ วิธีโฮลท์-วินเทอร์แบบคูณให้ผลการพยากรณ์แม่นยำที่สุด โดยมีค่า MAPE ต่ำสุดที่ 13.88% ในขณะที่วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าให้ผลคลาดเคลื่อนมากที่สุดที่ 70.21% วิธีที่คำนึงถึงฤดูกาล เช่น โฮลท์-วินเทอร์แบบบวกและแบบคูณมีแนวโน้มให้ผลแม่นยำกว่าวิธีที่ไม่พิจารณาฤดูกาล เนื่องจากข้อมูลมีลักษณะเป็นฤดูกาล จึงควรเลือกใช้วิธีที่สามารถสะท้อนรูปแบบซ้ำและความผันผวนได้ดี โดยเฉพาะโฮลท์-วินเทอร์แบบคูณที่แสดงประสิทธิภาพเหนือกว่า

4.2.6.3 ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE)

แผนภูมิแท่งรูปที่ 4.19 แสดงค่า Mean Square Error (MSE) ของวิธีการพยากรณ์เชิงเวลาแบบต่าง ๆ ซึ่งแต่ละวิธีมีค่าความคลาดเคลื่อนเฉลี่ยกำลังสอง (MSE) แตกต่างกันไปไม่มีรายละเอียดดังนี้

1) Simple Exponential มีค่า MSE 397,415,393,545,952.75 หรือ ประมาณ 3.97×10^{14} ซึ่งมีค่าสูงที่สุดในกลุ่ม

2) Double Exponential มีค่า MSE 374,258,212,963,707.25 หรือ ประมาณ 3.74×10^{14} ซึ่งมีค่าใกล้เคียงกับวิธี Simple Exponential



รูปที่ 4.19 แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก

3) Holt Winter Additive มีค่า MSE 111,592,967,442,311.08 หรือประมาณ 1.11×10^{14} ซึ่งมีค่าต่ำลงมาเมื่อเทียบกับสองวิธีแรก

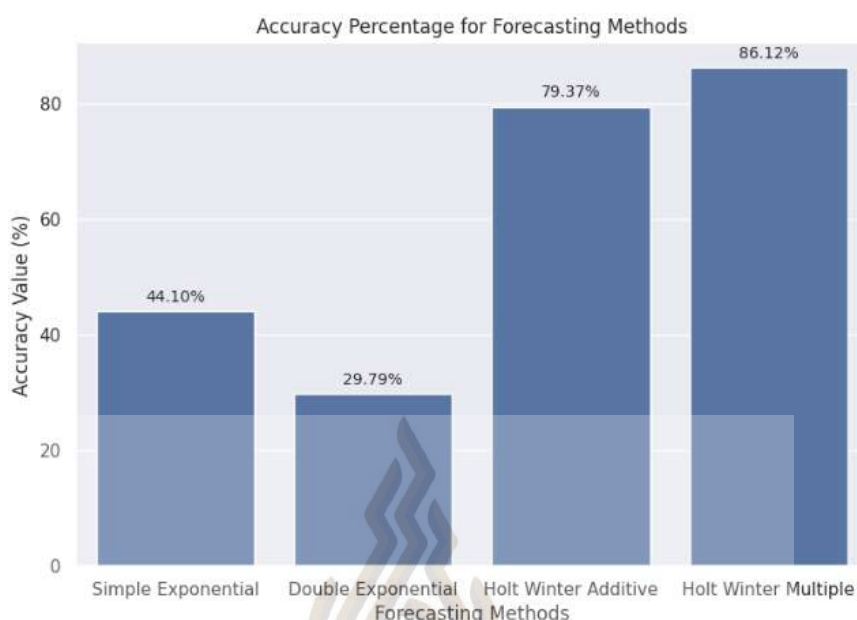
4) Holt Winter Multiple มีค่า MSE 15,753,398,355,683.73 หรือประมาณ 1.57×10^{13} ซึ่งเป็นค่าที่ต่ำที่สุดเมื่อเทียบกับวิธีการทั้งหมด

วิธีการคูณโฮลท์-วินเทอร์ มีค่าความคลาดเคลื่อนเฉลี่ยกำลังสองต่ำที่สุด ซึ่งหมายถึงประสิทธิภาพในการพยากรณ์ที่ดีที่สุดในกลุ่มนี้ ส่วนчимเปิดเอ็กซ์โพเนนเชียล และปรับเรียบเอ็กซ์โพเนนเชียลสองเท่ามีความคลาดเคลื่อนสูงกว่าวิธีอื่น ๆ อย่างมาก

4.2.7 ผลลัพธ์ความแม่นยำของการพยากรณ์กรมธรรม์หลัก

แผนภูมิแท่งรูปที่ 4.20 แสดงเปอร์เซ็นต์ความแม่นยำของวิธีการพยากรณ์ทั้ง 4 วิธี เพื่อเปรียบเทียบประสิทธิภาพในการพยากรณ์ข้อมูล

วิธีการพยากรณ์ที่แม่นยำที่สุดคือ Holt Winter Multiple ซึ่งมีความแม่นยำ 86.12% เนื่องจากสามารถพิจารณาแนวโน้มและฤดูกาลของข้อมูลได้อย่างครบถ้วน จึงให้ผลการพยากรณ์แม่นยำกว่าวิธีอื่น และตอบสนองต่อการเปลี่ยนแปลงได้ดี



รูปที่ 4.20 แผนภูมิแท่งแสดงผลลัพธ์ความแม่นยำของการพยากรณ์กรรมวิธีหลัก

- 1) Simple Exponential มีความแม่นยำ 44.10%
- 2) Double Exponential มีความแม่นยำ 29.79%
- 3) Holt Winter Additive มีความแม่นยำ 79.37%
- 4) Holt Winter Multiple มีความแม่นยำสูงสุดที่ 86.12%

4.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรรมวิธีสัญญาเพิ่มเติม

ตารางที่ 4.3 โดยการอัปโหลดจะใช้ไฟล์ชื่อ Sales_Additional.csv ส่งไปที่โปรแกรมภาษา Python และนำข้อมูลเข้าสู่กระบวนการพยากรณ์ด้วยแต่ละวิธี จากนั้นเปรียบเทียบค่าความแม่นยำด้วยค่า MAPE เพื่อประเมินประสิทธิภาพของการพยากรณ์ ทั้งนี้ วิธีการแต่ละแบบจะมีสมมติฐานและลักษณะการคำนวณที่แตกต่างกัน จึงต้องเลือกใช้ให้เหมาะสมกับรูปแบบข้อมูลเพื่อให้ได้ผลลัพธ์ที่แม่นยำมากที่สุด

ตารางที่ 4.3 ผลการพยากรณ์ยอดขายประกันชีวิตกรรมวิธีสัญญาเพิ่มเติมโดยวิธีต่าง ๆ

Period	Year	Quarter	Sales	ผลจาก HWA	ผลจาก HWM	ผลจาก SES	ผลจาก DES
1	2018	1	1459909.47	None	None	1459909.47	None
2	2018	2	3135763.08	None	None	1459909.47	None
3	2018	3	66552399.05	None	None	1627494.83	4811616.69

ตารางที่ 4.3 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีต่าง ๆ (ต่อ)

Period	Year	Quarter	Sales	ผลจาก HWA	ผลจาก HWM	ผลจาก SES	ผลจาก DES
4	2018	4	90247932.04	None	None	8119985.25	12661548.54
5	2019	1	24861304.10	1459909.47	1459909.47	16332779.93	22096040.50
6	2019	2	48072989.46	3369777.03	3135763.08	17185632.35	24048420.47
7	2019	3	72838021.79	67233445.12	66552399.05	20274368.06	28126730.98
8	2019	4	98665579.77	90985023.88	90247932.04	25530733.43	34273713.67
9	2020	1	28520342.94	33126187.85	24861304.10	32844218.06	42388753.89
10	2020	2	52265650.40	51940792.48	48072989.46	32411830.55	42677766.41
11	2020	3	78844226.85	76267894.39	72838021.79	34397212.54	45312408.42
12	2020	4	107022301.12	101644789.04	98665579.77	38841913.97	50341443.87
13	2021	1	28992683.84	32534616.48	28520342.94	45659952.69	57685383.20
14	2021	2	55743671.59	55940612.58	52265650.40	43993225.80	56491966.87
15	2021	3	83800510.25	82284011.40	78844226.85	45168270.38	58092990.95
16	2021	4	114310417.96	110107535.72	107022301.12	49031494.37	62339596.49
17	2022	1	31691762.82	32872205.00	28992683.84	55559386.73	69212532.25
18	2022	2	60425681.52	33742636.05	28992683.84	61434489.85	70888385.86
19	2022	3	90518857.46	34613067.09	28992683.84	66722082.66	72564239.47
20	2022	4	122999750.74	35483498.14	28992683.84	71480916.19	74240093.08

4.3.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมด้วยวิธีการบวกโฮลท์-วินเทอร์

ตารางที่ 4.3 แสดงผลการพยากรณ์ยอดขายประกันชีวิตสัญญาเพิ่มเติม โดยใช้วิธีการพยากรณ์แบบการบวกโฮลท์-วินเทอร์ สำหรับแต่ละไตรมาสในช่วงปี ค.ศ. 2018 - 2022 ซึ่งสามารถเปรียบเทียบความแม่นยำของการพยากรณ์กับข้อมูลยอดขายจริงได้อย่างชัดเจน

วิธีการพยากรณ์แบบโฮลท์-วินเทอร์ (Holt-Winter) เหมาะสมกับข้อมูลที่มีทั้งแนวโน้มเชิงเส้น (Trend) และความแปรปรวนตามฤดูกาล (Seasonality) โดยสามารถปรับขนาดของความแปรปรวนที่เพิ่มขึ้นหรือลดลงตามการเปลี่ยนแปลงของเวลา การพยากรณ์จะทำการปรับเรียบทั้งระดับ (Level) แนวโน้ม (Trend) และฤดูกาล (Seasonality) เพื่อให้ผลการพยากรณ์สามารถติดตามรูปแบบของข้อมูลได้อย่างแม่นยำมากขึ้น ส่งผลให้การวิเคราะห์ข้อมูลในระยะยาวมีความแม่นยำและเชื่อถือได้สูงขึ้น สามารถนำไปใช้ในการวางแผนเชิงกลยุทธ์และการตัดสินใจได้อย่างมีประสิทธิภาพ นอกจากนี้ยังสามารถนำไปประยุกต์ใช้ในหลายอุตสาหกรรมที่มีลักษณะข้อมูลซับซ้อนและมีการเปลี่ยนแปลงตามฤดูกาลได้อย่างเหมาะสม

4.3.1.1 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

- 1) ปี 2018 ข้อมูลเริ่มต้นไม่มีการพยากรณ์ เนื่องจากเป็นปีฐาน
- 2) ปี 2019 การพยากรณ์ในไตรมาสแรกต่ำกว่าความเป็นจริงอย่างมาก (ยอดขายจริง 24.86 ล้าน ในขณะที่ยอดขายอยู่ที่ 1.46 ล้าน) แต่การพยากรณ์ในไตรมาสถัดมาเริ่มใกล้เคียงกับยอดขายจริงมากขึ้น โดยเฉพาะไตรมาสที่ 3 และ 4 ที่ความแตกต่างน้อย
- 3) ปี 2020 การพยากรณ์ในปีนี้มีแนวโน้มความแม่นยำมากขึ้น โดยค่าเฉลี่ยของยอดขายที่พยากรณ์ใกล้เคียงกับยอดขายจริงในทุกไตรมาส การพยากรณ์ในไตรมาสที่ 3 และ 4 มีความใกล้เคียงกันมาก
- 4) ปี ค.ศ. 2021 การพยากรณ์ในปีนี้ยังคงมีความแม่นยำอยู่ ยกเว้นไตรมาสแรกที่ยอดขายจริงต่ำกว่าการพยากรณ์เล็กน้อย แต่ในไตรมาสถัดไปการพยากรณ์ใกล้เคียงกับยอดขายจริง
- 5) ปี ค.ศ. 2022 การพยากรณ์เริ่มมีความคลาดเคลื่อนมากขึ้น โดยเฉพาะในไตรมาสที่ 3 และ 4 ซึ่งยอดขายจริงสูงกว่าการพยากรณ์อย่างมาก โดยปัจจัยภายนอกและความผันผวนของตลาดสูงเกินขอบเขตการปรับตัวของโมเดลอย่างมีนัยสำคัญ

4.3.1.2 แนวโน้มของการพยากรณ์และยอดขายจริง

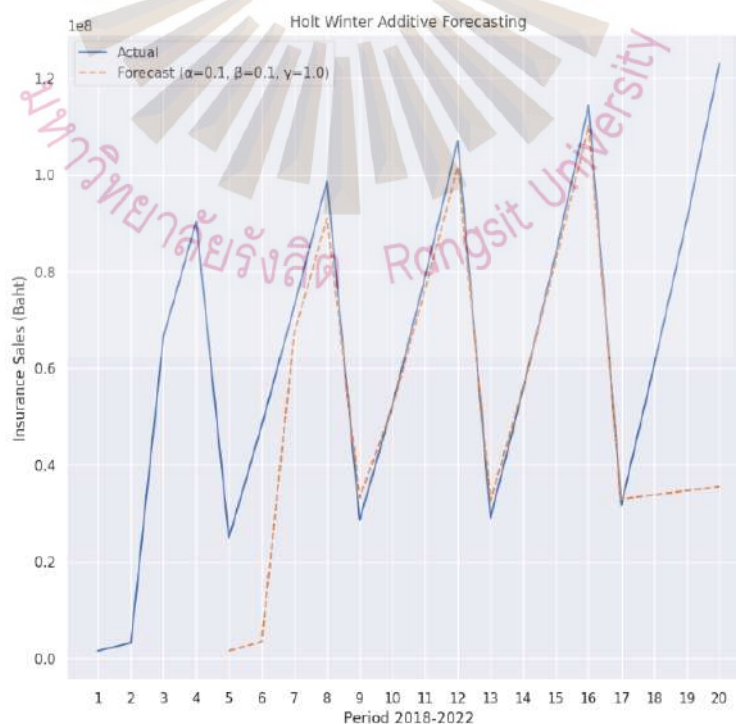
- 1) ปี ค.ศ. 2019 - 2021 การพยากรณ์โดยวิธีการบวกโฮสต์-วินเทอร์สามารถติดตามแนวโน้มของยอดขายจริงได้ค่อนข้างดี โดยเฉพาะช่วงไตรมาสที่ 2 ถึง 4 ที่การพยากรณ์มีความใกล้เคียงกับยอดขายจริง
- 2) ปี ค.ศ. 2022 ยอดขายจริงปรากฏการเติบโตอย่างรวดเร็วในไตรมาส 3 และ 4 โดยสูงกว่าค่าพยากรณ์อย่างมีนัยสำคัญ ซึ่งสะท้อนถึงความผันผวนของข้อมูลและปัจจัยภายนอกที่โมเดลไม่อาจตรวจจับได้
- 3) ความแม่นยำของการพยากรณ์ วิธีการบวกโฮสต์-วินเทอร์สามารถให้การพยากรณ์ที่แม่นยำในช่วงที่แนวโน้มของยอดขายไม่เปลี่ยนแปลงมากนัก (ปี ค.ศ. 2019 - 2021) แต่เมื่อมีการเพิ่มขึ้นหรือลดลงอย่างรวดเร็ว (ปี ค.ศ. 2022) วิธีการนี้อาจไม่สามารถติดตามได้ดี
 - 3.1) ข้อดี วิธีนี้มีความเหมาะสมในการพยากรณ์ข้อมูลที่มีทั้งแนวโน้มเชิงเส้นและฤดูกาล โดยเฉพาะอย่างยิ่งถ้าขนาดของความแปรปรวนตามฤดูกาลคงที่
 - 3.2) ข้อจำกัด หากข้อมูลมีการเปลี่ยนแปลงหรือมีความผันผวนสูง วิธีนี้อาจไม่สามารถให้การพยากรณ์ที่แม่นยำได้ จำเป็นต้องใช้วิธีที่ซับซ้อนหรือยืดหยุ่นมากขึ้นเพื่อรองรับการเปลี่ยนแปลงเชิงพลวัตในข้อมูลอย่างมีประสิทธิภาพ

การพยากรณ์โดยวิธีการบวกลบ-วินเทอร์สามารถใช้งานได้ในช่วงที่ข้อมูลมีแนวโน้มและรูปแบบเสถียร แต่จะมีข้อจำกัดเมื่อข้อมูลเปลี่ยนแปลงอย่างรวดเร็วหรือมีความผันผวนที่ไม่คงที่

แผนภูมิเส้นรูปที่ 4.21 แสดงผลการพยากรณ์แบบการบวกลบ-วินเทอร์ โดยใช้พารามิเตอร์ $\alpha = 0.1$ $\beta = 0.1$ และ $\gamma = 1.0$ ซึ่งเป็นวิธีการพยากรณ์ที่ใช้ในกรณีที่ข้อมูลมีแนวโน้มและฤดูกาล (Seasonality)

4.3.1.3 เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในช่วงปี ค.ศ. 2018 - 2022 มีลักษณะขึ้นลงเป็นรอบ ๆ ที่สอดคล้องกับฤดูกาล ซึ่งในบางช่วงค่าจะเพิ่มขึ้นและบางช่วงลดลงใกล้เคียงศูนย์ สะท้อนถึงการเปลี่ยนแปลงในแต่ละรอบฤดูกาลและปัจจัยฤดูกาลในการคาดการณ์ยอดขาย

4.3.1.4 เส้นประสีส้ม (Forecast) แสดงผลการพยากรณ์โดยใช้ Holt-Winters Additive Model ซึ่งเห็นได้ว่าการพยากรณ์ในช่วงต้นและกลางของแผนภูมิสามารถติดตามแนวโน้มของข้อมูลจริงได้ดี แต่ในช่วงปลาย (หลังจากช่วงที่ 17) การพยากรณ์ไม่สามารถจับทิศทางได้ เนื่องจากการพยากรณ์มีแนวโน้มคงที่ ในขณะที่ค่าจริงเพิ่มขึ้น



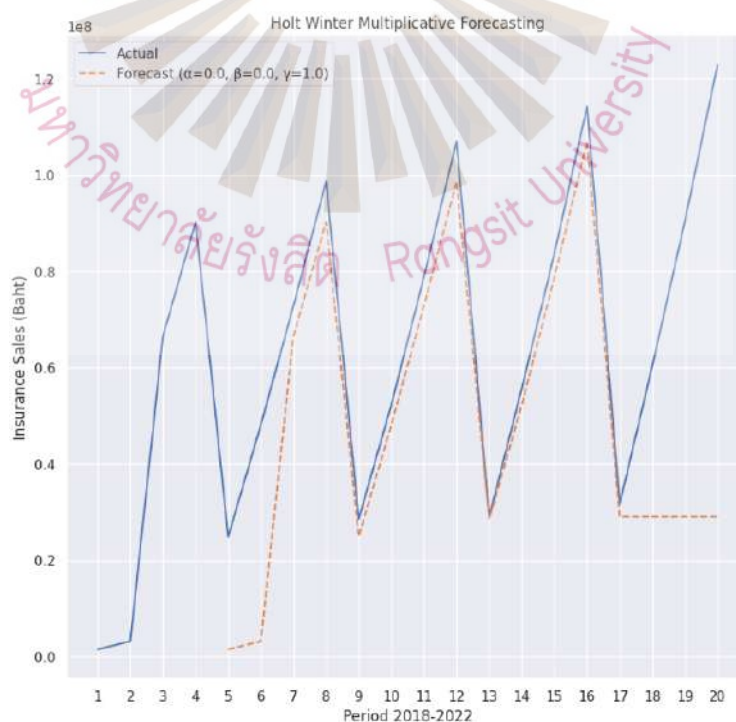
รูปที่ 4.21 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีการบวกลบ-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022

4.3.1.5 แนวโน้มและความแตกต่าง การพยากรณ์ด้วยพารามิเตอร์ที่ใช้ ($\alpha = 0.1$ $\beta = 0.1$ $\gamma = 1.0$) ทำให้ผลลัพธ์สามารถติดตามข้อมูลจริงได้ค่อนข้างดีในช่วงที่มีการเปลี่ยนแปลงฤดูกาลชัดเจน แต่ในช่วงที่ข้อมูลผันผวนสูง เช่น ช่วงหลัง ค่าพยากรณ์เริ่มแตกต่างจากค่าจริงมากขึ้น การตั้งค่าพารามิเตอร์ดังกล่าวอาจไม่ปรับตัวได้อย่างรวดเร็วในช่วงที่ข้อมูลมีความผันผวนสูง

การใช้การบวกโฮลท์-วินเทอร์ ให้ผลการพยากรณ์ที่ใกล้เคียงกับค่าจริงได้ดีในช่วงที่ข้อมูลมีรูปแบบที่ชัดเจนและสม่ำเสมอ อย่างไรก็ตาม การพยากรณ์ในช่วงที่ข้อมูลมีความผันผวนสูงจะมีความคลาดเคลื่อน การพยากรณ์จะมีความแม่นยำสูงเมื่อข้อมูลมีแนวโน้มหรือฤดูกาลที่ชัดเจน แต่ในช่วงที่มีความไม่แน่นอนหรือปัจจัยภายนอกแทรกซ้อน ความคลาดเคลื่อนจะเพิ่มขึ้น

4.3.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมด้วยวิธีการคูณโฮลท์-วินเทอร์

จากตารางที่ 4.3 การพยากรณ์ยอดขายรายไตรมาสด้วยวิธีการคูณโฮลท์-วินเทอร์ แสดงให้เห็นข้อมูลในแต่ละปีและไตรมาส รวมถึงยอดขายที่เกิดขึ้นจริง (Sales) และการพยากรณ์ในช่วงปี ค.ศ. 2018 - 2022 แยกตามไตรมาส



รูปที่ 4.22 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีการคูณโฮลท์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022

จากแผนภูมิเส้นรูปที่ 4.22 แผนภูมิเส้นนี้แสดงผลการพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ โดยใช้พารามิเตอร์ $\alpha = 0.0$ $\beta = 0.0$ $\gamma = 1.0$ ซึ่งเป็นรูปแบบการพยากรณ์ที่ใช้ในกรณีที่ข้อมูลมีความเป็นฤดูกาล (Seasonality) และรูปแบบคูณ (Multiplicative) สำหรับการพยากรณ์การเปลี่ยนแปลงของข้อมูลที่ไม่เป็นเส้นตรง (Non-Linear)

4.3.2.1 การพยากรณ์โดยวิธีการคูณโฮลท์-วินเทอร์

วิธีการนี้เหมาะสมกับข้อมูลที่มีแนวโน้มเชิงเส้น (Trend) และฤดูกาล (Seasonality) ที่มีรูปแบบการเปลี่ยนแปลงเชิงทวีคูณ (Multiplicative) หรือในกรณีที่ขนาดของความแปรปรวนตามฤดูกาลเพิ่มขึ้นหรือลดลงตามสัดส่วนของระดับข้อมูล (Level) ที่เปลี่ยนแปลง

การพยากรณ์โดยวิธีนี้จะใช้การปรับเรียบทั้งระดับ (Level) แนวโน้ม (Trend) และฤดูกาล (Seasonality) โดยใช้การคูณเพื่อสะท้อนถึงความผันผวนตามฤดูกาล

4.3.2.2 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

- 1) ปี ค.ศ. 2018 เป็นข้อมูลปีฐาน จึงไม่มีการพยากรณ์ในช่วงนี้
- 2) ปี ค.ศ. 2019 การพยากรณ์ในปีนี้เป็นเริ่มต้น โดยใช้ข้อมูลของปี 2018 เป็นพื้นฐาน การพยากรณ์ในไตรมาสที่ 1 และ 2 ต่ำกว่าความเป็นจริงอย่างมาก (ยอดขายจริงสูงกว่าการพยากรณ์) แต่ในไตรมาสที่ 3 และ 4 การพยากรณ์ใกล้เคียงกับยอดขายจริงมากขึ้น
- 3) ปี ค.ศ. 2020 การพยากรณ์ในปีนี้ยังคงสามารถติดตามแนวโน้มของยอดขายจริงได้ดี โดยมีความใกล้เคียงกับยอดขายจริงในทุกไตรมาส
- 4) ปี ค.ศ. 2021 การพยากรณ์ยังคงมีความแม่นยำที่ดีในช่วงต้นปี แต่มีความคลาดเคลื่อนเล็กน้อยในไตรมาสที่ 3 และ 4
- 5) ปี ค.ศ. 2022 การพยากรณ์ในปีนี้มีมีความคลาดเคลื่อนมากขึ้น โดยเฉพาะในไตรมาสที่ 2 ถึง 4 ซึ่งยอดขายจริงสูงกว่าการพยากรณ์อย่างมาก

4.3.2.3 แนวโน้มของการพยากรณ์และยอดขายจริง

- 1) การพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์สามารถติดตามแนวโน้มและความผันผวนของยอดขายจริงได้ดีในช่วงปี ค.ศ. 2019 - 2021
- 2) อย่างไรก็ตาม ในปี ค.ศ. 2022 การพยากรณ์มีความคลาดเคลื่อนเพิ่มมากขึ้น สะท้อนถึงการเปลี่ยนแปลงอย่างรวดเร็วของยอดขายจริงที่วิธีนี้ไม่สามารถติดตามได้อย่างถูกต้อง

3) ความแม่นยำของการพยากรณ์ วิธีการคูณโฮลท์-วินเทอร์มีความแม่นยำในการพยากรณ์ยอดขายที่มีแนวโน้มและความผันผวนตามฤดูกาลที่คงที่หรือเปลี่ยนแปลงตามสัดส่วนในช่วงปี ค.ศ. 2019 - 2021 แต่ไม่สามารถติดตามการเปลี่ยนแปลงที่ไม่คาดคิดในปี ค.ศ. 2022 ได้อย่างแม่นยำ

3.1) ข้อดี วิธีนี้เหมาะสำหรับข้อมูลที่มีรูปแบบการเปลี่ยนแปลงเชิงทวิคูณตามฤดูกาลและแนวโน้มที่ไม่ซับซ้อนเกินไป

3.2) ข้อจำกัด หากมีความผันผวนที่ไม่คาดคิดหรือการเปลี่ยนแปลงอย่างรวดเร็วในข้อมูล วิธีนี้อาจไม่สามารถให้ผลการพยากรณ์ที่แม่นยำได้ จำเป็นต้องพิจารณาใช้วิธีที่ซับซ้อนมากขึ้นหรือปรับพารามิเตอร์ให้เหมาะสมกับข้อมูลที่เปลี่ยนแปลง

การพยากรณ์โดยวิธีการคูณโฮลท์-วินเทอร์มีประสิทธิภาพในการพยากรณ์ข้อมูลที่มีแนวโน้มและความแปรปรวนตามฤดูกาลที่มีรูปแบบเชิงทวิคูณ แต่จะมีข้อจำกัดเมื่อข้อมูลมีการเปลี่ยนแปลงที่ไม่คาดคิดหรือผันผวนมาก

4.3.2.4 เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในช่วงปี ค.ศ. 2018 - 2022 ซึ่งค่าจริงมีลักษณะการขึ้นลงเป็นรอบอย่างชัดเจน โดยมีการเพิ่มขึ้นและลดลงในแต่ละช่วงเวลา คล้ายคลึงกับรูปแบบฤดูกาล

4.3.2.5 เส้นประสีส้ม (Forecast) แสดงผลการพยากรณ์โดยใช้ Holt-Winters Multiplicative Model แสดงผลใกล้เคียงกับค่าจริงในช่วงต้นและกลางแผนภูมิเส้น สามารถติดตามแนวโน้มของข้อมูลจริงได้แม่นยำในช่วงที่มีรูปแบบซ้ำ ๆ ตามฤดูกาล อย่างไรก็ตามในช่วงปลายของแผนภูมิเส้น (หลังจากช่วงที่ 17) ผลการพยากรณ์มีแนวโน้มคงที่ขณะที่ค่าจริงเพิ่มขึ้นอย่างมาก

4.3.2.6 แนวโน้มการพยากรณ์ การพยากรณ์ตามโมเดลนี้ทำงานได้ดีในช่วงที่ข้อมูลมีการเปลี่ยนแปลงตามฤดูกาลอย่างสม่ำเสมอ แต่ในช่วงที่ข้อมูลมีการเปลี่ยนแปลงที่สูงเกินไป โมเดลนี้ไม่สามารถสะท้อนการเพิ่มขึ้นที่รุนแรงของค่าจริงได้อย่างแม่นยำ

การคูณโฮลท์-วินเทอร์ทำงานได้ดีเมื่อข้อมูลมีรูปแบบซ้ำตามฤดูกาลชัดเจน แต่เมื่อข้อมูลมีความผันผวนหรือเพิ่มขึ้นมาก โมเดลนี้จะคลาดเคลื่อนในการพยากรณ์ เนื่องจากลักษณะการคูณจะขยายผลของฤดูกาลตามระดับข้อมูล

4.3.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

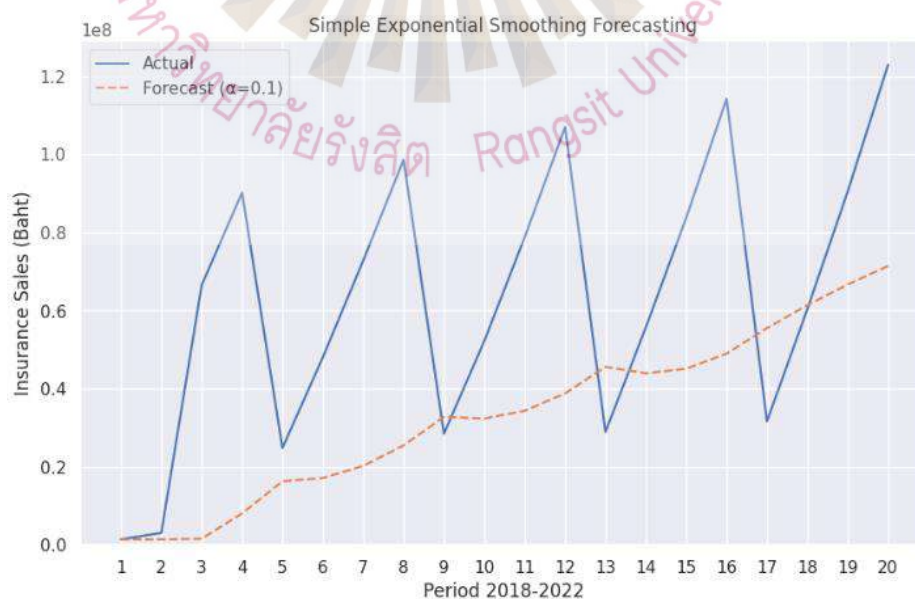
จากตารางที่ 4.3 ได้แสดงข้อมูลยอดขายจริงและยอดขายที่พยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม โดยใช้วิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายในช่วงปี ค.ศ. 2018 - 2022 แยกตามไตรมาสข้อมูลดังกล่าวทำให้เห็นภาพรวมของยอดขายในแต่ละช่วงเวลา พร้อมทั้งการเปรียบเทียบระหว่างผลการพยากรณ์และข้อมูลจริงที่เกิดขึ้น

4.3.3.1 การพยากรณ์โดยวิธีซิมเปิลเอ็กซ์โพเนนเชียล

1) วิธีนี้เป็นวิธีการพยากรณ์ที่สะดวกและเป็นที่ยอมรับสำหรับข้อมูลอนุกรมเวลา (Time Series) ที่ไม่มีฤดูกาลหรือแนวโน้มที่ชัดเจน โดยใช้การปรับเรียบเพื่อให้การพยากรณ์มีความสมดุลและไม่ซับซ้อน

2) ใช้ค่าเฉลี่ยถ่วงน้ำหนักที่ลดลงเรื่อย ๆ สำหรับค่าข้อมูลในอดีต โดยให้น้ำหนักที่มากขึ้นกับข้อมูลที่อยู่ใกล้กับช่วงเวลาที่พยากรณ์ วิธีนี้ส่งเสริมให้การพยากรณ์มีความยืดหยุ่นและตอบสนองต่อการเปลี่ยนแปลงในข้อมูลได้ดียิ่งขึ้น

3) การพยากรณ์ครั้งแรก (ในปี ค.ศ. 2018) เริ่มจากการใช้ค่าเฉลี่ยของยอดขายจริงที่มีอยู่ เพื่อกำหนดค่าพยากรณ์เริ่มต้นและฐานในการคำนวณการพยากรณ์ถัดไป โดยใช้การปรับเรียบตามข้อมูลในอดีต



รูปที่ 4.23 ผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายระหว่างปี ค.ศ. 2018 - 2022

รูปที่ 4.23 แผนภูมิเส้นนี้แสดงผลการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย โดยใช้พารามิเตอร์ $\alpha = 0.1$ ซึ่งเป็นวิธีการพยากรณ์ที่ใช้ข้อมูลในอดีตเพื่อคำนวณค่าเฉลี่ยแบบถ่วงน้ำหนัก โดยค่าใหม่จะมีน้ำหนักมากกว่าค่าก่อนหน้า

4.3.3.2 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

- 1) ปี ค.ศ. 2018 เริ่มต้นด้วยการใช้ยอดขายจริงในไตรมาสแรกเป็นค่าพยากรณ์ ซึ่งทำให้การพยากรณ์ในไตรมาสที่สองและสามต่างจากยอดขายจริงอย่างมาก เนื่องจากยอดขายจริงเพิ่มขึ้นอย่างรวดเร็ว (ไตรมาสที่ 3 และ 4 มีความผันผวนสูง)
- 2) ปี ค.ศ. 2019 การพยากรณ์เริ่มใกล้เคียงกับยอดขายจริงมากขึ้นในช่วงต้นปี แต่ยังคงมีความคลาดเคลื่อนเมื่อยอดขายจริงเพิ่มขึ้นอย่างมากในช่วงครึ่งหลังของปี
- 3) ปี ค.ศ. 2020 การพยากรณ์ยังคงใกล้เคียงกับยอดขายจริง โดยเฉพาะในช่วงครึ่งแรกของปี อย่างไรก็ตาม การพยากรณ์ในไตรมาสที่ 4 ยังคงต่ำกว่ายอดขายจริง
- 4) ปี ค.ศ. 2021 การพยากรณ์ยังคงตามแนวโน้มของยอดขายจริงได้ดี แต่มีความคลาดเคลื่อนในบางไตรมาส โดยเฉพาะไตรมาสที่ 4 ซึ่งยอดขายจริงสูงกว่าการพยากรณ์อย่างมาก
- 5) ปี ค.ศ. 2022 ความคลาดเคลื่อนเพิ่มขึ้นในช่วงไตรมาสที่ 3 และ 4 ซึ่งยอดขายจริงเพิ่มขึ้นอย่างมาก ในขณะที่การพยากรณ์ยังไม่สามารถตามความผันผวนนี้ได้

4.3.3.3 แนวโน้มของการพยากรณ์และยอดขายจริง

- 1) การพยากรณ์มีแนวโน้มที่จะตามแนวโน้มของยอดขายจริงได้ แต่มีความล่าช้าเมื่อยอดขายจริงเปลี่ยนแปลงอย่างรวดเร็ว เนื่องจากการพยากรณ์ใช้ข้อมูลในอดีตเป็นหลัก ซึ่งอาจไม่สามารถตอบสนองต่อการเปลี่ยนแปลงที่เกิดขึ้นทันทีได้
- 2) การพยากรณ์ในช่วงที่ยอดขายมีความผันผวนสูงอาจมีความคลาดเคลื่อนมาก เนื่องจากวิธีเอ็กซ์โพเนนเชียลอย่างง่ายไม่สามารถติดตามการเปลี่ยนแปลงเชิงแนวโน้มหรือฤดูกาลได้ดีเท่าวิธีอื่น ๆ
- 3) ความแม่นยำของการพยากรณ์ วิธีเอ็กซ์โพเนนเชียลอย่างง่ายมีความแม่นยำระดับหนึ่งในการพยากรณ์ยอดขายที่ไม่มีแนวโน้มที่ซับซ้อนหรือความผันผวนมาก แต่มีข้อจำกัดเมื่อข้อมูลมีแนวโน้มและความผันผวนที่เปลี่ยนแปลงอย่างรวดเร็ว
 - 3.1) ข้อดี วิธีนี้ง่ายต่อการคำนวณและใช้พยากรณ์ข้อมูลที่มีความผันผวนต่ำได้ดี เนื่องจากไม่ต้องการพารามิเตอร์ที่ซับซ้อน ทำให้การใช้งานสะดวก

3.2) ข้อจำกัด วิธีนี้ไม่เหมาะสมกับข้อมูลที่มีแนวโน้มที่ซับซ้อนหรือมีฤดูกาลที่ชัดเจน เนื่องจากไม่สามารถจับรูปแบบแนวโน้มหรือฤดูกาลได้ดี และไม่สามารถตอบสนองต่อการเปลี่ยนแปลงอย่างรวดเร็วในข้อมูลได้

3.3) ข้อเสนอแนะเพิ่มเติม ในกรณีที่ข้อมูลมีแนวโน้มหรือฤดูกาลที่ชัดเจน ควรพิจารณาใช้วิธีการพยากรณ์ที่ซับซ้อนขึ้น เช่น Holt-Winters หรือ ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

วิธีหิมปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายเหมาะสำหรับข้อมูลที่ไม่มีแนวโน้มที่ชัดเจนหรือความผันผวนต่ำ แต่จะมีข้อจำกัดเมื่อข้อมูลมีการเปลี่ยนแปลงที่ไม่คาดคิดหรือแนวโน้มที่ชัดเจน รูปที่ 4.23 แสดงผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายระหว่างปี ค.ศ. 2018 - 2022

4.3.3.4 เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในช่วงปี ค.ศ. 2018 - 2022 ซึ่งเห็นได้ว่าการขึ้นลงอย่างชัดเจนตามฤดูกาล โดยมีการเพิ่มขึ้นและลดลงเป็นรอบ ๆ และในช่วงปลายแผนภูมิเส้น ค่าจริงมีการเพิ่มขึ้นอย่างมาก

4.3.3.5 เส้นประสีส้ม (Forecast) เป็นการพยากรณ์จาก Simple Exponential Smoothing ซึ่งให้ผลการพยากรณ์ที่เรียบและค่อย ๆ เพิ่มขึ้นอย่างสม่ำเสมอ โมเดลนี้ไม่สามารถสะท้อนการเปลี่ยนแปลงที่ผันผวนของข้อมูลจริงได้ โดยเส้นพยากรณ์มีแนวโน้มเพิ่มขึ้นช้ากว่าค่าจริงในช่วงต้น และไม่สามารถสะท้อนการเพิ่มขึ้นอย่างรวดเร็วในช่วงหลังของข้อมูลจริงได้

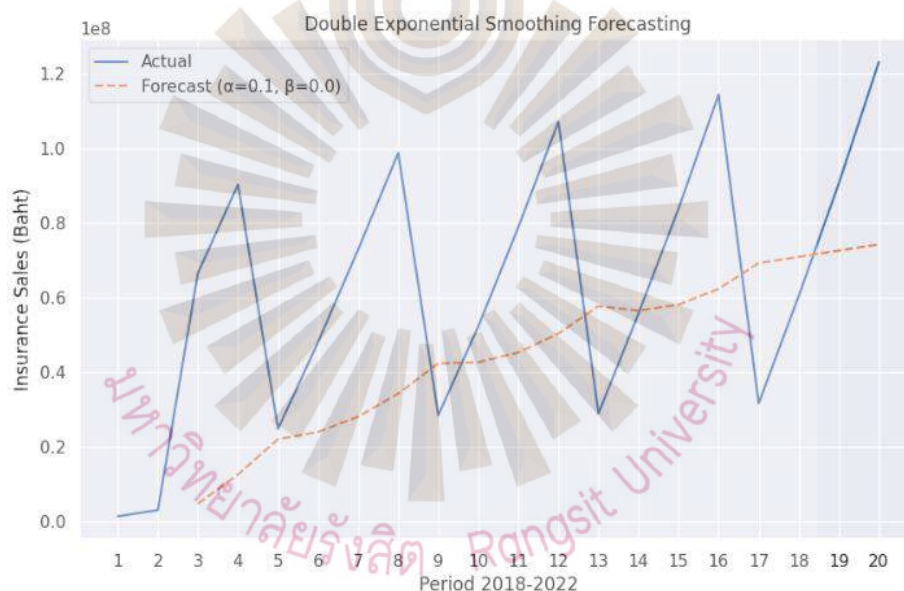
4.3.3.6 แนวโน้มการพยากรณ์ การใช้พารามิเตอร์ $\alpha = 0.1$ ทำให้การพยากรณ์ไม่ตอบสนองต่อการเปลี่ยนแปลงของข้อมูลจริงอย่างทันทีทันใด การพยากรณ์มีลักษณะค่อย ๆ เพิ่มขึ้นอย่างช้า ๆ และไม่สามารถจับแนวโน้มฤดูกาลหรือความผันผวนของข้อมูลได้ดี

การพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายไม่เหมาะสมกับข้อมูลที่มีแนวโน้มเปลี่ยนแปลงตามฤดูกาลหรือมีความผันผวนสูง เนื่องจากโมเดลไม่สามารถตอบสนองต่อการเปลี่ยนแปลงในข้อมูลได้ทัน ทำให้การพยากรณ์มีความแตกต่างจากค่าจริงอย่างมากในช่วงที่ข้อมูลเพิ่มขึ้นหรือลดลงอย่างรวดเร็ว จึงควรพิจารณาใช้โมเดลที่สามารถรองรับความผันผวนและลักษณะฤดูกาลของข้อมูลได้ดียิ่งขึ้น

4.3.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

จากตารางที่ 4.3 แสดงข้อมูลการพยากรณ์โดยวิธีการปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม เปรียบเทียบระหว่างค่าจริง (Actual Sales) และค่าพยากรณ์ (Forecast) ในช่วงปี ค.ศ. 2018 - 2022 เพื่อประเมินความแม่นยำและแนวโน้ม

จากรูปที่ 4.24 แผนภูมิเส้นนี้แสดงผลการพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม โดยใช้พารามิเตอร์ $\alpha = 0.1$ และ $\beta = 0.0$ ซึ่งเป็นการปรับปรุงจากวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย โดยเพิ่มส่วนประกอบในการจับแนวโน้ม และควบคุมการตอบสนองของโมเดลต่อการเปลี่ยนแปลงของข้อมูล



รูปที่ 4.24 ผลการพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม ระหว่างปี ค.ศ. 2018 - 2022

4.3.4.1 การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

- 1) เป็นวิธีการพยากรณ์ที่เพิ่มการจับแนวโน้ม (Trend) เข้าไปในสูตร เพื่อให้สามารถพยากรณ์ได้ดีขึ้นในกรณีที่ข้อมูลมีแนวโน้มเพิ่มหรือลดลง
- 2) เหมาะสำหรับข้อมูลอนุกรมเวลาที่มีลักษณะการเพิ่มขึ้นหรือลดลงเชิงเส้น โดยเฉพาะในกรณีที่ข้อมูลไม่มีฤดูกาลเข้ามาเกี่ยวข้องและแนวโน้มเปลี่ยนแปลงอย่างต่อเนื่อง โดยส่งผลให้สามารถติดตามพฤติกรรมของข้อมูลในอนาคตได้อย่างแม่นยำมากยิ่งขึ้น

4.3.4.2 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

1) ปี ค.ศ. 2018 เริ่มต้นโดยไม่มีการพยากรณ์ในสองไตรมาสแรก (ในช่วง “Warm-up” ของการคำนวณ) การพยากรณ์ในไตรมาสที่สามและสี่มีค่าต่ำกว่ายอดขายจริงอย่างมาก เนื่องจากยอดขายจริงเพิ่มขึ้นอย่างรวดเร็ว

2) ปี ค.ศ. 2019 การพยากรณ์ยังคงต่ำกว่ายอดขายจริงอย่างมีนัยสำคัญในไตรมาสแรกและที่สอง แต่ในไตรมาสที่สามและสี่ การพยากรณ์เริ่มปรับตัวให้เข้ากับแนวโน้มที่เพิ่มขึ้นของยอดขายจริง

3) ปี ค.ศ. 2020 การพยากรณ์เริ่มใกล้เคียงกับยอดขายจริงมากขึ้น แต่ยังคงมีความคลาดเคลื่อนอยู่บ้าง โดยเฉพาะในไตรมาสที่สามและสี่ที่ยอดขายจริงสูงขึ้นกว่าการพยากรณ์

4) ปี ค.ศ. 2021 การพยากรณ์ในปีนี้เริ่มสามารถติดตามแนวโน้มการเพิ่มขึ้นของยอดขายจริงได้ดีขึ้น แม้ว่ายอดขายจริงในไตรมาสที่สี่จะสูงกว่าการพยากรณ์อย่างมีนัยสำคัญ

5) ปี ค.ศ. 2022 ความคลาดเคลื่อนระหว่างการพยากรณ์และยอดขายจริงยังคงมีอยู่ แต่ลดลงเมื่อเทียบกับปีก่อนหน้า การพยากรณ์ไตรมาสที่สามและสี่ยังไม่สามารถตามยอดขายจริงที่เพิ่มขึ้นอย่างรวดเร็วได้ดี

4.3.4.3 แนวโน้มของการพยากรณ์และยอดขายจริง

1) การพยากรณ์ตามแนวโน้มของยอดขายจริงได้ดีขึ้นเมื่อเวลาผ่านไป แต่ยังคงมีความคลาดเคลื่อนอยู่บ้างเมื่อยอดขายจริงมีการเพิ่มขึ้นอย่างมากในบางไตรมาส

2) การใช้วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าสามารถปรับปรุงความแม่นยำในการพยากรณ์เมื่อเทียบกับวิธีเอ็กซ์โพเนนเชียลอย่างง่าย แต่ยังไม่สามารถจัดการกับความผันผวนที่รุนแรงได้ดี

3) ความแม่นยำของการพยากรณ์ วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่ามีความแม่นยำในการพยากรณ์สูงขึ้นเมื่อเทียบกับวิธีเอ็กซ์โพเนนเชียลอย่างง่าย โดยสามารถจับแนวโน้มของข้อมูลได้ดีกว่า แต่ยังคงมีความคลาดเคลื่อนอยู่บ้างเมื่อยอดขายจริงมีความผันผวนสูง

3.1) ข้อดี สามารถปรับปรุงการพยากรณ์เมื่อข้อมูลมีแนวโน้มที่เพิ่มขึ้นหรือลดลงเชิงเส้น และมีการตอบสนองต่อแนวโน้มที่เปลี่ยนแปลงได้ดีกว่าวิธีเอ็กซ์โพเนนเชียลอย่างง่าย วิธีนี้สามารถจับการเปลี่ยนแปลงในแนวโน้มได้อย่างมีประสิทธิภาพ

3.2) ข้อจำกัด ไม่สามารถจัดการกับความผันผวนที่รุนแรงหรือแนวโน้มที่ซับซ้อนได้ดีเท่าวิธีการที่มีการรวมฤดูกาล เช่น Holt-Winters

3.3) ข้อเสนอแนะเพิ่มเติม ในกรณีที่ข้อมูลมีแนวโน้มและฤดูกาลที่ชัดเจน ควรใช้วิธีการพยากรณ์ที่มีการปรับปรุงเพิ่มเติม เช่น การบวกโฮลท์-วินเทอร์ หรือ การคูณโฮลท์-วินเทอร์ ซึ่งจะให้ผลลัพธ์ที่แม่นยำกว่าในการจัดการกับความผันผวนและแนวโน้มที่หลากหลาย

วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าสามารถปรับปรุงความแม่นยำของการพยากรณ์ได้ดี ในกรณีที่ข้อมูลมีแนวโน้มเพิ่มขึ้นหรือลดลงเชิงเส้น แต่มีข้อจำกัดในการจัดการกับข้อมูลที่มีความผันผวนและแนวโน้มที่ซับซ้อน รูปที่ 4.24 แสดงผลการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมระหว่างปี ค.ศ. 2018 - 2022 โดยวิธีนี้เหมาะสำหรับข้อมูลที่มีแนวโน้มชัดเจน แต่ไม่สามารถจัดการกับข้อมูลที่มีความผันผวนได้ดี และอาจให้ผลลัพธ์ที่ไม่แม่นยำเมื่อข้อมูลมีการเปลี่ยนแปลงที่ไม่สามารถคาดเดาหรือมีลักษณะผันผวนสูง

4.3.4.4 เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในช่วงปี ค.ศ. 2018 - 2022 ซึ่งมีรูปแบบที่ชัดเจน โดยข้อมูลมีการขึ้นลงเป็นรอบตามฤดูกาล และมีการเพิ่มขึ้นอย่างมากในช่วงปลายของแผนภูมิเส้น ซึ่งสะท้อนถึงปัจจัยภายนอกที่มีผลต่อการเปลี่ยนแปลง และสามารถใช้ในการวิเคราะห์ความสัมพันธ์ระหว่างฤดูกาลและการเปลี่ยนแปลงยอดขาย

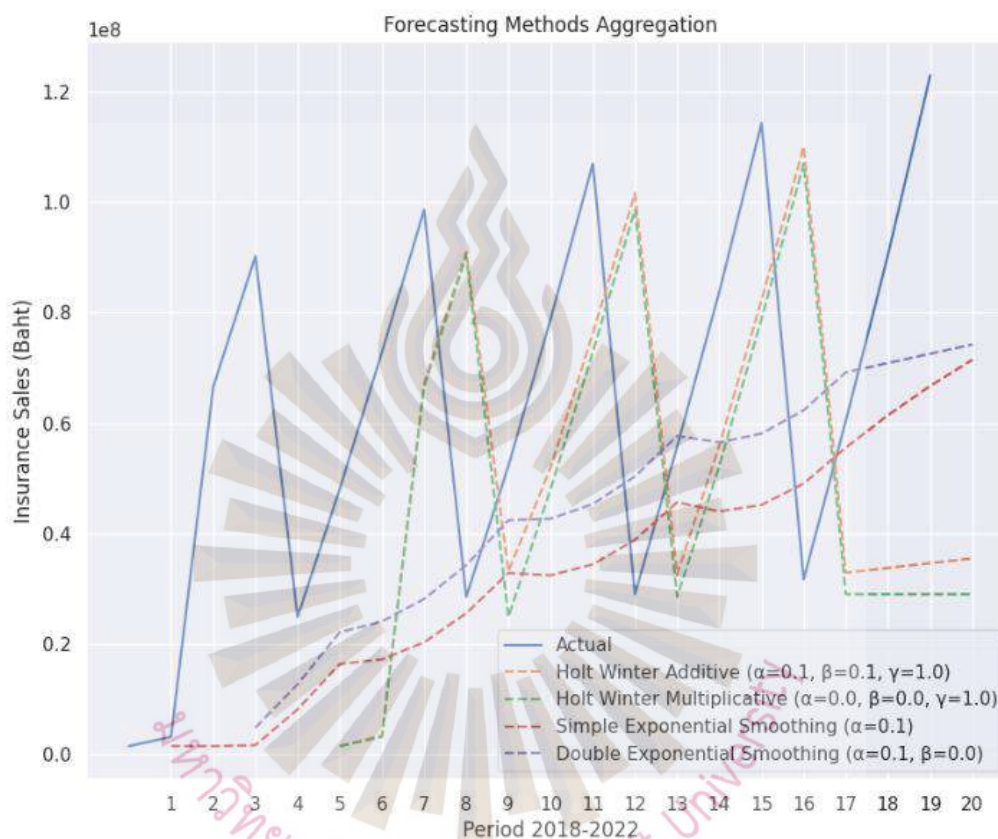
4.3.4.5 เส้นประสีส้ม (Forecast) แสดงผลการพยากรณ์โดยใช้ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ซึ่งค่าพยากรณ์ค่อย ๆ เพิ่มขึ้น แต่ยังคงมีความแตกต่างจากค่าจริง โดยเฉพาะในช่วงที่ค่าจริงเพิ่มขึ้นอย่างรวดเร็ว เส้นพยากรณ์แสดงการเพิ่มขึ้นที่ช้ากว่าค่าจริง เนื่องจาก $\beta = 0.0$ ทำให้โมเดลไม่สามารถจับแนวโน้ม (Trend) การเปลี่ยนแปลงของข้อมูลได้อย่างเต็มที่

4.3.4.6 แนวโน้มการพยากรณ์ การพยากรณ์นี้ยังคงทำงานได้ไม่ดีนักในกรณีที่ข้อมูลจริงมีการเปลี่ยนแปลงที่รุนแรงในระยะสั้น แม้ว่าจะมีการพยายามจับแนวโน้ม แต่การตั้งค่า $\beta = 0.0$ ส่งผลให้โมเดลนี้ไม่สามารถสะท้อนการเปลี่ยนแปลงอย่างรวดเร็วในข้อมูลจริงได้

การใช้วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าทำให้ความแม่นยำดีขึ้นเมื่อเทียบกับวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย แต่การตั้งค่า $\beta = 0.0$ ทำให้โมเดลไม่สามารถจับแนวโน้มการเพิ่มขึ้นหรือลดลงที่ชัดเจนได้ ส่งผลให้ผลการพยากรณ์ไม่สอดคล้องกับค่าจริงในช่วงที่ข้อมูลมีการเปลี่ยนแปลงอย่างรุนแรง

4.3.5 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมด้วยวิธีต่าง ๆ

เมื่อนำผลทั้ง 4 วิธี มารวมกัน จะได้ดังรูปที่ 4.25 ที่เป็นแผนภูมิเส้นที่แสดงการพยากรณ์วิธีต่าง ๆ เปรียบเทียบกับข้อมูลยอดขายจริงยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมในช่วงปี ค.ศ. 2018 - 2022



รูปที่ 4.25 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติมโดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022

4.3.5.1 ช่วงที่ยอดขายสูงมาก ช่วงที่เห็นยอดขายสูงชัดเจนคือในปี ค.ศ. 2020 - 2021 ซึ่งตรงกับช่วงที่ COVID-19 ระบาดหนัก ยอดขายการประกันชีวิตน่าจะพุ่งสูงขึ้นในช่วงนี้ เพราะผู้บริโภคมีความกังวลเกี่ยวกับสุขภาพและอนาคตของตัวเองและครอบครัว

4.3.5.2 ช่วงที่ยอดขายต่ำ ช่วงที่ยอดขายลดลงอาจเกิดจากสถานการณ์ที่ COVID-19 เริ่มคลี่คลายลง ซึ่งส่งผลให้ผู้บริโภคอาจไม่รู้สึกถึงความจำเป็นในการซื้อประกันเพิ่มขึ้นเหมือนช่วงที่ COVID-19 ระบาดหนัก การเปลี่ยนแปลงนี้สะท้อนถึงการปรับตัวของพฤติกรรมผู้บริโภคที่ได้รับผลกระทบจากสถานการณ์ต่าง ๆ ในช่วงเวลานั้น

4.3.5.3 สาเหตุที่เกี่ยวข้องกับ COVID-19 และพฤติกรรมผู้บริโภค

- 1) การระบาดของ COVID-19 ทำให้ผู้คนตระหนักและกังวลเกี่ยวกับสุขภาพมากขึ้น ส่งผลให้ยอดขายประกันชีวิตและสัญญาเพิ่มเติมเพิ่มขึ้นอย่างรวดเร็ว
- 2) ความไม่แน่นอนทางเศรษฐกิจ COVID-19 ทำให้ผู้คนกังวลเกี่ยวกับความมั่นคงทางการเงินในอนาคต จึงหันมาซื้อประกันชีวิตเพื่อเป็นการป้องกันความเสี่ยง การซื้อประกันชีวิตในช่วงวิกฤติเศรษฐกิจเป็นกลยุทธ์ที่เสริมสร้างความมั่นคงทางการเงิน
- 3) การกลับสู่สภาวะปกติ เมื่อการระบาดคลี่คลายลง ผู้บริโภครู้สึกปลอดภัยและมั่นใจมากขึ้น ส่งผลให้ความต้องการในการซื้อประกันชีวิตลดลง โดยผู้บริโภคเริ่มมองข้ามความเสี่ยงและความจำเป็นในการคุ้มครอง

ในช่วงปี ค.ศ. 2018 - 2022 ยอดขายประกันชีวิตสัญญาเพิ่มเติมเปลี่ยนแปลงอย่างชัดเจน โดยพุ่งสูงในช่วงปี ค.ศ. 2020 - 2021 ซึ่งเป็นช่วง COVID-19 ระบาดหนัก ยอดขายเพิ่มขึ้นมากเนื่องจากผู้บริโภคกังวลเรื่องสุขภาพและอนาคตทางการเงิน อย่างไรก็ตาม หลังจากสถานการณ์ COVID-19 คลี่คลาย ยอดขายมีแนวโน้มลดลง

4.3.6 ผลลัพธ์การวัดข้อผิดพลาดการพยากรณ์กรรมธรรม์เพิ่มเติม

ในการประเมินประสิทธิภาพของโมเดลพยากรณ์ที่ใช้กับข้อมูลของกรรมธรรม์เพิ่มเติม ได้มีการวัดค่าความผิดพลาดของการพยากรณ์ด้วยตัวชี้วัดมาตรฐาน ได้แก่ MAD (Mean Absolute Deviation), MAPE (Mean Absolute Percentage Error) และ MSE (Mean Squared Error) เพื่อวิเคราะห์ความแม่นยำของโมเดล ดังนี้

4.3.6.1 ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD)

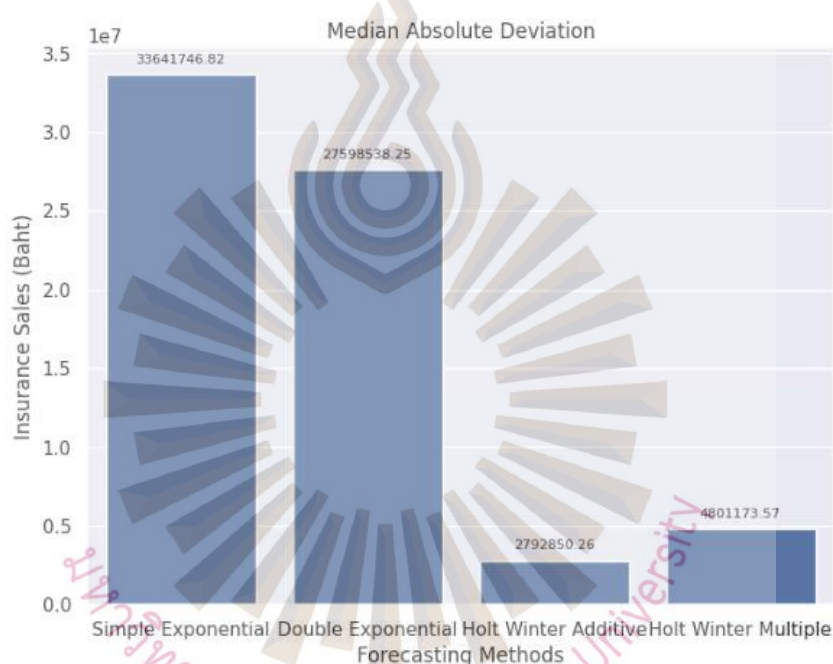
แผนภูมิแท่งรูปที่ 4.26 แสดงค่า Median Absolute Deviation (MAD) ของวิธีพยากรณ์เชิงเวลา ใช้วัดความเบี่ยงเบนสัมบูรณ์ของค่ากลางการพยากรณ์

- 1) Simple Exponential มีค่า MAD ประมาณ 3.36×10^7 หรือ 33,641,746.82 ซึ่งเป็นค่าความเบี่ยงเบนสูงที่สุดในกลุ่ม
- 2) Double Exponential มีค่า MAD ประมาณ 2.75×10^7 หรือ 27,598,538.25 ซึ่งมีค่าความเบี่ยงเบนต่ำกว่าวิธี Simple Exponential แต่ยังคงสูงอยู่
- 3) Holt-Winter Additive มีค่า MAD ประมาณ 2.79×10^6 ซึ่งต่ำกว่าวิธีอื่น สะท้อนความแม่นยำที่สูงขึ้น โดยโมเดลจับแนวโน้มและฤดูกาลได้ดี

4) Holt Winter Multiple แสดงผลการพยากรณ์ที่มีความแม่นยำในระดับสูง โดยมีค่า MAD อยู่ที่ประมาณ 4.80×10^6 หรือ 4,801,173.57 ซึ่งเป็นค่าความเบี่ยงเบนเฉลี่ยต่ำที่สุดเป็นอันดับสอง รองจากวิธี Holt Winter Additive

4.1) วิธี การบวกไฮลท์-วินเทอร์ มีค่าความเบี่ยงเบนสัมบูรณ์ของค่ากลางต่ำที่สุด แสดงถึงการพยากรณ์ที่แม่นยำที่สุดในแง่ของ MAD

4.2) ในทางกลับกัน ซิมเปิลเอ็กซ์โพเนนเชียล มีค่า MAD สูงที่สุด ซึ่งแสดงว่าการพยากรณ์ของวิธีนี้มีความคลาดเคลื่อนจากค่ากลางมากที่สุดในกลุ่ม



รูปที่ 4.26 แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม

4.3.6.2 ค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE)

แผนภูมิแท่งรูปที่ 4.27 แสดงค่า Mean Absolute Percentage Error (MAPE) ของวิธีการพยากรณ์เชิงเวลาแบบต่าง ๆ ซึ่งวัดข้อผิดพลาดสัมบูรณ์เฉลี่ยในรูปแบบเปอร์เซ็นต์ ดังนี้

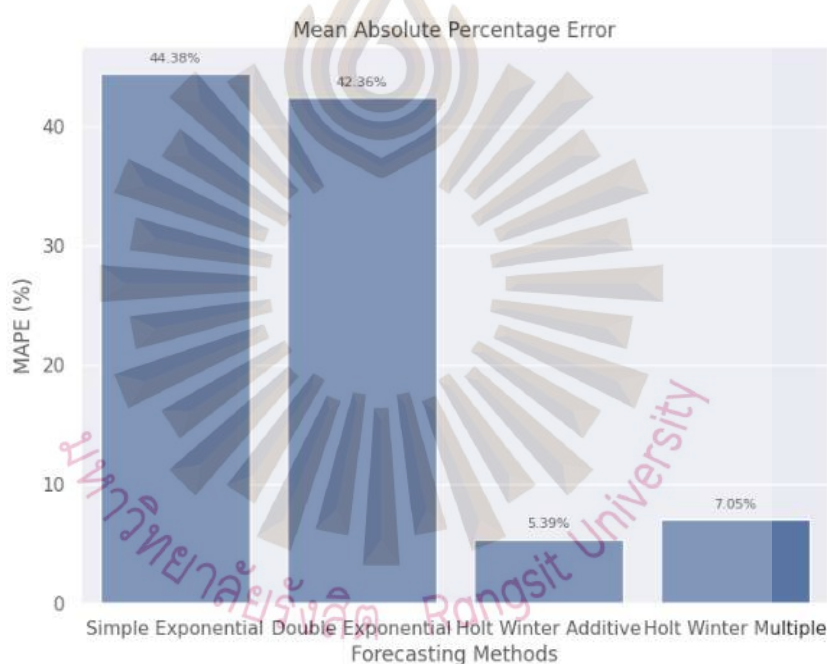
1) Simple Exponential มีค่า MAPE อยู่ที่ 44.38% ซึ่งเป็นค่าความคลาดเคลื่อนสัมบูรณ์เฉลี่ยสูงที่สุดในกลุ่ม

2) Double Exponential มีค่า MAPE อยู่ที่ 42.36% ซึ่งคล้ายคลึงกับวิธี Simple Exponential และยังถือว่าสูงมาก

3) Holt Winter Additive มีค่า MAPE อยู่ที่ 5.39% ซึ่งต่ำกว่าสองวิธีแรกมาก แสดงถึงประสิทธิภาพที่ดีกว่า

4) Holt Winter Multiple มีค่า MAPE อยู่ที่ 7.05% ซึ่งถือว่าต่ำมากเมื่อเทียบกับวิธี Simple และ Double Exponential แต่ยังสูงกว่าวิธี Holt Winter Additive เล็กน้อย

วิธีการบวกโฮลท์-วินเทอร์มีค่า MAPE ต่ำที่สุด แสดงถึงความสามารถในการจับแนวโน้มและฤดูกาลของข้อมูลได้แม่นยำกว่าวิธีอื่น ขณะที่วิธีอื่นไม่สามารถจัดการกับข้อมูลที่มีแนวโน้มหรือฤดูกาลได้ดีเท่า ทำให้โฮลท์-วินเทอร์มีความแม่นยำในการพยากรณ์สูงกว่า ซึ่งสะท้อนถึงประสิทธิภาพในการจับรูปแบบของข้อมูลที่มีความผันผวนและการเปลี่ยนแปลงอย่างต่อเนื่อง



รูปที่ 4.27 แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม

4.3.6.3 ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE)

แผนภูมิแท่งรูปที่ 4.28 แสดงค่าความผิดพลาดเฉลี่ยกำลังสอง (Mean Square Error, MSE) สำหรับวิธีการพยากรณ์ต่าง ๆ โดยแผนภูมิแสดงค่าความผิดพลาดของแต่ละวิธีด้วยความสูงต่างกัน ซึ่งแปลได้ว่าความแม่นยำของแต่ละวิธีมีความแตกต่างกันมาก ดังนี้

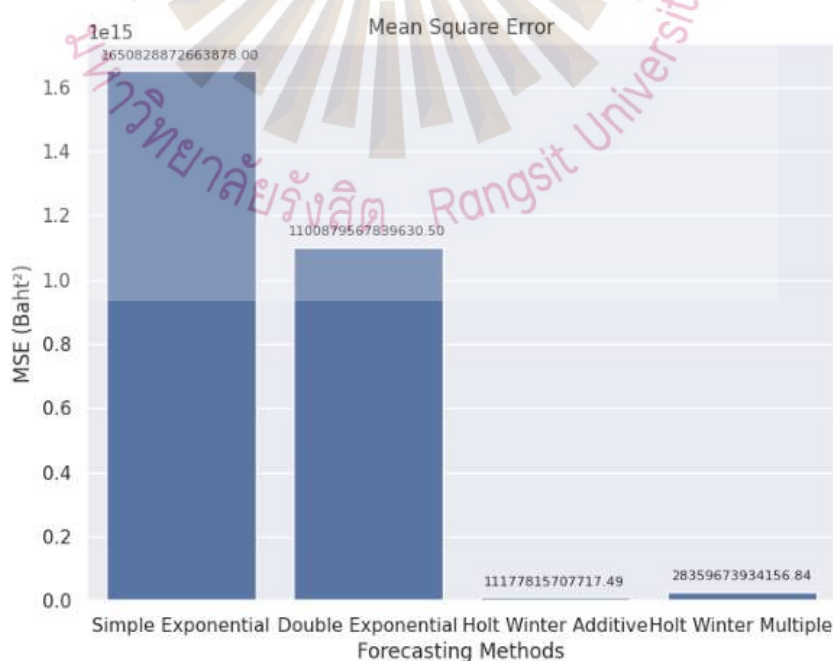
1) Simple Exponential มีค่าความผิดพลาดเฉลี่ยกำลังสองสูงที่สุด อยู่ที่ประมาณ 1.65×10^{15} ซึ่งหมายความว่าวิธีนี้มีความแม่นยำต่ำสุดในกลุ่มที่ทดลอง

2) Double Exponential มีค่าความผิดพลาดเฉลี่ยกำลังสองลดลงจาก Simple Exponential อยู่ที่ประมาณ 1.10×10^{15} ซึ่งแสดงให้เห็นถึงประสิทธิภาพที่ดีขึ้นในการติดตามแนวโน้มของข้อมูลที่มีการเปลี่ยนแปลงอย่างต่อเนื่อง

3) Holt Winter Additive มีค่าความผิดพลาดเฉลี่ยกำลังสองต่ำกว่าวิธีอื่น ๆ ที่ประมาณ 1.11×10^{15} ซึ่งสะท้อนถึงความสามารถในการจับทั้งแนวโน้มและฤดูกาลของข้อมูลได้อย่างแม่นยำ ทำให้สามารถติดตามการเปลี่ยนแปลงของข้อมูลได้อย่างใกล้เคียงกับค่าจริง และยังคงข้อผิดพลาดจากความผันผวนที่เกิดขึ้นในแต่ละช่วงเวลาได้ดี

4) Holt Winter Multiple มีค่าความผิดพลาดเฉลี่ยกำลังสองต่ำสุด อยู่ที่ประมาณ 2.83×10^{12} ซึ่งแสดงให้เห็นว่าวิธีนี้มีความแม่นยำสูงสุดในกลุ่มที่ทดลอง และสามารถสะท้อนการเปลี่ยนแปลงของข้อมูลได้อย่างแม่นยำในทุกช่วงเวลาที่มีความผันผวน

วิธี Holt Winter Multiple มีความแม่นยำสูงสุดในการพยากรณ์ เนื่องจากมีค่าความผิดพลาดเฉลี่ยกำลังสองต่ำที่สุดเมื่อเปรียบเทียบกับวิธีการอื่น ๆ ที่ทดลอง ซึ่งแสดงให้เห็นว่าการพยากรณ์ด้วยวิธีนี้สามารถให้ผลลัพธ์ที่แม่นยำและน่าเชื่อถือมากที่สุด โดยสามารถสะท้อนการเปลี่ยนแปลงในข้อมูลได้อย่างมีประสิทธิภาพ อีกทั้งยังสามารถจับแนวโน้มและฤดูกาลของข้อมูลได้อย่างเหมาะสม ทำให้เหมาะสมกับข้อมูลที่มีความผันผวนและการเปลี่ยนแปลงตามช่วงเวลา



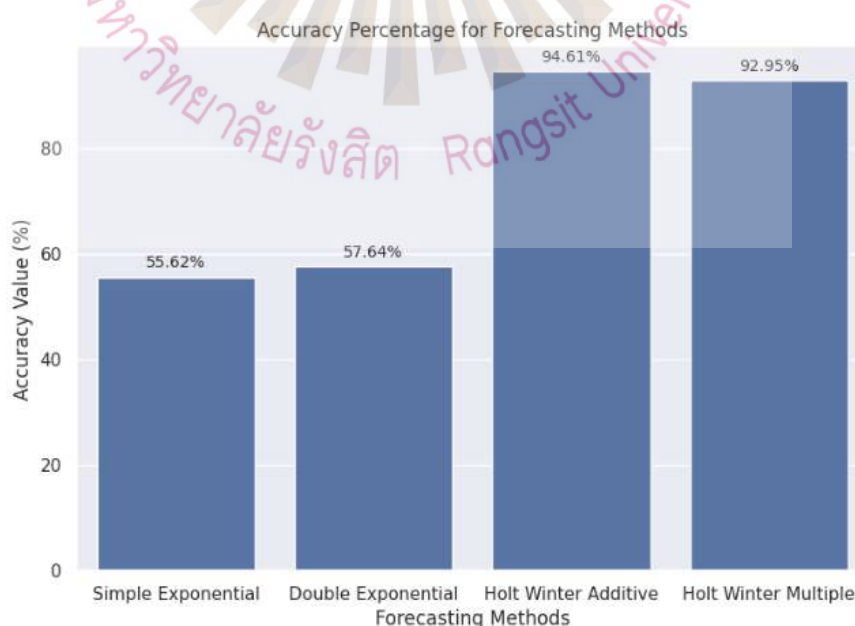
รูปที่ 4.28 แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) สำหรับการพยากรณ์ยอดขายประกันชีวิตกรมธรรม์สัญญาเพิ่มเติม

4.3.7 ผลลัพธ์ความแม่นยำของการพยากรณ์กรรมธรรม์เพิ่มเติม

แผนภูมิแท่งรูปที่ 4.29 นำเสนอแสดงถึงเปอร์เซ็นต์ความแม่นยำของวิธีการพยากรณ์ต่าง ๆ โดยแสดงให้เห็นถึงประสิทธิภาพในการพยากรณ์ของแต่ละวิธี และสามารถเปรียบเทียบค่าความแม่นยำเพื่อเลือกวิธีที่เหมาะสมที่สุด ดังนี้

- 1) Simple Exponential มีความแม่นยำอยู่ที่ 55.62%
- 2) Double Exponential มีความแม่นยำเพิ่มขึ้นเล็กน้อยที่ 57.64%
- 3) Holt Winter Additive มีความแม่นยำสูงถึง 94.61%
- 4) Holt Winter Multiple มีความแม่นยำที่ 92.95%

วิธีการพยากรณ์แบบโฮลท์-วินเทอร์ แอดดิทีฟมีความแม่นยำสูงสุดที่ 94.61% รองลงมาเป็นโฮลท์-วินเทอร์ มัลติพลีเคทีฟที่ 92.95% ส่วนซิมเปิลและดับเบิลเอ็กซ์โพเนนเชียลมีความแม่นยำต่ำกว่าที่ประมาณ 55 - 58% ซึ่งแสดงให้เห็นว่าผลลัพธ์จากวิธีการโฮลท์-วินเทอร์มีประสิทธิภาพสูงกว่าวิธีเอ็กซ์โพเนนเชียลในการจัดการกับข้อมูลที่มีลักษณะของแนวโน้มและฤดูกาล โดยเฉพาะในกรณีที่ข้อมูลมีความแปรผันตามฤดูกาลอย่างชัดเจน เนื่องจากวิธีนี้สามารถสะท้อนรูปแบบการเปลี่ยนแปลงในแต่ละช่วงเวลาได้อย่างแม่นยำ นอกจากนี้ยังสามารถปรับตัวได้ดีในกรณีที่ข้อมูลมีการผันผวนหรือมีลักษณะฤดูกาลที่ไม่เป็นระเบียบ ซึ่งส่งผลให้การพยากรณ์มีความแม่นยำสูงขึ้นเมื่อเปรียบเทียบกับวิธีการพยากรณ์ประเภทอื่น



รูปที่ 4.29 แผนภูมิแท่งแสดงผลลัพธ์ความแม่นยำของการพยากรณ์กรรมธรรม์เพิ่มเติม

4.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์

การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยใช้ 4 วิธี ผลลัพธ์แสดงดังตารางที่ 4.4 โดยการอัปโหลดจะใช้ไฟล์ชื่อ Sales_Total.csv ส่งไปที่โปรแกรมภาษา Python

ตารางที่ 4.4 ผลการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีต่าง ๆ

Period	Year	Quarter	Sales	ผลจาก HWA	ผลจาก HWM	ผลจาก SES	ผลจาก DES
1	2018	1	134482310.08	None	None	134482310.08	None
2	2018	2	262727733.34	None	None	134482310.08	None
3	2018	3	452900410.38	None	None	147306852.41	390973156.60
4	2018	4	617035583.60	None	None	177866208.21	32842575.70
5	2019	1	143967627.71	134482310.08	134482310.08	221783145.75	680849005.78
6	2019	2	288420585.07	262727733.34	262727733.34	214001593.95	693902931.11
7	2019	3	434417872.10	452900410.38	452900410.38	221443493.06	725127015.92
8	2019	4	598032844.76	617035583.60	617035583.60	242740930.96	773491558.30
9	2020	1	144558479.58	141122032.42	145648048.99	278270122.34	841397012.46
10	2020	2	277346871.55	280712729.55	277144341.26	264898958.06	791089732.09
11	2020	3	417580168.02	439962633.58	426284344.17	266143749.41	767126728.98
12	2020	4	584850027.07	603733666.41	589928733.24	281287391.27	769012054.57
13	2021	1	147558504.94	143527545.43	142518786.02	311643654.85	800291046.30
14	2021	2	283538582.69	278356628.95	278166469.17	295235139.86	724801284.43
15	2021	3	425328721.41	424294907.69	427220006.76	294065484.14	682780236.45
16	2021	4	595426000.22	590515118.87	599607869.50	307191807.87	672372395.51
17	2022	1	144721029.20	146349217.09	148135220.99	336015227.11	696526650.61
18	2022	2	278299004.51	146349217.09	148135220.99	361956304.42	736070184.77
19	2022	3	421803444.50	146349217.09	148135220.99	385303274.00	775613718.93
20	2022	4	589651266.19	146349217.09	148135220.99	406315546.62	815157253.09

4.4.1 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ด้วยวิธีการบวกโฮลท์-วินเทอร์

จากตารางที่ 4.4 ข้อมูลการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีการบวกโฮลท์-วินเทอร์ แสดงถึงยอดขายที่มีการเปลี่ยนแปลงตามฤดูกาลและแนวโน้มในแต่ละไตรมาสตลอดช่วงปี ค.ศ. 2018 - 2022 แยกตามไตรมาส ซึ่งผลการพยากรณ์แสดงความแตกต่างในความแม่นยำและแนวโน้มตามวิธีการที่ใช้ โดยเฉพาะอย่างยิ่งในช่วงเวลาที่เกิดความผันผวนของยอดขายอย่างมีนัยสำคัญ ดังนี้

4.4.1.1 การพยากรณ์โดยวิธีการบวกโฮลท์-วินเทอร์

- 1) วิธีการนี้เป็นการพยากรณ์ข้อมูลอนุกรมเวลาที่มีทั้งแนวโน้ม (Trend) และฤดูกาล (Seasonality) โดยใช้การรวมแนวโน้มเชิงเส้นเข้ากับฤดูกาลแบบแอดดิทีฟ
- 2) วิธีนี้เหมาะสมสำหรับข้อมูลที่มีแนวโน้มเพิ่มขึ้นหรือลดลงในช่วงเวลาหนึ่งและลักษณะของฤดูกาลที่คงที่ (ฤดูกาลไม่มีการเปลี่ยนแปลงมาก)

4.4.1.2 การเปรียบเทียบยอดขายจริงและยอดขายที่พยากรณ์

- 1) ปี ค.ศ. 2018 เป็นปีเริ่มต้นของชุดข้อมูล จึงยังไม่สามารถดำเนินการพยากรณ์ได้ เนื่องจากจำเป็นต้องมีข้อมูลย้อนหลังที่เพียงพอสำหรับการคำนวณและสร้างแบบจำลองการพยากรณ์ที่ถูกต้องแม่นยำ
- 2) ปี ค.ศ. 2019 การพยากรณ์ในแต่ละไตรมาสยังคงต่ำกว่ายอดขายจริง โดยเฉพาะไตรมาสที่ 3 และ 4 ซึ่งการพยากรณ์มีความคลาดเคลื่อนมากขึ้น
- 3) ปี ค.ศ. 2020 การพยากรณ์เริ่มเข้าใกล้ยอดขายจริงมากขึ้น แต่ยังคงมีความคลาดเคลื่อนในบางไตรมาส โดยเฉพาะในไตรมาสที่ 3 และ 4 ที่ยอดขายจริงสูงกว่าการพยากรณ์อย่างมีนัยสำคัญ
- 4) ปี ค.ศ. 2021 การพยากรณ์ยังคงมีความแม่นยำในระดับหนึ่ง โดยเฉพาะในไตรมาสที่ 3 และ 4 ซึ่งการพยากรณ์ใกล้เคียงกับยอดขายจริงมากขึ้น
- 5) ปี ค.ศ. 2022 การพยากรณ์ในไตรมาสที่ 2 3 และ 4 เริ่มมีความคลาดเคลื่อนมากขึ้น โดยคาดการณ์ยอดขายต่ำกว่าในทุกลไตรมาส แม้ว่ายอดขายจริงจะเพิ่มขึ้น

4.4.1.3 แนวโน้มของการพยากรณ์และยอดขายจริง

- 1) การพยากรณ์ใช้การเพิ่มแนวโน้มเชิงเส้นแบบแอดดิทีฟ แต่ไม่ได้คำนึงถึงความผันผวนที่มากขึ้นของยอดขายในช่วงเวลา ซึ่งทำให้ในระยะยาวมีการพยากรณ์ที่ต่ำกว่ายอดขายจริง
- 2) ยอดขายจริงแสดงการเพิ่มขึ้นอย่างรวดเร็วและผันผวนในบางช่วงที่อาจเกิดจากฤดูกาลหรือเหตุการณ์พิเศษที่ไม่ได้คำนวณในแบบจำลอง

4.4.1.4 ความแม่นยำของการพยากรณ์ วิธีการบวกโฮลท์-วินเทอร์ มีความแม่นยำในการพยากรณ์ได้ดีในช่วงต้น แต่มีความคลาดเคลื่อนสูงในช่วงที่ข้อมูลมีความผันผวนหรือมีการเปลี่ยนแปลงอย่างรวดเร็วของยอดขายจริง

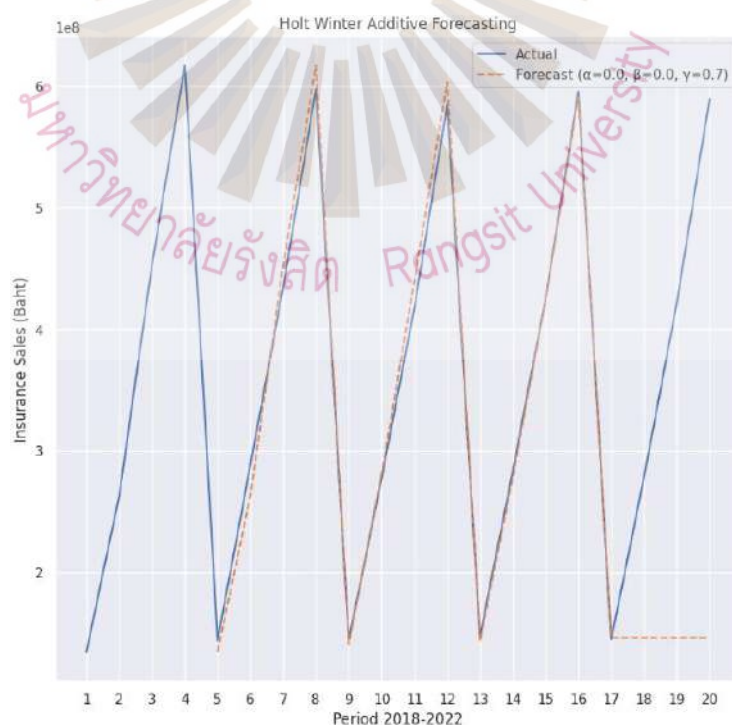
1) ข้อดี สามารถจับแนวโน้มเชิงเส้นและฤดูกาลของข้อมูลได้ดีในบางช่วง และให้ความแม่นยำสูงเมื่อแนวโน้มของข้อมูลมีความเสถียร เหมาะสำหรับการพยากรณ์ข้อมูลที่มีรูปแบบเป็นคาบและมีโครงสร้างที่ไม่ซับซ้อนมากนัก

2) ข้อจำกัดไม่สามารถจัดการกับความผันผวนหรือแนวโน้มที่มีการเปลี่ยนแปลงอย่างรวดเร็วได้ดี โดยเฉพาะในกรณีที่ข้อมูลมีฤดูกาลที่เปลี่ยนแปลง

3) ข้อเสนอแนะเพิ่มเติม อาจต้องปรับเปลี่ยนวิธีการพยากรณ์หรือใช้วิธีที่สามารถจัดการกับความผันผวนได้ดีกว่า เช่น การพยากรณ์แบบ Holt-Winter

วิธีการบวก Holt-Winter สามารถให้ผลลัพธ์ที่ดีในการพยากรณ์ข้อมูลที่มีแนวโน้มและฤดูกาล แต่มีข้อจำกัดเมื่อเจอความผันผวนที่รุนแรงหรือแนวโน้มที่เปลี่ยนแปลงอย่างรวดเร็วในช่วงเวลาหนึ่ง

รูปที่ 4.30 แสดงแผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีการบวก Holt-Winter ระหว่างปี ค.ศ. 2018 - 2022 โดยมีพารามิเตอร์ $\alpha = 0.0$ $\beta = 0.0$ และ $\gamma = 0.7$ เพื่อปรับปรุงความแม่นยำของการพยากรณ์



รูปที่ 4.30 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีการบวก Holt-Winter ระหว่างปี ค.ศ. 2018 - 2022

4.4.1.5 เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในช่วงปี ค.ศ. 2018 - 2022 ซึ่งมีรูปแบบการแกว่งตัวที่ชัดเจนตามฤดูกาล โดยมีความผันผวนสูงในแต่ละช่วงเวลา

1) เส้นประสีส้ม (Forecast) แสดงผลการพยากรณ์ที่มีความใกล้เคียงกับค่าจริงมาก โมเดลสามารถจับแนวโน้มและรูปแบบตามฤดูกาลได้ดีในหลายช่วงเวลา เช่น ช่วงที่ค่าจริงพุ่งสูงหรือช่วงที่ลดต่ำลง

2) การตั้งค่าพารามิเตอร์

2.1) $\alpha = 0.0$ การพยากรณ์ไม่ได้ใช้การปรับตัวตามค่าใหม่ของข้อมูล ทำให้ค่าเฉลี่ยเดิมถูกนำมาใช้อย่างต่อเนื่อง

2.2) $\beta = 0.0$ โมเดลไม่เน้นการจับแนวโน้ม (Trend) ของข้อมูล ซึ่งสอดคล้องกับลักษณะของข้อมูลที่มีการแกว่งตัวเป็นฤดูกาลมากกว่าแนวโน้ม

2.3) $\gamma = 0.7$ โมเดลให้ความสำคัญกับการจับความผันผวนตามฤดูกาลได้เป็นอย่างดี

4.4.1.6 การพยากรณ์โดยใช้การบวกโฮลท์-วินเทอร์สามารถจับรูปแบบการเคลื่อนไหวตามฤดูกาลได้ดี อย่างไรก็ตาม ในช่วงที่ข้อมูลจริงเปลี่ยนแปลงอย่างรวดเร็ว โมเดลอาจมีการเบี่ยงเบนเล็กน้อย เช่น ในช่วงเวลา 5 - 6 และ 19 - 20

การพยากรณ์ด้วยวิธีการบวกโฮลท์-วินเทอร์ในกรณีนี้ทำงานได้ดี โดยสามารถจับรูปแบบตามฤดูกาลได้อย่างแม่นยำ แต่แนวโน้มของข้อมูลไม่ได้ถูกจับได้มากนัก เนื่องจากการตั้งค่า $\beta = 0.0$ ซึ่งทำให้โมเดลไม่เน้นการปรับตามแนวโน้ม

4.4.2 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ด้วยวิธีการคูณโฮลท์-วินเทอร์

จากตารางที่ 4.4 ข้อมูลการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีการคูณโฮลท์-วินเทอร์ แสดงถึงยอดขายที่มีการเปลี่ยนแปลงตามฤดูกาลและแนวโน้มในแต่ละไตรมาสตลอดช่วงปี ค.ศ. 2018 - 2022 แยกตามไตรมาส

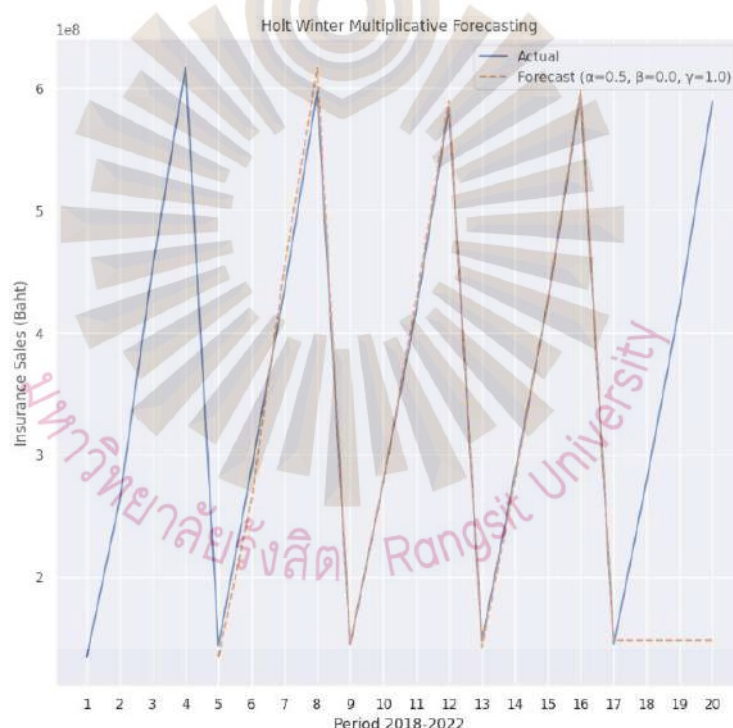
แผนภูมิเส้นจากรูปที่ 4.31 แสดงผลการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยใช้เทคนิค Holt-Winters แบบมีฤดูกาลซึ่งเหมาะสมกับข้อมูลที่มีความผันผวนสูง เพื่อเปรียบเทียบระหว่างค่าจริงกับค่าคาดการณ์ในช่วงปี ค.ศ. 2018 - 2022 มีรายละเอียดดังนี้

4.4.2.1 การพยากรณ์และยอดขาย

- 1) ในตารางนี้แสดงข้อมูลของยอดขายจริงและการพยากรณ์ยอดขายที่ใช้วิธีการคูณโฮลท์-วินเทอร์สำหรับแต่ละปีและไตรมาส ตั้งแต่ปี ค.ศ. 2018 - 2022
- 2) สำหรับปี ค.ศ. 2018 ยังไม่มีการพยากรณ์เพราะเป็นปีเริ่มต้น
- 3) ตั้งแต่ปี ค.ศ. 2019 - 2022 จะพบการพยากรณ์ของยอดขายตามไตรมาสที่คำนวณจากข้อมูลยอดขายที่มีในปีที่แล้ว

4.4.2.2 แนวโน้มยอดขาย

- 1) ปี ค.ศ. 2018 ยอดขายเริ่มต้นต่ำในไตรมาสแรกและเพิ่มสูงขึ้นอย่างรวดเร็วในไตรมาสที่ 4



รูปที่ 4.31 แผนภูมิเส้นการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีการคูณโฮลท์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022

- 2) ปี ค.ศ. 2019 - 2021 ยอดขายมีแนวโน้มที่สูงขึ้นในแต่ละปี โดยมีการเพิ่มขึ้นในแต่ละไตรมาส เมื่อเปรียบเทียบกับปีที่ผ่านมา
- 3) ปี ค.ศ. 2022 ยอดขายสูงมากในไตรมาสที่ 4 แต่มีแนวโน้มไม่สอดคล้องกับการพยากรณ์อย่างมากในไตรมาสอื่น ๆ

4.4.2.3 ความแม่นยำของการพยากรณ์

- 1) ปี ค.ศ. 2019 - 2021 การพยากรณ์ในปีเหล่านี้ค่อนข้างใกล้เคียงกับยอดขายจริง แสดงให้เห็นว่าการพยากรณ์ของ Holt-Winters' Multiplicative ค่อนข้างแม่นยำ
- 2) ปี ค.ศ. 2022 การพยากรณ์ในปีนี้ดูไม่สอดคล้องกับยอดขายจริง โดยยอดขายในปี 2022 ต่ำกว่าค่าพยากรณ์ในไตรมาส 1 - 4 ซึ่งอาจเป็นสัญญาณของการเปลี่ยนแปลงในแนวโน้มหรือฤดูกาลที่อาจไม่ได้รับการคำนึงถึงในโมเดล

4.4.2.4 การวิเคราะห์เชิงลึก

- 1) การพยากรณ์ในปี ค.ศ. 2022 ไม่สามารถจับแนวโน้มของยอดขายจริงได้ดี อาจเกิดจากความเปลี่ยนแปลงของข้อมูลที่ไม่สอดคล้องกับรูปแบบเดิมที่ใช้ฝึกโมเดล โดยเฉพาะปัจจัยภายนอก เช่น เศรษฐกิจหรือพฤติกรรมผู้บริโภคที่เปลี่ยนไป ส่งผลให้ความแม่นยำของการพยากรณ์ลดลง
- 2) ความไม่ตรงกันระหว่างการพยากรณ์และยอดขายจริงในปี ค.ศ. 2022 อาจชี้ให้เห็นถึงความจำเป็นในการปรับแต่งโมเดลหรือการพิจารณาปัจจัยอื่น ๆ ที่อาจมีผลกระทบต่อยอดขาย เช่น การเปลี่ยนแปลงในแนวโน้มตลาด หรือเหตุการณ์ภายนอกที่ไม่ได้ถูกพิจารณาในการสร้างโมเดล

การพยากรณ์ด้วยวิธีการคูณโฮลท์-วินเทอร์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ มีประสิทธิภาพดีในปี ค.ศ. 2019 - 2021 แต่มีข้อบกพร่องในปี ค.ศ. 2022 ซึ่งอาจบ่งชี้ถึงการเปลี่ยนแปลงในแนวโน้มยอดขายที่ไม่สามารถคาดการณ์ได้จากโมเดลที่ใช้ ข้อเสนอแนะคือการทบทวนและปรับปรุงโมเดลการพยากรณ์เพื่อให้สอดคล้องกับข้อมูลยอดขายจริงในอนาคต รูปที่ 4.31 แสดงแผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีการคูณโฮลท์-วินเทอร์ ระหว่างปี ค.ศ. 2018 - 2022 โดยมีพารามิเตอร์ $\alpha = 0.5$ $\beta = 0.0$ และ $\gamma = 1.0$ ข้อมูลในแผนภูมิเส้นแสดงผลการเปรียบเทียบระหว่างค่าจริงและการพยากรณ์

4.4.2.5 เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในช่วงปี ค.ศ. 2018 - 2022 มีรูปแบบการแกว่งตัวชัดเจนตามฤดูกาล โดยค่าจริงมีความผันผวนสูงและแกว่งตัวขึ้นลงอย่างรวดเร็วในแต่ละช่วงเวลา ซึ่งสะท้อนถึงปัจจัยภายนอกที่อาจส่งผลกระทบต่อยอดขายในแต่ละปี เช่น สถานการณ์ทางเศรษฐกิจและการเปลี่ยนแปลงในพฤติกรรมของผู้บริโภค การผันผวนนี้ยังอาจเกิดจากเหตุการณ์ที่ไม่สามารถคาดการณ์ล่วงหน้า เช่น การระบาดของโรคหรือวิกฤตเศรษฐกิจ

4.4.2.6 เส้นประสีส้ม (Forecast) แสดงผลการพยากรณ์ ซึ่งมีลักษณะใกล้เคียงกับค่าจริงในหลายช่วงเวลา โดยเฉพาะในช่วงที่มีการเปลี่ยนแปลงแบบเป็นฤดูกาล แต่ในบางช่วงเวลา เช่น ช่วงที่ข้อมูลจริงลดลงเป็นศูนย์ (เช่น ช่วงที่ 6 - 7 และ 16 - 17) การพยากรณ์กลับไม่สามารถจับการลดลงอย่างรวดเร็วของข้อมูลจริงได้ทั้งหมด

4.4.2.7 การตั้งค่าพารามิเตอร์

- 1) $\alpha = 0.5$ ไม่ได้ให้ความสำคัญกับการปรับค่าเฉลี่ยตามข้อมูลใหม่
- 2) $\beta = 0.0$ ไม่ได้เน้นการจับแนวโน้มระยะยาว
- 3) $\gamma = 1.0$ เน้นการจับความผันผวนตามฤดูกาล ซึ่งทำให้โมเดลสามารถพยากรณ์ได้ใกล้เคียงกับค่าจริงในช่วงที่มีการแกว่งตัวเป็นฤดูกาล

4.4.2.8 ความแม่นยำของการพยากรณ์ การพยากรณ์แบบทวิคูณนี้สามารถจับรูปแบบการแกว่งตัวตามฤดูกาลได้ดีในบางช่วงเวลา อย่างไรก็ตาม ในช่วงที่ข้อมูลจริงลดลงอย่างรวดเร็วหรือเป็นศูนย์ โมเดลไม่สามารถพยากรณ์ค่าได้อย่างแม่นยำ ซึ่งเป็นข้อจำกัดของการพยากรณ์แบบทวิคูณในบางกรณีที่มีข้อมูลมีการเปลี่ยนแปลงอย่างรุนแรง การใช้โมเดลที่ยืดหยุ่นและพิจารณาปัจจัยภายนอกจะสนับสนุนการปรับปรุงผลลัพธ์

การพยากรณ์ด้วยวิธีการโซลท์-วินเทอร์แบบคูณยังมีข้อจำกัดในกรณีที่ข้อมูลมีความเปลี่ยนแปลงอย่างฉับพลันจากปัจจัยภายนอกที่ไม่สามารถระบุแนวโน้มหรือฤดูกาลได้อย่างชัดเจน เช่น วิกฤตการณ์ทางเศรษฐกิจ การเปลี่ยนแปลงนโยบายของรัฐ หรือเหตุการณ์ไม่คาดคิดอื่น ๆ ซึ่งอาจส่งผลให้แบบจำลองไม่สามารถจับทิศทางของข้อมูลได้อย่างถูกต้องและส่งผลต่อความแม่นยำในการพยากรณ์

4.4.3 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

ตารางที่ 4.4 ได้แสดงผลวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายสำหรับการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ช่วงปี ค.ศ. 2018 - 2022 แยกตามไตรมาส

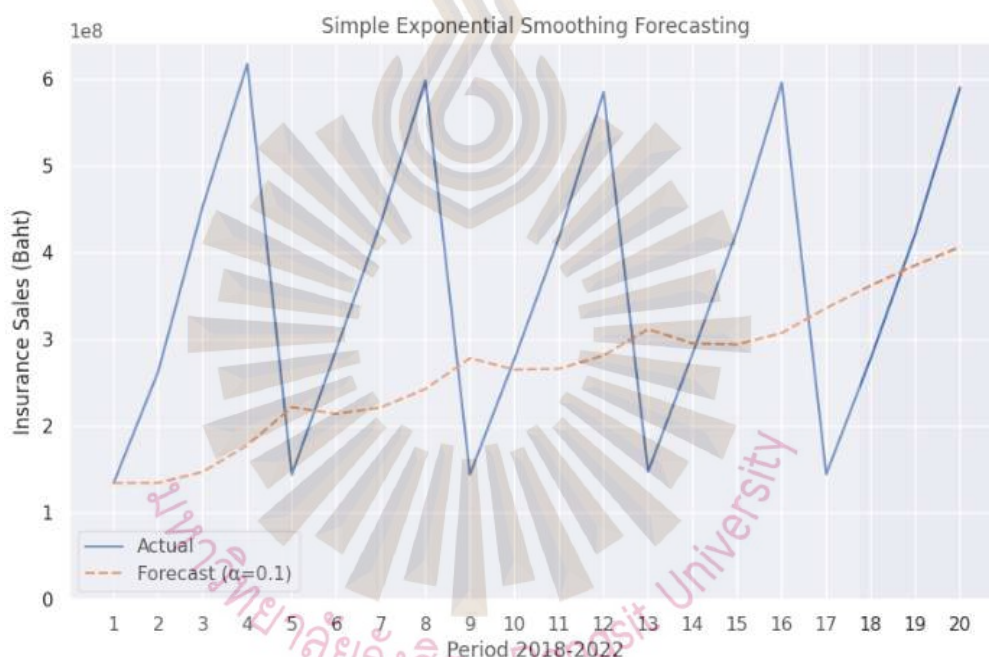
รูปที่ 4.32 แสดงแผนภูมิเส้นเปรียบเทียบระหว่างค่าจริงและค่าพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยใช้วิธีการปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (Simple Exponential Smoothing) ที่ค่า α เท่ากับ 0.1 ดังนี้

4.4.3.1 การพยากรณ์และยอดขาย

1) ปี ค.ศ. 2018

1.1) การพยากรณ์ปีนี้เริ่มจากยอดขายไตรมาสแรกและปรับขึ้นตามยอดขายถัดไป โดยพยากรณ์ดีขึ้นในไตรมาส 4 เมื่อยอดขายสูงขึ้น ซึ่งสะท้อนถึงการปรับตัวของโมเดลตามการเปลี่ยนแปลงในข้อมูลจริง

1.2) การพยากรณ์เริ่มจับสัญญาณการเพิ่มขึ้นของยอดขายจริงได้ดีในไตรมาสที่ 4 ซึ่งแสดงให้เห็นถึงความสามารถของโมเดลในการปรับตัวตามการเปลี่ยนแปลงของข้อมูลที่เกิดขึ้นในช่วงเวลานั้น



รูปที่ 4.32 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย ระหว่างปี ค.ศ. 2018 - 2022

2) ปี ค.ศ. 2019 - 2021

2.1) การพยากรณ์เริ่มมีความแม่นยำมากขึ้นเมื่อเวลาผ่านไป โดยการพยากรณ์สะท้อนการเพิ่มขึ้นของยอดขายจริงในแต่ละไตรมาส

2.2) ปี ค.ศ. 2020 - 2021 มีแนวโน้มการพยากรณ์ที่ดีขึ้น โดยสามารถจับการเพิ่มขึ้นของยอดขายได้ดีขึ้น ซึ่งสะท้อนให้เห็นถึงประสิทธิภาพที่ดีขึ้นของแบบจำลองในการติดตามแนวโน้มและการเปลี่ยนแปลงของข้อมูลในช่วงเวลานั้น

3) ปี ค.ศ. 2022

3.1) การพยากรณ์ค่อนข้างสูงกว่าในปี ค.ศ. 2022 โดยเฉพาะในไตรมาสที่ 3 และ 4 ซึ่งแสดงให้เห็นถึงการเพิ่มขึ้นอย่างรวดเร็วในยอดขายที่ไม่สอดคล้องกับค่าพยากรณ์

4.4.3.2 แนวโน้มของยอดขาย

- 1) ปี ค.ศ. 2018 ยอดขายมีการเพิ่มขึ้นอย่างรวดเร็วในไตรมาสที่ 3 และ 4 การพยากรณ์ค่อย ๆ ปรับตัวตามการเพิ่มขึ้นของยอดขาย
- 2) ปี ค.ศ. 2019 - 2021 การพยากรณ์สามารถจับแนวโน้มการเพิ่มขึ้นของยอดขายได้ดีขึ้น โดยเฉพาะในปี ค.ศ. 2020 และ ค.ศ. 2021
- 3) ปี ค.ศ. 2022 มีการพยากรณ์สูงเกินไปในแต่ละไตรมาส ซึ่งอาจสะท้อนถึงการพยากรณ์ไม่สามารถติดตามการเพิ่มขึ้นของยอดขายที่แท้จริงได้

4.4.3.3 ความแม่นยำของการพยากรณ์

- 1) ปี ค.ศ. 2018 การพยากรณ์ยังไม่แม่นยำมากในไตรมาสแรกและสอง แต่ดีขึ้นในไตรมาสที่ 4
- 2) ปี ค.ศ. 2019 - 2021 การพยากรณ์มีการปรับตัวให้ใกล้เคียงกับยอดขายจริงมากขึ้น โดยเฉพาะในปี ค.ศ. 2020 และ ค.ศ. 2021
- 3) ปี 2022 การพยากรณ์ในปีนี้คาดการณ์สูงเกินไปจากยอดขายจริง ซึ่งอาจเป็นผลจากการที่โมเดล ซิมเปิลเอ็กซ์โพเนนเชียล ไม่สามารถจับการเปลี่ยนแปลงอย่างรวดเร็วได้

4.4.3.4 การวิเคราะห์เชิงลึก

- 1) การพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายเหมาะสมสำหรับข้อมูลที่ไม่มีแนวโน้มที่ชัดเจนหรือฤดูกาล เนื่องจากโมเดลนี้ไม่สามารถจับแนวโน้มหรือฤดูกาลได้ดี จึงทำให้มีความแม่นยำต่ำในกรณีที่ข้อมูลมีการเปลี่ยนแปลงตามฤดูกาลหรือแนวโน้มที่ชัดเจน ส่งผลให้ไม่สามารถให้การพยากรณ์ที่มีประสิทธิภาพในสภาพแวดล้อมที่ซับซ้อน
- 2) การพยากรณ์ที่สูงเกินไปในปี ค.ศ. 2022 การพยากรณ์สูงเกินไปอาจเป็นเพราะโมเดลซิมเปิลเอ็กซ์โพเนนเชียล ใช้ค่าคาดการณ์เฉลี่ยซึ่งอาจไม่ทันกับการเปลี่ยนแปลงที่รวดเร็วในยอดขาย การพยากรณ์สูงเกินไปอาจเกิดจากการที่โมเดลไม่สามารถปรับตัวให้ทันกับการเปลี่ยนแปลงที่เกิดขึ้นอย่างรวดเร็วในข้อมูล

การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่ายสามารถจับแนวโน้มการเพิ่มขึ้นของยอดขายได้ดีในช่วงปี ค.ศ. 2019 - 2021 แต่มีการพยากรณ์สูงเกินไปในปี ค.ศ. 2022 เนื่องจากวิธีนี้มีข้อจำกัดในการจับแนวโน้มและฤดูกาลที่ซับซ้อน ซึ่งอาจทำให้การพยากรณ์ไม่แม่นยำเมื่อยอดขายมีการเปลี่ยนแปลงอย่างรวดเร็ว ดังนั้น การใช้วิธีพยากรณ์ที่มีความซับซ้อนมากขึ้น เช่น ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า หรือ Holt-Winters จะสามารถยกระดับความแม่นยำและความสามารถในการจับความผันผวนของข้อมูลในระยะยาวได้ดียิ่งขึ้น และสามารถรองรับการเปลี่ยนแปลงที่รวดเร็วในข้อมูลได้ดีกว่า

4.4.3.5 การพยากรณ์และยอดขาย

- 1) ปี ค.ศ. 2018 การพยากรณ์เริ่มต้นจากยอดขายจริงในไตรมาสแรกและค่อย ๆ ปรับตัวตามการเปลี่ยนแปลงของยอดขายจริงที่ตามมาในไตรมาสถัดไป
- 2) ตัวอย่าง ไตรมาสที่ 2 การพยากรณ์ยังคงต่ำกว่าค่าจริง และการพยากรณ์ในไตรมาสที่ 3 และ 4 ค่อย ๆ เข้าใกล้ยอดขายจริงมากขึ้น
- 3) ปี ค.ศ. 2019 - 2022 การพยากรณ์เริ่มมีความแม่นยำมากขึ้นในปีหลัง ๆ แม้ว่าจะมีความเบี่ยงเบนอยู่บ้าง

4.4.3.6 แนวโน้มของยอดขาย

- 1) ปี ค.ศ. 2018 มีการเพิ่มขึ้นของยอดขายในไตรมาสที่ 3 และ 4 ซึ่งไม่สามารถจับการเปลี่ยนแปลงนี้ได้อย่างรวดเร็ว
- 2) ปี ค.ศ. 2019 - 2021 การพยากรณ์ดูเหมือนจะมีแนวโน้มที่จะปรับให้ใกล้เคียงกับยอดขายจริงมากขึ้น
- 3) ปี ค.ศ. 2022 การพยากรณ์มีการเพิ่มขึ้นอย่างชัดเจนในแต่ละไตรมาส ซึ่งสะท้อนถึงการเติบโตของยอดขายที่เพิ่มขึ้น

4.4.3.7 ความแม่นยำของการพยากรณ์

- 1) ปี ค.ศ. 2018 การพยากรณ์ในช่วงต้นปีมีความเบี่ยงเบนจากยอดขายจริง แต่มีแนวโน้มที่ดีขึ้นในไตรมาสที่ 4 การพยากรณ์ในไตรมาสที่ 4 มีความแม่นยำมากขึ้น เนื่องจากข้อมูลเริ่มสะท้อนแนวโน้มที่ชัดเจน
- 2) ปี ค.ศ. 2019 - 2021 การพยากรณ์มีความแม่นยำดีขึ้น โดยเฉพาะในปี ค.ศ. 2020 และ ค.ศ. 2021 ซึ่งแนวโน้มยอดขายที่เพิ่มขึ้นสอดคล้องกับการพยากรณ์

3) ปี ค.ศ. 2022 การพยากรณ์ในปีนี้นี้น่าจะสูงกว่ายอดขายจริง แสดงให้เห็นว่าโมเดลมีแนวโน้มที่จะคาดการณ์ยอดขายสูงเกินไปในปี

4.4.3.8 การวิเคราะห์เชิงลึก

1) การพยากรณ์ด้วยวิธีซิมเพล็กซ์โพเนนเชียล ใช้การตั้งค่า Alpha ที่เป็นตัวกำหนดการถ่วงน้ำหนักของข้อมูลล่าสุดมากกว่าข้อมูลก่อนหน้า หากค่า Alpha สูง การพยากรณ์จะตอบสนองต่อการเปลี่ยนแปลงอย่างรวดเร็ว แต่มีความเสี่ยงของความผันผวนสูง

2) แนวโน้มการพยากรณ์ การพยากรณ์แสดงให้เห็นว่าโมเดลสามารถจับแนวโน้มทั่วไปของยอดขายได้ดีในปี ค.ศ. 2019 - 2021 แต่มีแนวโน้มคาดการณ์สูงในปี ค.ศ. 2022 อาจจำเป็นต้องปรับค่า Alpha หรือใช้โมเดลที่มีความซับซ้อนมากขึ้นเพื่อจับแนวโน้มของยอดขายที่เพิ่มขึ้นในปีล่าสุด

การพยากรณ์โดยวิธีซิมเพล็กซ์โพเนนเชียล มีแนวโน้มดีในการจับการเปลี่ยนแปลงของยอดขายในช่วงปี ค.ศ. 2019 - 2021 แต่มีข้อผิดพลาดในการพยากรณ์สูงในปี ค.ศ. 2022 การปรับค่า Alpha หรือพิจารณาการใช้โมเดลที่ซับซ้อนขึ้นอาจเสริมความแม่นยำของการพยากรณ์

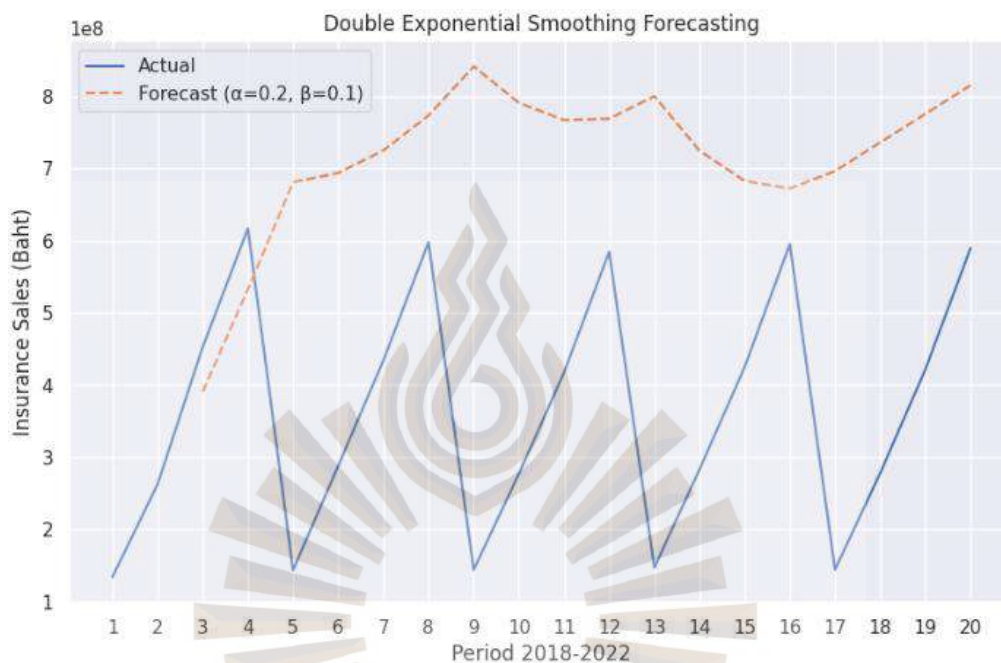
4.4.4 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า

ตารางที่ 4.4 แสดงผลการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ในแต่ละไตรมาสระหว่างปี ค.ศ. 2018 - 2022 โดยใช้วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าเพื่อตรวจสอบแนวโน้มและความแม่นยำของการพยากรณ์

รูปที่ 4.33 แสดงแผนภูมิเส้นที่เปรียบเทียบค่าจริงและค่าพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ โดยใช้วิธีการปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า (Double Exponential Smoothing) พร้อมค่าพารามิเตอร์ $\alpha = 0.2$ และ $\beta = 0.1$ ในช่วงระหว่างปี ค.ศ. 2018 - 2022

การพยากรณ์ด้วยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าแสดงให้เห็นถึงความสามารถในการจับแนวโน้มของยอดขายที่เพิ่มขึ้นได้อย่างมีประสิทธิภาพ โดยเฉพาะในช่วงปี ค.ศ. 2019 - 2021 และยังคงแสดงแนวโน้มที่ดีต่อเนื่องในปี ค.ศ. 2022 แม้ว่าค่าพยากรณ์อาจสูงเกินค่าจริงเล็กน้อยในปีดังกล่าว การใช้โมเดลนี้จึงเป็นเครื่องมือที่มีประสิทธิภาพในการพยากรณ์ยอดขายที่มีแนวโน้มเพิ่มขึ้น และสามารถสนับสนุนการวางแผนและการตัดสินใจในอนาคตได้อย่างเหมาะสม

ซึ่งจากรูปที่ 4.33 แสดงแผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ระหว่างปี ค.ศ. 2018 - 2022 โดยค่าพยากรณ์มีลักษณะเคลื่อนไหวสอดคล้องกับค่าจริงในช่วงที่ยอดขายมีแนวโน้มชัดเจน มีรายละเอียดดังนี้



รูปที่ 4.33 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า ระหว่างปี ค.ศ. 2018 - 2022

4.4.4.1 การพยากรณ์และยอดขาย

- 1) ปี ค.ศ. 2018 การพยากรณ์เริ่มต้นในปีนี้ไม่สามารถคำนวณได้สำหรับไตรมาสแรกและไตรมาสที่สอง เพราะข้อมูลยังไม่เพียงพอ การพยากรณ์เริ่มต้นในไตรมาสที่ 3 โดยคาดการณ์ต่ำกว่าค่าจริงในปี ค.ศ. 2018 และมีแนวโน้มคาดการณ์ที่สูงขึ้นในไตรมาสที่ 4
- 2) ปี ค.ศ. 2019 - 2022 การพยากรณ์ในปีนี้แสดงให้เห็นถึงการเพิ่มขึ้นอย่างต่อเนื่องของยอดขาย และมีการพยากรณ์ที่ใกล้เคียงกับแนวโน้มยอดขายจริง
- 3) ปี ค.ศ. 2022 การพยากรณ์มีการพยากรณ์ที่สูงขึ้นในแต่ละไตรมาส โดยมีการพยากรณ์ที่แม่นยำและสอดคล้องกับการเพิ่มขึ้นของยอดขายจริง

4.4.4.2 ความแม่นยำของการพยากรณ์

- 1) ปี ค.ศ. 2018 การพยากรณ์ในปีนี้ยังไม่แม่นยำมาก แต่มีแนวโน้มปรับตัวให้ดีขึ้นในไตรมาสที่ 4

2) ปี ค.ศ. 2019 - 2021 การพยากรณ์ด้วย Double Exponential Smoothing สามารถจับแนวโน้มการเพิ่มขึ้นของยอดขายได้ดี โดยเฉพาะในปี ค.ศ. 2020 และ ค.ศ. 2021 ซึ่งการพยากรณ์ใกล้เคียงกับยอดขายจริง

3) ปี ค.ศ. 2022 การพยากรณ์มีแนวโน้มที่ดีในการจับการเพิ่มขึ้นของยอดขายในแต่ละไตรมาส แต่มีแนวโน้มคาดการณ์ที่สูงเกินไปเล็กน้อย

4.4.4.3 การวิเคราะห์เชิงลึก

1) การปรับปรุงการพยากรณ์ วิธีปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าเหมาะสมสำหรับข้อมูลที่มีแนวโน้มเพิ่มขึ้นหรือ ลดลงอย่างต่อเนื่อง โดยการใช้สองขั้นตอนการเรียบเรียง (Level และ Trend) สามารถจับแนวโน้มได้ดี

2) การพยากรณ์ที่สูงเกินไป การพยากรณ์ในปี ค.ศ. 2022 ดูเหมือนว่าจะคาดการณ์สูงกว่ายอดขายจริงเล็กน้อย อาจเป็นผลมาจากการที่โมเดลต้องการเวลาในการปรับตัวให้เข้ากับการเปลี่ยนแปลงในระดับยอดขาย

4.4.4.4 เส้นสีน้ำเงิน (Actual) แสดงค่าจริงของข้อมูลในช่วงปี ค.ศ. 2018 - 2022 โดยมีรูปแบบการแกว่งตัวตามฤดูกาลที่ชัดเจน โดยค่าจริงมีความผันผวนสูงและมีการแกว่งขึ้นลงในลักษณะที่เป็นวัฏจักรอย่างต่อเนื่อง

4.4.4.5 เส้นประสีส้ม (Forecast) แสดงผลการพยากรณ์ที่มีลักษณะเป็นแนวโน้มเพิ่มขึ้นแบบเส้นตรง (Linear) อย่างต่อเนื่องตลอดช่วงเวลา โดยไม่สามารถสะท้อนรูปแบบการแกว่งตัวของข้อมูลจริงได้อย่างชัดเจน ซึ่งสะท้อนถึงข้อจำกัดของโมเดลในการติดตามความผันผวนของข้อมูล

4.4.4.6 การตั้งค่าพารามิเตอร์

1) $\alpha = 0.0$ หมายความว่าโมเดลไม่ได้ให้ความสำคัญกับข้อมูลใหม่มากนัก ซึ่งทำให้การพยากรณ์ไม่สามารถปรับตัวตามการเปลี่ยนแปลงของข้อมูลจริงได้ เนื่องจากโมเดลจะอ้างอิงเฉพาะข้อมูลในอดีตเป็นหลัก โดยไม่ตอบสนองต่อแนวโน้มล่าสุด

2) $\beta = 0.0$ ไม่เน้นการจับแนวโน้มระยะยาว ทำให้การพยากรณ์เป็นเพียงเส้นตรงโดยไม่มีการเพิ่มหรือลดลงอย่างชัดเจน ส่งผลให้โมเดลไม่สามารถสะท้อนการเปลี่ยนแปลงของทิศทางข้อมูลเมื่อเวลาผ่านไป โดยเฉพาะในชุดข้อมูลที่มีแนวโน้มชัดเจน

4.4.4.7 ข้อสังเกต

- 1) ผลการพยากรณ์ไม่สามารถจับรูปแบบฤดูกาลหรือความผันผวนของข้อมูลจริงได้ เนื่องจากพารามิเตอร์ที่ตั้งไว้ไม่ให้ความสำคัญกับการปรับตัวตามข้อมูลใหม่และแนวโน้ม ซึ่งอาจไม่เหมาะสมสำหรับข้อมูลที่มีความผันผวนตามฤดูกาลอย่างในแผนภูมิเส้นนี้
- 2) เส้นพยากรณ์มีแนวโน้มเพิ่มขึ้นอย่างช้า ๆ แต่ไม่ตอบสนองต่อการเปลี่ยนแปลงของข้อมูลจริง

การใช้ปรับเรียบเอ็กซ์โพเนนเชียลสองเท่ากับพารามิเตอร์ $\alpha = 0.2$ และ $\beta = 0.1$ ไม่เหมาะสมสำหรับข้อมูลที่มีความผันผวนตามฤดูกาลอย่างในแผนภูมิเส้นนี้ เนื่องจากโมเดลไม่สามารถจับการแกว่งตัวหรือแนวโน้มได้อย่างแม่นยำ แม้ค่าพารามิเตอร์จะถูกตั้งให้มีความไวต่อข้อมูลใหม่ แต่ยังไม่สามารถสะท้อนรูปแบบที่เปลี่ยนแปลงช้า ๆ ตามฤดูกาลได้อย่างมีประสิทธิภาพ ดังนั้น การเลือกใช้โมเดลที่สามารถจัดการกับองค์ประกอบเชิงฤดูกาล เช่น Holt-Winters จึงอาจเหมาะสมกว่าในกรณีนี้ เพื่อเพิ่มความแม่นยำในการพยากรณ์

4.4.5 ผลลัพธ์การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์ด้วยวิธีต่าง ๆ

เมื่อนำผลทั้ง 4 วิธี มารวมกัน จะได้ดังรูปที่ 4.34 ที่เป็นแผนภูมิเส้นที่แสดงการพยากรณ์วิธีต่าง ๆ เปรียบเทียบกับข้อมูลยอดขายจริงยอดขายประกันชีวิตรวมทุกกรมธรรม์ในช่วงปี ค.ศ. 2018 - 2022 มีรายละเอียดดังนี้

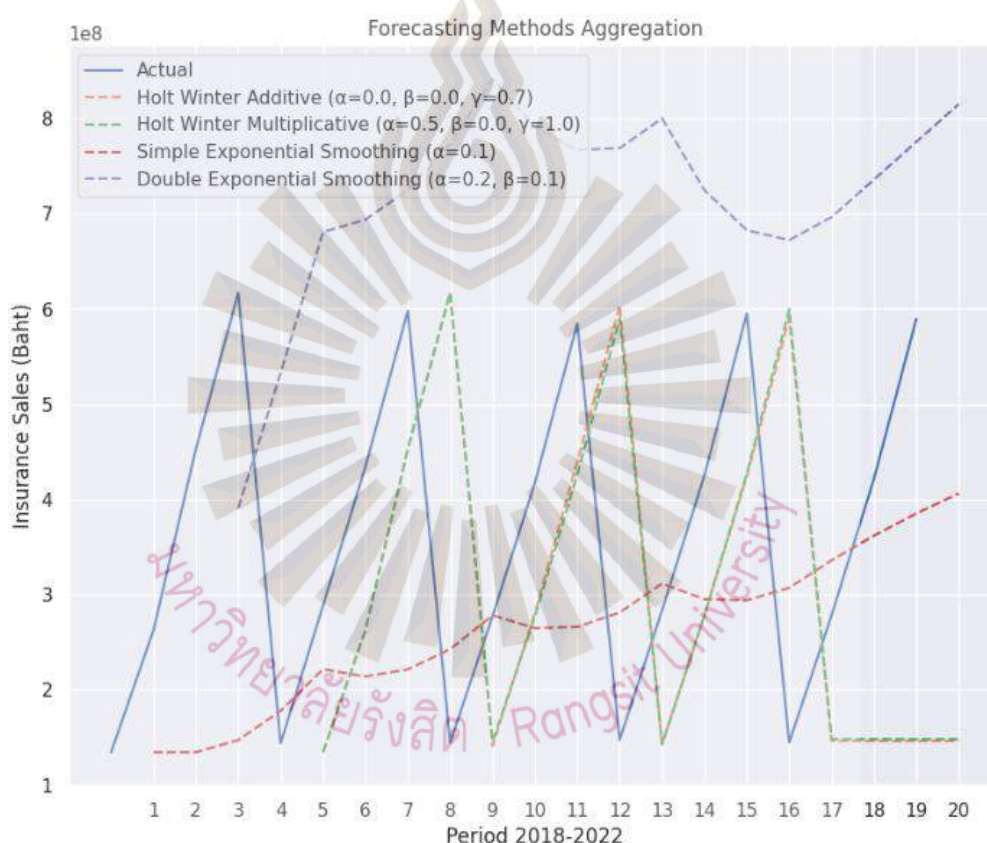
4.4.5.1 ช่วงที่ยอดขายสูงมาก ช่วงที่เห็นยอดขายสูงชัดเจนคือในปี ค.ศ. 2020 - 2021 ซึ่งตรงกับช่วงที่ COVID-19 ระบาดหนัก ยอดขายการประกันชีวิตน่าจะพุ่งสูงขึ้นในช่วงนี้ เพราะผู้บริโภคมีความกังวลเกี่ยวกับสุขภาพและอนาคตของตัวเองและครอบครัว

4.4.5.2 ช่วงที่ยอดขายต่ำ ช่วงที่ยอดขายลดลงอาจเกิดจากสถานการณ์ที่ COVID-19 เริ่มคลี่คลายลง ซึ่งส่งผลให้ผู้บริโภคอาจไม่รู้สึกถึงความจำเป็นในการซื้อประกันเพิ่มขึ้นเหมือนช่วงที่ COVID-19 ระบาดหนัก เนื่องจากความกังวลเกี่ยวกับความเสี่ยงทางสุขภาพลดลงในช่วงเวลานั้น ส่งผลให้ความจำเป็นในการซื้อผลิตภัณฑ์ประกันลดลงตามลำดับ

4.4.5.3 ความกังวลเรื่องสุขภาพ การระบาดของ COVID-19 ทำให้ผู้คนมีความตระหนักและกังวลเกี่ยวกับสุขภาพมากขึ้น ส่งผลให้ยอดขายประกันชีวิตและสัญญาเพิ่มเติมเพิ่มขึ้น

4.4.5.4 ความไม่แน่นอนทางเศรษฐกิจ COVID - 19 ทำให้ผู้คนกังวลเกี่ยวกับความมั่นคงทางการเงินในอนาคต จึงหันมาซื้อประกันชีวิตเพื่อเป็นการป้องกันความเสี่ยง

4.4.5.5 การกลับสู่สภาวะปกติ เมื่อการระบาดคลี่คลายลง ความต้องการในการซื้อประกันก็ลดลงตามไปด้วย เนื่องจากผู้บริโภครู้สึกปลอดภัยมากขึ้น และไม่เห็นความจำเป็นเร่งด่วนเหมือนในช่วงวิกฤต ส่งผลให้พฤติกรรมการใช้จ่ายกลับมาใกล้เคียงกับช่วงก่อนเกิดการระบาด โดยเฉพาะในกลุ่มที่มีรายได้น้อยซึ่งให้ความสำคัญกับค่าใช้จ่ายจำเป็นก่อน



รูปที่ 4.34 แผนภูมิเส้นแสดงการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์โดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022

ในช่วงปี ค.ศ. 2018 - 2022 ยอดขายการประกันชีวิตสัญญาเพิ่มเติมมีการเปลี่ยนแปลงอย่างชัดเจน โดยยอดขายพุ่งสูงในช่วงปี ค.ศ. 2020 - 2021 ซึ่งเป็นช่วงที่ COVID - 19 ระบาดหนัก สาเหตุที่ยอดขายเพิ่มขึ้นอย่างมากเป็นเพราะผู้บริโภคมีความกังวลเกี่ยวกับสุขภาพและอนาคตทางการเงินของตนเอง จึงหันมาซื้อประกันมากขึ้น อย่างไรก็ตาม หลังจากสถานการณ์ COVID - 19

คลี่คลายลง ยอดขายก็มีแนวโน้มลดลงเนื่องจากผู้คนรู้สึกปลอดภัยและไม่จำเป็นต้องซื้อประกันเพิ่มขึ้นเหมือนช่วงที่โรคระบาดหนัก

วิธีที่เหมาะสมที่สุด คือ Multiplicative ซึ่งเหมาะสมกว่าสำหรับข้อมูลที่มีฤดูกาล เนื่องจากสามารถปรับรูปแบบให้เข้าใกล้ค่าจริงได้ดีกว่า

วิธีที่ไม่เหมาะสม คือ Simple Exponential Smoothing อาจเหมาะกับข้อมูลที่ไม่มีฤดูกาล แต่สำหรับกราฟนี้ ข้อมูลมีการเปลี่ยนแปลงตามฤดูกาลชัดเจน

การเปรียบเทียบ Holt Winter Additive และ Multiplicative ทำให้ Holt Winter Multiplicative เหมาะสำหรับข้อมูลที่มีการเปลี่ยนแปลงตามสัดส่วน (Proportional Seasonality) เนื่องจากจัดการกับการเปลี่ยนแปลงที่มีอัตราส่วนสม่ำเสมอระหว่างฤดูกาลและแนวโน้มที่แตกต่างกันในแต่ละช่วงเวลาได้ดีกว่า ในขณะที่ Holt Winter Additive เหมาะกับข้อมูลที่มีฤดูกาลคงที่

4.4.6 ผลลัพธ์การวัดข้อผิดพลาดการพยากรณ์รวมทุกกรรมธรรม์

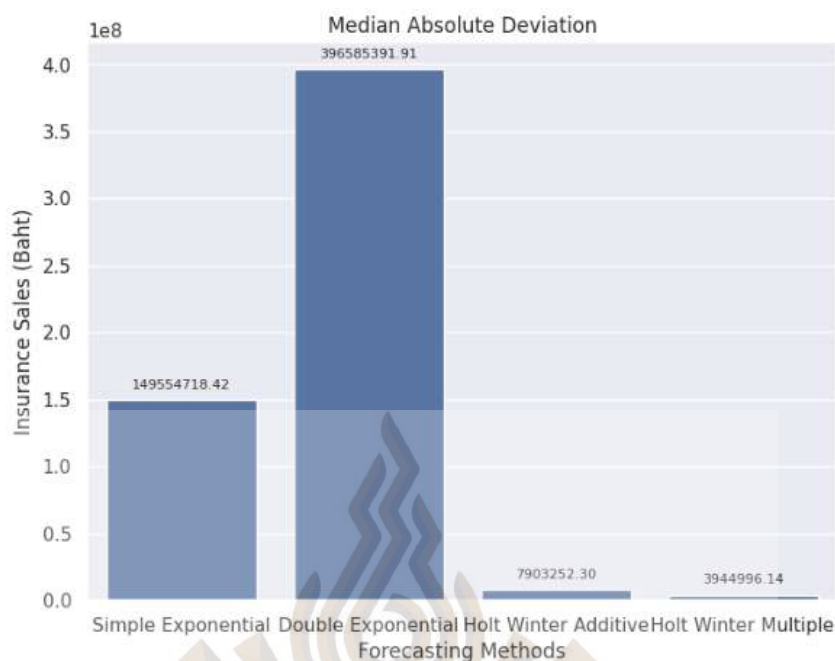
ในการประเมินประสิทธิภาพของโมเดลพยากรณ์ที่ใช้กับข้อมูลรวมของกรรมธรรม์ ได้มีการวัดค่าความผิดพลาดของการพยากรณ์ด้วยตัวชี้วัดมาตรฐาน ได้แก่ MAD (Mean Absolute Deviation) MAPE (Mean Absolute Percentage Error) และ MSE (Mean Squared Error) เพื่อวิเคราะห์ความแม่นยำของโมเดล ดังนี้

4.4.6.1 ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD)

แผนภูมิแท่งรูปที่ 4.35 แสดงค่าเบี่ยงเบนสัมบูรณ์มัธยฐาน (Median Absolute Deviation) สำหรับวิธีการพยากรณ์ ดังนี้

1) Simple Exponential ค่า MAD 149,554,718.42 (ประมาณ 149.55 ล้าน) จำนวนที่ระบุในกราฟเป็น 42% ความคลาดเคลื่อนของวิธีนี้รองลงมาจาก Double Exponential Smoothing แต่ไม่เหมาะสมสำหรับข้อมูลที่มีฤดูกาลชัดเจน

2) Double Exponential ค่า MAD 396,585,391.91 (ประมาณ 396.59 ล้าน) จำนวนที่ระบุในกราฟเป็น 91% ของ MAD รวม วิธีนี้มีค่าความคลาดเคลื่อนสูงสุดในกราฟ สะท้อนว่าวิธีนี้ไม่มีประสิทธิภาพ เนื่องจากไม่สามารถจับรูปแบบข้อมูลที่มีฤดูกาลหรือแนวโน้มซับซ้อนได้



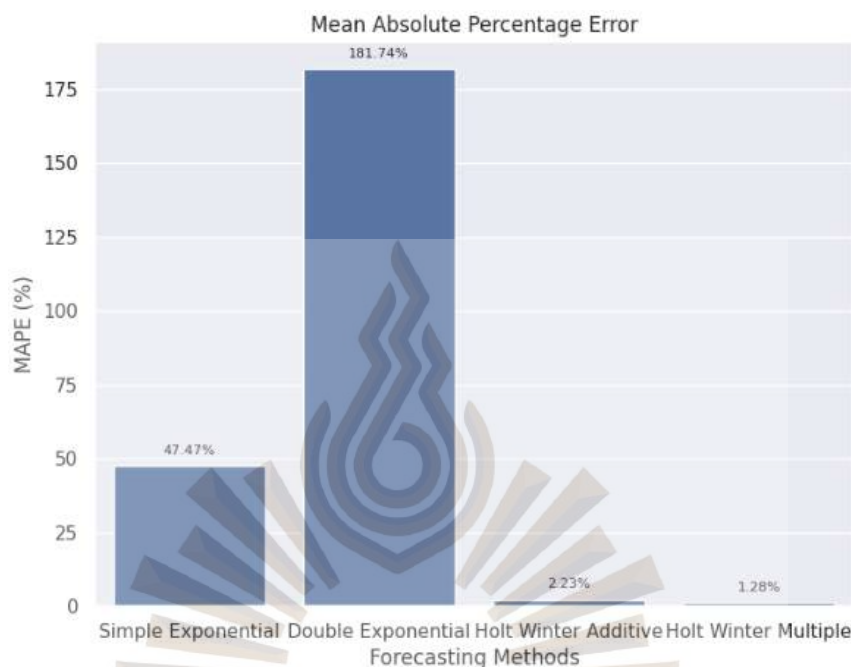
รูปที่ 4.35 แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) สำหรับยอดขายประกันชีวิตรวมทุกกรมธรรม์

3) Holt Winter Additive มีค่า MAD เท่ากับ 7,903,252.30 (ประมาณ 7.90 ล้านบาท) และมีค่าความคลาดเคลื่อนตามกราฟที่ 2.30% ทำให้ปรับปรุงความแม่นยำในการพยากรณ์ โดยการลดความคลาดเคลื่อนอย่างมีนัยสำคัญ เมื่อเปรียบเทียบกับ Exponential Smoothing ทั้งสองแบบ และเหมาะสมกับข้อมูลที่มีลักษณะฤดูกาลแบบ Additive

4) Holt Winter Multiplicative ค่า MAD 3,944,996.14 (ประมาณ 3.94 ล้านบาท) จำนวนที่ระบุในกราฟ 1.14% มีความคลาดเคลื่อนต่ำที่สุดในกราฟ คิดเป็นเพียง 1.14% ของ MAD รวม แสดงให้เห็นว่าเป็นวิธีที่เหมาะสมที่สุดกับข้อมูล โดยสามารถจับรูปแบบฤดูกาลแบบ Multiplicative ได้อย่างมีประสิทธิภาพ ค่า MAD ที่ต่ำสุดในกราฟบ่งชี้ว่า Holt Winter Multiplicative สามารถคำนวณพยากรณ์ได้ใกล้เคียงกับข้อมูลจริงที่สุด

Double Exponential Smoothing (91%) มีค่าความคลาดเคลื่อนมากที่สุด และไม่เหมาะสมสำหรับการพยากรณ์ข้อมูลชุดนี้ Simple Exponential Smoothing (42%) มีความคลาดเคลื่อนน้อยกว่า แต่ยังคงสูง Holt Winter Additive (2.30%) และ Multiplicative (1.14%) เป็นวิธีที่เหมาะสมสำหรับข้อมูลที่มีลักษณะฤดูกาล โดยเฉพาะ Multiplicative ซึ่งมีความแม่นยำที่สุด ในกรณีนี้ ควรเลือกใช้ Holt Winter Multiplicative สำหรับการพยากรณ์ข้อมูลที่มีรูปแบบฤดูกาลและแนวโน้มร่วมกัน

4.4.6.2 ค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE)



รูปที่ 4.36 แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) สำหรับยอดขายประกันชีวิตรวมทุกกรมธรรม์

แผนภูมิแท่งรูปที่ 4.36 แสดงค่าความผิดพลาดเฉลี่ยเปอร์เซ็นต์สัมบูรณ์ (Mean Absolute Percentage Error) สำหรับวิธีการพยากรณ์ต่าง ๆ โดยค่าที่แสดงในแผนภูมิดังนี้

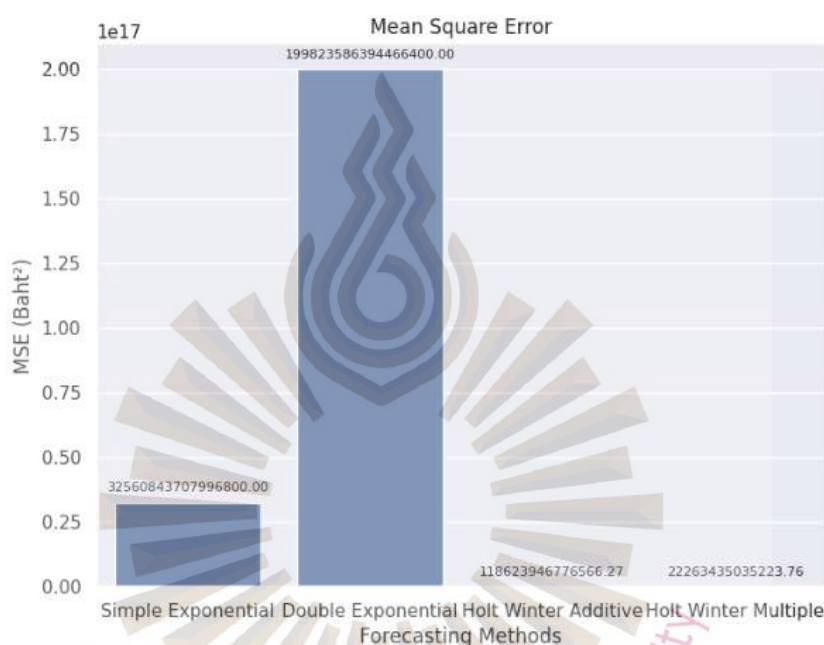
1) ประสิทธิภาพของโมเดล Holt Winter Multiplicative มีค่าความผิดพลาด (MAPE) ต่ำที่สุด (1.28%) ซึ่งแสดงถึงความแม่นยำในการพยากรณ์ที่ดีที่สุด ในกราฟนี้ Double Exponential มีค่า MAPE สูงที่สุด (181.74%) บ่งชี้ว่าโมเดลนี้ไม่เหมาะสมสำหรับข้อมูลนี้

2) เปรียบเทียบระหว่าง Additive กับ Multiplicative พบว่า Holt-Winter Additive (2.23%) มีประสิทธิภาพที่ดีในการพยากรณ์ข้อมูลที่มีฤดูกาลคงที่ แต่ยังด้อยกว่า Multiplicative เล็กน้อย ซึ่งสามารถจัดการกับข้อมูลที่มีความผันผวนตามฤดูกาลในลักษณะสัดส่วนได้ดีกว่า

3) Exponential Smoothing โมเดลพื้นฐาน ซึ่ง Simple Exponential (47.47%) มีประสิทธิภาพดีกว่า Double Exponential (181.74%) แต่ยังคงมีความผิดพลาดสูงกว่า Holt-Winter เนื่องจากไม่สามารถจับแนวโน้มและฤดูกาลของข้อมูลได้อย่างแม่นยำเหมือนในโมเดลที่ซับซ้อนกว่า

4) Holt-Winter Multiplicative เป็นโมเดลที่เหมาะสมที่สุดสำหรับการพยากรณ์ในกรณีนี้ เนื่องจากค่าความผิดพลาด MAPE ต่ำสุด และ Double Exponential ควรถูกพิจารณาว่าไม่เหมาะสมสำหรับชุดข้อมูลนี้

4.4.6.3 ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE)



รูปที่ 4.37 แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) สำหรับยอดขายประกันชีวิตรวมทุกกรมธรรม์

จากแผนภูมิแท่งรูปที่ 4.37 แสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error: MSE) ของโมเดลพยากรณ์เวลา (Time Series Forecasting) 4 โมเดล ดังนี้

1) ประสิทธิภาพของโมเดล Holt Winter Multiplicative มีค่า MSE ต่ำที่สุด ซึ่งบ่งชี้ว่าโมเดลนี้เหมาะสมที่สุดสำหรับการพยากรณ์ ขณะที่โมเดล Double Exponential มีค่า MSE สูงผิดปกติ (1.99×10^{17}) แสดงว่าโมเดลไม่สามารถจับแนวโน้มและฤดูกาลของข้อมูลได้อย่างแม่นยำ จึงไม่เหมาะสมกับข้อมูลนี้

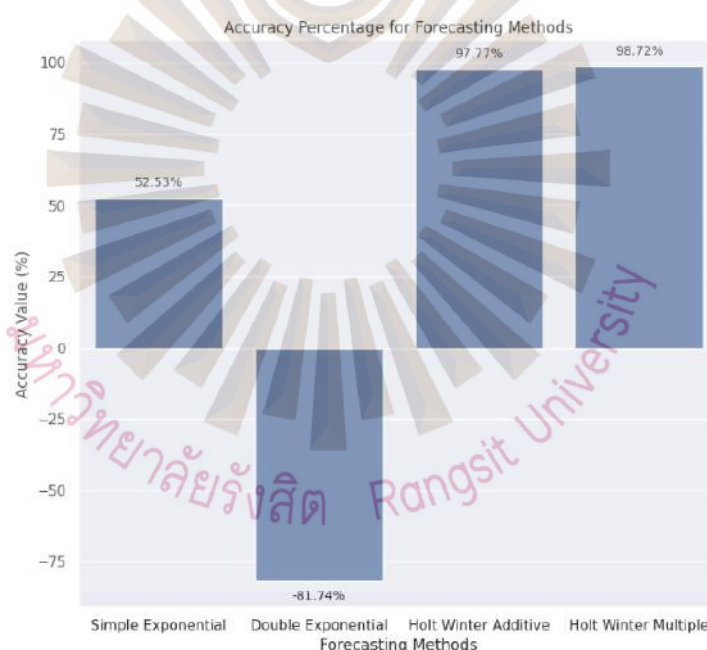
2) เปรียบเทียบ Holt Winter Additive กับ Holt Winter Multiplicative ซึ่ง Holt Winter Additive มี MSE ต่ำ แต่ยังสูงกว่า Multiplicative อยู่ประมาณ 5 เท่า Holt Winter Multiplicative แสดงให้เห็นถึงประสิทธิภาพที่ดีที่สุด ด้วยค่าความผิดพลาด MSE ที่ต่ำที่สุดในการพยากรณ์ ขณะที่ Double Exponential ควรถูกตัดจากการพิจารณา เนื่องจากมีค่าความผิดพลาดสูง

ถึงขั้นที่ไม่สามารถใช้งานได้ สำหรับโมเดล Holt Winter Additive และ Simple Exponential อาจเหมาะสมในบางกรณีหากต้องการทางเลือกที่ซับซ้อนน้อยกว่า

3) Exponential Smoothing โมเดลพื้นฐาน โดย Simple Exponential มีค่า MSE ต่ำกว่า Double Exponential มาก แต่ยังสูงกว่า Holt-Winter ทั้งแบบ Additive และ Multiplicative เนื่องจาก Simple Exponential ไม่สามารถจับแนวโน้มหรือฤดูกาลของข้อมูลได้ดีเท่ากับ Holt - Winter ซึ่งสามารถปรับตามแนวโน้มและฤดูกาลได้อย่างมีประสิทธิภาพมากกว่า

4.4.7 ผลลัพธ์ความแม่นยำของการพยากรณ์รวมทุกกรรมธรรม์

แผนภูมิแท่งรูปที่ 4.38 แผนภูมิแท่งนี้แสดงค่าความแม่นยำของวิธีการพยากรณ์ 4 วิธีที่แตกต่างกันดังนี้ ซึ่งสะท้อนถึงประสิทธิภาพในการพยากรณ์ของแต่ละวิธีการและแสดงความสามารถในการจับรูปแบบของข้อมูลที่แตกต่างกันในแต่ละกรณี ดังนี้



รูปที่ 4.38 แผนภูมิแท่งแสดงผลลัพธ์ความแม่นยำของการพยากรณ์รวมทุกกรรมธรรม์

1) ประสิทธิภาพของโมเดล

ประสิทธิภาพของโมเดล Holt Winter Multiplicative มีความแม่นยำสูงสุดที่ 98.72% ซึ่งหมายถึงเป็นโมเดลที่เหมาะสมที่สุด ส่วน Holt Winter Additive มีความแม่นยำรองลงมาที่ 97.77% แสดงให้เห็นว่าโมเดล Additive ให้ผลลัพธ์ใกล้เคียงกับ Multiplicative ได้ดี โดยทั้งสองโมเดลพยากรณ์ข้อมูลที่มีฤดูกาลและแนวโน้มได้อย่างแม่นยำและมีประสิทธิภาพสูง

2) ประสิทธิภาพปานกลาง

Simple Exponential (52.53%) มีความแม่นยำปานกลาง เหมาะสำหรับการพยากรณ์ที่ไม่ซับซ้อน (ไม่มีแนวโน้มหรือฤดูกาลที่ชัดเจน) โดยสามารถนำไปใช้กับข้อมูลที่มีความผันผวนต่ำและมีรูปแบบสม่ำเสมอในช่วงเวลาสั้น ๆ ได้อย่างเหมาะสม

3) ประสิทธิภาพต่ำ

Double Exponential มีความแม่นยำต่ำมาก โดยค่าติดลบ (-81.74%) แสดงว่าโมเดลนี้ไม่เหมาะสมกับข้อมูลนี้อย่างสิ้นเชิง

Holt Winter Multiplicative เหมาะที่สุดสำหรับการพยากรณ์ในกรณีนี้ และ Double Exponential ควรถูกตัดออกจากการใช้งาน เนื่องจากค่าความแม่นยำติดลบ บ่งบอกว่าการพยากรณ์ผิดพลาดมากเกินไป ในกรณีที่ไม่มีแนวโน้มหรือฤดูกาล การใช้ Simple Exponential อาจเป็นทางเลือกที่สะดวกกว่า แต่ความแม่นยำจะลดลงเมื่อเทียบกับ Holt Winter

ควรตรวจสอบลักษณะข้อมูล (เช่น แนวโน้ม และฤดูกาล) เพื่อยืนยันว่าการเลือก Multiplicative หรือ Additive เหมาะสมที่สุด หรือปรับค่าพารามิเตอร์เพิ่มเติมเพื่อปรับปรุงประสิทธิภาพของโมเดล Simple และ Double Exponential การเลือกวิธีพยากรณ์ที่เหมาะสมกับลักษณะข้อมูลสามารถเพิ่มความแม่นยำในการพยากรณ์และลดข้อผิดพลาดที่อาจเกิดขึ้น โดยวิธีพยากรณ์ที่เหมาะสมจะทำให้โมเดลสามารถจับแนวโน้มและฤดูกาลได้ดีขึ้น เพิ่มความแม่นยำในการพยากรณ์

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

วิทยานิพนธ์นี้นำเสนอการพยากรณ์ยอดขายประกันชีวิตโดยใช้วิธีการ 4 วิธี ได้แก่ การบวกโสมทวินเทอร์ การคูณโสมทวินเทอร์ ปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย และปรับเรียบเอ็กซ์โพเนนเชียลสองเท่า จากนั้นได้ใช้วิธีคำนวณค่าเฉลี่ย 3 วิธีในการประเมินค่าความคลาดเคลื่อนสำหรับการพยากรณ์ ได้แก่ ค่าความผิดพลาดสัมบูรณ์เฉลี่ย ค่าความผิดพลาดเปอร์เซ็นต์สัมบูรณ์เฉลี่ย และค่าความผิดพลาดกำลังสองเฉลี่ย แล้วทำการวัดค่าความแม่นยำของการพยากรณ์แต่ละวิธีเพื่อเลือกวิธีการพยากรณ์ที่แม่นยำและเหมาะสมที่สุด

ผลการวิจัยพบว่า การพยากรณ์ยอดขายประกันชีวิตจากการประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป การใช้การคูณโสมทวินเทอร์ มีความแม่นยำสูงสุดที่ 98.43% การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก วิธีการคูณโสมทวินเทอร์มีความแม่นยำสูงสุดถึง 86.12% การพยากรณ์ยอดขายประกันชีวิตสัญญาเพิ่มเติมวิธีที่แม่นยำที่สุดคือการบวกโสมทวินเทอร์ มีเปอร์เซ็นต์ความแม่นยำสูงสุดถึง 94.61% และการพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์วิธีการคูณโสมทวินเทอร์มีผลผิดพลาดที่พหุ มีความแม่นยำสูงสุดที่ 98.72% ซึ่งสูงกว่าวิธีการอื่น ๆ อย่างชัดเจน และแสดงให้เห็นถึงประสิทธิภาพการพยากรณ์ที่มีประสิทธิภาพสูง

การระบาดของโควิด-19 ทำให้ผู้บริโภคมีความต้องการในผลิตภัณฑ์ประกันชีวิตมากขึ้น เนื่องจากมีความกังวลเกี่ยวกับสุขภาพและความมั่นคงทางการเงิน ยอดขายของผลิตภัณฑ์ประกันชีวิตแบบทั่วไปและกรมธรรม์หลักจึงมีแนวโน้มเพิ่มขึ้นในช่วงที่เกิดการระบาด การซื้อประกันชีวิตยังสามารถใช้เป็นส่วนหนึ่งในการ ลดหย่อนภาษี ได้ ซึ่งทำให้ผู้บริโภคมีแรงจูงใจในการซื้อเพิ่มขึ้นในช่วงปลายปี การระบาดของโควิด-19 ทำให้ยอดขายประกันชีวิตเพิ่มขึ้นเนื่องจากความกังวลเรื่องสุขภาพและอนาคตทางการเงินของผู้บริโภค ประกันชีวิตยังสามารถใช้ลดหย่อนภาษีได้ จึงเป็นแรงจูงใจที่สำคัญในการซื้อประกัน

5.2 ข้อเสนอแนะ

5.2.1 ควรใช้วิธี Holt Winter Multiplicative ในการพยากรณ์ยอดขายประกันชีวิตเนื่องจากมีความแม่นยำสูงสุด

5.2.2 การพยากรณ์ยอดขายประกันชีวิตจากข้อมูลเป็นระยะเวลา 5 ปี สามารถทำการพยากรณ์ยอดขายในปีที่ 5 รายไตรมาสได้ในอีกรูปแบบ

5.2.3 พิจารณาปัจจัยภายนอก เช่น การระบาดของโรค และมาตรการทางเศรษฐกิจในการวางแผนการตลาดและกลยุทธ์การขายประกันชีวิตในอนาคต

5.2.4 พิจารณาการใช้โมเดลการพยากรณ์ที่ผสมผสานหลายวิธี (Hybrid Models) การพยากรณ์โดยใช้โมเดลเดียวอาจมีความแม่นยำไม่เพียงพอในสถานการณ์ที่มีความผันผวนสูง เช่น ในช่วงการระบาดของโควิด-19 การใช้วิธีการผสมผสานหลายวิธี (Hybrid Models) ที่สามารถประยุกต์ใช้จุดเด่นของแต่ละวิธีร่วมกัน อาจส่งเสริมให้การพยากรณ์ยอดขายประกันชีวิตมีความแม่นยำสูงขึ้น เช่น การผสมผสานระหว่าง Holt Winter Multiplicative กับวิธีการอื่น ๆ ที่สามารถคำนึงถึงปัจจัยความไม่แน่นอนได้ดีขึ้น

5.2.5 เพิ่มการศึกษาปัจจัยที่มีผลกระทบต่อพฤติกรรมผู้บริโภคในระยะยาว ปัจจัยที่ส่งผลกระทบต่อพฤติกรรมผู้บริโภคซื้อประกันชีวิตของผู้บริโภคไม่ได้จำกัดเฉพาะสถานการณ์วิกฤติเท่านั้น ควรศึกษาเพิ่มเติมเกี่ยวกับปัจจัยอื่น ๆ ที่อาจส่งผลต่อการซื้อประกันชีวิต เช่น การเปลี่ยนแปลงในกฎหมายภาษี การเปลี่ยนแปลงในระบบสาธารณสุข การรณรงค์ของรัฐบาลหรือหน่วยงานทางการเงินต่าง ๆ เพื่อให้สามารถวางแผนกลยุทธ์การตลาดที่ยั่งยืน

5.2.6 การพัฒนาโปรแกรมการตลาดที่ยืดหยุ่นและเน้นความต้องการผู้บริโภคในช่วงเวลาที่แตกต่างกัน จากการศึกษาพบว่ายอดขายประกันชีวิตมีการเปลี่ยนแปลงตามสถานการณ์ทางเศรษฐกิจและสังคม ดังนั้น การพัฒนาโปรแกรมการตลาดและโปรโมชั่นที่สามารถปรับตัวตามความต้องการและความเชื่อมั่นของผู้บริโภคในช่วงเวลาที่แตกต่างกัน เช่น โปรโมชั่นพิเศษในช่วงปลายปีที่เน้นลดหย่อนภาษีหรือแผนประกันที่เน้นการคุ้มครองสุขภาพในช่วงที่มีความเสี่ยงสูง จะเป็นปัจจัยที่สนับสนุนการเพิ่มยอดขายได้อย่างมีประสิทธิภาพมากขึ้น

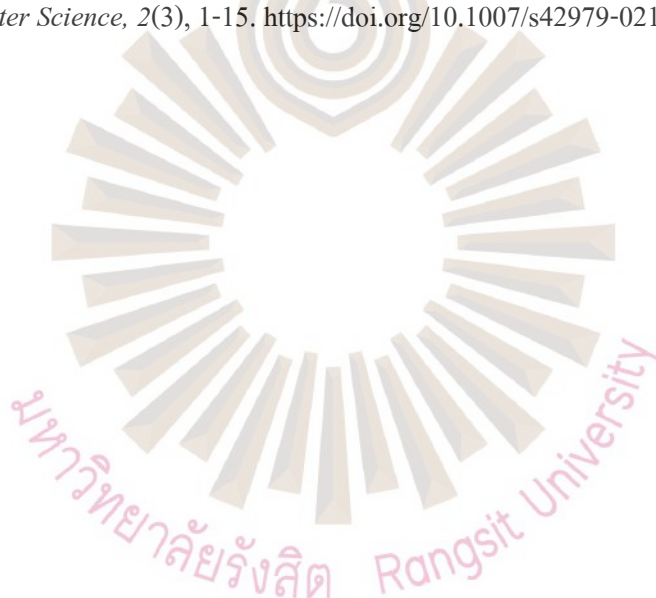
การเสนอแนะเหล่านี้มีส่วนในการยกระดับความแม่นยำในการพยากรณ์ ส่งผลให้เพิ่มยอดขายผลิตภัณฑ์ประกันชีวิตให้สอดคล้องกับความต้องการของผู้บริโภคที่เปลี่ยนแปลงต่อเนื่อง นอกจากนี้ การเสนอแนะได้เสริมสร้างประสิทธิภาพในการวางแผนกลยุทธ์ในบริษัทธุรกิจ และสนับสนุนการปรับกลยุทธ์ให้เหมาะสมกับสภาพแวดล้อมทางธุรกิจที่เปลี่ยนแปลงอย่างเหมาะสม

บรรณานุกรม

- ธีระพงษ์ ทับพร, ยอดนภา เกษเมือง, เอกพล ทับพร, และภรดิษฐ์ แปงจิตต์. (2561). การพยากรณ์ยอดขายและการบริหารสินค้าคงคลังของสินค้าคงหมักยักษ์แช่แข็ง : บริษัท สยามแม็คโคร จำกัด มหาชน. *วารสารวิชาการมหาวิทยาลัยธนบุรี (วิทยาศาสตร์และเทคโนโลยี)*, 2(2), 28-41.
- เรวัฒน์ ไทยทอง. (2560). การพยากรณ์การผลิตเชื้อและทางแกงกรณีศึกษา บริษัท อามสการ์เมนต์ จำกัด. 1449-1458. สืบค้นจาก <https://grad.dpu.ac.th/upload/content/files/year8-1/8120.pdf>
- สำนักข่าวอีไฟแนนซ์ไทย. (2565). ส่องมาร์เก็ตแชร์ “ธุรกิจประกันชีวิต” ปี 65. สืบค้นจาก https://www.efinancethai.com/efinReview/efinReviewMain.aspx?name=er_2022062816
- 26
- สำนักงานคณะกรรมการกำกับและส่งเสริมการประกอบธุรกิจประกันภัย (คปภ.). (2565). สถิติธุรกิจประกันชีวิตรายเดือนของปี 2546 ถึง 2558 และไตรมาสปี 2559 ถึงปัจจุบัน. สืบค้นจาก <https://www.oic.or.th/th/industry/statistic/data/31/2>
- แอมะซอนเว็บเซอร์วิส. (2567). Python คืออะไร? สืบค้นจาก <https://aws.amazon.com/th/what-is/python>
- Google. (2023a). *Google Colaboratory*. Retrieved from <https://colab.research.google.com/>
- Google. (2023b). *Sign in - Google Accounts*. Retrieved from <https://accounts.google.com/>
- Nurhamidah, Wan, N., & Faisol, A. (2020). Forecasting Seasonal Time Series Data using The Holt-Winters Exponential Smoothing Method of Additive Models. *Journal Matematika Integratif*, 16(2), 151-157.
- Ostertagová, E., & Oskar, O. (2012). Forecasting Using Simple Exponential Smoothing Method. *Acta Electrotechnica et Informatica*, 12(3), 62-66.
- Pleños, M.C.F. (2022). Time Series Forecasting Using Holt-Winters Exponential Smoothing: Application to Abaca Fiber Data. *Problems of World Agriculture*, 22(2), 17-29. <https://doi.org/10.22630/PRS.2022.22.2.6>
- Rahman, H., Salma, U., Hossain, M., & Khan, T. F. (2016). Revenue Forecasting using Holt-Winters Exponential Smoothing. *STN Journals*, 5(3), 19-25.

บรรณานุกรม (ต่อ)

- Sidqi, F., & Sumitra, I. D. (2019). Forecasting Product Selling Using Single Exponential Smoothing and Double Exponential Smoothing Methods. *IOP Conference Series: Materials Science and Engineering*, 662(3), 1-6. <https://doi.org/10.1088/1757-899X/662/3/032031>
- Suwanvijit, W., Lumley, T., Choonpradub, C., & McNeil, N. (2011). Long-term sales forecasting using Lee-Carter and Holt-Winters methods. *Journal of Applied Business Research*, 27(1), 87-102. <https://doi.org/10.19030/jabr.v27i1.913>
- Swapnarekha, H., Behera, S. B., Nayak, J., Naik, B., & Kumar, P. S. (2021). Multiplicative Holts Winter Model for Trend Analysis and Forecasting of COVID-19 Spread in India. *SN Computer Science*, 2(3), 1-15. <https://doi.org/10.1007/s42979-021-00808-0>







```
# arrays and dataframes
```

```
import pandas as pd
```

```
import numpy as np
```

```
# visualization imports
```

```
from mpl_toolkits.mplot3d import Axes3D
```

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns
```

```
%matplotlib inline
```

```
Load data
```

```
# Load the data into a Pandas DataFrame
```

```
path = 'Sales_Primary-General.csv' #change data
```

```
path accordingly
```

```
data = pd.read_csv(path)
```

```
data.head()
```

	Year	quarter	period	Sales
0	2018	1	1	1.209574e+08
1	2018	2	2	2.371767e+08
2	2018	3	3	3.483160e+08
3	2018	4	4	4.736409e+08
4	2019	1	5	1.116968e+08

รูปภาคผนวก ก ที่ 1 ผลลัพธ์จากการเรียกใช้งาน

คำสั่ง data.head() ข้อมูล 5

แถวแรก

```
data.info(verbose=True)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Year        20 non-null    int64
1   quarter     20 non-null    int64
2   period      20 non-null    int64
3   Sales       20 non-null    float64
dtypes: float64(1), int64(3)
memory usage: 768.0 bytes
```

รูปภาคผนวก ก ที่ 2 ข้อมูลเกี่ยวกับ DataFrame

```
#Count number of null and NA values in each
```

```
column
```

```
data.isnull().sum().sort_values(ascending=False)
```

```
Year      0
quarter    0
period     0
Sales      0
dtype: int64
```

รูปภาคผนวก ก ที่ 3 ผลลัพธ์จากการตรวจสอบ

ค่าที่เป็น null (หรือค่า

missing) ใน DataFrame

```
Error Functions
```

```
def mape(y_test, pred):
```

```
    y_test, pred = np.array(y_test), np.array(pred)
```

```
    mape = np.mean(np.abs((y_test - pred) / y_test))
```

```
    return mape * 100
```

```
def mad(y_true, predictions):
```

```
    y_true, predictions = np.array(y_true),
```

```
    np.array(predictions)
```

```
    return np.mean(np.abs(y_true - predictions))
```

```
# MAE (mean absolute error) is also scale-
```

```
dependent and so cannot be used for comparisons
```

```
across series of different units.
```

```
# MAD (mean absolute deviation) is just another
```

```
name for the MAE
```

```
#
```

```
https://robjhyndman.com/hyndsight/forecastmse/#:
```

```
~:text=MAE%20(mean%20absolute%20error)%20is,
```

```
often%20useful%20for%20forecast%20evaluation.
```

```
def mse(actual, predicted)
```

```

    return np.square(np.subtract(np.array(actual),
np.array(predicted))).mean()
Simple Exp Smoothing
def SES(data_input, alpha=0.0, predict_period=1):
    Y = data_input.tolist()
    Yt = []
    for idx ,value in enumerate(Y):
        if idx == 0:
            Yt.append(Y[idx])
        else :
            cal_Yt = round((alpha*Y[idx-1])+((1-
alpha)*Yt[idx-1]),2)
            Yt.append(cal_Yt)
            # print ("Y: ",Y[idx],"Yt: ",Yt[idx])

    lastValueY = Y[len(Y)-1]
    lastValueYt = Yt[len(Yt)-1]
    predictList = []
    for i in range(predict_period):
        Y.append(None)
        if i == 0:
            cal_predict = round((alpha*lastValueY)+((1-
alpha)*lastValueYt),2)
            predictList.append(cal_predict)
        else:
            cal_predict = round((alpha*lastValueY)+((1-
alpha)*predictList[i-1]),2)
            predictList.append(cal_predict)
            # print ("predictPeriod ",i+1, ":",predictList[i])

    #start calculating error from period 9th
    MSE =
    round(mse(data_input[8:],Yt[8:len(Yt)]),2)
    # print(MSE)

    MAD =
    round(mad(data_input[8:],Yt[8:len(Yt)]),2)
    # print(MAD)

    MAPE =
    round(mape(data_input[8:],Yt[8:len(Yt)]),2)
    # print(MAPE)

    Yt.extend(predictList)

    return predictList,MSE,MAD,MAPE,Yt,Y

SES_predict_values,SES_MSE, SES_MAD,
SES_MAPE, SES_Yt, SES_Y = SES(data.loc[:15,
'Sales'],alpha=0.0,predict_period=4)
SES_Y[-4:] = data.loc[16:, 'Sales'].values.tolist()

SES_data =
{'Period':list(range(1,len(SES_Yt)+1)),
'Sales': SES_Y,
'Predict': SES_Yt}
SES_df = pd.DataFrame(SES_data)
SES_df = SES_df.set_index('Period')

# parameter linestyle สามารถกำหนดรูปแบบของ
เส้นที่ plot ได้ (- เส้นทึบ, -- เส้นประ)
linestyle = '--'

sns.set_style("whitegrid")
sns.set(rc={"figure.figsize":(10, 6)})
# sns.lineplot(data = SES_df[['Sales',
'Predict']]).set(title='Simple Exponential
Smoothing',xlabel='Period', ylabel='Sales Value')
sns.lineplot(data = SES_df['Sales'],
label='Actual').set(title='Simple Exponential

```

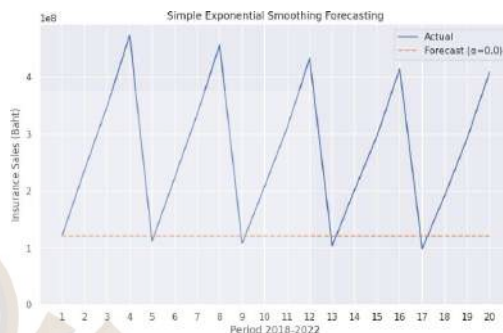
```
Smoothing Forecasting', xlabel='Period 2018 -
2022', ylabel='Insurance Sales (Baht)')
sns.lineplot(data = SES_df['Predict'],
label=f'Forecast ( $\alpha=\{0.0\}$ )',
linestyle=linestyle).set(title='Simple Exponential
Smoothing Forecasting', xlabel='Period 2018 -
2022', ylabel='Insurance Sales (Baht)')
```

```
plt.ylim(0, None)
plt.xlim(0, None)
plt.xticks(range(1, len(SES_Yt)+1))
plt.legend()
```

```
print("Y: ", SES_Y)
print("Yt: ", SES_Yt)
print("predict: ", SES_predict_values)
print("MSE: ", SES_MSE)
print("MAPE: ", SES_MAPE)
print("MAD: ", SES_MAD)
```

```
Y: [120957430.86, 237176683.26, 348315999.34,
473640932.31, 111696754.49, 223261561.44,
334183042.8, 456844390.25, 107824503.86,
207898856.27, 311040107.44, 432801114.79,
103863695.65, 201741391.53, 297642101.18,
413999403.31, 98786804.7, 191643743.94,
292947075.59, 408846655.82]
Yt: [120957430.86, 120957430.86,
120957430.86, 120957430.86, 120957430.86,
120957430.86, 120957430.86, 120957430.86,
120957430.86, 120957430.86, 120957430.86,
120957430.86, 120957430.86, 120957430.86,
120957430.86, 120957430.86, 120957430.86]
```

```
predict: [120957430.86, 120957430.86,
120957430.86, 120957430.86]
MSE: 3.3127313349399492e+16
MAPE: 46.73
MAD: 146200631.45
```



รูปภาคผนวก ก ที่ 4 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตแบบทั่วไปโดยวิธี

ปรับเรียบเอ็กซ์โพเนนเชียล

อย่างง่ายระหว่างปี ค.ศ.

2018 - 2022

Double Exp Smoothing

```
def DES(data_input, alpha=0.2, beta=1.0,
predict_period=1):
```

```
Y = data_input.tolist()
```

```
Lt = []
```

```
Tt = []
```

```
Yt_1 = []
```

```
for idx ,value in enumerate(Y):
```

```
if idx == 0:
```

```
    Lt.append(None)
```

```
    Tt.append(None)
```

```
    Yt_1.append(None)
```

```
elif idx == 1:
```

```
    Lt.append(round(Y[idx],2))
```

```
    Tt.append(round(Y[idx]-Y[idx-1],2))
```

```

Yt_1.append(None)
else :
    cal_Lt = round((alpha*Y[idx])+((1-
alpha)*(Lt[idx-1]+Tt[idx-1])),2)
    Lt.append(cal_Lt)
    cal_Tt = round((beta*(Lt[idx]-Lt[idx-
1]))+((1-beta)*Tt[idx-1])),2)
    Tt.append(cal_Tt)
    cal_Yt_1 = round(Lt[idx-1]+Tt[idx-1],2)
    Yt_1.append(cal_Yt_1)

# print ("Y: ",Y[idx],"Lt: ",Lt[idx],"Tt:
",Tt[idx],"Yt_1: ",Yt_1[idx])

lastValueY = Y[len(Y)-1]
lastValueLt = Lt[len(Lt)-1]
lastValueTt = Tt[len(Tt)-1]
lastValueYt_1 = Yt_1[len(Yt_1)-1]
predictList = []
for i in range(predict_period):
    Y.append(None)
    Lt.append(None)
    Tt.append(i+1)
    if i == 0:
        cal_predict =
round(lastValueLt+lastValueTt,2)
        predictList.append(cal_predict)
    else:
        cal_predict = round(lastValueLt+(Tt[len(Tt)-
1]*lastValueTt),2)
        predictList.append(cal_predict)
    # print ("predictPeriod ",i+1, ":",predictList[i])
#start calculating error from period 9th

#put starting to 2 if wanted to calculate error
from the first forecast value
MSE =
round(mse(data_input[8:],Yt_1[8:len(Yt_1)]),2)
# print(MSE)
MAD =
round(mad(data_input[8:],Yt_1[8:len(Yt_1)]),2)
# print(MAD)
MAPE =
round(mape(data_input[8:],Yt_1[8:len(Yt_1)]),2)
# print(MAPE)

Yt_1.extend(predictList)

return predictList,MSE,MAD,MAPE,Yt_1,Y
DES_predict_values,DES_MSE,DES_MAD,
DES_MAPE,DES_Yt_1,DES_Y =
DES(data.loc[:15,
'Sales'],alpha=0.2,beta=1.0,predict_period=4)
DES_Y[-4:] = data.loc[16:,'Sales'].values.tolist()
DES_data =
{'Period':list(range(1,len(DES_Yt_1)+1)),
'Sales': DES_Y,
'Predict': DES_Yt_1}
DES_df = pd.DataFrame(DES_data)
DES_df = DES_df.set_index('Period')
# parameter linestyle สามารถกำหนดรูปแบบของ
เส้นที่ plot ได้ (- เส้นทึบ, -- เส้นประ)
linestyle = '--'

sns.set_style("whitegrid")
sns.set(rc={"figure.figsize":(10, 6)})

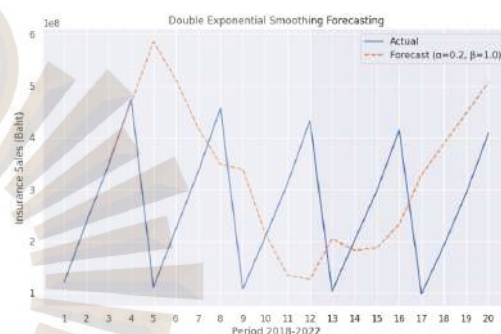
```

```
# sns.lineplot(data = DES_df[['Sales',
'Predict']]).set(title='Double Exponential
Smoothing',xlabel='Period', ylabel='Sales Value')
sns.lineplot(data = DES_df['Sales'],
label='Actual').set(title='Double Exponential
Smoothing Forecasting',xlabel='Period 2018 -
2022', ylabel='Insurance Sales (Baht)')
sns.lineplot(data = DES_df['Predict'],
label=f'Forecast ( $\alpha=\{0.2\}$ ,  $\beta=\{1.0\}$ )',
linestyle=linestyle).set(title='Double Exponential
Smoothing Forecasting',xlabel='Period 2018 -
2022', ylabel='Insurance Sales (Baht)')
```

```
plt.ylim(None, None)
plt.ylim(None, None)
plt.xticks(range(1,len(SES_Yt)+1))
plt.legend(loc='upper right') #Lock legend for
parameters
```

```
print("Y: ",DES_Y)
print("Yt_1: ",DES_Yt_1)
print("predict: ",DES_predict_values)
print("MSE: ",DES_MSE)
print("MAPE: ",DES_MAPE)
print("MAD: ",DES_MAD)
Y: [120957430.86, 237176683.26, 348315999.34,
473640932.31, 111696754.49, 223261561.44,
334183042.8, 456844390.25, 107824503.86,
207898856.27, 311040107.44, 432801114.79,
103863695.65, 201741391.53, 297642101.18,
413999403.31, 98786804.7, 191643743.94,
292947075.59, 408846655.82]
Yt_1: [None, None, 353395935.66,
467583213.54, 585209566.18, 512219250.39,
```

```
418348421.36, 348602978.7, 338987176.37,
215258022.73, 134817737.01, 126338232.76,
205199407.24, 182233720.67, 187338244.76,
232662777.24, 328461188.86, 387992275.27,
447523361.68, 507054448.09]
predict: [328461188.86, 387992275.27,
447523361.68, 507054448.09]
MSE: 2.9270443594557756e+16
MAPE: 66.69
MAD: 141711369.55
```



รูปภาคผนวก ก ที่ 5 แผนภูมิเส้นแสดงการพยากรณ์สำหรับยอดขายประกันชีวิตแบบทั่วไปโดยปรับเรียบเอ็กซ์โพเนนเชียลสองเท่าระหว่างปี ค.ศ. 2018 - 2022

Holt Additive

```
def HWA(data_input, alpha=0.1,
beta=0.1,gamma=0.1, predict_period=1):
    Y = data_input.tolist()
    E = []
    T = []
    S = []
    countS = 0
    Yhat = []
    for idx ,value in enumerate(Y):
```

```

if idx < predict_period-1:
    E.append(None)
    T.append(None)
    S.append(None)
    Yhat.append(None)
elif idx == predict_period-1:
    E.append(np.mean(Y[:idx+1]))
    T.append(round(0,2))
    S.append(None)
    Yhat.append(None)
    S[countS] = Y[countS]-E[countS+
        predict_period-1]
    countS +=1
    Yhat.append(None)
else :
    cal_E = (alpha*(Y[idx]-S[idx-
predict_period]))+((1-alpha)*(E[idx-1]+T[idx-1]))
    E.append(cal_E)
    cal_T = (beta*(E[idx]-E[idx-1]))+((1-
beta)*T[idx-1])
    T.append(cal_T)

    if countS < predict_period:
        S[countS] = Y[countS]-
E[countS+predict_period-1]
        countS +=1

        cal_S = (gamma*(Y[idx]-E[idx]))+((1-
gamma)*S[idx-predict_period])
        S.append(cal_S)

        cal_Yhat = E[idx-1]+(1*T[idx-1])+S[idx-
predict_period]
        Yhat.append(cal_Yhat)

# print ("Y: ",Y[idx],"E: ",E[idx],"T: ",T[idx],"S:
",S[idx],"Yhat: ",Yhat[idx])

lastValueY = Y[len(Y)-1]
lastValueE = E[len(E)-1]
lastValueT = T[len(T)-1]
lastValueS = S[len(S)-1]
lastValueYhat = Yhat[len(Yhat)-1]
predictList = []
for i in range(predict_period):
    Y.append(None)
    E.append(None)
    T.append(None)
    cal_predict =
lastValueE+((i+1)*lastValueT)+S[len(S)-
predict_period]
    predictList.append(cal_predict)

#start calculating error from period 9th
#put starting to 'predict_period' if wanted to
calculate error from the first forecast value
MSE =
round(mse(data_input[8:],Yhat[8:len(Yhat)]),2)
# print(MSE)

MAD =
round(mad(data_input[8:],Yhat[8:len(Yhat)]),2)
# print(MAD)

MAPE =
round(mape(data_input[8:],Yhat[8:len(Yhat)]),2)
# print(MAPE)

Yhat.extend(predictList)

return predictList,MSE,MAD,MAPE,Yhat,Y

```

```

HWA_predict_values,HWA_MSE, HWA_MAD,
HWA_MAPE, HWA_Yhat, HWA_Y =
HWA(data.loc[:15,
'Sales'],alpha=0.1,beta=0.1,gamma=0.1,predict_per
iod=4)
HWA_Y[-4:] = data.loc[16:, 'Sales'].values.tolist()

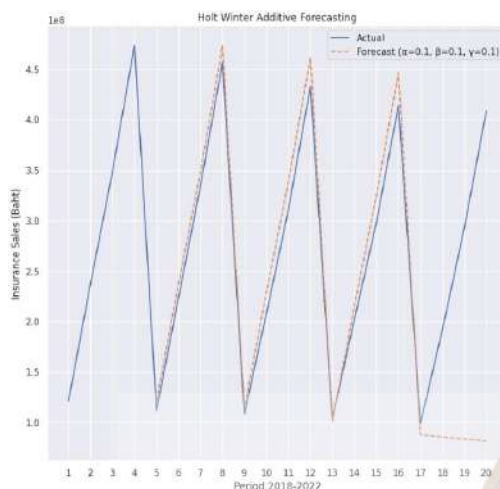
HWA_data =
{'Period':list(range(1,len(SSES_Yt)+1)),
'Sales': HWA_Y,
'Predict': HWA_Yhat}
HWA_df = pd.DataFrame(HWA_data)
HWA_df = HWA_df.set_index('Period')
# parameter linestyle สามารถกำหนดรูปแบบของ
เส้นที่ plot ได้ (- เส้นทึบ, -- เส้นประ)
linestyle = '--'

sns.set_style("whitegrid")
sns.set(rc={"figure.figsize":(10, 9.5)})
# sns.lineplot(data = HWA_df[['Sales',
'Predict']]).set(title='Holt Winter Additive
Forecasting',xlabel='Period', ylabel='Sales Value')
sns.lineplot(data = HWA_df['Sales'],
label='Actual').set(title='Holt Winter Additive
Forecasting',xlabel='Period 2018 - 2022',
ylabel='Insurance Sales (Baht)')
sns.lineplot(data = HWA_df['Predict'],
label=f'Forecast ( $\alpha$ =0.1,  $\beta$ =0.1,  $\gamma$ =0.1)',
linestyle=linestyle).set(title='Holt Winter Additive
Forecasting',xlabel='Period 2018 - 2022',
ylabel='Insurance Sales (Baht)')
plt.ylim(None, None)
plt.ylim(None, None)
plt.xticks(range(1,len(SSES_Yt)+1))

plt.legend(loc='upper right') #Lock legend for
parameters
print("Y: ",HWA_Y)
print("Yhat: ",HWA_Yhat)
print("predict: ",HWA_predict_values)
print("MSE: ",HWA_MSE)
print("MAPE: ",HWA_MAPE)
print("MAD: ",HWA_MAD)

Y: [120957430.86, 237176683.26, 348315999.34,
473640932.31, 111696754.49, 223261561.44,
334183042.8, 456844390.25, 107824503.86,
207898856.27, 311040107.44, 432801114.79,
103863695.65, 201741391.53, 297642101.18,
413999403.31, 98786804.7, 191643743.94,
292947075.59, 408846655.82]
Yhat: [None, None, None, None,
120957430.86000001, 237084076.49629998,
348085167.42573696, 473271079.1494796,
113555357.60354972, 229125598.1974771,
338806237.4865846, 461667033.73743474,
100834117.90409255, 215137479.5786563,
324342653.7319586, 446551696.91317344,
87524544.7520794, 85458334.69133496,
83392124.63059053, 81325914.5698461]
predict: [87524544.7520794,
85458334.69133496, 83392124.63059053,
81325914.5698461]
MSE: 506102669207793.8
MAPE: 7.19
MAD: 19908519.58

```



รูปภาคผนวก ก ที่ 6 แผนภูมิเส้นแสดงการ
พยากรณ์ยอดขายประกันชีวิต
แบบทั่วไปโดยวิธีการบวก
โฮลท์-วินเทอร์ระหว่างปี ค.ศ.
2018 - 2022

Holt Multiplicative

```
def HWM(data_input, alpha=0.2,  
beta=0.6,gamma=0.6, predict_period=1):
```

```
    Y = data_input.tolist()
```

```
    E = []
```

```
    T = []
```

```
    S = []
```

```
    countS = 0
```

```
    Yhat = []
```

```
    for idx ,value in enumerate(Y):
```

```
        if idx < predict_period-1 :
```

```
            E.append(None)
```

```
            T.append(None)
```

```
            S.append(None)
```

```
            Yhat.append(None)
```

```
        elif idx == predict_period-1:
```

```
            E.append(np.mean(Y[:idx+1]))
```

```
            T.append(round(0,2))
```

```
            S.append(None)
```

```
        S[countS] = Y[countS]/E[countS+  
                                predict_period-1]
```

```
        countS +=1
```

```
        Yhat.append(None)
```

```
    else :
```

```
        cal_E = (alpha*(Y[idx]/S[idx-  
predict_period]))+((1-alpha)*(E[idx-1]+T[idx-1]))
```

```
        E.append(cal_E)
```

```
        cal_T = (beta*(E[idx]-E[idx-1]))+((1-beta)  
                                *T[idx-1])
```

```
        T.append(cal_T)
```

```
    if countS < predict_period:
```

```
        S[countS] =
```

```
        Y[countS]/E[countS+predict_period-1]
```

```
        countS +=1
```

```
        cal_S = (gamma*(Y[idx]/E[idx]))+((1-  
gamma)*S[idx-predict_period])
```

```
        S.append(cal_S)
```

```
        cal_Yhat = (E[idx-1]+(1*T[idx-1]))*S[idx-  
predict_period]
```

```
        Yhat.append(cal_Yhat)
```

```
        # print ("Y: ",Y[idx],"E: ",E[idx],"T:  
",T[idx],"S: ",S[idx],"Yhat: ",Yhat[idx])
```

```
    lastValueY = Y[len(Y)-1]
```

```
    lastValueE = E[len(E)-1]
```

```
    lastValueT = T[len(T)-1]
```

```
    lastValueS = S[len(S)-1]
```

```
    lastValueYhat = Yhat[len(Yhat)-1]
```

```
    predictList = []
```

```

for i in range(predict_period):
    Y.append(None)
    E.append(None)
    T.append(None)
    cal_predict =
    (lastValueE+((i+1)*lastValueT))*S[len(S)-
    predict_period]

    predictList.append(cal_predict)

    #start calculating error from period 9th
    #put starting to 'predict_period' if wanted to
    calculate error from the first forecast value
    MSE =
    round(mse(data_input[8:],Yhat[8:len(Yhat)]),2)
    # print(MSE)
    MAD =
    round(mad(data_input[8:],Yhat[8:len(Yhat)]),2)
    # print(MAD)
    MAPE =
    round(mape(data_input[8:],Yhat[8:len(Yhat)]),2)
    # print(MAPE)

    Yhat.extend(predictList)

    return predictList,MSE,MAD,MAPE,Yhat,Y
HWM_predict_values,HWM_MSE, HWM_MAD,
HWM_MAPE, HWM_Yt_1, HWM_Y =
HWM(data.loc[:15,
'Sales'],alpha=0.2,beta=0.6,gamma=0.6,predict_pe
riod=4)
HWM_Y[-4:] = data.loc[16:, 'Sales'].values.tolist()

HWM_data =
{'Period':list(range(1,len(SSES_Yt)+1)),
'Sales': HWM_Y,
'Predict': HWM_Yt_1}
HWM_df = pd.DataFrame(HWM_data)
HWM_df = HWM_df.set_index('Period')
HWM_df.loc[17:, 'Sales'] = data.loc[16:,
'Sales'].values

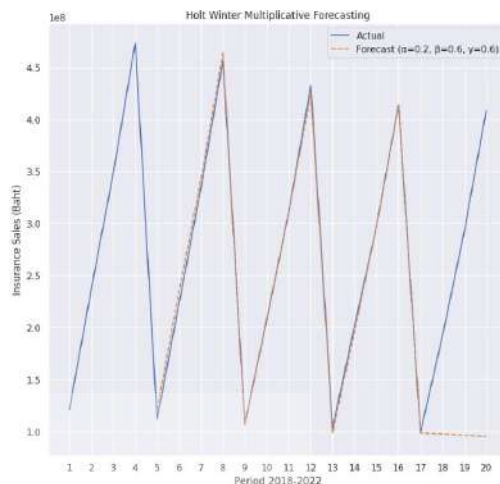
# parameter linestyle สามารถกำหนดรูปแบบของ
เส้นที่ plot ได้ (- เส้นทึบ, -- เส้นประ)
linestyle = '--'

sns.set_style("whitegrid")
sns.set(rc={"figure.figsize":(10, 9.5)})
# sns.lineplot(data = HWM_df[['Sales',
'Predict']]).set(title='Holt Winter Multiple
Forecasting',xlabel='Period', ylabel='Sales Value')
sns.lineplot(data = HWM_df['Sales'],
label='Actual').set(title='Holt Winter
Multiplicative Forecasting',xlabel='Period',
ylabel='Insurance Sales (Baht)')
sns.lineplot(data = HWM_df['Predict'],
label=f'Forecast ( $\alpha=\{0.2\}$ ,  $\beta=\{0.6\}$ ,  $\gamma=0.6\}$ ),
linestyle=linestyle).set(title='Holt Winter
Multiplicative Forecasting',xlabel='Period 2018 -
2022', ylabel='Insurance Sales (Baht)')

plt.ylim(None, None)
plt.ylim(None, None)
plt.xticks(range(1,len(SSES_Yt)+1))
plt.legend(loc='upper right')
print("Y: ",HWM_Y)
print("Yhat: ",HWM_Yt_1)
print("predict: ",HWM_predict_values)
print("MSE: ",HWM_MSE)

```

```
print("MAPE: ",HWM_MAPE)
print("MAD: ",HWM_MAD)
Y: [120957430.86, 237176683.26, 348315999.34,
473640932.31, 111696754.49, 223261561.44,
334183042.8, 456844390.25, 107824503.86,
207898856.27, 311040107.44, 432801114.79,
103863695.65, 201741391.53, 297642101.18,
413999403.31, 98786804.7, 191643743.94,
292947075.59, 408846655.82]
Yhat: [None, None, None, None, 120957430.86,
234963767.54164392, 342899853.66212714,
464669258.091373, 105392240.37494506,
209660465.92508233, 310742947.90019363,
424331497.62000185, 99097365.37282494,
196332187.23939222, 297549784.0565526,
414429643.1265142, 98491670.15442815,
97470499.67469828, 96449329.19496839,
95428158.7152385]
predict: [98491670.15442815,
97470499.67469828, 96449329.19496839,
95428158.7152385]
MSE: 16626614644459.72
MAPE: 1.57
MAD: 2957342.67
```



รูปภาคผนวก ก ที่ 7 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตแบบทั่วไปโดย

วิธีการคูณโฮลท์-วินเทอร์

ระหว่างปี ค.ศ. 2018 - 2022

parameter linestyle สามารถกำหนดรูปแบบของ
เส้นที่ plot ได้ (- เส้นทึบ, -- เส้นประ)

```
linestyle = '--'
```

```
sns.set_style("whitegrid")
```

```
sns.set(rc={"figure.figsize":(10, 8)})
```

```
sns.lineplot(data = data['Sales'],
label='Actual').set(xlabel='Period 2018 - 2022',
ylabel='Insurance Sales (Baht)')
```

```
sns.lineplot(data = HWA_df.loc[:, 'Predict'],
label=f'Holt Winter Additive ( $\alpha=\{0.1\}$ ,  $\beta=\{0.1\}$ ,
 $\gamma=0.1$ ), linestyle=linestyle)
```

```
sns.lineplot(data = HWM_df.loc[:, 'Predict'],
label=f'Holt Winter Multiplicative ( $\alpha=\{0.2\}$ ,
 $\beta=\{0.6\}$ ,  $\gamma=0.6$ ), linestyle=linestyle)
```

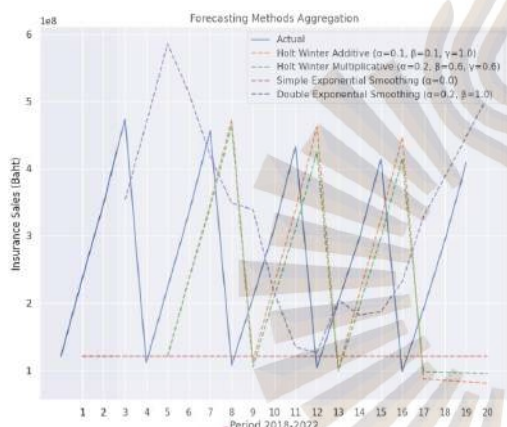
```
sns.lineplot(data = SES_df.loc[:, 'Predict'],
label=f'Simple Exponential Smoothing
( $\alpha=\{0.0\}$ ), linestyle=linestyle)
```

```
sns.lineplot(data = DES_df.loc[:, 'Predict'],
label=f'Double Exponential Smoothing ( $\alpha=\{0.2\}$ ,
 $\beta=\{1.0\}$ )', linestyle=linestyle)

plt.title('Forecasting Methods Aggregation')

plt.ylim(None, None)
plt.xlim(None, None)
plt.xticks(range(1,len(SSES_Yt)+1))

plt.show()
```



รูปภาคผนวก ก ที่ 8 แผนภูมิเส้นแสดงการพยากรณ์สำหรับยอดขายประกันชีวิตแบบทั่วไปโดยวิธีต่าง ๆ ระหว่างปี ค.ศ. 2018 - 2022

```
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.pyplot as plt
plt.rcParams.update({'font.size': 8})

labels = ['Simple Exponential','Double
Exponential','Holt Winter Additive','Holt Winter
Multiple']
```

```
values = [SES_MSE, DES_MSE, HWA_MSE,
HWM_MSE]

# Generate the barplot

plt.figure(figsize=(8, 6))

ax = sns.barplot(x=labels, y=values)

sns.lineplot(label='Actual').set(xlabel='Forecasting
Methods', ylabel='MSE (Baht²)')

ax.set(title='Mean Square Error')
```

```
# Annotating each bar with its height value
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}',
                (p.get_x() + p.get_width() / 2.,
p.get_height()),
                ha = 'center', va = 'center',
                xytext = (0, 9),
                textcoords = 'offset points')
```

```
plt.show()
```



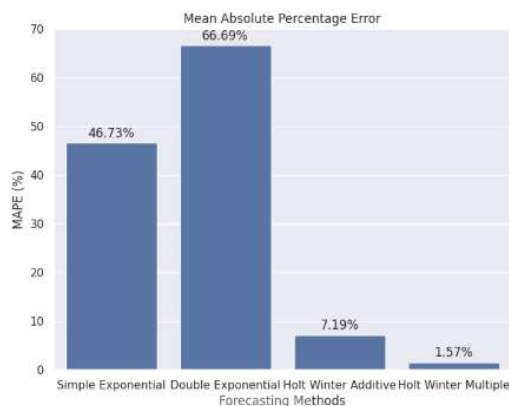
รูปภาคผนวก ก ที่ 9 แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป

```
import seaborn as sns
import matplotlib.pyplot as plt
labels = ['Simple Exponential','Double
Exponential','Holt Winter Additive','Holt Winter
Multiple']
values = [SES_MAPE, DES_MAPE,
HWA_MAPE, HWM_MAPE]
```

```
# Generate the barplot
plt.figure(figsize=(8, 6))
ax = sns.barplot(x=labels, y=values)
sns.lineplot(label='Actual').set(xlabel='Forecasting
Methods', ylabel='MAPE (%)')
ax.set(title='Mean Absolute Percentage Error')
```

```
# Annotating each bar with its height value
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}%',
                (p.get_x() + p.get_width() / 2.,
p.get_height()),
                ha = 'center', va = 'center',
                xytext = (0, 9),
                textcoords = 'offset points')
```

```
plt.show()
```



รูปภาคผนวก ก ที่ 10 แผนภูมิแท่งแสดงค่าเฉลี่ย
ของร้อยละความผิดพลาด
สัมบูรณ์ (Mean Absolute
Percentage Error, MAPE)
การพยากรณ์ยอดขายประกัน
ชีวิตแบบทั่วไป

```
import seaborn as sns
import matplotlib.pyplot as plt
labels = ['Simple Exponential','Double
Exponential','Holt Winter Additive','Holt Winter
Multiple']
values = [SES_MAD, DES_MAD, HWA_MAD,
HWM_MAD]
```

```
# Generate the barplot
plt.figure(figsize=(8, 6))
ax = sns.barplot(x=labels, y=values)
sns.lineplot(label='Actual').set(xlabel='Forecasting
Methods', ylabel='Insurance Sales (Baht)')
ax.set(title='Median Absolute Deviation')
```

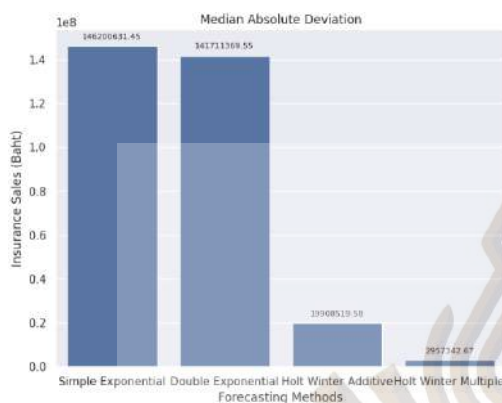
```
# Annotating each bar with its height value
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}',
                (p.get_x() + p.get_width() / 2.,
p.get_height()),
```

```

ha = 'center', va = 'center',
xytext = (0, 9),
textcoords = 'offset points')

plt.show()

```



รูปภาคผนวก ก ที่ 11 แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป

```

import seaborn as sns
import matplotlib.pyplot as plt

```

```

labels = ['Simple Exponential', 'Double Exponential', 'Holt Winter Additive', 'Holt Winter Multiple']
values = [100-SES_MAPE, 100-DES_MAPE, 100-HWA_MAPE, 100-HWM_MAPE]

```

```

# Generate the barplot
plt.figure(figsize=(9, 6))
ax = sns.barplot(x=labels, y=values)
sns.lineplot(label='Actual').set(xlabel='Forecasting Methods')

```

```

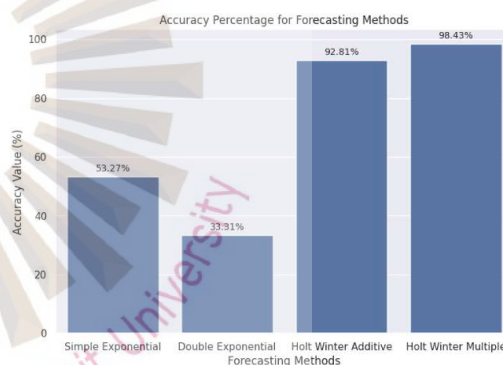
ax.set(title='Accuracy Percentage for Forecasting Methods')

```

```

# Annotating each bar with its height value
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}%',
                (p.get_x() + p.get_width() / 2.,
                 p.get_height()),
                ha = 'center', va = 'center',
                xytext = (0, 9 if p.get_height() > 0 else
                        -9),
                textcoords = 'offset points')
plt.ylabel('Accuracy Value (%)')
plt.show()

```



รูปภาคผนวก ก ที่ 12 แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) การพยากรณ์ยอดขายประกันชีวิตแบบทั่วไป

Error Optimization

```

def
errOptHW_nastedLoop(dataInput,tsFunc,predict_p
eriod=1):
    dicMSE = {}

```

```

dicMAD = {}
dicMAPE = {}

for a in np.round(np.linspace(0,1,11,),2):
    for b in np.round(np.linspace(0,1,11,),2):
        for g in np.round(np.linspace(0,1,11,),2):
            _,HW_MSE, HW_MAD, HW_MAPE, _, _ =
tsFunc(dataInput,alpha=a,beta=b,gamma=g,predict
_period=predict_period)

            dicMSE[tuple([a,1,1])] = HW_MSE
            dicMAD[tuple([a,1,1])] = HW_MAD
            dicMAPE[tuple([a,1,1])] = HW_MAPE

            minMSE = [min(dicMSE, key=dicMSE.get),
dicMSE[min(dicMSE, key=dicMSE.get)]]
            minMAD = [min(dicMAD, key=dicMAD.get),
dicMAD[min(dicMAD, key=dicMAD.get)]]
            minMAPE = [min(dicMAPE,
key=dicMAPE.get), dicMAPE[min(dicMAPE,
key=dicMAPE.get)]]

            return
            minMSE,minMAD,minMAPE,dicMSE,dicMAD,
dicMAPE

minMSE = [min(dicMSE, key=dicMSE.get),
dicMSE[min(dicMSE, key=dicMSE.get)]]
minMAD = [min(dicMAD, key=dicMAD.get),
dicMAD[min(dicMAD, key=dicMAD.get)]]
minMAPE = [min(dicMAPE,
key=dicMAPE.get), dicMAPE[min(dicMAPE,
key=dicMAPE.get)]]

return
minMSE,minMAD,minMAPE,dicMSE,dicMAD,
dicMAPE

Simple Exponential
def
errOptHW_nastedLoop(dataInput,tsFunc,predict_p
eriod=1):
    dicMSE = {}
    dicMAD = {}
    dicMAPE = {}

    for a in np.round(np.linspace(0,1,11,),2):
        _,HW_MSE, HW_MAD, HW_MAPE, _, _ =
tsFunc(dataInput,alpha=a,predict_period=predict_
period)

        dicMSE[tuple([a,1,1])] = HW_MSE
        dicMAD[tuple([a,1,1])] = HW_MAD
        dicMAPE[tuple([a,1,1])] = HW_MAPE

        minMSE = [min(dicMSE, key=dicMSE.get),
dicMSE[min(dicMSE, key=dicMSE.get)]]
        minMAD = [min(dicMAD, key=dicMAD.get),
dicMAD[min(dicMAD, key=dicMAD.get)]]
        minMAPE = [min(dicMAPE,
key=dicMAPE.get), dicMAPE[min(dicMAPE,
key=dicMAPE.get)]]

        return
        minMSE,minMAD,minMAPE,dicMSE,dicMAD,
dicMAPE

fig_MSE = plt.figure(figsize=(10, 10), dpi= 100)
ax_MSE = fig_MSE.add_subplot(111,
projection='3d')
ax_MSE.set_xlabel('Alpha')
ax_MSE.set_ylabel('Beta')
ax_MSE.set_zlabel('gamma')
x_MSE = [t[0] for t in dicMSE.keys()]
y_MSE = [t[1] for t in dicMSE.keys()]
z_MSE = [t[2] for t in dicMSE.keys()]
c_MSE = list(dicMSE.values())

```

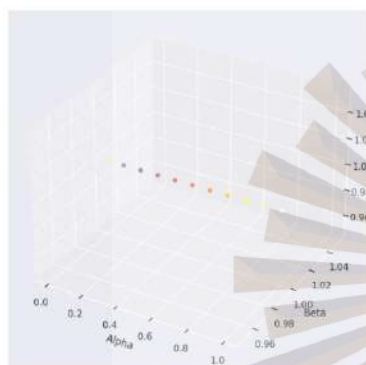
```
img_MSE = ax_MSE.scatter(x_MSE, y_MSE,
z_MSE, c=c_MSE, cmap=plt.hot())
fig_MSE.colorbar(img_MSE,fraction=0.025,pad=
0.2)
plt.show()
```

Smallest MSE: [(0.1, 1, 1),

1.5323081149235914e+16]

Smallest MAD: [(0.1, 1, 1), 106540723.64]

Smallest MAPE: [(0.0, 1, 1), 46.86]



รูปภาคผนวก ก ที่ 13 กราฟสามมิติที่แสดง

ความสัมพันธ์ระหว่างตัวแปรสามตัว แอลฟา (แกน X) เบตา (แกน Y) แกมมา (แกน Z) สำหรับยอดขายประกันชีวิตแบบทั่วไป โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย

Double Exponential

```
def
errOptHW_nastedLoop(dataInput,tsFunc,predict_p
eriod=1):
dicMSE = {}
dicMAD = {}
dicMAPE = {}
```

```
for a in np.round(np.linspace(0,1,11,),2):
```

```
for b in np.round(np.linspace(0,1,11,),2):
```

```
_HW_MSE, HW_MAD, HW_MAPE, _,_ =
tsFunc(dataInput,alpha=a,beta=b,predict_period=p
redict_period)
```

```
dicMSE[tuple([a,b,1])] = HW_MSE
```

```
dicMAD[tuple([a,b,1])] = HW_MAD
```

```
dicMAPE[tuple([a,b,1])] = HW_MAPE
```

```
minMSE = [min(dicMSE, key=dicMSE.get),
```

```
dicMSE[min(dicMSE, key=dicMSE.get)]]
```

```
minMAD = [min(dicMAD, key=dicMAD.get),
```

```
dicMAD[min(dicMAD, key=dicMAD.get)]]
```

```
minMAPE = [min(dicMAPE,
key=dicMAPE.get), dicMAPE[min(dicMAPE,
key=dicMAPE.get)]]
```

```
return
```

```
minMSE,minMAD,minMAPE,dicMSE,dicMAD,d
icMAPE
```

```
minMSE,minMAD,minMAPE,dicMSE,dicMAD,d
icMAPE =
```

```
errOptHW_nastedLoop(data['Sales'],DES,4)
```

```
print("Smallest MSE: ", minMSE)
```

```
print("Smallest MAD: ", minMAD)
```

```
print("Smallest MAPE: ", minMAPE)
```

```
fig_MSE = plt.figure(figsize=(10, 10), dpi= 100)
```

```
ax_MSE = fig_MSE.add_subplot(111,
projection='3d')
```

```
ax_MSE.set_xlabel('Alpha')
```

```
ax_MSE.set_ylabel('Beta')
```

```
ax_MSE.set_zlabel('gamma')
```

```
x_MSE = [t[0] for t in dicMSE.keys()]
```

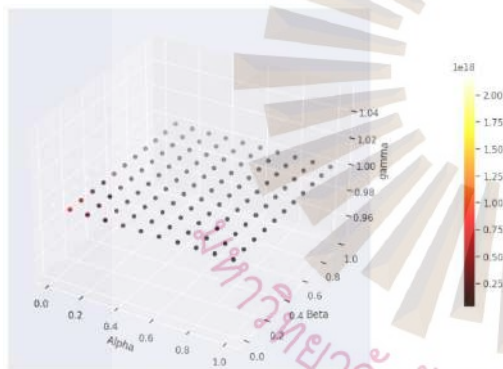
```

y_MSE = [t[1] for t in dicMSE.keys()]
z_MSE = [t[2] for t in dicMSE.keys()]
c_MSE = list(dicMSE.values())

img_MSE = ax_MSE.scatter(x_MSE, y_MSE,
z_MSE, c=c_MSE, cmap=plt.hot())
fig_MSE.colorbar(img_MSE, fraction=0.025, pad=
0.2)
plt.show()

```

Smallest MSE: [(0.3, 0.7, 1),
2.628956882707435e+16]
Smallest MAD: [(1.0, 0.0, 1), 121165697.72]
Smallest MAPE: [(0.2, 1.0, 1), 71.9]



รูปภาคผนวก ก ที่ 14 กราฟสามมิติที่แสดง
ความสัมพันธ์ระหว่างตัว
แปรสามตัว แอลฟา (แกน
X) เบตา (แกน Y) แกมมา
(แกน Z) สำหรับยอดขาย
ประกันชีวิตแบบทั่วไป โดย
วิธีปรับเรียบเอ็กซ์โพเนน
เชียลสองเท่า

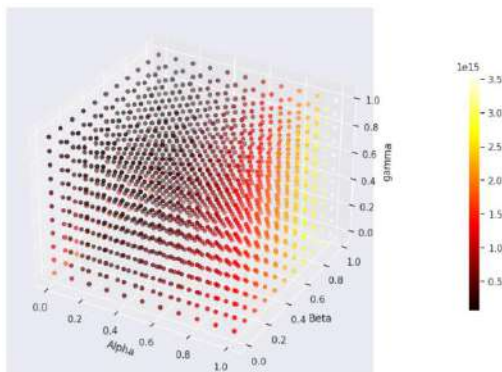
Holt Additive

```

minMSE,minMAD,minMAPE,dicMSE,dicMAD,d
icMAPE =
errOptHW_nastedLoop(data['Sales'],HWA,4)
print("Smallest MSE: ", minMSE)
print("Smallest MAD: ", minMAD)
print("Smallest MAPE: ", minMAPE)
fig_MSE = plt.figure(figsize=(10, 10), dpi= 100)
ax_MSE = fig_MSE.add_subplot(111,
projection='3d')
ax_MSE.set_xlabel('Alpha')
ax_MSE.set_ylabel('Beta')
ax_MSE.set_zlabel('gamma')
x_MSE = [t[0] for t in dicMSE.keys()]
y_MSE = [t[1] for t in dicMSE.keys()]
z_MSE = [t[2] for t in dicMSE.keys()]
c_MSE = list(dicMSE.values())
img_MSE = ax_MSE.scatter(x_MSE, y_MSE,
z_MSE, c=c_MSE, cmap=plt.hot())
fig_MSE.colorbar(img_MSE, fraction=0.025, pad=
0.2)
plt.show()

```

Smallest MSE: [(0.1, 0.9, 1.0),
60363511875071.79]
Smallest MAD: [(0.1, 1.0, 1.0), 6505295.19]
Smallest MAPE: [(0.1, 0.1, 1.0), 2.6]



รูปภาคผนวก ก ที่ 15 กราฟสามมิติที่แสดงความสัมพันธ์ระหว่างตัวแปรสามตัว แอลฟา (แกน X) เบตา (แกน Y) แกมมา (แกน Z) สำหรับยอดขายประกันชีวิตแบบทั่วไป โดยวิธีการบวกโฮลท์-วินเทอร์

Holt Multiplicative

```
minMSE,minMAD,minMAPE,dicMSE,dicMAD,dicMAPE =
```

```
errOptHW_nastedLoop(data['Sales'],HWM,4)
```

```
print("Smallest MSE: ", minMSE)
```

```
print("Smallest MAD: ", minMAD)
```

```
print("Smallest MAPE: ", minMAPE)
```

```
fig_MSE = plt.figure(figsize=(10, 10), dpi= 100)
```

```
ax_MSE = fig_MSE.add_subplot(111,
```

```
projection='3d')
```

```
ax_MSE.set_xlabel('Alpha')
```

```
ax_MSE.set_ylabel('Beta')
```

```
ax_MSE.set_zlabel('gamma')
```

```
x_MSE = [t[0] for t in dicMSE.keys()]
```

```
y_MSE = [t[1] for t in dicMSE.keys()]
```

```
z_MSE = [t[2] for t in dicMSE.keys()]
```

```
c_MSE = list(dicMSE.values())
```

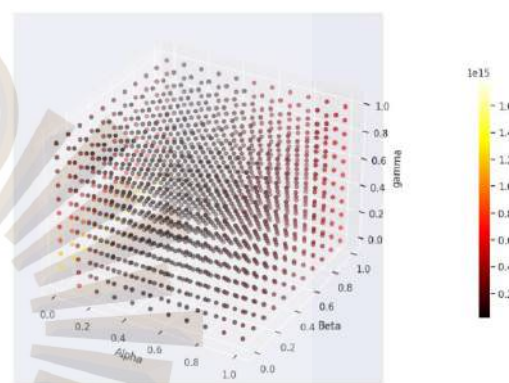
```
img_MSE = ax_MSE.scatter(x_MSE, y_MSE,
z_MSE, c=c_MSE, cmap=plt.hot())
fig_MSE.colorbar(img_MSE,fraction=0.025,pad=
0.2)
plt.show()
```

Smallest MSE: [(0.4, 0.3, 0.3),

15022174686675.37]

Smallest MAD: [(0.3, 0.4, 0.4), 3259536.46]

Smallest MAPE: [(0.2, 0.6, 0.6), 1.47]



รูปภาคผนวก ก ที่ 16 กราฟสามมิติที่แสดงความสัมพันธ์ระหว่างตัวแปรสามตัว แอลฟา (แกน X) เบตา (แกน Y) แกมมา (แกน Z) สำหรับยอดขายประกันชีวิตแบบทั่วไป โดยวิธีการคูณโฮลท์-วินเทอร์

The logo of Rangsit University is a circular emblem. At the top is a stylized flame or sunburst. Below it are several radiating lines of varying lengths, creating a sunburst effect. The bottom half of the circle is filled with a series of vertical lines of increasing height from left to right, resembling a rising sun or a stylized 'R'. The text 'มหาวิทยาลัยรังสิต' is written in Thai script along the bottom arc, and 'Rangsit University' is written in English along the bottom arc.

ภาคผนวก ข

ค่าการเพิ่มประสิทธิภาพข้อผิดพลาดของการพยากรณ์ ค่าการพยากรณ์ยอดขายประกันชีวิต
ค่าการคำนวณข้อผิดพลาดการพยากรณ์ เปอร์เซ็นต์ความแม่นยำสำหรับวิธีการพยากรณ์

1. ค่าการเพิ่มประสิทธิภาพข้อผิดพลาดของการพยากรณ์

1) ค่าการเพิ่มประสิทธิภาพและลดข้อผิดพลาดของการพยากรณ์ สำหรับเซตข้อมูล การประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป

Parameter	Models			
	Holt Winter Additive	Holt Winter Multiplicative	Simple Exponential	Double Exponential
α	0.1	0.2	0.0	0.2
β	0.1	0.6	-	1.0
γ	0.1	0.6	-	-

ตารางภาคผนวก ข ที่ 1 ค่าประสิทธิภาพการลดข้อผิดพลาดในการพยากรณ์สำหรับข้อมูลกรมธรรม์ประกันชีวิตหลัก-ผลิตภัณฑ์ประกันชีวิตทั่วไป

2) ค่าการเพิ่มประสิทธิภาพและลดข้อผิดพลาดของการพยากรณ์สำหรับเซตข้อมูลการประกันชีวิตกรมธรรม์หลัก

Parameter	Models			
	Holt Winter Additive	Holt Winter Multiplicative	Simple Exponential	Double Exponential
α	0.0	0.3	0.1	0.3
β	0.0	0.0	-	0.7
γ	0.8	0.5	-	-

ตารางภาคผนวก ข ที่ 2 ค่าประสิทธิภาพการลดข้อผิดพลาดในการพยากรณ์สำหรับข้อมูลกรมธรรม์ประกันชีวิตกรมธรรม์หลัก

3) ค่าการเพิ่มประสิทธิภาพและลดข้อผิดพลาด

Parameter	Models			
	Holt Winter Additive	Holt Winter Multiplicative	Simple Exponential	Double Exponential
α	0.1	0.0	0.1	0.1
β	0.1	0.0	-	0.0
γ	1.0	1.0	-	-

ตารางภาคผนวก ข ที่ 3 ค่าประสิทธิภาพการ

ลดข้อผิดพลาดในการพยากรณ์สำหรับข้อมูลกรมธรรม์ประกันชีวิตสัญญาเพิ่มเติม

4.) ค่าการเพิ่มประสิทธิภาพและลดข้อผิดพลาดของการพยากรณ์ สำหรับเซตข้อมูลประกันชีวิต รวมทุกกรมธรรม์

Parameter	Models			
	Holt Winter Additive	Holt Winter Multiplicative	Simple Exponential	Double Exponential
α	0.0	0.5	0.1	0.2
β	0.0	0.0	-	0.1
γ	0.7	1.0	-	-

ตารางภาคผนวก ข ที่ 4 ค่าประสิทธิภาพการ

ลดข้อผิดพลาดในการพยากรณ์สำหรับข้อมูลกรมธรรม์ประกันชีวิตรวมทุกกรมธรรม์

2. การพยากรณ์ยอดขายประกันชีวิต

1) การพยากรณ์ยอดขายประกันชีวิตสำหรับเซตข้อมูล การประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป

1.1) “การพยากรณ์โดยวิธีการ
บอกโฮลท์-วินเทอร์ (Holt Winters' Additive)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	120957430.86	None
2018	2	2	237176683.26	None
2018	3	3	348315999.34	None
2018	4	4	473640932.31	None
2019	1	5	111696754.49	120957430.86
2019	2	6	223261561.44	237084076.50
2019	3	7	334183042.80	348085167.43
2019	4	8	456844390.25	473271079.15
2020	1	9	107824503.86	113555357.60
2020	2	10	207898856.27	229125598.20
2020	3	11	311040107.44	338806237.49
2020	4	12	432801114.79	461667033.74
2021	1	13	103863695.65	100834117.90
2021	2	14	00041391.53	215137479.58
2021	3	15	297642101.18	324342653.73
2021	4	16	413999403.31	446551696.91
2022	1	17	98786804.70	87524544.75
2022	2	18	191643743.94	85458334.69
2022	3	19	292947075.59	83392124.63
2022	4	20	408846655.82	81325914.57

ตารางภาคผนวก ข ที่ 5 การพยากรณ์ด้วย

วิธีการบอกโฮลท์-วิน

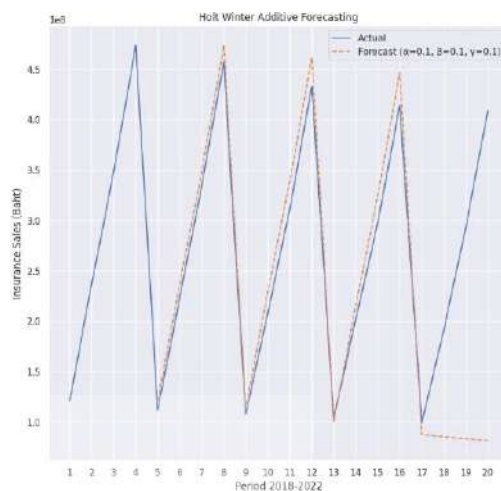
เทอร์ (Holt-Winters'

Additive Method)

สำหรับยอดขายประกัน

ชีวิตแบบทั่วไประหว่าง

ปี ค.ศ. 2018 – 2022



รูปภาคผนวก ข ที่ 1 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตแบบทั่วไปโดย

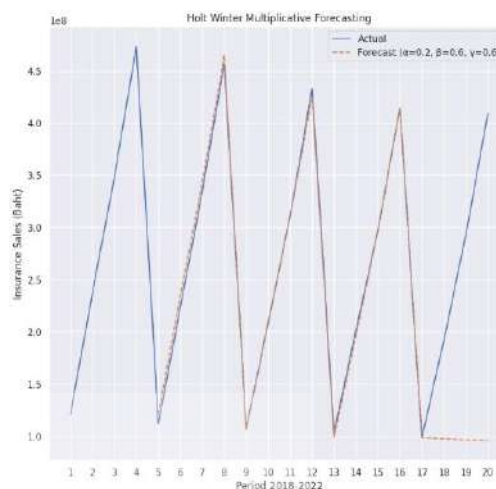
วิธีการบอกโฮลท์-วินเทอร์

ระหว่างปี ค.ศ. 2018 - 2022

1.2) การพยากรณ์ด้วยวิธีการคูณ โฮลท์-วินเทอร์ (Holt Winters' Multiplicative)''

Year	Quarter	Period	Sales	Forecast
Year	Quarter	Period	Sales	Forecast
2018	1	1	120957430.86	None
2018	2	2	237176683.26	None
2018	3	3	348315999.34	None
2018	4	4	473640932.31	None
2019	1	5	111696754.49	120957430.86
2019	2	6	223261561.44	234963767.54
2019	3	7	334183042.80	342899853.66
2019	4	8	456844390.25	464669258.09
2020	1	9	107824503.86	105392240.37
2020	2	10	207898856.27	209660465.93
2020	3	11	311040107.44	310742947.90
2020	4	12	432801114.79	424331497.62
2021	1	13	103863695.65	99097365.37
2021	2	14	00041391.53	196332187.24
2021	3	15	297642101.18	297549784.06
2021	4	16	413999403.31	414429643.13
2022	1	17	98786804.70	98491670.15
2022	2	18	191643743.94	97470499.67
2022	3	19	292947075.59	96449329.19
2022	4	20	408846655.82	95428158.72

ตารางภาพผนวก ข ที่ 6 การพยากรณ์ด้วยวิธีการ
คูณโฮลท์-วินเทอร์ (Holt-
Winters' Multiplicative
Method) สำหรับยอดขาย
ประกันชีวิตแบบทั่วไป
ระหว่างปี ค.ศ.
2018 - 2022

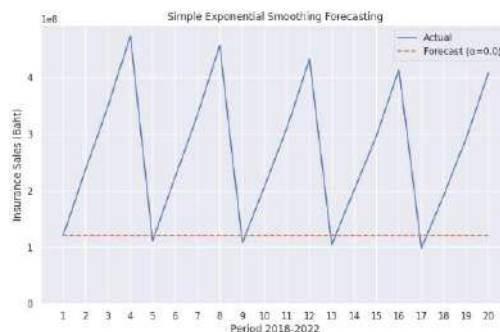


รูปภาพผนวก ข ที่ 2 แผนภูมิเส้นแสดงการ
พยากรณ์สำหรับยอดขาย
ประกันชีวิตแบบทั่วไปโดย
วิธีการคูณโฮลท์-วินเทอร์
ระหว่างปี ค.ศ. 2018 - 2022

1.3) “การพยากรณ์โดยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียลอย่างง่าย (Simple
Exponential Smoothing)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	120957430.86	120957430.86
2018	2	2	237176683.26	120957430.86
2018	3	3	348315999.34	120957430.86
2018	4	4	473640932.31	120957430.86
2019	1	5	111696754.49	120957430.86
2019	2	6	223261561.44	120957430.86
2019	3	7	334183042.80	120957430.86
2019	4	8	456844390.25	120957430.86
2020	1	9	107824503.86	120957430.86
2020	2	10	207898856.27	120957430.86
2020	3	11	311040107.44	120957430.86
2020	4	12	432801114.79	120957430.86
2021	1	13	103863695.65	120957430.86
2021	2	14	00041391.53	120957430.86
2021	3	15	297642101.18	120957430.86
2021	4	16	413999403.31	120957430.86
2022	1	17	98786804.70	120957430.86
2022	2	18	191643743.94	120957430.86
2022	3	19	292947075.59	120957430.86
2022	4	20	408846655.82	120957430.86

ตารางภาคผนวก ข ที่ 7 การพยากรณ์ด้วยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียล
อย่างง่าย (Simple
Exponential Smoothing)
สำหรับยอดขายประกัน
ชีวิตแบบทั่วไประหว่างปี
ค.ศ. 2018 - 2022



รูปภาคผนวก ข ที่ 3 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตแบบทั่วไปโดยวิธี

ปรับเรียบเอ็กซ์โพเนนเชียล

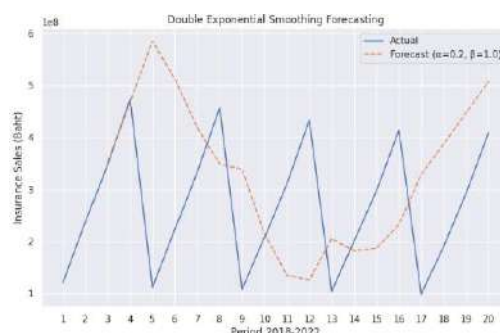
อย่างง่ายระหว่างปี ค.ศ.

2018 - 2022

1.4) “การพยากรณ์โดยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียลสองเท่า (Double
Exponential Smoothing)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	120957430.86	None
2018	2	2	237176683.26	None
2018	3	3	348315999.34	353395935.66
2018	4	4	473640932.31	467583213.54
2019	1	5	111696754.49	585209566.18
2019	2	6	223261561.44	512219250.39
2019	3	7	334183042.80	418348421.36
2019	4	8	456844390.25	348602978.70
2020	1	9	107824503.86	338987176.37
2020	2	10	207898856.27	215258022.73
2020	3	11	311040107.44	134817737.01
2020	4	12	432801114.79	126338232.76
2021	1	13	103863695.65	205199407.24
2021	2	14	00041391.53	182233720.67
2021	3	15	297642101.18	187338244.76
2021	4	16	413999403.31	232662777.24
2022	1	17	98786804.70	328461188.86
2022	2	18	191643743.94	387992275.27
2022	3	19	292947075.59	447523361.68
2022	4	20	408846655.82	507054448.09

ตารางภาคผนวก ข ที่ 8 การพยากรณ์ด้วยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียล
สองเท่า (Double
Exponential Smoothing)
สำหรับยอดขายประกัน
ชีวิตแบบทั่วไประหว่างปี
ค.ศ. 2018 - 2022



รูปภาคผนวก ข ที่ 4 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตแบบทั่วไปโดยวิธี

ปรับเรียบเอ็กซ์โพเนนเชียล

สองเท่าระหว่างปี ค.ศ.

2018 - 2022

2) การพยากรณ์ยอดขายประกันชีวิต
สำหรับเขตข้อมูลการประกันชีวิตกรมธรรม์หลัก

2.1) “การพยากรณ์โดยวิธีการ

บอกโฮลท์-วินเทอร์ (Holt Winters' Additive)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	12064969.75	None
2018	2	2	22415287.00	None
2018	3	3	38032011.99	None
2018	4	4	53146719.25	None
2019	1	5	7409569.12	12064969.75
2019	2	6	17086034.17	22415287.00
2019	3	7	27396807.51	38032011.99
2019	4	8	42522874.74	53146719.25
2020	1	9	8213632.78	8340649.25
2020	2	10	17182364.88	18151884.74
2020	3	11	27695833.73	29523848.41
2020	4	12	45026611.16	44647643.64
2021	1	13	14702125.45	8239036.07
2021	2	14	27753519.57	17376268.85
2021	3	15	43886109.98	28061436.67
2021	4	16	67116178.95	44950817.66
2022	1	17	14242461.68	13409507.57
2022	2	18	26229579.05	13409507.57
2022	3	19	38337511.45	13409507.57
2022	4	20	57804859.63	13409507.57

ตารางภาคผนวก ข ที่ 9 การพยากรณ์ด้วยวิธีการ

บอกโฮลท์-วินเทอร์

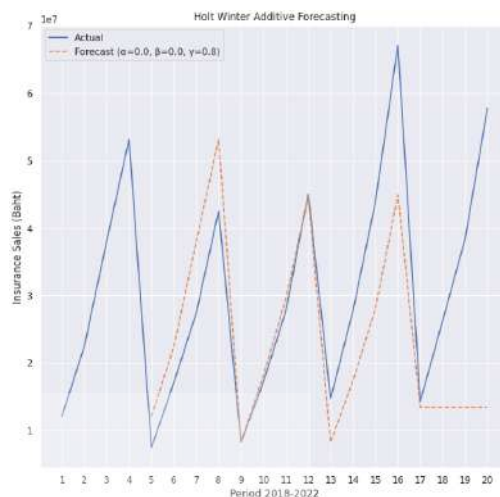
(Holt-Winters' Additive

Method) สำหรับยอดขาย

ประกันชีวิตกรมธรรม์

หลักระหว่างปี ค.ศ.

2018 - 2022



รูปภาคผนวก ข ที่ 5 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตกรมธรรม์หลัก

โดยวิธีการบอกโฮลท์-วิน

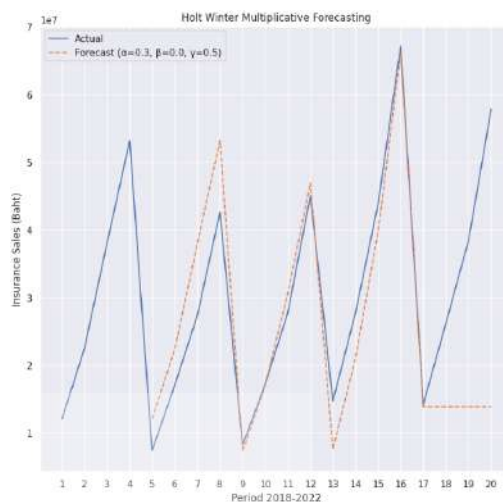
เทอร์ระหว่างปี ค.ศ.

2018 - 2022

2.2) “การพยากรณ์ด้วยวิธีการ คูณโฮลท์-วินเทอร์ (Holt Winters' Multiplicative)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	12064969.75	None
2018	2	2	22415287.00	None
2018	3	3	38032011.99	None
2018	4	4	53146719.25	None
2019	1	5	7409569.12	12064969.75
2019	2	6	17086034.17	22415287.00
2019	3	7	27396807.51	38032011.99
2019	4	8	42522874.74	53146719.25
2020	1	9	8213632.78	7228883.00
2020	2	10	17182364.88	16987241.78
2020	3	11	27695833.73	30553359.67
2020	4	12	45026611.16	46941731.16
2021	1	13	14702125.45	7580608.75
2021	2	14	27753519.57	21060880.07
2021	3	15	43886109.98	39796936.99
2021	4	16	67116178.95	66137668.23
2022	1	17	14242461.68	13847001.62
2022	2	18	26229579.05	13847001.62
2022	3	19	38337511.45	13847001.62
2022	4	20	57804859.63	13847001.62

ตารางภาคผนวก ข ที่ 10 การพยากรณ์ด้วยวิธีการ
คูณโฮลท์-วินเทอร์ (Holt
Winters' Multiplicative)
สำหรับยอดขายประกัน
ชีวิตกรมธรรม์หลัก
ระหว่างปี ค.ศ.
2018 - 2022



รูปภาคผนวก ข ที่ 6 แผนภูมิเส้นแสดงการ
พยากรณ์สำหรับยอดขาย
ประกันชีวิตกรมธรรม์หลัก
โดยวิธีการคูณโฮลท์-วินเทอร์
ระหว่างปี ค.ศ. 2018 - 2022

2.3) “การพยากรณ์โดยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียลอย่างง่าย (Simple
Exponential Smoothing)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	12064969.75	12064969.75
2018	2	2	22415287.00	12064969.75
2018	3	3	38032011.99	13100001.48
2018	4	4	53146719.25	15593202.53
2019	1	5	7409569.12	19348554.20
2019	2	6	17086034.17	18154655.69
2019	3	7	27396807.51	18047793.54
2019	4	8	42522874.74	18982694.94
2020	1	9	8213632.78	21336712.92
2020	2	10	17182364.88	20024404.91
2020	3	11	27695833.73	19740200.91
2020	4	12	45026611.16	20535764.19
2021	1	13	14702125.45	22984848.89
2021	2	14	27753519.57	22156576.55
2021	3	15	43886109.98	22716270.85
2021	4	16	67116178.95	24833254.76
2022	1	17	14242461.68	29061547.18
2022	2	18	26229579.05	32867010.36
2022	3	19	38337511.45	36291927.22
2022	4	20	57804859.63	39374352.39

ตารางภาคผนวก ข ที่ 11 การพยากรณ์ด้วยวิธีปรับ

เรียบเอ็กซ์โพเนนเชียล

อย่างง่าย (Simple

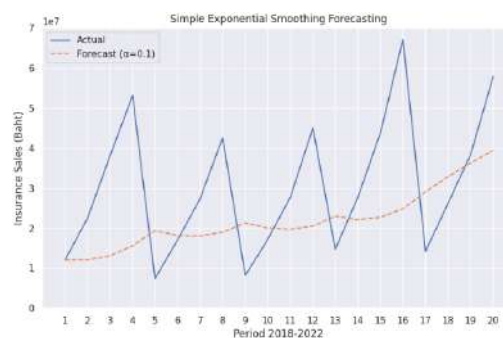
Exponential

Smoothing) สำหรับ

ยอดขายประกันชีวิต

กรมธรรม์หลักระหว่างปี

ค.ศ. 2018 - 2022



รูปภาคผนวก ข ที่ 7 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตกรมธรรม์หลัก

โดยวิธีปรับเรียบเอ็กซ์โพเนน

เชียลอย่างง่ายระหว่างปี ค.ศ.

2018 - 2022

2.4) “การพยากรณ์โดยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียลสองเท่า (Double
Exponential Smoothing)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	12064969.75	None
2018	2	2	22415287.00	None
2018	3	3	38032011.99	32765604.25
2018	4	4	53146719.25	45801789.44
2019	1	5	7409569.12	61003966.51
2019	2	6	17086034.17	46669521.97
2019	3	7	27396807.51	33325817.87
2019	4	8	42522874.74	25833364.82
2020	1	9	8213632.78	28631264.95
2020	2	10	17182364.88	16009319.69
2020	3	11	27695833.73	10110917.13
2020	4	12	45026611.16	12828908.48
2021	1	13	14702125.45	26692253.21
2021	2	14	27753519.57	24781321.98
2021	3	15	43886109.98	27983249.86
2021	4	16	67116178.95	38403977.13
2022	1	17	14242461.68	58697069.30
2022	2	18	26229579.05	70376500.92
2022	3	19	38337511.45	82055932.54
2022	4	20	57804859.63	93735364.16

ตารางภาคผนวก ข ที่ 12 การพยากรณ์ด้วยวิธีปรับ

เรียบเอ็กซ์โพเนนเชียล

สองเท่า (Double

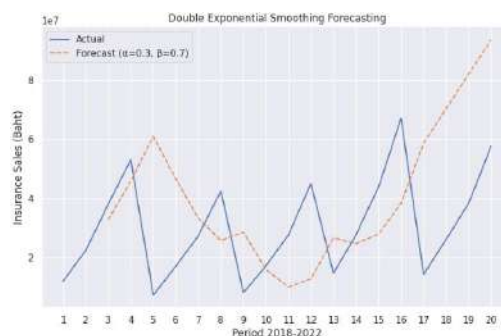
Exponential Smoothing)

สำหรับยอดขายประกัน

ชีวิตกรมธรรม์หลัก

ระหว่างปี ค.ศ.

2018 - 2022



รูปภาคผนวก ข ที่ 8 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตกรมธรรม์หลัก

โดยวิธีปรับเรียบเอ็กซ์โพเนน

เชียลสองเท่าระหว่างปี ค.ศ.

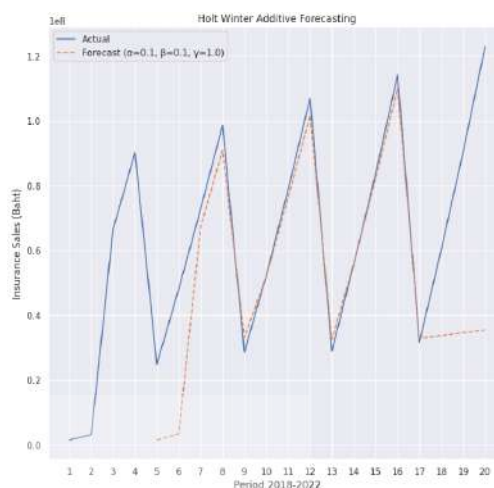
2018 - 2022

3) การพยากรณ์ยอดขายประกันชีวิต
สำหรับเซตข้อมูล สัญญาเพิ่มเติม

3.1) “การพยากรณ์โดยวิธีการ
บวกโฮลท์-วินเทอร์ (Holt Winters' Additive)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	1459909.47	None
2018	2	2	3135763.08	None
2018	3	3	66552399.05	None
2018	4	4	90247932.04	None
2019	1	5	24861304.10	1459909.47
2019	2	6	48072989.46	3369777.03
2019	3	7	72838021.79	67233445.12
2019	4	8	98665579.77	90985023.88
2020	1	9	28520342.94	33126187.85
2020	2	10	52265650.40	51940792.48
2020	3	11	78844226.85	76267894.39
2020	4	12	107022301.12	101644789.04
2021	1	13	28992683.84	32534616.48
2021	2	14	55743671.59	55940612.58
2021	3	15	83800510.25	82284011.40
2021	4	16	114310417.96	110107535.72
2022	1	17	31691762.82	32872205.00
2022	2	18	60425681.52	33742636.05
2022	3	19	90518857.46	34613067.09
2022	4	20	122999750.74	35483498.14

ตารางภาคผนวก ข ที่ 13 การพยากรณ์ด้วยวิธีการ
บวกโฮลท์-วินเทอร์
(Holt Winters' Additive)
สำหรับยอดขายประกัน
ชีวิตกรมธรรม์สัญญา
เพิ่มเติมระหว่างปี ค.ศ.
2018 - 2022



รูปภาคผนวก ข ที่ 9 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตกรมธรรม์สัญญา

เพิ่มเติมโดยวิธีการบวก

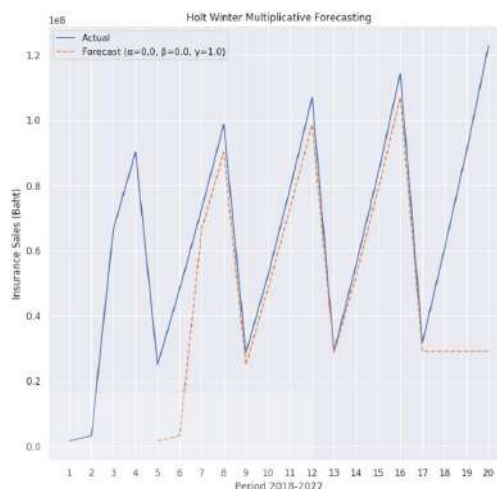
โฮลท์-วินเทอร์ระหว่างปี ค.ศ.

2018 - 2022

3.2) “การพยากรณ์ด้วยวิธีคูณ
โฮลท์-วินเทอร์ มัลติพลีเคทีฟ (Holt Winters'
Multiplicative)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	1459909.47	None
2018	2	2	3135763.08	None
2018	3	3	66552399.05	None
2018	4	4	90247932.04	None
2019	1	5	24861304.10	1459909.47
2019	2	6	48072989.46	3135763.08
2019	3	7	72838021.79	66552399.05
2019	4	8	98665579.77	90247932.04
2020	1	9	28520342.94	24861304.10
2020	2	10	52265650.40	48072989.46
2020	3	11	78844226.85	72838021.79
2020	4	12	107022301.12	98665579.77
2021	1	13	28992683.84	28520342.94
2021	2	14	55743671.59	52265650.40
2021	3	15	83800510.25	78844226.85
2021	4	16	114310417.96	107022301.12
2022	1	17	31691762.82	28992683.84
2022	2	18	60425681.52	28992683.84
2022	3	19	90518857.46	28992683.84
2022	4	20	122999750.74	28992683.84

ตารางภาคผนวก ข ที่ 14 การพยากรณ์ด้วยวิธีการ
คูณโฮลท์-วินเทอร์ (Holt
Winters' Multiplicative)
สำหรับยอดขายประกัน
ชีวิตกรมธรรม์สัญญา
เพิ่มเติมระหว่างปี ค.ศ.
2018 - 2022



รูปภาคผนวก ข ที่ 10 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตกรมธรรม์สัญญา

เพิ่มเติมโดยวิธีการคูณ

โฮลท์-วินเทอร์ระหว่างปี

ค.ศ. 2018 - 2022

3.3) “การพยากรณ์โดยวิธีปรับเรียบเอ็กซ์โพเนนเชียลอย่างง่าย (Simple Exponential Smoothing)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	1459909.47	1459909.47
2018	2	2	3135763.08	1459909.47
2018	3	3	66552399.05	1627494.83
2018	4	4	90247932.04	8119985.25
2019	1	5	24861304.10	16332779.93
2019	2	6	48072989.46	17185632.35
2019	3	7	72838021.79	20274368.06
2019	4	8	98665579.77	25530733.43
2020	1	9	28520342.94	32844218.06
2020	2	10	52265650.40	32411830.55
2020	3	11	78844226.85	34397212.54
2020	4	12	107022301.12	38841913.97
2021	1	13	28992683.84	45659952.69
2021	2	14	55743671.59	43993225.80
2021	3	15	83800510.25	45168270.38
2021	4	16	114310417.96	49031494.37
2022	1	17	31691762.82	55559386.73
2022	2	18	60425681.52	61434489.85
2022	3	19	90518857.46	66722082.66
2022	4	20	122999750.74	71480916.19

ตารางภาคผนวก ข ที่ 15 การพยากรณ์ด้วยวิธีปรับ

เรียบเอ็กซ์โพเนนเชียล

อย่างง่าย (Simple

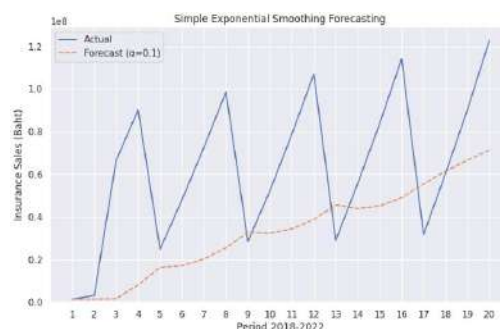
Exponential Smoothing)

สำหรับยอดขายประกัน

ชีวิตกรมธรรม์สัญญา

เพิ่มเติมระหว่างปี ค.ศ.

2018 - 2022



รูปภาคผนวก ข ที่ 11 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตกรมธรรม์

สัญญาเพิ่มเติมโดยวิธีปรับ

เรียบเอ็กซ์โพเนนเชียลอย่าง

ง่ายระหว่างปี ค.ศ. 2018 -

2022

3.4) “การพยากรณ์โดยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียลสองเท่า (Double
Exponential Smoothing)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	1459909.47	None
2018	2	2	3135763.08	None
2018	3	3	66552399.05	4811616.69
2018	4	4	90247932.04	12661548.54
2019	1	5	24861304.10	22096040.50
2019	2	6	48072989.46	24048420.47
2019	3	7	72838021.79	28126730.98
2019	4	8	98665579.77	34273713.67
2020	1	9	28520342.94	42388753.89
2020	2	10	52265650.40	42677766.41
2020	3	11	78844226.85	45312408.42
2020	4	12	107022301.12	50341443.87
2021	1	13	28992683.84	57685383.20
2021	2	14	55743671.59	56491966.87
2021	3	15	83800510.25	58092990.95
2021	4	16	114310417.96	62339596.49
2022	1	17	31691762.82	69212532.25
2022	2	18	60425681.52	70888385.86
2022	3	19	90518857.46	72564239.47
2022	4	20	122999750.74	74240093.08

ตารางภาคผนวก ข ที่ 16 การพยากรณ์ด้วยวิธีปรับ

เรียบเอ็กซ์โพเนนเชียล

สองเท่า (Double

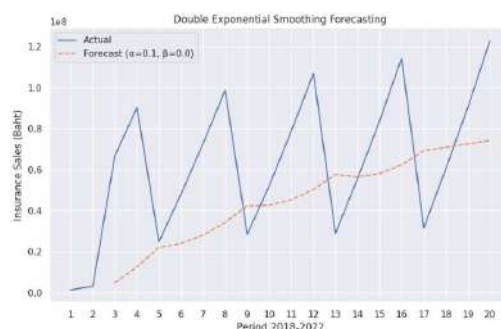
Exponential Smoothing)

สำหรับยอดขายประกัน

ชีวิตกรมธรรม์สัญญา

เพิ่มเติมระหว่างปี ค.ศ.

2018 - 2022



รูปภาคผนวก ข ที่ 12 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตกรมธรรม์สัญญา

เพิ่มเติมโดยวิธีปรับเรียบ

เอ็กซ์โพเนนเชียลสองเท่า

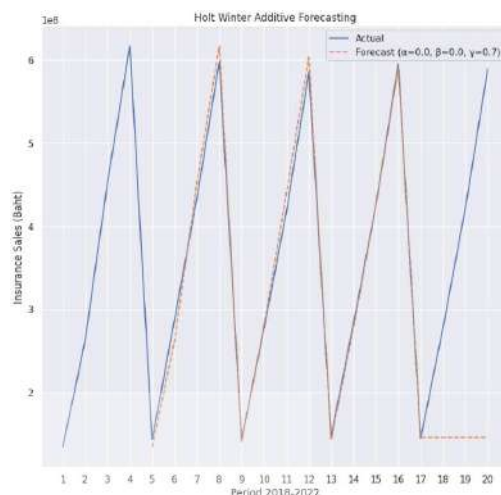
ระหว่างปี ค.ศ. 2018 - 2022

4) การพยากรณ์ยอดขายประกันชีวิต
สำหรับเซตข้อมูล รวมทุกกรรมธรรม์

4.1) “การพยากรณ์โดยวิธีการ
บวกโฮลท์-วินเทอร์ (Holt Winters' Additive)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	134482310.08	None
2018	2	2	262727733.34	None
2018	3	3	452900410.38	None
2018	4	4	617035583.60	None
2019	1	5	143967627.71	134482310.08
2019	2	6	288420585.07	262727733.34
2019	3	7	434417872.10	452900410.38
2019	4	8	598032844.76	617035583.60
2020	1	9	144558479.58	141122032.42
2020	2	10	277346871.55	280712729.55
2020	3	11	417580168.02	439962633.58
2020	4	12	584850027.07	603733666.41
2021	1	13	147558504.94	143527545.43
2021	2	14	283538582.69	278356628.95
2021	3	15	425328721.41	424294907.69
2021	4	16	595426000.22	590515118.87
2022	1	17	144721029.20	146349217.09
2022	2	18	278299004.51	146349217.09
2022	3	19	421803444.50	146349217.09
2022	4	20	589651266.19	146349217.09

ตารางภาคผนวก ข ที่ 17 การพยากรณ์ด้วยวิธีการ
บวกโฮลท์-วินเทอร์
(Holt Winters' Additive
สำหรับยอดขายประกัน
ชีวิตรวมทุกกรรมธรรม์
ระหว่างปี ค.ศ.
2018 - 2022



รูปภาคผนวก ข ที่ 13 แผนภูมิเส้นแสดงการ

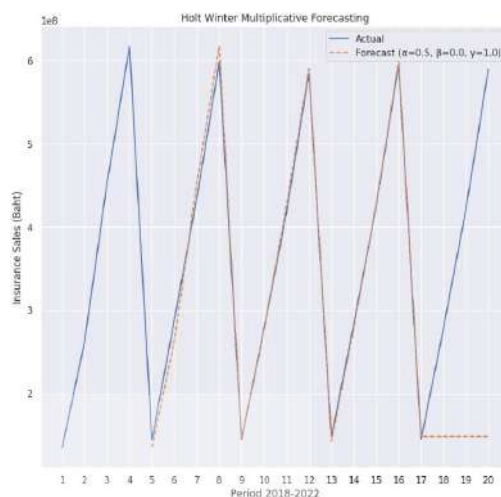
พยากรณ์สำหรับยอดขาย
ประกันชีวิตรวมทุก
กรรมธรรม์โดยวิธีการบวก
โฮลท์-วินเทอร์ระหว่างปี
ค.ศ. 2018 - 2022

4.2) “การพยากรณ์ด้วยวิธีการ

คลุณโฮลท์-วินเทอร์ (Holt Winters' Multiplicative)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	134482310.08	None
2018	2	2	262727733.34	None
2018	3	3	452900410.38	None
2018	4	4	617035583.60	None
2019	1	5	143967627.71	134482310.08
2019	2	6	288420585.07	262727733.34
2019	3	7	434417872.10	452900410.38
2019	4	8	598032844.76	617035583.60
2020	1	9	144558479.58	145648048.99
2020	2	10	277346871.55	277144341.26
2020	3	11	417580168.02	426284344.17
2020	4	12	584850027.07	589928733.24
2021	1	13	147558504.94	142518786.02
2021	2	14	283538582.69	278166469.17
2021	3	15	425328721.41	427220006.76
2021	4	16	595426000.22	599607869.50
2022	1	17	144721029.20	148135220.99
2022	2	18	278299004.51	148135220.99
2022	3	19	421803444.50	148135220.99
2022	4	20	589651266.19	148135220.99

ตารางภาคผนวก ข ที่ 18 การพยากรณ์ด้วยวิธีการ
คลุณโฮลท์-วินเทอร์ (Holt
Winters' Multiplicative)
สำหรับยอดขายประกัน
ชีวิตรวมทุกกรมธรรม์
ระหว่างปี ค.ศ.
2018 - 2022



รูปภาคผนวก ข ที่ 14 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตรวมทุก

กรมธรรม์โดยวิธีการคลุณ

โฮลท์-วินเทอร์ระหว่างปี

ค.ศ. 2018 - 2022

4.3) “การพยากรณ์โดยวิธีเอ็กซ์

โพนเนนเชียล (Simple Exponential Smoothing)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	134482310.08	134482310.08
2018	2	2	262727733.34	134482310.08
2018	3	3	452900410.38	147306852.41
2018	4	4	617035583.60	177866208.21
2019	1	5	143967627.71	221783145.75
2019	2	6	288420585.07	214001593.95
2019	3	7	434417872.10	221443493.06
2019	4	8	598032844.76	242740930.96
2020	1	9	144558479.58	278270122.34
2020	2	10	277346871.55	264898958.06
2020	3	11	417580168.02	266143749.41
2020	4	12	584850027.07	281287391.27
2021	1	13	147558504.94	311643654.85
2021	2	14	283538582.69	295235139.86
2021	3	15	425328721.41	294065484.14
2021	4	16	595426000.22	307191807.87
2022	1	17	144721029.20	336015227.11
2022	2	18	278299004.51	361956304.42
2022	3	19	421803444.50	385303274.00
2022	4	20	589651266.19	406315546.62

ตารางภาคผนวก ข ที่ 19 การพยากรณ์ด้วยวิธีปรับ

เรียบเอ็กซ์โพนเนนเชียล

อย่างง่าย (Simple

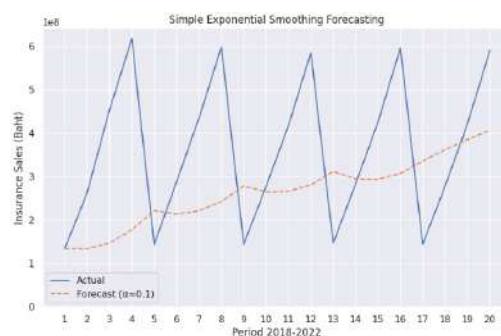
Exponential Smoothing)

สำหรับยอดขายประกัน

ชีวิตรวมทุกกรมธรรม์

ระหว่างปี ค.ศ.

2018 - 2022



รูปภาคผนวก ข ที่ 15 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตรวมทุก

กรมธรรม์โดยวิธีปรับเรียบ

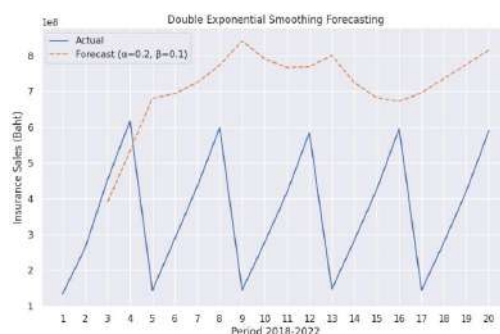
เอ็กซ์โพนเนนเชียลอย่างง่าย

ระหว่างปี ค.ศ. 2018 - 2022

4.4) “การพยากรณ์โดยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียลสองเท่า (Double
Exponential Smoothing)”

Year	Quarter	Period	Sales	Forecast
2018	1	1	134482310.08	None
2018	2	2	262727733.34	None
2018	3	3	452900410.38	390973156.60
2018	4	4	617035583.60	32842575.70
2019	1	5	143967627.71	680849005.78
2019	2	6	288420585.07	693902931.11
2019	3	7	434417872.10	725127015.92
2019	4	8	598032844.76	773491558.30
2020	1	9	144558479.58	841397012.46
2020	2	10	277346871.55	791089732.09
2020	3	11	417580168.02	767126728.98
2020	4	12	584850027.07	769012054.57
2021	1	13	147558504.94	800291046.30
2021	2	14	283538582.69	724801284.43
2021	3	15	425328721.41	682780236.45
2021	4	16	595426000.22	672372395.51
2022	1	17	144721029.20	696526650.61
2022	2	18	278299004.51	736070184.77
2022	3	19	421803444.50	775613718.93
2022	4	20	589651266.19	815157253.09

ตารางภาคผนวก ข ที่ 20 การพยากรณ์ด้วยวิธีปรับ
เรียบเอ็กซ์โพเนนเชียล
สองเท่า (Double
Exponential Smoothing)
สำหรับยอดขายประกัน
ชีวิตรวมทุกกรมธรรม์
ระหว่างปี ค.ศ.
2018 - 2022



รูปภาคผนวก ข ที่ 16 แผนภูมิเส้นแสดงการ

พยากรณ์สำหรับยอดขาย

ประกันชีวิตรวมทุก

กรมธรรม์โดยวิธีปรับเรียบ

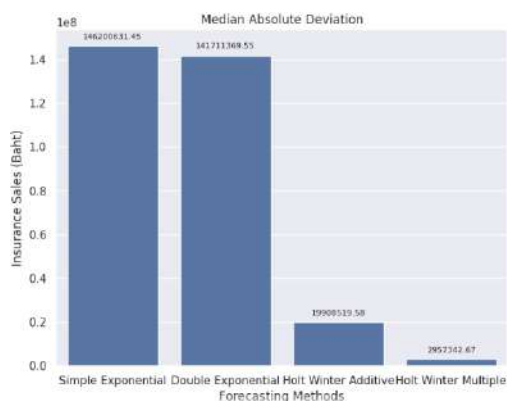
เอ็กซ์โพเนนเชียลสองเท่า

ระหว่างปี ค.ศ. 2018 - 2022

3. เปอร์เซ็นต์การวัดข้อผิดพลาดการ คาดการณ์

1) การคำนวณค่าพยากรณ์ประกันชีวิต
กรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป
โดยการอัปโหลด Sales_Primary-General.csv ที่
โปรแกรมภาษา Python

1.1) ค่าเฉลี่ยความผิดพลาด
สัมบูรณ์ (Mean Absolute Deviation, MAD)



รูปภาคผนวก ข ที่ 17 แผนภูมิแท่งแสดงค่า

เบี่ยงเบนสัมบูรณ์เฉลี่ย

(Mean Absolute Deviation,

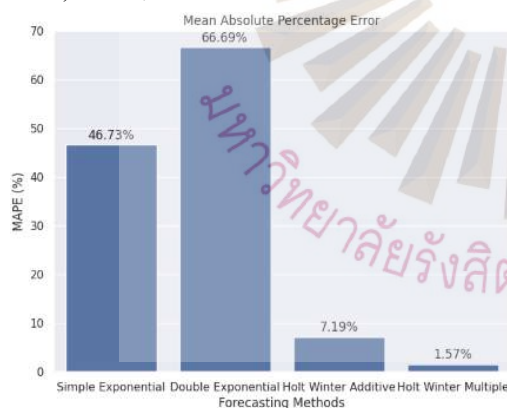
MAD) การพยากรณ์

ยอดขายประกัน

ชีวิตแบบทั่วไป

1.2) ค่าเฉลี่ยของร้อยละความ

ผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE)



รูปภาคผนวก ข ที่ 18 แผนภูมิแท่งแสดงค่าเฉลี่ย

ของร้อยละความผิดพลาด

สัมบูรณ์ (Mean Absolute

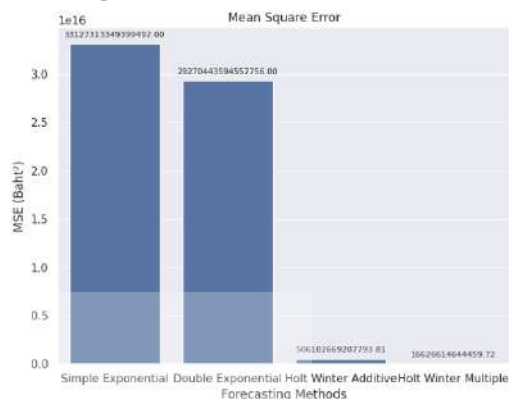
Percentage Error, MAPE)

การพยากรณ์ยอดขาย

ประกันชีวิตแบบทั่วไป

1.3) ค่าเฉลี่ยความผิดพลาดกำลังสอง

(Mean Square Error, MSE)



รูปภาคผนวก ข ที่ 19 แผนภูมิแท่งแสดงค่าความ

ผิดพลาดกำลังสองเฉลี่ย

(Mean Square Error, MSE)

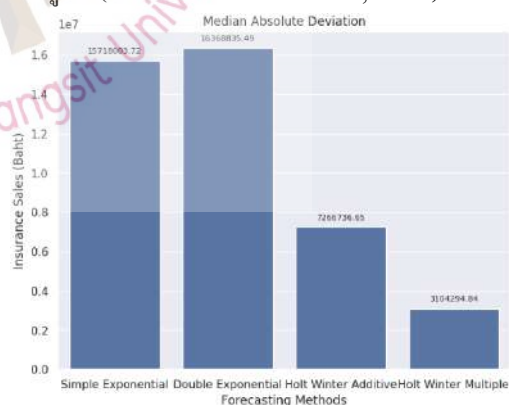
การพยากรณ์ยอดขายประกัน

ชีวิตแบบทั่วไป

2) การคำนวณค่าพยากรณ์ยอดขายประกัน

ชีวิตกรมธรรม์หลัก โดยการอัปโหลด Sales_Primary.csv ที่โปรแกรมภาษา Python

2.1) ค่าเฉลี่ยความผิดพลาดสัมบูรณ์ (Mean Absolute Deviation, MAD)



รูปภาคผนวก ข ที่ 20 แผนภูมิแท่งแสดงค่า

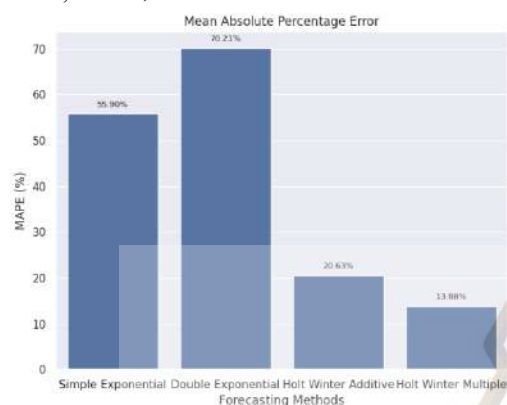
เบี่ยงเบนสัมบูรณ์เฉลี่ย

(Mean Absolute Deviation,

MAD) การพยากรณ์ยอดขาย

ประกันชีวิตกรมธรรม์หลัก

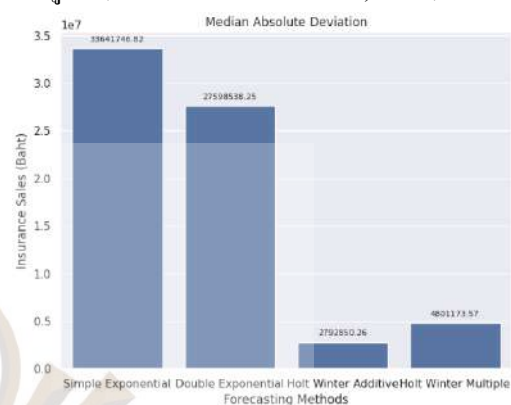
2.2) ค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE)



รูปภาคผนวก ข ที่ 21 แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก

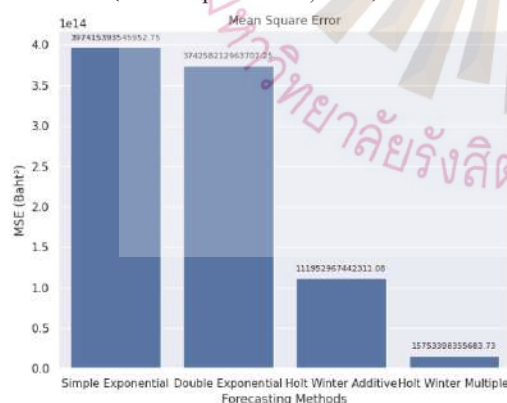
3) การคำนวณค่าพยากรณ์ยอดขายประกันชีวิตประกันสัญญาเพิ่มเติม โดยการอัปโหลด Sales_Additional.csv ที่โปรแกรมภาษา Python

3.1) ค่าเฉลี่ยความผิดพลาดสัมบูรณ์ (Mean Absolute Deviation, MAD)



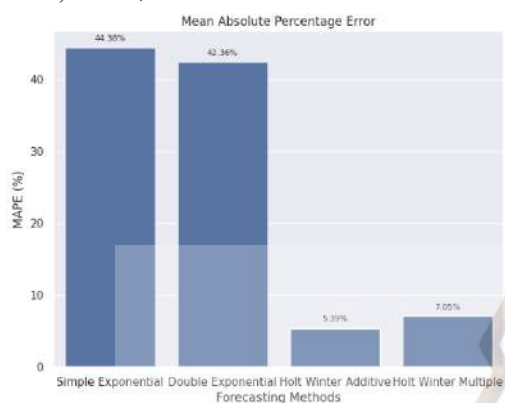
รูปภาคผนวก ข ที่ 23 แผนภูมิแท่งแสดงค่าเบี่ยงเบนสัมบูรณ์เฉลี่ย (Mean Absolute Deviation, MAD) การพยากรณ์ยอดขายประกันชีวิตประกันสัญญาเพิ่มเติม

2.3) ค่าเฉลี่ยความผิดพลาดกำลังสอง (Mean Square Error, MSE)



รูปภาคผนวก ข ที่ 22 แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) การพยากรณ์ยอดขายประกันชีวิตกรมธรรม์หลัก

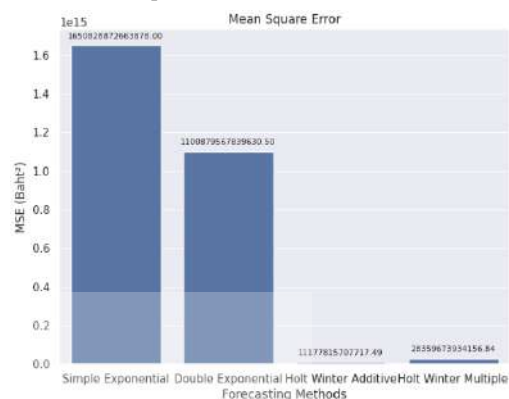
3.2) ค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE)



รูปภาคผนวก ข ที่ 24 แผนภูมิแท่งแสดงค่าเฉลี่ยของร้อยละความผิดพลาดสัมบูรณ์ (Mean Absolute Percentage Error, MAPE) การพยากรณ์ยอดขายประกันชีวิตประกันชีวิตสัญญาเพิ่มเติม

3.3) ค่าเฉลี่ยความผิดพลาดกำลังสอง (Mean Square Error, MSE)

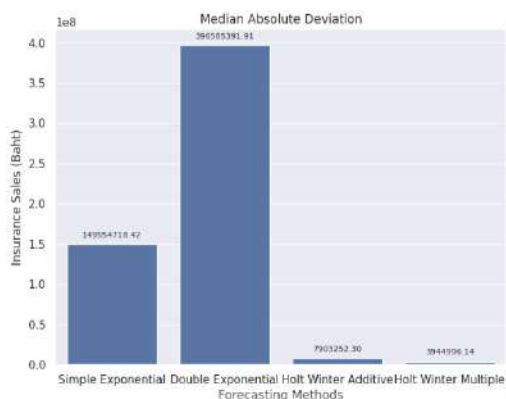
สอง (Mean Square Error, MSE)



รูปภาคผนวก ข ที่ 25 แผนภูมิแท่งแสดงค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error, MSE) การพยากรณ์ยอดขายประกันชีวิตประกันชีวิตสัญญาเพิ่มเติม

4) การคำนวณค่าพยากรณ์รวมทุกกรรมธรรม์โดยการอัปโหลด Sales_Total.csv ที่โปรแกรมภาษา Python

4.1) ค่าเฉลี่ยความผิดพลาดสัมบูรณ์ (Mean Absolute Deviation, MAD)

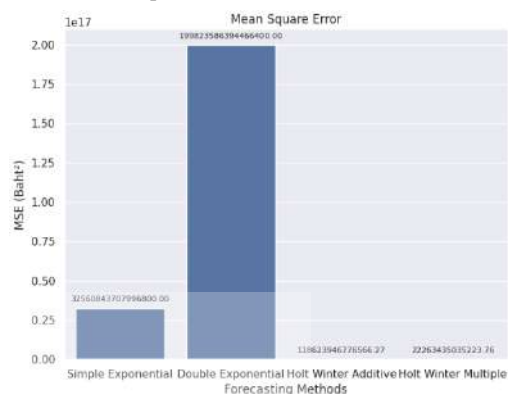


รูปภาคผนวก ข ที่ 26 แผนภูมิแท่งแสดงค่า

เบี่ยงเบนสัมบูรณ์เฉลี่ย

(Mean Absolute Deviation, MAD) การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์

4.3) ค่าเฉลี่ยความผิดพลาดกำลังสอง (Mean Square Error, MSE)



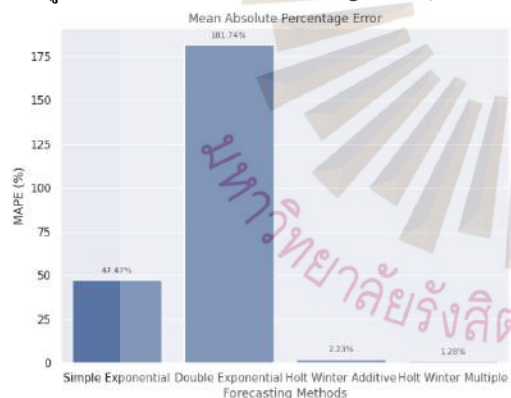
รูปภาคผนวก ข ที่ 28 แผนภูมิแท่งแสดงค่าความ

ผิดพลาดกำลังสองเฉลี่ย

(Mean Square Error, MSE) การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์

4.2) ค่าเฉลี่ยของร้อยละความผิดพลาด

สัมบูรณ์ (Mean Absolute Percentage Error, MAPE)



รูปภาคผนวก ข ที่ 27 แผนภูมิแท่งแสดงค่าเฉลี่ย

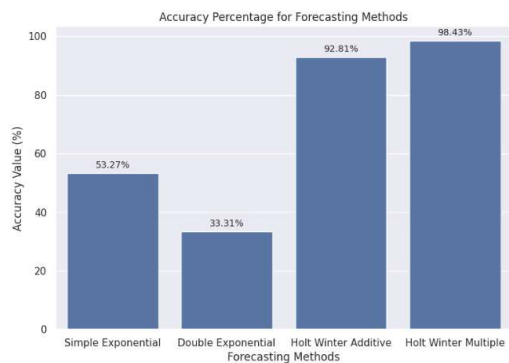
ของร้อยละความผิดพลาด

สัมบูรณ์ (Mean Absolute Percentage Error, MAPE)

การพยากรณ์ยอดขายประกันชีวิตรวมทุกกรมธรรม์

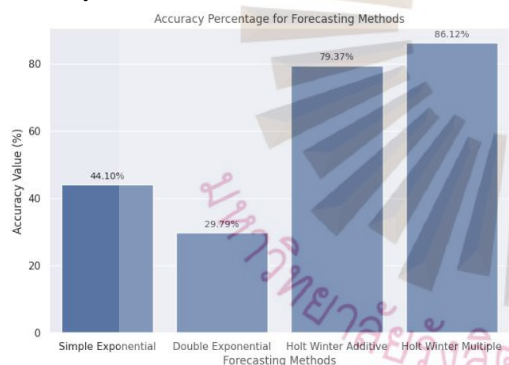
4. ผลลัพธ์จากการคำนวณความแม่นยำของค่าพยากรณ์

1) การคำนวณความแม่นยำของค่าพยากรณ์ยอดขายประกันชีวิตจากการประกันชีวิตกรมธรรม์หลัก-ผลิตภัณฑ์ประกันชีวิตแบบทั่วไป โดยการอัปโหลด Sales_Primary-General.csv ที่โปรแกรมภาษา Python



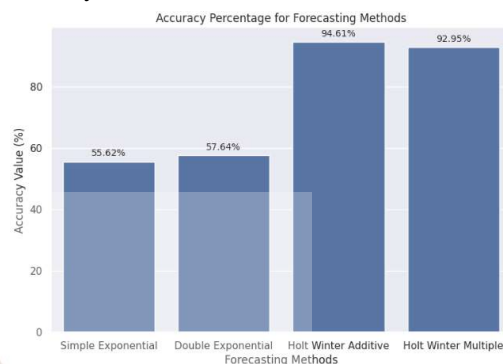
รูปภาคผนวก ข ที่ 29 แผนภูมิแท่งแสดงผลลัพธ์
ความแม่นยำสำหรับการ
พยากรณ์ยอดขายประกัน
ชีวิตแบบทั่วไป

2) การคำนวณความแม่นยำของค่า
พยากรณ์ยอดขายการประกันชีวิตกรมธรรม์หลัก
โดยการอัปโหลด Sales_Primary.csv ที่โปรแกรม
ภาษา Python



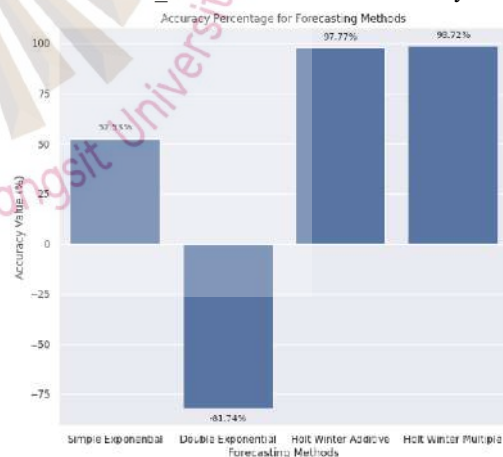
รูปภาคผนวก ข ที่ 30 แผนภูมิแท่งแสดงผลลัพธ์
ความแม่นยำสำหรับการ
พยากรณ์ยอดขายประกัน
ชีวิตกรมธรรม์หลัก

3) การคำนวณความแม่นยำของค่า
พยากรณ์ยอดขายการประกันชีวิตสัญญาเพิ่มเติม
โดยการอัปโหลด Sales_Additional.csv ที่โปรแกรม
ภาษา Python



รูปภาคผนวก ข ที่ 31 แผนภูมิแท่งแสดงผลลัพธ์
ความแม่นยำสำหรับการ
พยากรณ์ยอดขายประกัน
ชีวิตสัญญาเพิ่มเติม

4) การคำนวณความแม่นยำของค่า
พยากรณ์ยอดขายรวมทุกกรมธรรม์โดยการ
อัปโหลด Sales_Total.csv ที่โปรแกรมภาษา Python



รูปภาคผนวก ข ที่ 32 แผนภูมิแท่งแสดงผลลัพธ์
ความแม่นยำสำหรับการ
พยากรณ์ยอดขายประกัน
รวมทุกกรมธรรม์

ประวัติผู้วิจัย

ชื่อ	ปานรวี บุญสม
วัน เดือน ปีเกิด	15 กรกฎาคม 2534
สถานที่เกิด	จังหวัด จันทบุรี ประเทศไทย
ประวัติการศึกษา	มหาวิทยาลัยอัสสัมชัญ วิทยาเขตสุวรรณภูมิ Bachelor of Business Administration in Business Information System, 2558 มหาวิทยาลัยรังสิต ปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์, 2567
ที่อยู่ปัจจุบัน	678/448 ถ.เฉลิมพระเกียรติ ร.9 แขวงประเวศ เขต ประเวศ กรุงเทพมหานคร 10250

