



การพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐโดยใช้ปัจจัยนอกเสริม

นางสาวกุลธิดา มีกำ
นายกลย์ธัช วงศ์วิทยานนท์

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาสถิติ

ภาคคณิตศาสตร์ คณะวิทยาศาสตร์

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ปีการศึกษา 2566



Forecasting the exchange rate of the Thai Baht against the US dollar Using External Factors

Kuntida Meekam
Kolathat Vongvittayanont

A PROJECT SUBMITTED PARTIAL FULLFILLMENT OF THE REQUIREMENT FOR

THE BACHELOR DEGREE OF SCIENCE

DEPARTMENT OF MATHIMATICS FACULTY OF SCIENCE

KING MONGKUT'S UNIVERSITY OF TECHNOLOGY THONBURI

2023

การพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ

โดยใช้ปัจจัยนอกเสริม

นางสาวกุลธิดา มีก่า วท.บ. (สถิติ)
นายกฤษณ์ วงศ์วิทยานนท์ วท.บ. (สถิติ)

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาสถิติ
ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ปีการศึกษา 2566

คณะกรรมการสอบโครงการ

..... อาจารย์ที่ปรึกษาโครงการ
(ผศ. ดร.ณรรธคุณ วิรุฬห์ศรี)

..... กรรมการ
(ดร.พรทิพย์ เดชพิชัย)

..... กรรมการ
(ดร.ธเนศ จิตต์สุภาพรรณ)

..... กรรมการ
(ผศ. ดร.ณภัฏจันทร์ ด่านสวัสดิ์)

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

หัวข้อโครงการ	การพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐโดยใช้ปัจจัยนอกเสริม
หน่วยกิต	3
นักศึกษา	นางสาวกุลธิดา มีกำ
	นายกฤษณ์ วงศ์วิทยานนท์
อาจารย์ที่ปรึกษา	ผศ. ดร.ณรรฐคุณ วิรุฬห์ศรี
หลักสูตร	วิทยาศาสตร์บัณฑิต
สาขาวิชา	สถิติ
ภาควิชา	คณิตศาสตร์
คณะ	วิทยาศาสตร์
ปีการศึกษา	2566

บทคัดย่อ

งานวิจัยครั้งนี้มีวัตถุประสงค์เพื่อศึกษาปัจจัยที่มีอิทธิพลต่ออัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐและพยากรณ์อัตราการแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ โดยใช้แบบจำลอง Autoregressive Integrated Moving Average with exogenous (ARIMAX) และแบบจำลอง Support Vector Regression (SVR) แล้วเปรียบเทียบประสิทธิภาพแบบจำลองด้วยค่า RMSE, MAE และ MAPE โดยศึกษาชุดข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ เป็นข้อมูลรายวันระยะเวลา 19 ปี 1 เดือน ระหว่างเดือนกุมภาพันธ์ พ.ศ. 2548 ถึงเดือนกุมภาพันธ์ พ.ศ.2567 จำนวนทั้งสิ้น 6,968 วัน โดยใช้ปัจจัยนอกเสริม นั่นคือ อัตราดอกเบี้ยนโยบาย เป็นข้อมูลรายเดือนระยะเวลา 19 ปี 1 เดือน ระหว่างเดือนกุมภาพันธ์ พ.ศ. 2548 ถึงเดือนกุมภาพันธ์ พ.ศ.2567 จำนวนทั้งสิ้น 229 เดือน

ผลการวิจัยครั้งนี้พบว่าข้อมูลมีอิทธิพลของฤดูกาล เมื่อพิจารณาค่า RMSE, MAE และ MAPE ของตัวแบบ จะพบว่าแบบจำลองที่ดีที่สุดคือ ARIMA(0,1,1) หรือ แบบจำลอง ARIMAX โดยมีค่า MAPE เท่ากับ 1.59 % แต่เมื่อเปรียบเทียบในกรณีที่มิใช่ปัจจัยภายนอกเสริม พบว่าแบบจำลอง ARIMAX ให้ค่าการพยากรณ์มีความคลาดเคลื่อนเปอร์เซ็นต์ต่ำที่สุด

คำสำคัญ : อัตราแลกเปลี่ยน, การพยากรณ์, ความแม่นยำ, อัตราดอกเบี้ยนโยบาย

Project Title	Forecasting the exchange rate of the Thai Baht against the US dollar Using External Factors	
Project Credits	3	
Students	Kuntida	Meekam
	Kolathat	Vongvittayanont
Advisor	Asst. Prof. Dr.Nathakhun Wiroonsri	
Program	Bachelor of Science	
Field of Study	Statistics	
Department	Mathematics	
Faculty	Science	
Academic Year	2023	

Abstract

This research investigates the factors influencing the exchange rate of the Thai baht against the US dollar. It employs the Autoregressive Integrated Moving Average with exogenous variables (ARIMAX) and Support Vector Regression (SVR) models to forecast the exchange rate. The performance of these models is compared using the RMSE, MAE, and MAPE metrics. Daily exchange rate data spanning 19 years and 1 month (February 2005 to February 2024; 6,968 days) is utilized. The policy interest rate serves as the exogenous variable, with monthly data for the corresponding period (229 months) incorporated into the analysis.

The results reveal a seasonal influence on the exchange rate data. By evaluating the RMSE, MAE, and MAPE values, the ARIMA(0,1,1) or ARIMAX model emerges as the most effective, achieving a MAPE of 1.59%. Notably, when the exogenous variable is excluded, the ARIMAX model demonstrates the lowest percentage error in forecasting.

Keywords : Exchange Rate, Forecasting, Accuracy, Policy rate

กิตติกรรมประกาศ

โครงการเรื่อง “การพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐโดยใช้ปัจจัยนอกเสริม” ฉบับนี้สำเร็จไปได้ด้วยดีโดยได้รับคำปรึกษาอย่างดีจาก ผศ. ดร.ณรรฐคุณ วิรุฬห์ศรี อาจารย์ที่ปรึกษาโครงการ ให้คำแนะนำด้านแนวคิด แนวทางปฏิบัติ และช่วยตรวจทานแก้ไขข้อบกพร่องของงาน พร้อมทั้งให้ความรู้ในเชิงลึก ความชัดเจนในทฤษฎีของโครงการฉบับนี้ ทางคณะผู้จัดทำรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านอาจารย์ และขอกราบขอบพระคุณไว้อย่างสูง

ขอกราบขอบพระคุณ ดร.พรทิพย์ เดชพิชัย ดร.ธเนศ จิตต์สุภาพรรณ และ ผศ.ดร.ณภัฏจันทร์ ด่านสวัสดิ์ ที่ให้คำปรึกษาในเรื่องต่าง ๆ รวมทั้งเป็นกำลังใจที่ดีเสมอมา

สุดท้ายขอขอบคุณเพื่อนๆ ในสาขาทุกคนที่ช่วยให้คำแนะนำดี ๆ และให้ความช่วยเหลือในการให้คำปรึกษาแลกเปลี่ยนความคิด แบ่งปันประสบการณ์ต่าง ๆ และส่งเสริมในทุก ๆ ด้านตลอดการทำวิจัย

คณะผู้วิจัย

สารบัญ

บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ.....	ง
สารบัญตาราง	ฉ
สารบัญรูปภาพ	ช
บทที่ 1.....	1
บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย	1
1.3 ขอบเขตของการวิจัย	2
1.4 นิยามศัพท์เฉพาะ	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	3
1.6 ขั้นตอนและแผนการดำเนินการวิจัย.....	3
บทที่ 2.....	4
แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	4
2.1 อัตราแลกเปลี่ยนระหว่างเงินตรา.....	4
2.2 แนวคิดและทฤษฎีที่เกี่ยวข้องกับการพยากรณ์.....	5
2.3 งานวิจัยที่เกี่ยวข้องกับการพยากรณ์.....	20
2.4 กรอบแนวคิดการวิจัย.....	21
บทที่ 3.....	22
การดำเนินงานวิจัย	22
3.1 ชุดข้อมูลที่ใช้.....	23
3.2 เครื่องมือที่ใช้ในการวิเคราะห์ข้อมูล.....	27

3.3 การวิเคราะห์ข้อมูลและสร้างแบบจำลองเพื่อเปรียบเทียบประสิทธิภาพ	27
บทที่ 4.....	33
ผลการวิจัยและการอภิปราย	33
4.1 ผลการพยากรณ์ชุดข้อมูลตัวแปรภายนอก..... ผิดพลาด! ไม่ได้กำหนดบู๊กมาร์ก	
4.2. ผลการพยากรณ์จากแบบจำลอง ARIMAX (Autoregressive Integrated Moving Average with exogenous)	33
4.3. ผลการพยากรณ์จากแบบจำลอง SVR (Support Vector Regression).....	34
4.4. เปรียบเทียบผลการพยากรณ์.....	35
บทที่ 5.....	37
ผลการศึกษา	37
5.1 สรุปผลการศึกษา	37
5.2 อภิปรายผล	38
5.3 ข้อเสนอแนะ	38
เอกสารอ้างอิง	38
ภาคผนวก	40

สารบัญตาราง

ตารางที่ 2.1	วิธีการทำ DIFFERENCING DATA	14
ตารางที่ 2.2	แสดงลักษณะของ ACF และ PACF ของตัวแบบ ARMA	17
ตารางที่ 2.3	แสดงการสรุปผลจากการอ่านกราฟ ACF และ PACF	17
ตารางที่ 2.4	แสดงข้อมูลงานวิจัยที่เกี่ยวข้อง	21
ตารางที่ 3.1	โครงสร้างข้อมูลอัตราแลกเปลี่ยน THB/USD	23
ตารางที่ 3.2	โครงสร้างข้อมูลอัตราดอกเบี้ยนโยบาย	24
ตารางที่ 3.3	ตารางการแปลงข้อมูลอัตราแลกเปลี่ยน THB/USD จากรายวันเป็นรายเดือนโดยใช้ค่าเฉลี่ย	25
ตารางที่ 3.4	ชุดข้อมูลอัตราแลกเปลี่ยน THB/USD เฉพาะราคาปิดที่สนใจศึกษา	25
ตารางที่ 3.5	ชุดข้อมูลอัตราแลกเปลี่ยน THB/USD เฉพาะราคาปิดและข้อมูลอัตราดอกเบี้ยนโยบาย	26
ตารางที่ 3.6	ค่า ADF ที่ได้จากการทำ UNIT ROOT TEST	30
ตารางที่ 3.7	แสดงค่า AKAIKE INFORMATION CRITERION (AIC) และค่า SCHWARZ'S BAYESIAN INFORMATION CRITERION (BIC) ของแต่ละ ARIMA(P,D,Q)	31
ตารางที่ 4.1	ผลการพยากรณ์ 3 ช่วงเวลาสำหรับการพยากรณ์อัตราดอกเบี้ยนโยบาย . ผิดพลาด! ไม่ได้กำหนด	
	บักมารัก	
ตารางที่ 4.2	ค่าความคลาดเคลื่อนที่ได้จากแบบจำลองสำหรับการพยากรณ์อัตราดอกเบี้ยนโยบาย ... ผิดพลาด!	
	ไม่ได้กำหนดบักมารัก	
ตารางที่ 4.3	ผลการพยากรณ์ 1 ช่วงเวลาโดยใช้ตัวแปรภายนอก	33
ตารางที่ 4.4	ผลการพยากรณ์ 1 ช่วงเวลาโดยใช้แบบจำลอง SVR	34
ตารางที่ 4.6	เปรียบเทียบค่าความคลาดเคลื่อนของแบบจำลอง ARIMAX และแบบจำลอง SVR	35
ตารางที่ 4.7	เปรียบเทียบค่าความคลาดเคลื่อนของแบบจำลอง ARIMA และแบบจำลอง ARIMAX	36
ตารางที่ 5.1	เปรียบเทียบค่าความคลาดเคลื่อนของแต่ละแบบจำลอง	37

สารบัญรูปภาพ

ภาพที่ 2.1 ภาพตัวอย่างการแบ่งข้อมูลโดยไม่คำนึงถึง AUTOCORRELATION ของ K-FOLD CROSS- VALIDATION โดยค่า $K = 4$ หรือแบ่งข้อมูลเป็น 4 กลุ่ม	8
ภาพที่ 2.2 แสดงขั้นตอนการทดสอบความเสถียรด้วยเทคนิค WALK FORWARD ANALYSIS.....	9
ภาพที่ 2.3 แสดงการแยกองค์ประกอบของข้อมูล	11
ภาพที่ 2.4 แสดงกราฟการกระจายตัวของข้อมูลโดยใช้ Q-Q PLOT	12
ภาพที่ 2.5 แสดงข้อมูล STATIONARY และ NONSTATIONARY	13
ภาพที่ 2.6 แสดงการพล็อตกราฟ ACF และ PACF	17
ภาพที่ 3.1 วิธีการดำเนินการวิเคราะห์และสร้างตัวแบบของงานวิจัย	22
ภาพที่ 3.2 แสดงลักษณะของข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐเฉพาะราคาปิดรายเดือน และอัตราดอกเบี้ยนโยบาย	26
ภาพที่ 3.3 แสดงองค์ประกอบของชุดข้อมูล.....	28
ภาพที่ 3.4 แสดงการแจกแจงของชุดข้อมูล.....	29
ภาพที่ 3.5 แสดงความนิ่ง (STATIONARY) ของข้อมูล	30
ภาพที่ 3.6 แสดงความสัมพันธ์ (AUTOCORRELATION) ของข้อมูล	31
ภาพที่ 4.1 แสดงการพยากรณ์ค่าจริงเทียบกับค่าพยากรณ์โดยใช้ตัวแปรภายนอกจากแบบจำลอง ARIMAX.	34
ภาพที่ 4.2 แสดงการพยากรณ์ค่าจริงเทียบกับค่าพยากรณ์โดยใช้ตัวแปรภายนอกจากแบบจำลอง SVR.....	35

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

อัตราแลกเปลี่ยนเงินตรามีความสำคัญอย่างมากในการทำธุรกิจและธุรกรรมระหว่างประเทศ เนื่องจากอัตราแลกเปลี่ยนมีผลต่อการแลกเปลี่ยนซื้อขายสินค้า ซึ่งมีปัจจัยหลายอย่างส่งผลต่อการผันผวนของค่าเงิน เช่น สภาวะเศรษฐกิจโลก, นโยบายการเงินของรัฐบาล, อัตราเงินเฟ้อ ฯลฯ ซึ่งทำให้การแลกเปลี่ยนซื้อขายสินค้ากับต่างชาติมีความเสี่ยงเพิ่มสูงขึ้น แต่ถ้าหากสามารถคาดการณ์อัตราแลกเปลี่ยนเงินตราได้ จะลดความเสี่ยงและเพิ่มโอกาสในการทำธุรกิจเพิ่มยิ่งขึ้น ซึ่งจะช่วยในการสร้างแผนการจัดการ การลงทุนและการทำธุรกรรมได้อย่างรอบคอบยิ่งขึ้น

ในภาวะเศรษฐกิจของประเทศไทย ที่ปัจจุบันมีความผันผวนของค่าเงินอย่างสูง อันเนื่องมาจากอัตราแลกเปลี่ยนนั้นมีความผันแปรจากสภาวะเศรษฐกิจโลกเป็นอย่างมาก ซึ่งอัตราแลกเปลี่ยนเงินตราเป็นกลไกราคาที่สำคัญที่ช่วยรักษาเสถียรภาพเศรษฐกิจที่มีความผันผวนอยู่เสมอ ทำให้อัตราเงินเฟ้อของประเทศไทยเพิ่มสูงขึ้น หากธุรกิจที่มีการซื้อขายกับต่างชาติไม่สามารถวางแผนรับมือกับความผันผวนของค่าเงินได้ ก็อาจส่งผลกระทบต่อธุรกิจเป็นอย่างมาก ซึ่งหากสามารถคาดการณ์ความเป็นไปได้ของอัตราแลกเปลี่ยน ก็สามารถช่วยลดความเสี่ยงของความผันผวนลง

โดยผู้ทำวิจัยได้เล็งเห็นถึงปัญหานี้จึงได้นำเอาวิธีการทางสถิติและตัวแบบการพยากรณ์ 2 รูปแบบ นั่นคือ SARIMAX (Seasonal Auto Regressive Integrated Moving Average with eXogenous) และ SVR (Support Vector Regression) ซึ่งเป็นตัวแบบที่มีความเหมาะสมในการวิเคราะห์ข้อมูลที่มีการเปลี่ยนแปลงเป็นฤดูกาลมาใช้ในการวิเคราะห์อัตราแลกเปลี่ยนบาทต่อเงินดอลลาร์สหรัฐ เพื่อเปรียบเทียบโมเดลที่ให้ผลการพยากรณ์ที่แม่นยำและคลาดเคลื่อนน้อยที่สุด โดยวิเคราะห์จากตัวแปรอิสระนั้นคือ เวลา (Time) และ อัตราดอกเบี้ยนโยบาย (Policy rate) ที่อาจส่งผลต่อตัวแปรตามนั้นคือ อัตราแลกเปลี่ยนบาทต่อเงินดอลลาร์สหรัฐ (THB/USD Exchange rate) จากข้อมูลรายเดือนตั้งแต่กุมภาพันธ์ พ.ศ. 2548 ถึงกุมภาพันธ์ พ.ศ.2567 รวมทั้งสิ้น 229 เดือน แล้วเปรียบเทียบตัวแบบการพยากรณ์โดยตัววัดประสิทธิภาพนั้นคือ RMSE, MAE และ MAPE และเปรียบเทียบเพื่อสรุปตัวแบบการพยากรณ์ที่ให้ค่าความคลาดเคลื่อนน้อยที่สุด

1.2 วัตถุประสงค์ของการวิจัย

1. เพื่อพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ โดยใช้แบบจำลอง ARIMAX และ SVR และเปรียบเทียบแบบจำลองที่ดีที่สุด

1.3 ขอบเขตของการวิจัย

1. ขอบเขตด้านข้อมูล

- ข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐจากธนาคารแห่งประเทศไทย (Bank of Thailand) ทั้งหมด 6,968 วัน
- ข้อมูลอัตราดอกเบี้ยนโยบายจากธนาคารแห่งประเทศไทย (Bank of Thailand) ทั้งหมด 229 เดือน

2. ขอบเขตด้านเวลา

- ข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐรายวันระยะเวลา 19 ปี 1 เดือน (กุมภาพันธ์ พ.ศ. 2548 ถึงกุมภาพันธ์ พ.ศ.2567)
- ข้อมูลอัตราดอกเบี้ยนโยบายรายเดือนระยะเวลา 19 ปี 1 เดือน (กุมภาพันธ์ พ.ศ. 2548 ถึงกุมภาพันธ์ พ.ศ.2567)

1.4 นิยามศัพท์เฉพาะ

- 1 **อัตราแลกเปลี่ยน (Exchange Rate)** หมายถึง เงินตราระหว่างประเทศหนึ่งหน่วยเมื่อเปรียบเทียบกับเงินตราภายในประเทศของประเทศใดประเทศหนึ่ง หรืออีกนัยหนึ่ง อัตราการแลกเปลี่ยนเป็นการเปรียบเทียบอำนาจการซื้อของเงินตราสองสกุล
- 2 **การพยากรณ์ (Forecasting)** หมายถึง การคาดการณ์ทิศทางหรือแนวโน้มของข้อมูลที่สนใจที่จะเกิดขึ้นในอนาคต เพื่อใช้เป็นสารสนเทศประกอบการตัดสินใจในการวางแผนต่าง ๆ การพยากรณ์มีวิธีการอยู่หลายวิธีการตามความเหมาะสมของข้อมูล เช่น การพยากรณ์ข้อมูลในอนาคตโดยใช้ข้อมูลในอดีตจนถึงปัจจุบันมาพิจารณา เป็นต้น ซึ่งปัจจัยที่มีความสำคัญต่อการพิจารณารูปแบบการพยากรณ์ที่เหมาะสม คือ ระยะเวลา แนวโน้มและผลกระทบที่มีต่อข้อมูลที่สนใจ
- 3 **ความแม่นยำ (Accuracy)** หมายถึง ประสิทธิภาพการทำนายผลของตัวแบบการพยากรณ์ โดยวัดประสิทธิภาพจากค่าคลาดเคลื่อนที่เกิดขึ้นว่ามีความคลาดเคลื่อนคิดเป็นร้อยละหรือเปอร์เซ็นต์เท่าไร แล้วนำมาเปรียบเทียบระหว่างตัวแบบเพื่อให้ได้ตัวแบบที่เหมาะสมและมีความแม่นยำสูงที่สุด
- 4 **อัตราดอกเบี้ยนโยบาย (Policy rate)** หมายถึง อัตราดอกเบี้ยนโยบาย เป็นอัตราที่ธนาคารกลางจ่ายดอกเบี้ยให้กับธนาคารพาณิชย์ที่เอาเงินมาฝาก หรือเป็นอัตราที่ธนาคารกลางเก็บดอกเบี้ยจากธนาคารพาณิชย์ที่มากู้เงิน ซึ่งอัตราดอกเบี้ยนโยบายจะส่งผลกับอัตราดอกเบี้ยที่ธนาคารพาณิชย์คิดกับลูกค้าที่เป็นผู้กู้หรือผู้ฝากเงินต่อไป

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถมองเห็นแนวโน้มการเปลี่ยนแปลงและนำมาวางแผนรับมือได้ดียิ่งขึ้น
2. สามารถนำแนวคิดไปประยุกต์ใช้กับค่าเงินอื่นได้หลากหลาย
3. เพิ่มโอกาสให้กับธุรกิจที่หลากหลายยิ่งขึ้น

1.6 ขั้นตอนและแผนการดำเนินการวิจัย

[illegible]

บทที่ 2

แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

งานวิจัยฉบับนี้เกี่ยวข้องกับการพยากรณ์อัตราแลกเปลี่ยนเงินตราต่างประเทศในรูปแบบต่าง ๆ โดยมีเป้าหมายเพื่อนำข้อมูลมาวิเคราะห์ เพื่อคาดการณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ และนำมาประยุกต์ใช้ในการวางแผนกำหนดราคาขายสินค้า หรือสั่งซื้อสินค้า โดยผู้วิจัยได้ทำการศึกษาค้นคว้าเอกสาร แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง เพื่อใช้ในการทำวิจัย โดยมีรายละเอียดดังต่อไปนี้

2.1 อัตราแลกเปลี่ยนระหว่างเงินตรา

2.2 แนวคิดและทฤษฎีที่เกี่ยวข้องกับการพยากรณ์

2.2.1. ความหมายและความสำคัญของการพยากรณ์

2.2.2. การพยากรณ์อนุกรมเวลา

2.2.3. การเตรียมข้อมูล

2.2.4. การพิจารณาเลือกตัวแบบการพยากรณ์อนุกรมเวลา

2.2.5. วิธีการพยากรณ์อนุกรมเวลา

2.2.6. การหาค่าคาดเคลื่อนของการพยากรณ์อนุกรมเวลา

2.3 งานวิจัยที่เกี่ยวข้องกับการพยากรณ์

2.4 กรอบแนวคิดการวิจัย

2.1 อัตราแลกเปลี่ยนระหว่างเงินตรา

การแลกเปลี่ยนสกุลเงิน หมายถึง กระบวนการแปลงสกุลเงินหนึ่งเป็นอีกสกุลเงินหนึ่งตามอัตราแลกเปลี่ยนที่เป็นอยู่ มีบทบาทสำคัญในการค้าระหว่างประเทศ การเดินทาง และระบบการเงินการธนาคาร ยกตัวอย่างเช่นเมื่อต้องทำธุรกรรมข้ามประเทศ มักจะต้องแปลงสกุลเงินท้องถิ่นเป็นสกุลเงินต่างประเทศที่ประเทศปลายทางยอมรับ

อัตราแลกเปลี่ยนเงินตราเป็นตัวชี้วัดที่บ่งบอกถึงความเปลี่ยนแปลงในมูลค่าของสกุลเงินต่าง ๆ ต่อกันในตลาดการเงินระหว่างประเทศ อัตราแลกเปลี่ยนสามารถดูได้จากอัตราที่กำหนดในตลาดการเงินหรือธนาคาร ที่ทำการแลกเปลี่ยนเงินตรา ราคาที่กำหนดนี้จะส่งผลต่อความคุ้มค่าและความสมดุลของการทำธุรกรรมทางการเงินระหว่างประเทศ เช่น การนำเงินไปท่องเที่ยวในต่างประเทศ การซื้อขายสินค้าระหว่างประเทศ หรือการลงทุนในต่างประเทศ

อัตราแลกเปลี่ยนสามารถระบุได้โดยใช้คำนวณเป็นอัตราส่วนระหว่างสกุลเงินสองสกุลเงิน ตัวอย่างเช่น อัตราแลกเปลี่ยนระหว่างดอลลาร์สหรัฐ (USD) และยูโร (EUR) อาจถูกแสดงเป็น USD/EUR หรือ EUR/USD และมักมีการประมาณการราคาขาย (bid price) และราคาซื้อ (ask price) ที่ธนาคารหรือตลาดการเงินต่าง ๆ ซึ่งอัตราเหล่านี้จะเปลี่ยนแปลงตามความต้องการและข้อเสนอของตลาดและผู้ซื้อขาย เพื่อรักษาความสมดุลในตลาดการเงิน

2.2 แนวคิดและทฤษฎีที่เกี่ยวข้องกับการพยากรณ์

2.2.1 ความหมายและความสำคัญของการพยากรณ์

การพยากรณ์ คือ การคาดการณ์สิ่งที่จะเกิดขึ้นในอนาคต โดยศึกษาและวิเคราะห์จากข้อมูลในอดีต ข้อมูลปัจจุบันและประสบการณ์ สามารถนำไปใช้เพื่อให้ทราบถึงแนวโน้มการเปลี่ยนแปลงของสถานการณ์หรือสภาพแวดล้อมที่จะมีผลในอนาคต หลายธุรกิจใช้เครื่องมือและระบบซอฟต์แวร์เพื่อวิเคราะห์ข้อมูลจำนวนมากที่เก็บรวบรวมมาเป็นระยะเวลานาน จากนั้นซอฟต์แวร์จะคาดการณ์ความต้องการและแนวโน้มในอนาคต เพื่อช่วยให้บริษัทต่าง ๆ วางแผนหรือตัดสินใจด้านการเงิน การตลาด และการดำเนินงานได้แม่นยำยิ่งขึ้น

การพยากรณ์ทำหน้าที่เป็นเครื่องมือช่วยในการวางแผนเพื่อให้องค์กรสามารถเตรียมความพร้อมในการรับมือกับความไม่แน่นอนที่อาจเกิดขึ้นในอนาคต ทำให้สามารถตอบสนองต่อการเปลี่ยนแปลงได้อย่างมั่นใจ ควบคุมการดำเนินงานทางธุรกิจ และตัดสินใจเชิงกลยุทธ์ที่ขับเคลื่อนการเติบโตในอนาคตได้ ตัวอย่างเช่น วิธีการใช้ทรัพยากรอย่างมีประสิทธิภาพมากขึ้น, การแสดงข้อมูลภาพผลการดำเนินงานของธุรกิจ และการประมาณการซื้อขายและส่งออกในอนาคต เป็นต้น

2.2.2 การพยากรณ์อนุกรมเวลา

อนุกรมเวลา คือ อนุกรมของจุดข้อมูลที่เกิดขึ้นตามลำดับในช่วงเวลาหนึ่ง ซึ่งสามารถใช้กับตัวแปรใดก็ได้ที่เปลี่ยนแปลงตลอดเวลา เช่น ราคาหุ้น การลงทุนในตลาดหุ้นจะใช้ออนุกรมเวลา เพื่อติดตามราคาของหลักทรัพย์เมื่อเวลาผ่านไป สามารถติดตามได้ทั้งระยะสั้น เช่น ราคาหลักทรัพย์รายชั่วโมงในวันทำการ หรือระยะยาว เช่น ราคาหลักทรัพย์ที่ปิดในวันสุดท้ายของทุกเดือน เป็นต้น นอกจากราคาหุ้นแล้วยังมีข้อมูลอนุกรมเวลาที่เปลี่ยนแปลงในแต่ละวันก็เป็นข้อมูลอนุกรมเวลาเช่นเดียวกัน

เทคนิคในการพยากรณ์ข้อมูลจากอนุกรมเวลามีทั้งเทคนิคการพยากรณ์เชิงปริมาณ (Quantitative Forecasting Techniques) และเทคนิคการพยากรณ์เชิงคุณภาพ (Qualitative Forecasting Techniques)

ซึ่งจะกล่าวถึงเฉพาะการพยากรณ์เชิงปริมาณ ซึ่งเป็นการใช้ตัวเลขในอดีตนำมาพยากรณ์ในอนาคต โดยมีข้อมูลเพียงพอสำหรับการวิเคราะห์ทางสถิติและเหมาะสำหรับการพยากรณ์ระยะสั้นและระยะกลาง โดยสูตรการคำนวณอนุกรมเวลา มีดังนี้

$$Y = T \times S \times C \times I \quad (2.1)$$

Y = ค่าการพยากรณ์

T = ค่าอิทธิพลแนวโน้ม

S = ค่าอิทธิพลฤดูกาล

C = ค่าอิทธิพลวัฏจักร

I = ค่าผันแปรเนื่องจากเหตุการณ์ไม่ปกติ

องค์ประกอบของอนุกรมเวลา ในการวิเคราะห์อนุกรมเวลา จะสามารถแยกองค์ประกอบของอนุกรมเวลาเป็น 4 ส่วนดังนี้

1. **ค่าแนวโน้ม (Secular trend)** คือ ค่าที่แสดงถึงการเปลี่ยนแปลงของอนุกรมเวลาในระยะยาว โดยไม่ส่งผลต่อฤดูกาลหรือวัฏจักร แสดงถึงแนวโน้มของข้อมูลว่ามีการเพิ่มขึ้นหรือลดลงในช่วงเวลาที่กำหนด
2. **ค่าความผันแปรตามฤดูกาล (Seasonal Variation)** คือ ค่าที่แสดงถึงการเปลี่ยนแปลงที่เกิดขึ้นในอนุกรมเวลาตามฤดูกาลหรือรอบการเปลี่ยนแปลงที่เกิดขึ้นเป็นระยะ ๆ ซึ่งอาจเกี่ยวข้องกับสภาวะอากาศ ฤดูกาลการส่งเสริมการขาย หรือการเปลี่ยนแปลงที่เกิดขึ้นในธุรกิจ เช่น ยอดขายเสื้อกันหนาว, ยอดขายทุเรียน เป็นต้น
3. **ค่าความผันแปรตามวัฏจักร (Cyclical Variation)** คือ ค่าที่แสดงถึงการเปลี่ยนแปลงที่เกิดขึ้นในอนุกรมเวลาตามวัฏจักรหรือรอบการเปลี่ยนแปลงที่เกิดขึ้นเป็นระยะ ๆ ซึ่งอาจส่งผลต่อการเปลี่ยนแปลงในเศรษฐกิจ การเงิน หรือตลาดทุน
4. **ค่าความผันแปรเนื่องจากเหตุการณ์ผิดปกติ (Irregular Variation)** คือ ค่าที่แสดงถึงการเปลี่ยนแปลงที่ไม่สามารถอธิบายได้โดยใช้แนวโน้ม ฤดูกาล หรือวัฏจักร ซึ่งอาจเกี่ยวข้องกับเหตุการณ์ที่ไม่ปกติ เช่น ภัยพิบัติ สภาวะเศรษฐกิจผิดปกติ หรือเหตุการณ์ที่ไม่คาดคิด เป็นต้น

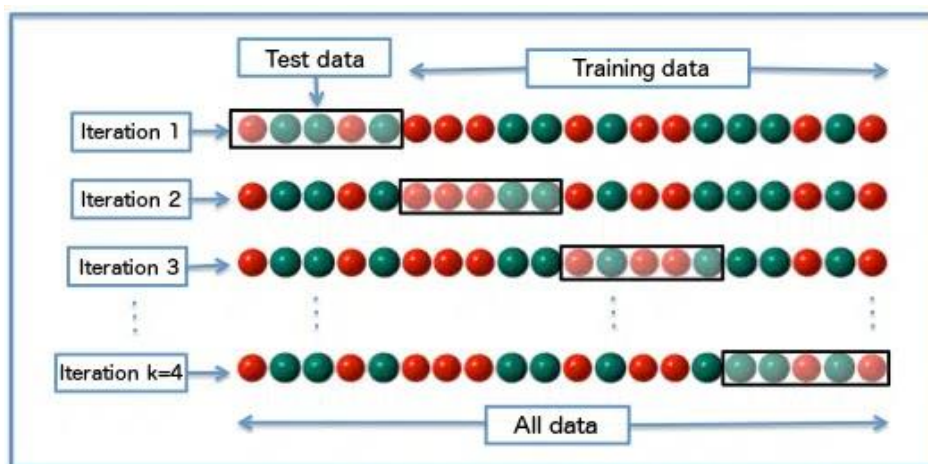
2.2.3 การเตรียมข้อมูล

ในการวิเคราะห์การพยากรณ์จำเป็นจะต้องเตรียมข้อมูลให้อยู่ในรูปแบบที่เหมาะสมที่จะสามารถนำไปใช้ในการพยากรณ์ข้อมูล เนื่องจากข้อมูลอาจเก็บรวบรวมมาจากหลายแหล่งและอาจมีรูปแบบที่แตกต่างกัน จึงจำเป็นที่จะต้องจัดรูปแบบให้ข้อมูลมีรูปแบบเดียวกัน เพื่อป้องกันไม่ให้เกิดความผิดพลาดระหว่างที่ทำการวิเคราะห์และการพยากรณ์ ซึ่งมีหลายกระบวนการเพื่อเตรียมข้อมูลให้มีความเหมาะสมสำหรับใช้เพื่อการพยากรณ์ ดังนี้

1. การทำความสะอาดข้อมูล (Data Cleansing) เป็นการทำให้ข้อมูลที่เก็บรวบรวมมาจากหลายแหล่งข้อมูลมีความสมบูรณ์โดยการกำจัดข้อมูลที่ขาดหาย (Missing Data) ออกไป หรือการแปลงรูปแบบการเก็บข้อมูลให้อยู่ในรูปแบบเดียวกันเพื่อป้องกันข้อผิดพลาดที่เกิดขึ้นระหว่างทำการวิเคราะห์ข้อมูล
2. การลดข้อมูล (Data Reduction) เป็นการกำจัดข้อมูลที่ไม่จำเป็นที่ต้องใช้ออก เช่น ในกรณีที่ต้องการพยากรณ์อัตราแลกเปลี่ยนค่าเงินบาทต่อค่าเงินดอลลาร์สหรัฐแบบรายวัน แต่ข้อมูลที่เก็บรวบรวมมาได้นั้นมีข้อมูลราคาต่ำสุดและสูงสุดในแต่ละวันมาด้วย ซึ่งไม่ใช่ข้อมูลที่เป็นปัจจัยภายนอกที่เลือกใช้ในการพยากรณ์ให้ทำการลบข้อมูลเหล่านั้นออกเพื่อให้ข้อมูลมีขนาดเล็กลงซึ่งช่วยเพิ่มความเร็วในการประมวลผลได้

การแก้ปัญหาข้อมูลทั้งการทำความสะอาดข้อมูลและการลดข้อมูล โดยวิธีต่าง ๆ มีข้อดีและข้อเสียแตกต่างกันออกไปโดยคำนึงถึงปัจจัย ได้แก่ ทรัพยากรที่ใช้ในการแก้ปัญหาหากต้องการใช้เวลานานขึ้นก็ควรเลือกวิธีอื่นในการแก้ปัญหาเกี่ยวกับข้อมูล เวลาที่ใช้ในการประมวลผลหากแก้ปัญหาข้อมูลแล้วทำให้ประมวลผลพยากรณ์ใช้เวลานานมากเกินไปก็ควรใช้วิธีการอื่นในการแก้ปัญหของข้อมูล ความแม่นยำของการพยากรณ์หากความแม่นยำเพิ่มขึ้นได้เปอร์เซ็นต์ที่ไม่มาก แต่ต้องใช้เวลาในการประมวลผลนานและเสียทรัพยากรมากขึ้นควรที่จะเลือกวิธีแก้ปัญหข้อมูลรูปแบบอื่นแทน

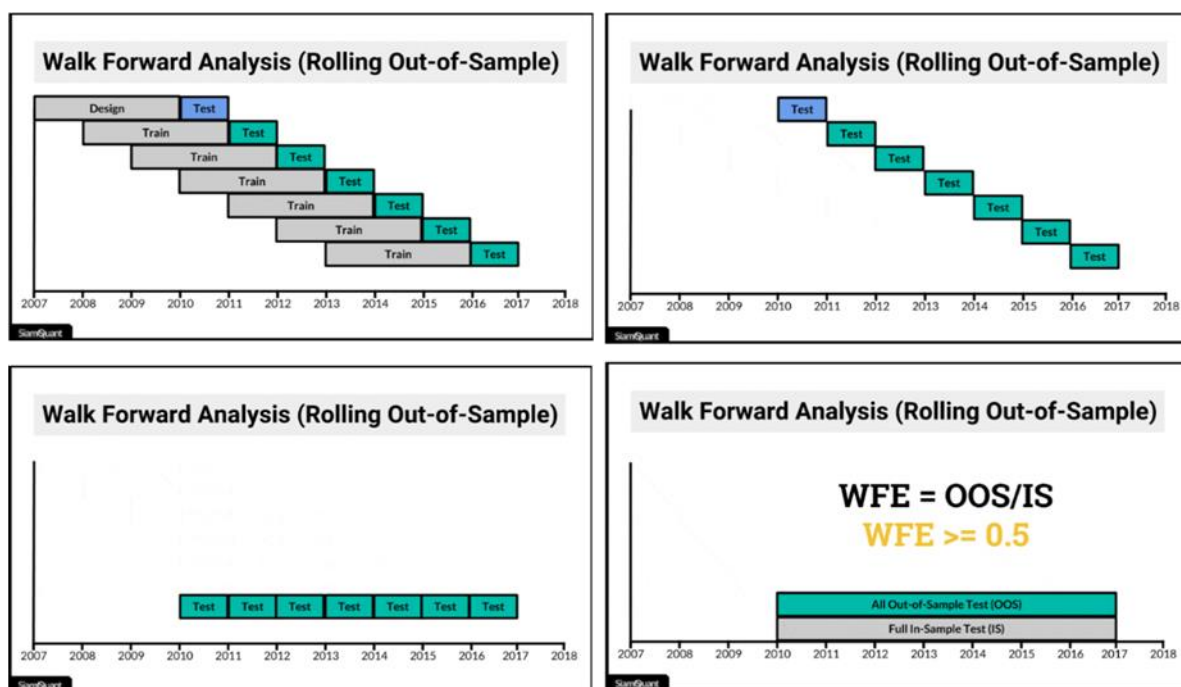
หลังจากตรวจสอบข้อมูลแล้ว ในการแบ่งข้อมูลสำหรับการวิเคราะห์จะทำการแบ่งข้อมูลสำหรับการเทรนนิ่งตัวแบบ (Training data) และข้อมูลสำหรับการทดสอบตัวแบบ (Test data) ซึ่งได้นำเทคนิคช่วยตรวจสอบความเสถียรภาพของแบบจำลองมาช่วยในการวิเคราะห์นั่นคือ Cross-Validation Technique ซึ่งอธิบายวิธีการทดสอบได้ตามภาพที่ 2.1



ภาพที่ 2.1 ภาพตัวอย่างการแบ่งข้อมูลโดยไม่คำนึงถึง Autocorrelation ของ K-Fold Cross-Validation โดย
ค่า $K = 4$ หรือแบ่งข้อมูลเป็น 4 กลุ่ม

ที่มา : (https://www.siamquant.com/wp-content/uploads/2018/11/fig4_Kfold_CrossValidation.jpg)

โดยหลักการของกระบวนการ Cross-Validation คือการแบ่งชุดข้อมูลในการวิจัยออกเป็น 2 ส่วน ส่วนแรกคือข้อมูลที่เรียกว่า “In-Sample” หรือ “Training data” ที่นำมาใช้ในกระบวนการออกแบบและทดสอบแบบจำลอง ส่วนที่สองคือข้อมูล “Out-of-Sample” หรือ “Test data” ที่ถูกแยกออกมาเพื่อนำมาใช้ทดสอบความเสถียรของแบบจำลองที่ถูกออกแบบมาจากข้อมูล In-Sample โดยแบบจำลองที่มีความเสถียรภาพ จะต้องมีประสิทธิภาพในพยากรณ์จากข้อมูล Out-of-Sample ในระดับที่ใกล้เคียงกับการทดสอบในข้อมูล In-Sample ได้มากที่สุด ซึ่งเพื่อให้การทำ Cross-Validation มีเสถียรภาพที่สุด จึงได้มีกระบวนการ Rolling Cross-Validation หรือเทคนิค Walk Forward Analysis (หรืออาจเรียกอีกชื่อว่า Walk Forward Optimization) โดยมาช่วยในการตรวจสอบแบบจำลอง และยังช่วยลด Bias ที่เกิดขึ้นจากการเลือกค่าพารามิเตอร์ที่ดีที่สุด โดยเทคนิค Walk Forward Analysis ใช้วิธีการเลื่อนกรอบช่วงเวลาทั้งของ Training Data และ Test Data ไปข้างหน้าทีละช่วง (Rolling Window) ซึ่งจำเป็นจะต้องกำหนดค่าให้กับกรอบช่วงเวลาของทั้ง 2 ชุดข้อมูลรวมถึงกำหนดค่าให้กับช่วงเวลาที่ขยับไปข้างหน้าหรือ Step Size ด้วย



ภาพที่ 2.2 แสดงขั้นตอนการทดสอบความเสถียรด้วยเทคนิค Walk Forward Analysis
ที่มา : (<https://www.siamquant.com/wp-content/uploads/2018/12/WFA-Feature-Image-1.png>)

โดยที่กระบวนการ Walk Forward Analysis สามารถระบุเป็นขั้นตอนได้ดังนี้

1. การทดสอบจะเริ่มต้นด้วยกระบวนการ Optimize Parameter ใน Training Data เพื่อหาค่า Parameter ที่ให้ผลลัพธ์ที่ดีที่สุด
2. นำค่า Parameter ที่ได้จาก Training Data ไปแทนค่า Parameter ในการทดสอบกับ Test Data
3. เริ่มทดสอบอีกครั้งกับช่วงของ Training Data ใหม่ที่ถูกขยับไปข้างหน้าตาม Step Size ที่กำหนดไว้ โดยจะรวม Test Data ในการทดสอบครั้งก่อนเข้าไปใน Training Data ชุดใหม่ด้วย
4. กระบวนการนี้จะถูกทำซ้ำไปเรื่อย ๆ จนกระทั่งหมดชุดข้อมูล
5. นำค่า Equity ของช่วง Test Data เพียงอย่างเดียวมาเชื่อมต่อกันและคำนวณค่าสถิติ
6. นำผล Equity ของ Test Data ทั้งหมดที่ต่อกัน (Out-of-Sample) ในข้อ 5 มาเปรียบเทียบกับผลการทดสอบที่มาจากขั้นตอนการออกแบบที่ใช้ตัวแปรที่กำหนดเอง (Best or Arbitrary Parameter) ซึ่งอ้างอิงมาจากค่าเริ่มต้นที่ใช้ในการทดสอบแบบจำลอง (In-Sample)
7. นำค่าสถิติจากทั้ง 2 ระบบมาทำการคำนวณค่า Walk Forward Efficiency (WFE)

โดยที่ค่า Walk Forward Efficiency หรือ WFE นั้นคือค่าที่ใช้เป็นมาตรวัดความเสถียรของแบบจำลองโดยการเปรียบเทียบประสิทธิภาพในการทำงานของแบบจำลองในข้อมูลที่ไม่เคยพบมาก่อน (Test

Data) กับประสิทธิภาพของแบบจำลองในข้อมูลที่ใช้ในการวิจัย (Training Data) ซึ่งแบบจำลองที่มีความเสถียรภาพนั้นต้องมีค่า WFE ที่มากกว่า 0.5 ขึ้นไป ซึ่งเราสามารถคำนวณค่า WFE ได้ด้วยสูตรดังนี้

$$WFE = \text{Out-of-Sample Statistic} / \text{In-Sample Statistic} \quad (2.2)$$

2.2.4 การพิจารณาเลือกตัวแบบการพยากรณ์อนุกรมเวลา

การเลือกตัวแบบหรือวิธีการพยากรณ์ ลำดับแรกผู้พยากรณ์จำเป็นต้องศึกษารูปแบบของชุดข้อมูลอย่างละเอียดก่อนก่อน เช่น ทำการตรวจสอบว่าชุดข้อมูลมีรูปแบบอย่างไร รูปแบบองค์ประกอบของแนวโน้ม วัฏจักร ฤดูกาล หรือว่ามีเพียงตัวแปรสุ่มเพียงอย่างเดียว ซึ่งวิธีการที่จะทำให้ทราบถึงองค์ประกอบของข้อมูลเหล่านี้ สามารถทำได้จากการวาดกราฟและวิเคราะห์ค่าสัมประสิทธิ์สหสัมพันธ์ เมื่อทราบรูปแบบของชุดข้อมูลแล้ว จึงนำไปเลือกตัวแบบหรือวิธีการพยากรณ์โดยเกณฑ์ในการเลือกวิธีการพยากรณ์ที่เหมาะสม

2.2.4.1 การทดสอบแนวโน้มและฤดูกาลของข้อมูล (Trend and Seasonal)

การทดสอบแนวโน้ม (Mann-Kendall Test)

สมมติฐานการทดสอบ

H_0 : อนุกรมเวลาไม่มีอิทธิพลของแนวโน้ม

H_1 : อนุกรมเวลาที่มีอิทธิพลของแนวโน้ม

สถิติทดสอบ

$$S = \sum_{i=1}^{n-1} \sum_{j=i+1}^n \text{sgn}(x_j - x_i) \quad (2.3)$$

โดยที่ x_i คือ ค่าของตัวแปรสุ่ม ณ เวลาที่ i

x_j คือ ค่าของตัวแปรสุ่ม ณ เวลาที่ j

$$\text{sgn}(x_j - x_i) = \begin{cases} 1 & \text{If } (x_j - x_i) > 0 \\ 0 & \text{If } (x_j - x_i) = 0 \\ -1 & \text{If } (x_j - x_i) < 0 \end{cases} \quad (2.4)$$

ขอบเขตการตัดสินใจ

ปฏิเสธ H_0 เมื่อ $|z_S| > z_{\alpha/2}$ หรือ P-value น้อยกว่า 0.05

การทดสอบฤดูกาล (Kruskal-Wallis Test)

สมมติฐานการทดสอบ

H_0 : อนุกรมเวลาไม่มีอิทธิพลของฤดูกาล

H_1 : อนุกรมเวลาที่มีอิทธิพลของฤดูกาล

สถิติทดสอบ

$$H = \frac{12}{n(n+1)} \left[\sum_{i=1}^L \frac{R_i^2}{n_i} \right] - 3(n+1) \quad (2.5)$$

โดยที่ L คือ จำนวนของฤดูกาล

R_i คือ ผลรวมของอันดับ ณ ฤดูกาลที่ i

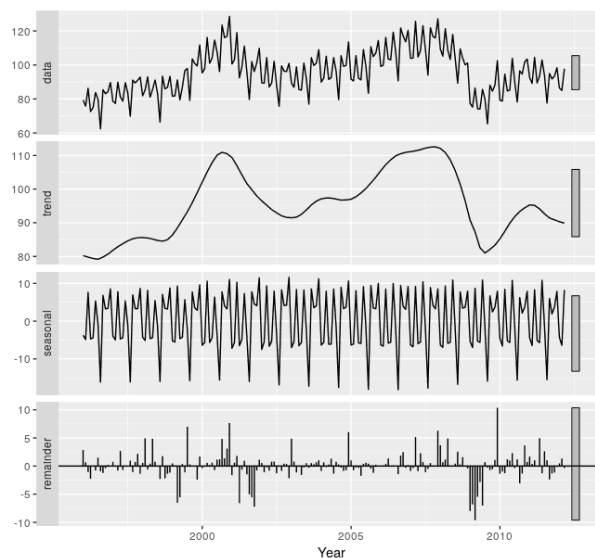
n_i คือ จำนวนของข้อมูล ณ ฤดูกาลที่ i

ขอบเขตการตัดสินใจ

ปฏิเสธ H_0 เมื่อ $H > \chi_{\alpha, L-1}^2$ หรือ P-value น้อยกว่า 0.05

การแยกส่วนองค์ประกอบของข้อมูล (Classical Decomposition Method)

การแยกส่วนองค์ประกอบของข้อมูล เป็นวิธีที่การพล็อตกราฟเพื่อแยกข้อมูลว่ามีองค์ประกอบของแนวโน้มและฤดูกาลโดยการใช้ฟังก์ชันแยกองค์ประกอบของข้อมูลออกมาเป็นกราฟ แสดงได้ดังภาพที่ 2.3



ภาพที่ 2.3 แสดงการแยกองค์ประกอบของข้อมูล

ที่มา : (<https://blog.datath.com/wp-content/uploads/2018/06/elecequip-stl-decomposition.png>)

2.2.4.2 การแจกแจงปกติของค่าคลาดเคลื่อนของข้อมูล (Normal Distribution)

เป็นการแจกแจงของข้อมูลรูปแบบหนึ่ง โดยข้อมูลมีการกระจายตัวรอบ ๆ ค่าเฉลี่ยหรือตำแหน่งตรงกลาง และค่อย ๆ ลดหลั่นไปทั้งซ้ายและขวาในลักษณะที่ใกล้เคียงหรือเท่ากัน โดยการทดสอบสามารถทดสอบได้การใช้สถิติทดสอบ Shapiro-Wilk test และการทำ Q-Q plot

การทดสอบโดย Shapiro-Wilk test

สมมติฐานการทดสอบ

H_0 : ค่าความคลาดเคลื่อนมีการแจกแจงแบบปกติ

H_1 : ค่าความคลาดเคลื่อนไม่มีการแจกแจงแบบปกติ

สถิติทดสอบ

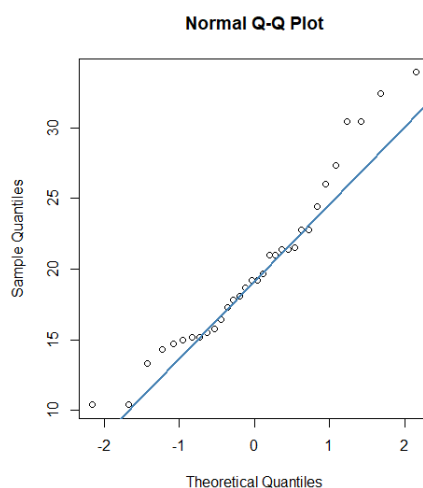
$$W = \frac{(\sum_{i=1}^n a_i x_{(i)})^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.6)$$

ขอบเขตการตัดสินใจ

ปฏิเสธ H_0 เมื่อ P-value น้อยกว่า 0.05

กราฟ Q-Q plot

เป็นการนำข้อมูลมาพล็อตกราฟเพื่อดูการแจกแจงปกติของข้อมูลเบื้องต้น โดยจะพิจารณาว่า ข้อมูลมีการกระจายตัวเกาะกลุ่มในแนวเส้นของกราฟ โดยแสดงดังภาพที่ 2.4



ภาพที่ 2.4 แสดงกราฟการกระจายตัวของข้อมูลโดยใช้ Q-Q plot

ที่มา : (<https://cdn.buttercms.com/3Av1YayDSkGttHUTuRf>)

2.2.5 วิธีการพยากรณ์อนุกรมเวลา

1. แบบจำลอง ARIMAX (Autoregressive Integrated Moving Average with exogenous)

เป็นตัวแบบ ARIMA(p,d,q) ที่มีการเพิ่มตัวแปรภายนอก (Exogenous variable) เข้ามาในตัวแบบ เพื่อเพิ่มประสิทธิภาพในการพยากรณ์ ซึ่งสามารถเขียนในรูปแบบสมการได้ดังนี้

$$\phi(B)\nabla^d Y_t = \theta(B)W_t + \sum_{h=1}^m (a_h X_{ht}) \quad (2.7)$$

หรือ

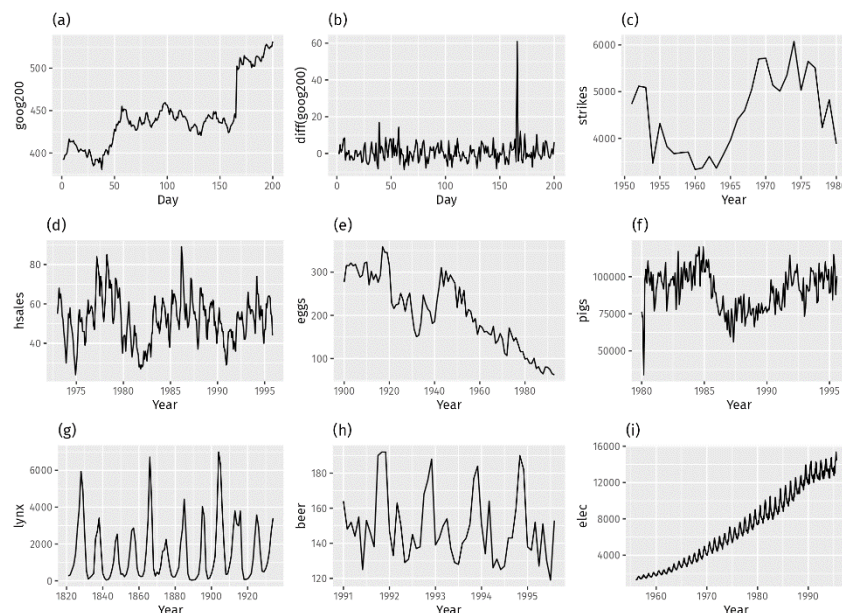
$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p)(1 - B)^d Y_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q)W_t + (a_1 X_{1t} + a_2 X_{2t} + \dots + a_m X_{mt}) \quad (2.8)$$

เมื่อ $X_{1t}, X_{2t}, \dots, X_{mt}$ คือ ตัวแปรภายนอก m ตัว ณ เวลาที่ t

และ a_1, a_2, \dots, a_m คือ ค่าสัมประสิทธิ์การถดถอยของตัวแปรภายนอก

1.1 ความนิ่งของข้อมูล (Stationary Data)

ข้อมูลที่มีลักษณะนิ่ง คือ ข้อมูลที่มีค่าเฉลี่ย (Mean) และค่าความแปรปรวน (Variance) คงที่ ในกรณี ที่เลือกสุ่มข้อมูลออกมาบางส่วนจากข้อมูลที่ได้เก็บรวบรวมไว้สำหรับการทดลอง ก่อนการสร้างแบบจำลองจึง ต้องมีการทำข้อมูลที่ไม่นิ่ง (Nonstationary) ให้เป็นข้อมูลที่นิ่งเสียก่อน (Stationary)



ภาพที่ 2.5 แสดงข้อมูล Stationary และ Nonstationary

ที่มา : (https://otexts.com/fpp2/fpp_files/figure-html/stationary-1.png)

จากภาพที่ 2.5 มีข้อมูลที่เป็น Stationary อยู่สองชุดข้อมูล คือ (b) และ (g) ส่วนที่เหลือเป็นข้อมูลที่เป็น Nonstationary โดยวิธีการทำให้ข้อมูลที่ไม่นิ่งเป็นข้อมูลที่นิ่งเหมาะสำหรับการสร้างแบบจำลอง สามารถทำได้โดยวิธีการ Differencing Data

ตารางที่ 2.1 วิธีการทำ Differencing Data

Date	Data	Diff1
1	10	
2	10.5	0.5
3	12.25	1.75
4	13.5	1.25
5	15	1.5
6	16	1

ตารางที่ 2.1 แสดงการ Differencing Data เพื่อให้ข้อมูลเป็น Stationary โดยทำการนำข้อมูลก่อนหน้ามาลบกับข้อมูลปัจจุบัน $10 - 10.5$ เท่ากับ 0.5 ในกรณีที่ทำ Differencing Data แล้วข้อมูลยังไม่เป็น Stationary Data ก็ยังสามารถทำ Differencing Data ได้อย่างรวดเร็ว ข้อมูลที่จะเป็น Stationary หรือไม่เป็น จะใช้สมการ Augmented Dickey-Fuller test ซึ่งเป็นหนึ่งในวิธี Unit root tests ในการทดสอบ

$$\Delta x_t = \gamma x_{t-1} + \sum_{i=1}^p \phi_i \Delta x_{t-i} + \varepsilon_t \quad (\text{random walk process}) \quad (2.9)$$

$$\Delta x_t = \alpha + \gamma x_{t-1} + \sum_{i=1}^p \phi_i \Delta x_{t-i} + \varepsilon_t \quad (\text{random walk with drift}) \quad (2.10)$$

$$\Delta x_t = \alpha + \beta_t + \gamma x_{t-1} + \sum_{i=1}^p \phi_i \Delta x_{t-i} + \varepsilon_t \quad (\text{random walk with linear and trend}) \quad (2.11)$$

สมมติฐานการทดสอบ

$$H_0 : \gamma = 0$$

$$H_1 : \gamma \neq 0$$

ถ้ายอมรับ H_0 คือ P-value มีค่ามากกว่าระดับนัยสำคัญ 0.05 แสดงว่า x_t ข้อมูลมีลักษณะไม่นิ่ง (Nonstationary) และถ้าปฏิเสธ H_0 คือ P-value มีค่าน้อยกว่าระดับนัยสำคัญ 0.05 แสดงว่า x_t ข้อมูลนั้นมีลักษณะนิ่ง (Stationary)

1.2 การทดสอบสหสัมพันธ์ในค่าคลาดเคลื่อนของข้อมูล

การทดสอบสหสัมพันธ์ในค่าคลาดเคลื่อนของข้อมูล (Ljung-Box Test)

สมมติฐานการทดสอบ

H_0 : ข้อมูลอนุกรมเวลา ณ เวลา $t + h$ ไม่มีความสัมพันธ์เชิงเส้นตรงกับข้อมูลอนุกรมเวลา ณ เวลา t

H_1 : ข้อมูลอนุกรมเวลา ณ เวลา $t + h$ มีความสัมพันธ์เชิงเส้นตรงกับข้อมูลอนุกรมเวลา ณ เวลา t

สถิติทดสอบ

$$Q^* = T(T+2) \sum_{k=1}^h (T-k)^{-1} r_k^2 \quad (2.12)$$

ขอบเขตการตัดสินใจ

ปฏิเสธ H_0 เมื่อ $Q^* > \chi_{\alpha, L-1}^2$ หรือ P-value น้อยกว่า 0.05

1.2.1 ฟังก์ชันสหสัมพันธ์ในตัวเอง (Autocorrelation Function : ACF)

กำหนด X_1, X_2, \dots, X_n คืออนุกรมเวลาชุดหนึ่งที่คงที่ (Stationary) มีจำนวนข้อมูลเท่ากับ n ค่าสหสัมพันธ์ในตัวเอง ณ h ช่วงเวลาที่แล้วเขียนแทนสัญลักษณ์ $\rho(h)$ คำนวณได้จาก

$$\rho(h) = \frac{\text{cov}(X_t, X_{t+h})}{\sqrt{\text{var}(X_t)}\sqrt{\text{var}(X_{t+h})}} = \frac{E[(X_t - E(X_t))(X_{t+h} - E(X_{t+h}))]}{\sqrt{\text{var}(X_t)}\sqrt{\text{var}(X_{t+h})}} \quad (2.13)$$

เมื่ออนุกรมเวลามีความนิ่ง (Stationary) จะได้ว่า $E(X_t) = E(X_{t+h})$ และ $\text{var}(X_t) = \text{var}(X_{t+h})$ จะได้ว่า

$$\rho(h) = \frac{\text{cov}(X_t, X_{t+h})}{\text{var}(X_t)} = \frac{\gamma_h}{\gamma_0} \quad (2.14)$$

ACF เป็นค่าที่ใช้วัดความสัมพันธ์เชิงเส้นตรงของจุด 2 จุดเวลาที่ต่างกัน กล่าวคือค่า ACF ก็คือค่าสหสัมพันธ์ (correlation) ระหว่างอนุกรมเวลา ณ ช่วงเวลาปัจจุบัน (X_t) กับอนุกรมเวลาที่อยู่ห่างกัน h ช่วงเวลา ซึ่งอาจเป็น h ช่วงเวลาถัดไป (X_{t+h}) หรือ ณ h ช่วงเวลาก่อนหน้า (X_{t-h}) ก็ได้ และมีคุณสมบัติดังนี้

- $\rho(h)$ มีค่าอยู่ระหว่าง -1 ถึง 1

- $|\rho(h)|$ มีค่าเข้าใกล้ 1 แสดงว่าค่าสังเกตที่อยู่ห่างกัน h ช่วงเวลามีสหสัมพันธ์กันสูง
- $|\rho(h)|$ มีค่าเข้าใกล้ 0 แสดงว่าค่าสังเกตที่อยู่ห่างกัน h ช่วงเวลามีสหสัมพันธ์กันต่ำ
- $\rho(h) > 0$ แสดงว่าค่าสังเกตที่อยู่ห่างกัน h ช่วงเวลามีสหสัมพันธ์ในทิศทางเดียวกัน
- $\rho(h) < 0$ แสดงว่าค่าสังเกตที่อยู่ห่างกัน h ช่วงเวลามีสหสัมพันธ์ในทิศทางตรงกันข้าม
- $\rho(0) = 1$ เสมอ เนื่องจาก $\rho(0) = \frac{cov(X_t, X_t)}{var(X_t)} = \frac{var(X_t)}{var(X_t)} = 1$ หรือจะพิจารณาว่าค่าสังเกตที่อยู่ห่างกัน $h = 0$ ช่วงเวลา ก็คือค่าสังเกตตัวเองจะมีความสัมพันธ์กับตัวเองมากที่สุดซึ่งเท่ากับ 1 นั่นเอง

แต่เนื่องจากอนุกรมเวลาที่น่าสนใจมาพิจารณาเป็นเพียงตัวอย่างสุ่ม ดังนั้น ในการวิเคราะห์ข้อมูลอนุกรมเวลาใด ๆ จึงจะพิจารณาจากค่า Sample Autocorrelation Function (Sample ACF) ซึ่งอาจจะแตกต่างไปบ้างเมื่อเทียบกับ ACF จากทฤษฎี

การคำนวณค่า Sample ACF : $\hat{\rho}(h)$ นั้นมีจุดประสงค์เพื่อนำมาใช้ประมาณค่า Theoretical Autocorrelation Function : $\rho(h)$ และค่า Sample ACF ที่ได้นั้นจะถูกนำไปพล็อตลงใน “Correlogram” ซึ่งเป็นกราฟเพื่อใช้ประเมินว่าอนุกรมเวลาคงที่หรือไม่ นอกจากนี้ค่า Sample ACF ยังช่วยในการตัดสินใจเบื้องต้นว่าควรเลือกแบบจำลองของ Box-Jenskin ชนิดใดกับอนุกรมเวลาที่กำลังพิจารณาอยู่

1.2.2 ฟังก์ชันสหสัมพันธ์ในตัวเองบางส่วน (Partial Autocorrelation Function : PACF)

เป็นค่าที่ใช้วัดความสัมพันธ์เชิงเส้นตรงระหว่าง X_t กับ X_{t+h} โดยไม่มีอิทธิพลของ $X_{t+h}, \dots, X_{t+h-1}$ เข้ามาเกี่ยวข้อง เขียนแทนด้วยสัญลักษณ์ ϕ_{hh} , $h = 1, 2, 3, \dots$ เช่น ϕ_{33} เป็นความสัมพันธ์เชิงเส้นตรงระหว่าง X_t กับ X_{t-3} โดยไม่มีอิทธิพลของ X_{t-1} และ X_{t-2} เข้ามาเกี่ยวข้อง

$$\phi_{11} = \text{corr}(X_1, X_0) = \rho(1)$$

.

.

$$\phi_{11} = \text{corr}(X_h - X_h^{h-1}, X_0 - X_0^{h-1}) \quad , h = 2, 3, \dots \quad (2.15)$$

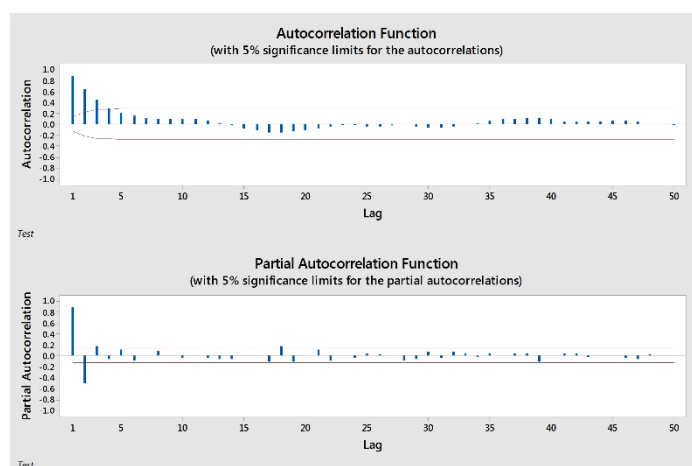
แต่เนื่องจากอนุกรมเวลาที่น่าสนใจมาพิจารณาเป็นเพียงตัวอย่างสุ่ม ดังนั้น ในการวิเคราะห์ข้อมูลอนุกรมเวลาใด ๆ จึงจะพิจารณาจากค่า Sample Partial Autocorrelation Function (Sample PACF) ซึ่งอาจแตกต่างไปบ้างเมื่อเทียบกับ PACF จากทฤษฎี ซึ่งค่า PACF ของตัวอย่างที่ได้ (Sample PACF : $\hat{\phi}_{hh}$) นั้นถูกนำไปพล็อตกราฟ PACF เพื่อช่วยในการตัดสินใจเลือกตัวแบบอนุกรมเวลาประกอบกับกราฟ Sample ACF

ซึ่งลักษณะของ ACF และ PACF ของตัวแบบ ARMA(p,q) สามารถสรุปได้ดังตารางที่ 2.2

ตารางที่ 2.2 แสดงลักษณะของ ACF และ PACF ของตัวแบบ ARMA

	AR (p)	MA(q)	ARMA(p,q)
ACF	มีค่าลดลงอย่างรวดเร็ว (Tail off)	สิ้นสุดหลังจาก q ช่วงเวลาที่แล้ว	มีค่าลดลงอย่างรวดเร็ว (Tail off)
PACF	สิ้นสุดหลังจาก p ช่วงเวลาที่แล้ว	มีค่าลดลงอย่างรวดเร็ว (Tail off)	มีค่าลดลงอย่างรวดเร็ว (Tail off)

ในการพล็อตกราฟของทั้งสองฟังก์ชันจะสร้างกราฟจากข้อมูลอนุกรมเวลา โดยนำค่าสหสัมพันธ์ (correlation) ของข้อมูลที่ถูกแบ่งเป็นส่วน ๆ โดยแต่ละส่วนจะถูกแบ่งในช่วง k หน่วยเวลา ซึ่งจะสร้างกราฟออกมาได้ดังภาพที่ 2.6



ภาพที่ 2.6 แสดงการพล็อตกราฟ ACF และ PACF

ที่มา : (https://blog.minitab.com/hubfs/Imported_Blog_Media/acf_pacf.jpg)

ซึ่งในการสรุปผลจากกราฟ สามารถสรุปได้ดังตารางที่ 2.3

ตารางที่ 2.3 แสดงการสรุปผลจากการอ่านกราฟ ACF และ PACF

ACF	PACF	Conclusion
มีค่าเข้าใกล้ 1 แบบคงที่ ไม่มี ลักษณะเรียวเล็กลง (taper)	มีค่าเข้าใกล้ 1 แบบคงที่ ไม่มี ลักษณะเรียวเล็กลง (taper)	ข้อมูลไม่มีความนิ่ง (Nonstationary) ต้องทำการ Differencing
ไม่มีความสัมพันธ์อย่างมีนัยสำคัญ	ไม่มีความสัมพันธ์อย่างมีนัยสำคัญ	ข้อมูลมีลักษณะแบบสุ่ม (Random Series)

1.3 เกณฑ์การคัดเลือกตัวแบบที่เหมาะสม

ในการพิจารณาแบบจำลองอนุกรมเวลาในบางครั้งอาจมีแบบจำลองที่เหมาะสมกับข้อมูลมากกว่า 1 ตัวแบบ ดังนั้นเครื่องมือที่ช่วยตัดสินใจว่ารูปแบบสมการใดเหมาะสมมากกว่ากัน สามารถพิจารณาได้จากค่า Akaike Information Criterion (AIC) และ ค่า Schwarz's Bayesian Information Criterion (BIC) โดยรูปแบบที่เหมาะสมกว่าจะมีค่า AIC และ BIC น้อยกว่า ซึ่งสรุปได้ว่าแบบจำลองสามารถใช้เป็นตัวแทนของข้อมูลจริงได้อย่างเหมาะสม

2. แบบจำลอง SVR (Support Vector Regression)

Support Vector Regression เป็นเทคนิคที่ใช้วิธีการของ Support Vector Machine (SVM) มาวิเคราะห์ความถดถอยระหว่าง Input vector และ Output variable ซึ่งนำมาใช้กับการพยากรณ์อนุกรมเวลาได้ โดยเปลี่ยนการจำแนกคลาสด้วย SVM เป็นการทำนายค่าด้วย SVR โดยมีเป้าหมายคือต้องการค้นหาความสัมพันธ์เชิงเส้นระหว่าง Input vector ในมิติขนาด n ($x \in R^n$) และ Output variable ($y \in R$) เนื่องจาก SVR ถูกดัดแปลงมาจาก SVM ดังนั้นสมการความถดถอยของ SVR จึงมีความคล้ายคลึงกับสมการ Hyperplane ของ SVM

$$f(x) = \langle w, x \rangle + b \text{ โดยที่ } w \in X, b \in \mathbb{R} \quad (2.16)$$

โดยที่

$\langle \cdot, \cdot \rangle$ คือ การทำ Dot product

w คือ ค่าน้ำหนักของ Support Vector

b คือ ค่าคงที่

ข้อมูลที่ใช้ในการพยากรณ์จะอยู่ในรูปแบบของ $\{(x_1, y_1), \dots, (x_l, y_l)\} \subset X \times \mathbb{R}$ โดยที่ X คือ ขนาดชนิดของข้อมูลนำเข้า และ \mathbb{R} คือ จำนวนจริง โดยเป้าหมายที่ต้องการ คือ การหาค่าของฟังก์ชัน $f(x)$ โดยสามารถหาค่าได้โดยใช้สมการของลากรานจ์ (Lagrange multipliers) ซึ่งผลลัพธ์ที่ได้ดังสมการ

$$f(x) = \sum_{i=1}^l (\alpha_i - \alpha_i^*) \langle x_i, x \rangle + b \quad \text{โดยที่ } \alpha_i \text{ และ } \alpha_i^* \text{ คือ ตัวคูณลากรานจ์} \quad (2.15)$$

แต่ในกรณีที่ไม่สามารถทำการพยากรณ์ข้อมูลได้ใน 2 มิติจะมีการนำเคอร์เนลเข้ามาช่วย โดยมีสมการ

$$f(x) = \sum_{i=1}^l (\alpha_i - \alpha_i^*) k(x_i, x) + b \quad \text{โดยที่ } k(x_i, x) \text{ คือ ค่าเคอร์เนล} \quad (2.17)$$

2.2.6 การหาค่าคลาดเคลื่อนของการพยากรณ์อนุกรมเวลา

1. ค่ารากที่สองของค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (Root Mean Squared Error : RMSE)

เป็นวิธีการวัดค่าความคลาดเคลื่อนแบบมาตรฐาน นิยมใช้กันอย่างแพร่หลาย โดยวัดผลความแม่นยำจากค่าที่แสดง โดยค่าที่ต่ำกว่าจะบ่งชี้ถึงประสิทธิภาพของแบบจำลองที่ดีกว่า โดยมีสมการดังนี้

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \quad (2.18)$$

โดย n คือ จำนวนข้อมูลที่ใช้
 y_t คือ ค่าจริงที่เวลา t ใด ๆ
 \hat{y}_t คือ ค่าที่ได้จากการพยากรณ์ที่เวลา t ใด ๆ

2. ค่าความคลาดเคลื่อนสัมบูรณ์เฉลี่ย (Mean Absolute Error : MAE)

เป็นตัวชี้วัดที่ใช้กันทั่วไปสำหรับการวัดความแม่นยำของแบบจำลองการถดถอย โดยจะวัดความแตกต่างสัมบูรณ์เฉลี่ยระหว่างค่าที่คาดการณ์และค่าจริงของตัวแปร คือค่าที่ไม่เป็นลบ โดยค่าที่ต่ำกว่าจะบ่งชี้ถึงประสิทธิภาพของแบบจำลองที่ดีกว่า โดยมีสมการดังนี้

$$MAE = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t| \quad (2.19)$$

โดย n คือ จำนวนข้อมูลที่ใช้
 y_t คือ ค่าจริงที่เวลา t ใด ๆ
 \hat{y}_t คือ ค่าที่ได้จากการพยากรณ์ที่เวลา t ใด ๆ

3. ค่าสัมบูรณ์ของเปอร์เซ็นต์ของความคลาดเคลื่อน (Mean Absolute Percentage Error : MAPE)

เป็นเมตริกที่ใช้กันทั่วไปสำหรับการวัดความแม่นยำของแบบจำลองการคาดการณ์ โดยจะวัดความแตกต่างของเปอร์เซ็นต์สัมบูรณ์เฉลี่ยระหว่างค่าจริงและค่าที่คาดการณ์ของอนุกรมเวลา MAPE

จะแสดงเป็นเปอร์เซ็นต์ โดยค่าที่ต่ำกว่าจะบ่งชี้ถึงประสิทธิภาพของแบบจำลองที่ดีขึ้น โดยมีสมการดังนี้

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{y_t - \hat{y}_t}{y_t} \right| \times 100 \quad (2.20)$$

โดย n คือ จำนวนข้อมูลที่ใช้
 y_t คือ ค่าจริงที่เวลา t ใด ๆ
 \hat{y}_t คือ ค่าที่ได้จากการพยากรณ์ที่เวลา t ใด ๆ

2.3 งานวิจัยที่เกี่ยวข้องกับการพยากรณ์

ประวีณา ศาลิคุปต์ และกิตติพันธ์ คงสวัสดิ์เกียรติ, 2556, "ปัจจัยที่กำหนดอัตราแลกเปลี่ยนระหว่างเงินบาทกับดอลลาร์สหรัฐอเมริกา (FACTORS DETERMINING THB/ USD EXCHANGE RATE)" โดยศึกษาความสัมพันธ์ของตัวแปรทั้ง 6 ตัวแปร ได้แก่ ดัชนีค่าเงินบาทที่แท้จริง (REER), อัตราดอกเบี้ย, อัตราการเปลี่ยนแปลงของปริมาณเงิน, อัตราการเปลี่ยนแปลงของทุนสำรองระหว่างประเทศ, อัตราการเปลี่ยนแปลงของมูลค่าการค้าสุทธิระหว่างประเทศไทยกับสหรัฐอเมริกา และอัตราการเปลี่ยนแปลงของอุปสงค์ที่มีต่อดอลลาร์สหรัฐอเมริกา ที่มีผลต่ออัตราแลกเปลี่ยนระหว่างเงินบาทกับดอลลาร์สหรัฐอเมริกา โดยศึกษาข้อมูลทุติยภูมิแบบรายเดือนตั้งแต่เดือนมกราคม พ.ศ. 2547 ถึงเดือนตุลาคม พ.ศ. 2555 และวิเคราะห์ข้อมูลโดยใช้การวิเคราะห์การถดถอยพหุคูณ (MRA) ผลการวิจัยพบว่า การทดสอบโดยใช้การวิเคราะห์การถดถอยพหุคูณ ได้ค่าสัมประสิทธิ์สหสัมพันธ์พหุคูณ เท่ากับ 0.975 ประสิทธิภาพในการพยากรณ์ เท่ากับ ร้อยละ 95 และอิสระของความคลาดเคลื่อนมาตรฐานในการพยากรณ์ เท่ากับ 0.373 พบว่า ดัชนีเงินบาทที่แท้จริงและอัตราดอกเบี้ยมีความสัมพันธ์กับอัตราแลกเปลี่ยนระหว่างเงินบาทกับดอลลาร์สหรัฐอเมริกา ส่วนปัจจัยอื่น ๆ ไม่มีความสัมพันธ์กับอัตราแลกเปลี่ยนอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.05

วัลลภ คุ่มประดิษฐ์, 2561, “การพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อดอลลาร์สหรัฐด้วยเทคนิคซีพอร์ตเวกเตอร์แมชชีน” ทำการศึกษาโดยใช้แบบจำลอง SVM ในการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อดอลลาร์สหรัฐ และนำมาเปรียบเทียบประสิทธิภาพการพยากรณ์ด้วยแบบจำลอง Random Walk และแบบจำลอง ARIMA ผลการวิจัยพบว่า เมื่อพิจารณาความถูกต้องของทิศทางในการคาดการณ์ (Mean Directional Accuracy) แบบจำลอง SVM เกือบทุกกรณีมีประสิทธิภาพสูงกว่าแบบจำลอง ARIMA และแบบจำลอง Random Walk โดยได้ความแม่นยำของ Mean Directional Accuracy สูงที่สุดเท่ากับ 51.943% แต่หากพิจารณาด้วยค่า RMSE พบว่า แบบจำลอง ARIMA มีประสิทธิภาพสูงสุด รองลงมาคือแบบจำลอง SVM และแบบจำลอง Random Walk โดยได้ค่า RMSE ต่ำที่สุดเท่ากับ 0.004912313

กมลวรรณ สารพานิช, 2555, “การพยากรณ์ราคาน้ำมันดิบล่วงหน้าในตลาดฟิวเจอร์สในเม็กซิโกโดยวิธีอาร์มาและอาร์แมกซ์” เป็นการศึกษาราคาน้ำมันดิบโดยใช้แบบจำลอง ARIMA และแบบจำลอง ARIMAX แล้ว

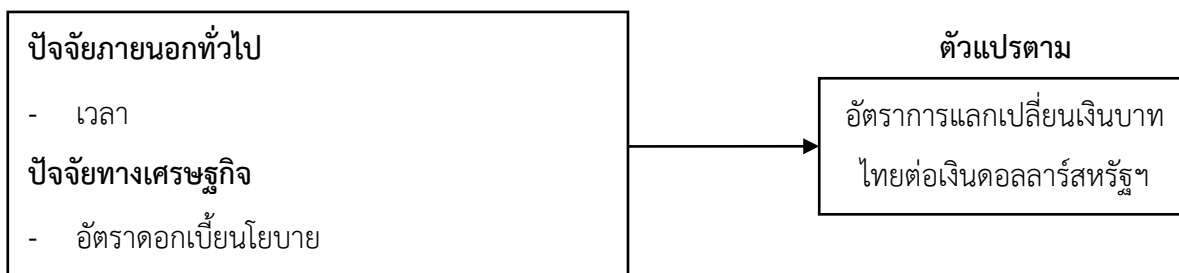
นำมาเปรียบเทียบประสิทธิภาพด้วยค่า RMSE ผลการวิจัยพบว่า เมื่อพิจารณาค่าคลาดเคลื่อนด้วย RMSE พบว่าแบบจำลอง ARIMA และแบบจำลอง ARIMAX ได้ค่า RMSE เท่ากับ 1.004 เท่ากัน จึงสรุปได้ว่าแบบจำลอง ARIMA และแบบจำลอง ARIMAX ให้ผลการพยากรณ์ที่มีความแม่นยำไม่แตกต่างกัน

ตารางที่ 2.4 แสดงข้อมูลงานวิจัยที่เกี่ยวข้อง

ชื่อ	ผู้แต่ง	ช่วง พยากรณ์	วิธีใช้	ตรวจสอบ	ผลการ วิเคราะห์
ปัจจัยที่กำหนดอัตราแลกเปลี่ยนระหว่างเงินบาทกับดอลลาร์สหรัฐอเมริกา	ประวีณา ศาลิคุปต์ และกิตติพันธ์ คงสวัสดิ์เกียรติ	-	MRA	-	correlation
การพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อดอลลาร์สหรัฐด้วยเทคนิคซัพพอร์ตเวกเตอร์แมชชีน	วัลลภ คุ่มประดิษฐ์	วัน	Random walk, ARIMA, Support Vector Machines	RMSE, Mean Directional Accuracy	ARIMA, Support Vector Machines
การพยากรณ์ราคาน้ำมันดิบล่วงหน้าในตลาดฟิวเจอร์สในเม็กซิโกโดยวิธีอาร์มาและอาร์แมกซ์	กมลวรรณ สารพานิช	สัปดาห์	ARIMA, ARIMAX	RMSE	ARIMA, ARIMAX

2.4 กรอบแนวคิดการวิจัย

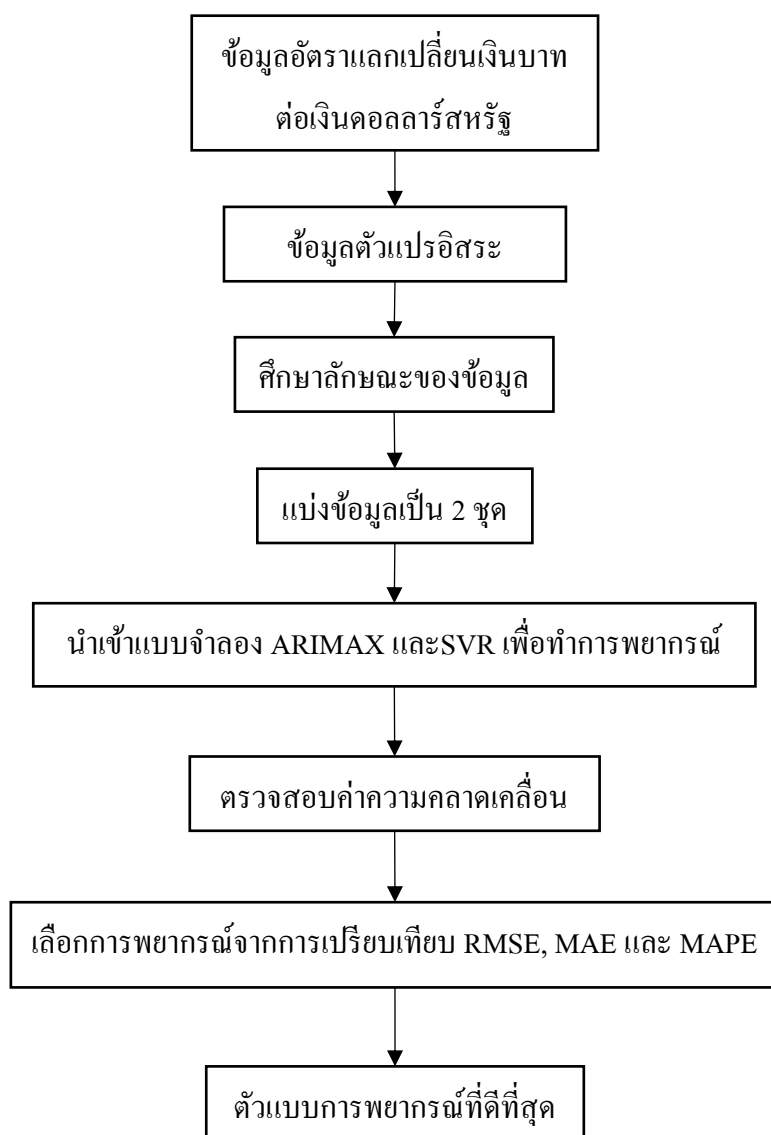
ตัวแปรอิสระ



บทที่ 3

การดำเนินงานวิจัย

ในงานวิจัยฉบับนี้ศึกษาการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ โดยใช้แบบจำลอง ARIMAX และแบบจำลอง SVR ซึ่งมีตัวแปรอิสระ 1 ตัวแปร คือ อัตราดอกเบี้ยนโยบาย แล้วเตรียมข้อมูลด้วยกระบวนการ Data Preprocessing เพื่อให้ข้อมูลมีความเหมาะสมต่อการพยากรณ์ ในการเลือกตัวแบบที่มีประสิทธิภาพที่สุดจะเลือกจากการนำผลพยากรณ์ไปคำนวณในวิธี RMSE, MAE และ MAPE และนำมาเปรียบเทียบกับระหว่าง 2 ตัวแบบเพื่อหาตัวแบบที่ให้ค่าความผิดพลาดน้อยที่สุด



ภาพที่ 3.1 วิธีการดำเนินการวิเคราะห์และสร้างตัวแบบของงานวิจัย

3.1 ชุดข้อมูลที่ใช้

3.1.1 รวบรวมข้อมูล

3.1.2 การเตรียมข้อมูล

3.1.3 การแบ่งข้อมูล

3.2 เครื่องมือที่ใช้ในการวิเคราะห์ข้อมูล

3.3 การวิเคราะห์ข้อมูลและสร้างแบบจำลองเพื่อเปรียบเทียบประสิทธิภาพ

3.3.1 การวิเคราะห์อนุกรมเวลาของข้อมูล

3.3.2 การตรวจสอบค่าคลาดเคลื่อนของข้อมูล

3.3.3 แบบจำลอง **ARIMAX**

3.3.4 แบบจำลอง SVR

3.3.5 การเปรียบเทียบประสิทธิภาพการพยากรณ์

3.1 ชุดข้อมูลที่ใช้

ในการวิจัยจะต้องทำการเก็บรวบรวมข้อมูลตัวแปรอิสระและตัวแปรตามที่ใช้ในการทำการวิเคราะห์ โดยนำข้อมูลมาทำความสะอาด แก้ไขปัญหาของข้อมูลให้เหมาะสมกับการวิเคราะห์ โดยมีกระบวนการดังนี้

3.1.1 รวบรวมข้อมูล

นำเข้าข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐจากธนาคารแห่งประเทศไทย (Bank of Thailand) ซึ่งเป็นข้อมูลทุดิถีภูมิและเป็นข้อมูลเชิงปริมาณ ระหว่างเดือนกุมภาพันธ์ พ.ศ. 2548 ถึงเดือนกุมภาพันธ์ พ.ศ.2567 จำนวนทั้งสิ้น 6,968 วันโดยรายละเอียดของข้อมูลมีดังนี้

ตารางที่ 3.1 โครงสร้างข้อมูลอัตราแลกเปลี่ยน THB/USD

Date	Close	Open	High	Low
2/22/2548	38.355	38.55	38.56	38.27
2/23/2548	38.49	38.34	38.519	38.34
2/24/2548	38.45	38.475	38.485	38.32
2/25/2548	38.36	38.44	38.465	38.34
2/28/2548	38.24	38.345	38.4	38.2
3/1/2548	38.275	38.23	38.33	38.23
3/2/2548	38.35	38.26	38.405	38.26
3/3/2548	38.46	38.31	38.485	38.31

จากตารางที่ 3.1 เป็นตัวอย่างบางส่วนของข้อมูลอัตราแลกเปลี่ยนรายวันของ THB/USD จากทั้งหมด โดยแต่ละหัวข้อมีรายละเอียดดังนี้

Date คือ วันที่ของอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ

Close คือ ราคาปิดของช่วงเวลานั้น

Open คือ ราคาเมื่อเปิดตลาดของช่วงเวลานั้น

High คือ ราคาสูงสุดของช่วงเวลานั้น

Low คือ ราคาต่ำสุดของช่วงเวลานั้น

ตารางที่ 3.2 โครงสร้างข้อมูลอัตราดอกเบี้ยนโยบาย

Date	Policy Rate
2/1/2548	2
3/1/2548	2.25
4/1/2548	2.25
5/1/2548	2.25
6/1/2548	2.5
7/1/2548	2.75
8/1/2548	2.75
9/1/2548	3.25

จากตารางที่ 3.2 เป็นตัวอย่างบางส่วนของข้อมูลอัตราดอกเบี้ยนโยบายรายเดือน นำข้อมูลมาจากรธนาคารแห่งประเทศไทย (Bank of Thailand) ระหว่างเดือนกุมภาพันธ์ พ.ศ. 2548 ถึงเดือนกุมภาพันธ์ พ.ศ. 2567 จำนวนทั้งสิ้น 229 เดือน

3.1.2 การเตรียมข้อมูล

1. การทำความสะอาดข้อมูล (Data Cleansing)

เนื่องจากข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐถูกเก็บข้อมูลในรูปแบบรายวัน แต่ข้อมูลอัตราดอกเบี้ยนโยบายถูกเก็บข้อมูลในรูปแบบรายเดือน จึงทำให้ข้อมูลไม่สามารถนำมาวิเคราะห์ร่วมกันได้ เพื่อกำจัดข้อมูลที่ไม่สมบูรณ์ จึงทำการจัดรูปแบบของข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐใหม่ให้อยู่ในรูปแบบรายเดือนโดยการหาค่าเฉลี่ยราคาเปิดรายเดือนในแต่ละเดือนมาเพื่อปรับให้ข้อมูลตัวแปรทั้ง 2 ตัวแปรมีรูปแบบเดียวกัน จะได้ค่าตัวอย่างตามตารางที่ 3.3

ตารางที่ 3.3 ตารางการแปลงข้อมูลอัตราแลกเปลี่ยน THB/USD จากรายวันเป็นรายเดือนโดยใช้ค่าเฉลี่ย

Date	Close	Open	High	Low
2/1/2548	38.4572	38.4595	38.56265	38.353
3/1/2548	31.83022	31.78848	31.91261	31.74457
4/1/2548	31.31864	31.33409	31.41273	31.26795
5/1/2548	31.26667	31.27571	31.34405	31.21833
6/1/2548	31.435	31.40591	31.50568	31.35977
7/1/2548	32.63591	32.60727	32.70295	32.55795
8/1/2548	33.08455	33.12295	33.20909	32.99682
9/1/2548	33.07545	33.01682	33.16205	32.92909
10/1/2548	33.43571	33.46976	33.57452	33.35619
11/1/2548	33.09091	33.075	33.20045	32.98136

2. การลดข้อมูล (Data Reduction)

เนื่องจากชุดข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐมีการรวบรวมข้อมูลที่ไม่จำเป็นต่อการวิเคราะห์ จึงทำการลบสถิติที่ไม่ต้องการเพื่อเพิ่มความเร็วในการวิเคราะห์ข้อมูลให้รวดเร็วยิ่งขึ้น ซึ่งข้อมูลที่เรานำมาวิเคราะห์เราใช้เป็นข้อมูลราคาปิด ณ เวลานั้นเท่านั้น เราจึงทำการลบชุดข้อมูลอื่นให้เหลือเพียง 2 สถิติดังนี้

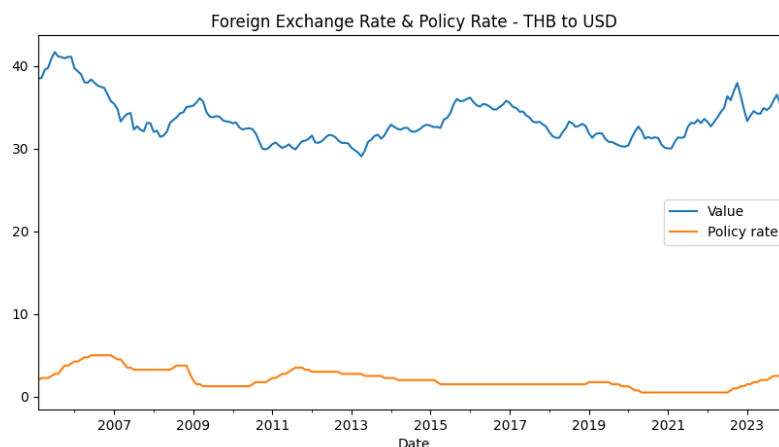
ตารางที่ 3.4 ชุดข้อมูลอัตราแลกเปลี่ยน THB/USD เฉพาะราคาปิดที่สนใจศึกษา

Date	Close
2/1/2548	38.4572
3/1/2548	31.83022
4/1/2548	31.31864
5/1/2548	31.26667
6/1/2548	31.435
7/1/2548	32.63591
8/1/2548	33.08455
9/1/2548	33.07545
10/1/2548	33.43571
11/1/2548	33.09091

เมื่อทำการลดข้อมูลที่ไม่จำเป็นออกแล้ว จากนั้นทำการรวมข้อมูลทั้งสองชุดเข้าด้วยกันเพื่อนำไปวิเคราะห์ต่อไป

ตารางที่ 3.5 ชุดข้อมูลอัตราแลกเปลี่ยน THB/USD เฉพาะราคาปิดและข้อมูลอัตราดอกเบี้ยนโยบาย

Date	Close	Policy Rate
2/1/2548	38.4572	2
3/1/2548	31.83022	2.25
4/1/2548	31.31864	2.25
5/1/2548	31.26667	2.25
6/1/2548	31.435	2.5
7/1/2548	32.63591	2.75
8/1/2548	33.08455	2.75
9/1/2548	33.07545	3.25



ภาพที่ 3.2 แสดงลักษณะของข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐเฉพาะราคาปิดรายเดือนและอัตราดอกเบี้ยนโยบาย

3.1.3 การแบ่งข้อมูล

ในการแบ่งข้อมูลสำหรับการวิเคราะห์ด้วยเทคนิค Walk Forward Analysis จะทำการแบ่งข้อมูลสำหรับชุดข้อมูลในการวิจัยออกเป็น 2 ส่วน ส่วนแรกคือข้อมูลที่เรียกว่า “In-Sample” หรือ “Training data” ที่นำมาใช้ในกระบวนการออกแบบและทดสอบแบบจำลอง ส่วนที่สองคือข้อมูล “Out-of-Sample” หรือ “Test data” ที่ถูกแยกออกมาเพื่อนำมาใช้ทดสอบความเสถียรของแบบจำลองที่ถูกออกแบบมาจากข้อมูล In-Sample โดยกำหนดกรอบช่วงเวลาของ Training Data เท่ากับ 5 เดือน และ Test Data เท่ากับ 1 เดือน แล้วทำการเลื่อนกรอบช่วงเวลาทั้งของ Training Data และ Test Data ไปข้างหน้าทีละช่วงจนครบ

3.2 เครื่องมือที่ใช้ในการวิเคราะห์ข้อมูล

ในการทำกรวิเคราะห์ข้อมูลและสร้างแบบจำลองนั้น จะใช้โปรแกรม Excel และโปรแกรม VSCode โดยมีรายละเอียดดังนี้

3.2.1 โปรแกรม Excel นำมาใช้ในการรวบรวมข้อมูล ทำความสะอาดข้อมูล ลดข้อมูลและรวมข้อมูล

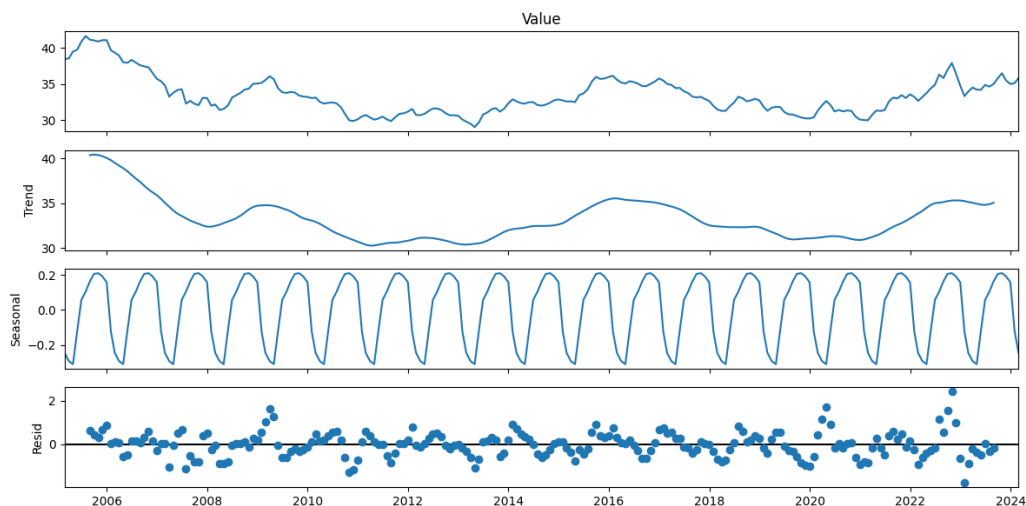
3.2.2 โปรแกรม VSCode (Visual Studio Code) เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไขและปรับแต่งโค้ด จากค่ายไมโครซอฟท์ ซึ่งเหมาะสำหรับพัฒนาโปรแกรมที่ต้องการใช้งานข้ามแพลตฟอร์ม โดยในงานวิจัยนำมาใช้ในการจัดการข้อมูล และสร้างแบบจำลองเพื่อหาตัวแบบที่ดีที่สุด โดยใช้ภาษา Python ซึ่งจะประกอบไปด้วย module ต่าง ๆ ดังนี้

1. Pandas ใช้สำหรับการวิเคราะห์ข้อมูลพื้นฐาน
2. Numpy ใช้สำหรับการคำนวณทางคณิตศาสตร์
3. sklearn.linear_model ใช้สำหรับการนำเข้าฟังก์ชัน LinearRegression
4. sklearn.model_selection ใช้สำหรับการนำเข้าฟังก์ชัน train_test_split
5. sklearn.preprocessing ใช้สำหรับการนำเข้าฟังก์ชัน StandardScaler
6. sklearn.svm ใช้สำหรับการนำเข้าฟังก์ชัน SVR
7. sklearn.metrics ใช้สำหรับการนำเข้าฟังก์ชัน mean_squared_error, mean_absolute_error, r2_score
8. matplotlib.pyplot ใช้สำหรับการ plot graph

3.3 การวิเคราะห์ข้อมูลและสร้างแบบจำลองเพื่อเปรียบเทียบประสิทธิภาพ

3.3.1 การวิเคราะห์อนุกรมเวลาของข้อมูล

ในการตรวจสอบค่าคลาดเคลื่อนของข้อมูลจะทำการแยกส่วนองค์ประกอบของข้อมูล เพื่อตรวจสอบว่าข้อมูลมีองค์ประกอบของแนวโน้มและฤดูกาล โดยสามารถแยกองค์ประกอบข้อมูลดังภาพที่ 3.3



ภาพที่ 3.3 แสดงองค์ประกอบของชุดข้อมูล

จากนั้นทำการทดสอบอนุกรมเวลา เพื่อตรวจสอบแนวโน้มและฤดูกาล

การทดสอบแนวโน้ม (Mann-Kendall Test)

สมมติฐานการทดสอบ

H_0 : อนุกรมเวลาไม่มีอิทธิพลของแนวโน้ม

H_1 : อนุกรมเวลาที่มีอิทธิพลของแนวโน้ม

ได้ค่า P-value เท่ากับ 0.018 ซึ่งน้อยกว่า 0.05 จึงปฏิเสธ H_0 สรุปได้ว่า อนุกรมเวลาที่มีอิทธิพลของแนวโน้ม

การทดสอบฤดูกาล (Kruskal-Wallis Test)

สมมติฐานการทดสอบ

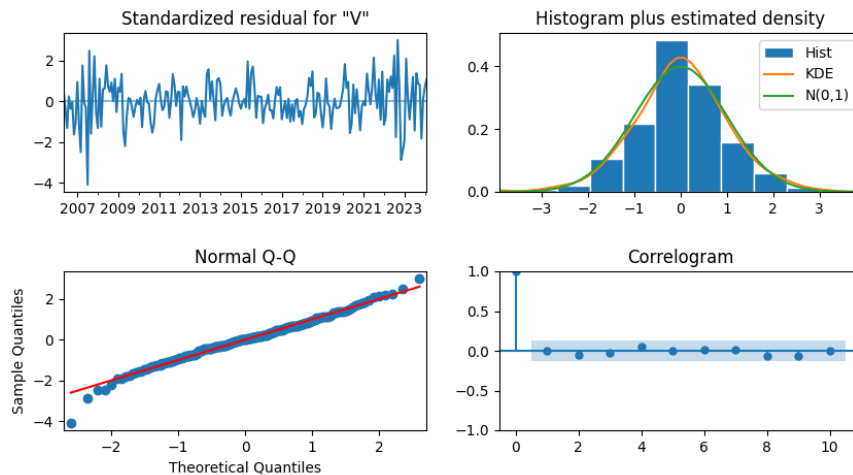
H_0 : อนุกรมเวลาไม่มีอิทธิพลของฤดูกาล

H_1 : อนุกรมเวลาที่มีอิทธิพลของฤดูกาล

ได้ค่า P-value เท่ากับ 1.12 ซึ่งมากกว่า 0.05 จึงยอมรับ H_0 สรุปได้ว่า อนุกรมเวลาไม่มีอิทธิพลของฤดูกาล

3.3.2 การตรวจสอบค่าคลาดเคลื่อนของข้อมูล

ในการทดสอบการแจกแจงแบบปกติของค่าคลาดเคลื่อนของข้อมูล ได้ทดลองพล็อตกราฟเพื่อดูกราฟแจกแจงเบื้องต้น แสดงดังภาพที่ 3.4



ภาพที่ 3.4 แสดงการแจกแจงของชุดข้อมูล

จากนั้นทำการทดสอบการแจกแจงปกติของค่าคลาดเคลื่อนของข้อมูล

การทดสอบโดย Shapiro-Wilk test

สมมติฐานการทดสอบ

H_0 : ค่าความคลาดเคลื่อนมีการแจกแจงแบบปกติ

H_1 : ค่าความคลาดเคลื่อนไม่มีการแจกแจงแบบปกติ

ได้ค่า P-value เท่ากับ 0.929 ซึ่งมากกว่า 0.05 จึงยอมรับ H_0 สรุปได้ว่า ค่าความคลาดเคลื่อนมีการแจกแจงแบบปกติ

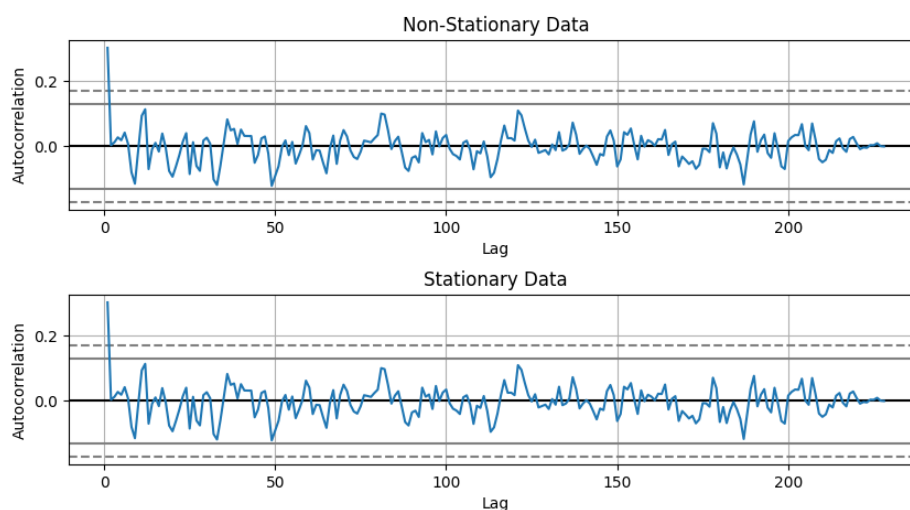
3.3.3 แบบจำลอง ARIMAX

ในการสร้างแบบจำลอง ARIMAX เพื่อให้แบบจำลองมีประสิทธิภาพที่สุดจะต้องทำการ Unit Root Test ด้วยวิธีการ ADF เพื่อตรวจสอบและทำให้ข้อมูลเป็น Stationary จากนั้นหาใช้เทคนิค AIC และ BIC เป็นดัชนีเพื่อกำหนดค่า p และ q ที่เหมาะสม จากนั้นประมาณค่าพารามิเตอร์เพื่อตรวจสอบว่าแบบจำลองมีความ

เหมาะสมกับชุดข้อมูลหรือไม่ เมื่อได้แบบจำลองที่เหมาะสมแล้วก็ทำการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ และประเมินประสิทธิภาพของแบบจำลองด้วยวิธี RMSE, MAE และ MAPE

3.3.3.1 การทดสอบ Unit Root Test

เป็นการตรวจสอบลักษณะของชุดข้อมูลนั้นว่า ข้อมูลมีความนิ่ง (Stationary) หรือไม่ โดยการดูจากรูปแบบของชุดข้อมูลและใช้วิธีการทดสอบ Augmented Dickey-Fuller (ADF)



ภาพที่ 3.5 แสดงความนิ่ง (Stationary) ของข้อมูล

จากภาพที่ 3.5 พบว่าชุดข้อมูลเป็นข้อมูลที่มีลักษณะนิ่ง (Stationary) ซึ่งไม่ต้องปรับค่าข้อมูล (Differencing Data) สามารถนำไปใช้ในการสร้างแบบจำลอง ARIMAX ต่อไป

ตารางที่ 3.6 ค่า ADF ที่ได้จากการทำ Unit Root Test

Augmented Dickey-Fuller	
p-value	1.479677e-17

จากตารางที่ 3.6 ค่า ADF ที่ได้จากการทำ Unit Root Test มีค่า p-value < 0.05 แสดงว่าชุดข้อมูลมีลักษณะนิ่ง (Stationary)

3.3.3.2 การทดสอบสหสัมพันธ์ในค่าคลาดเคลื่อนของข้อมูล (Ljung-Box Test)

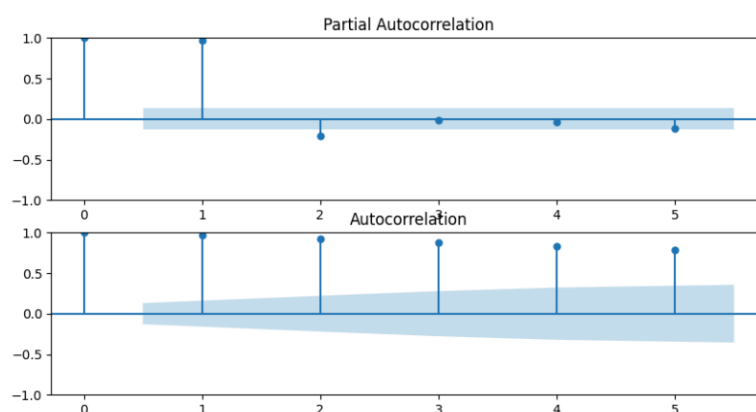
เป็นการตรวจสอบลักษณะของชุดข้อมูลนั้นว่า มีความสัมพันธ์เชิงเส้นตรง (Autocorrelation) หรือไม่ โดยการดูจากรูปแบบของชุดข้อมูลและใช้วิธีการ Ljung-Box Test

สมมติฐานการทดสอบ

H_0 : ข้อมูลอนุกรมเวลา ณ เวลา $t+h$ ไม่มีความสัมพันธ์เชิงเส้นตรงกับข้อมูลอนุกรมเวลา ณ เวลา t

H_1 : ข้อมูลอนุกรมเวลา ณ เวลา $t+h$ มีความสัมพันธ์เชิงเส้นตรงกับข้อมูลอนุกรมเวลา ณ เวลา t

ได้ค่า P-value เท่ากับ 0.9541 ซึ่งมากกว่า 0.05 จึงยอมรับ H_0 สรุปได้ว่า ข้อมูลอนุกรมเวลา ณ เวลา $t+h$ ไม่มีความสัมพันธ์เชิงเส้นตรงกับข้อมูลอนุกรมเวลา ณ เวลา t



ภาพที่ 3.6 แสดงความสัมพันธ์ (Autocorrelation) ของข้อมูล

3.3.3.3 การคัดเลือกตัวแบบที่เหมาะสมสำหรับแบบจำลอง ARIMAX

ในการพิจารณาแบบจำลอง ARIMAX จากค่า Akaike Information Criterion (AIC) และค่า Schwarz's Bayesian Information Criterion (BIC) เพื่อให้ได้ตัวแบบที่เหมาะสมที่สุด สามารถแสดงค่าของแต่ละตัวแบบได้ดังตารางที่ 3.7

ตารางที่ 3.7 แสดงค่า Akaike Information Criterion (AIC) และค่า Schwarz's Bayesian Information Criterion (BIC) ของแต่ละ ARIMA(p,d,q)

ARIMA(p,d,q)	AIC	BIC
ARIMA(0,0,0)	1466.343	1473.517
ARIMA(0,1,1)	409.953	417.120
ARIMA(0,1,2)	411.876	422.627
ARIMA(1,1,0)	412.609	419.776
ARIMA(1,1,1)	411.890	422.641
ARIMA(1,1,2)	411.732	426.066
ARIMA(2,1,2)	413.699	431.617

จากตารางที่ 3.7 จะพบว่าแบบจำลองที่ให้ค่า AIC และค่า BIC ต่ำที่สุดคือ แบบจำลอง ARIMA(0,1,1) จึงเลือกใช้แบบจำลองนี้สำหรับการสร้างแบบจำลอง ARIMAX เพื่อพยากรณ์อัตราแลกเปลี่ยน

3.3.4 แบบจำลอง SVR

วิเคราะห์ความถดถอยระหว่าง Input vector และ Output variable ซึ่งนำมาใช้กับการพยากรณ์อนุกรมเวลาได้ โดยหาความสัมพันธ์เชิงเส้นระหว่าง Input vector ในมิติขนาด n ($x \in R^n$) และ Output variable ($y \in R$) ข้อมูลที่ใช้ในการพยากรณ์จะอยู่ในรูปแบบของ $\{(x_1, y_1), \dots, (x_l, y_l)\} \subset X \times \mathbb{R}$ โดยที่ X คือ ขนาดชนิดของข้อมูลนำเข้า และ \mathbb{R} คือ จำนวนจริง โดยเป้าหมายที่ต้องการ คือ การหาค่าของฟังก์ชัน $f(x)$

3.3.5 การเปรียบเทียบประสิทธิภาพการพยากรณ์

การประเมินประสิทธิภาพผลการทดลองของแบบจำลองทั้งสองแบบจำลองจะใช้การเปรียบเทียบค่า RMSE, MAE และ MAPE และเลือกแบบจำลองที่ให้ค่าความคลาดเคลื่อนต่ำที่สุดเป็นแบบจำลองที่เหมาะสมต่อการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐโดยใช้ตัวแปรภายนอก นั่นคือ อัตราดอกเบี้ยนโยบาย

บทที่ 4

ผลการวิจัยและการอภิปราย

ในการศึกษาการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐโดยใช้ปัจจัยนอกเสริม นั่นคือ อัตราดอกเบี้ยนโยบาย ระหว่างเดือนกุมภาพันธ์ พ.ศ. 2548 ถึงเดือนกุมภาพันธ์ พ.ศ.2567 จำนวนทั้งสิ้น 229 เดือน โดยแบ่งข้อมูลเป็น 2 ส่วน คือ ส่วนชุดข้อมูลที่ใช้ในการทดลอง (กุมภาพันธ์ พ.ศ. 2548 - เมษายน พ.ศ. 2563) และส่วนการประเมินประสิทธิภาพ (พฤษภาคม พ.ศ. 2563 - กุมภาพันธ์ พ.ศ.2567) ซึ่งเลือกใช้แบบจำลองการพยากรณ์ 2 ตัวแบบ คือ แบบจำลอง ARIMAX (Autoregressive Integrated Moving Average with exogenous) และแบบจำลอง SVR (Support Vector Regression) ได้ทำการสร้างแบบจำลองเพื่อนำมาประเมินประสิทธิภาพ และสรุปผลได้ดังนี้

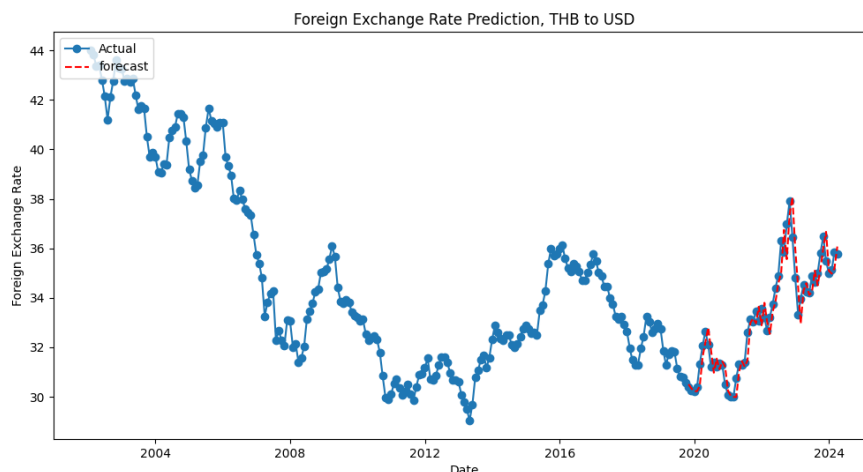
4.1 ผลการพยากรณ์จากแบบจำลอง ARIMAX (Autoregressive Integrated Moving Average with exogenous)

จากการทดสอบการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ ในระยะเวลา 1 เดือน โดยใช้ข้อมูลตั้งแต่เดือนกุมภาพันธ์ พ.ศ. 2548 ถึงเดือนกุมภาพันธ์ พ.ศ.2567 และเพิ่มตัวแปรภายนอก 1 ตัว คือ อัตราดอกเบี้ยนโยบาย ซึ่งในการใช้แบบจำลอง ARIMA(p,d,q) จะแทนค่าด้วยตัวแปร $p = 0$, $d = 1$, $q = 1$ ได้เป็นแบบจำลอง ARIMA(0,1,1) เมื่อนำมาพยากรณ์ 3 ช่วงเวลา ได้ผลการพยากรณ์ดังตารางที่ 4.3

ตารางที่ 4.1 ผลการพยากรณ์ 3 ช่วงเวลาโดยใช้ตัวแปรภายนอก

จำนวนเดือน พยากรณ์ ล่วงหน้า	ค่าจริง	ค่าพยากรณ์โดย ใช้ตัวแปร	RMSE	MAE	MAPE
1	35.133043	35.680255	0.54501	0.41567	1.24 %
2	35.852381	35.168520	0.54501	0.41567	1.24 %
3	35.852381	35.148830	0.54501	0.41567	1.24 %

จากตารางที่ 4.3 ผลการทดลองแสดงว่าการใช้ตัวแปรภายนอกส่งผลให้คลาดเคลื่อนของการพยากรณ์ RMSE เท่ากับ 0.54501, MAE เท่ากับ 0.41567 และ MAPE เท่ากับ 1.24 % ซึ่งความคลาดเคลื่อนน้อยมาก สามารถแสดงการพยากรณ์เปรียบเทียบได้ดังภาพที่ 4.1



ภาพที่ 4.1 แสดงการพยากรณ์ค่าจริงเทียบกับค่าพยากรณ์โดยใช้ตัวแปรภายนอกจากแบบจำลอง ARIMAX

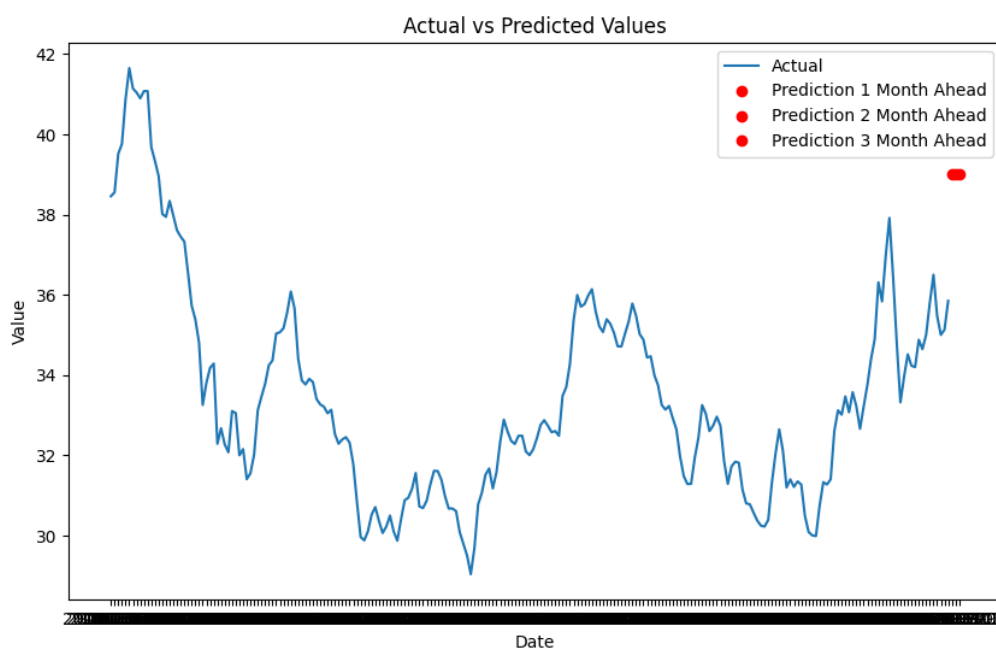
4.3. ผลการพยากรณ์จากแบบจำลอง SVR (Support Vector Regression)

จากการทดสอบการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐ ในระยะเวลา 1 เดือน โดยใช้ข้อมูลตั้งแต่เดือนกุมภาพันธ์ พ.ศ. 2548 ถึงเดือนกุมภาพันธ์ พ.ศ.2567 และเพิ่มตัวแปรภายนอก 1 ตัว คือ อัตราดอกเบี้ยนโยบาย ซึ่งในการสร้างแบบจำลอง SVR จะกำหนดพารามิเตอร์ $C = 1.0$ และ $\epsilon = 0.1$ และสามารถประเมินผลแบบจำลองด้วย RMSE, MAE และ MAPE โดยการใช้ดอกเบี้ยนโยบายเป็นตัวแปรภายนอก

ตารางที่ 4.4 ผลการพยากรณ์ 1 ช่วงเวลาโดยใช้แบบจำลอง SVR

จำนวนวัน พยากรณ์ล่วงหน้า	ค่าจริง	ค่าพยากรณ์	RMSE	MAE	MAPE
1	35.852381	39.024448	0.96868	0.65347	1.95 %

จากตารางที่ 4.4 จะได้ค่าความผิดพลาดจากการพยากรณ์โดยแบบจำลอง SVR ที่น้อยที่สุด คือ RMSE เท่ากับ 0.96868, MAE เท่ากับ 0.65347 และ MAPE เท่ากับ 1.95 % และแสดงการพยากรณ์เปรียบเทียบได้ดังภาพที่ 4.2



ภาพที่ 4.2 แสดงการพยากรณ์ค่าจริงเทียบกับค่าพยากรณ์โดยใช้ตัวแปรภายนอกจากแบบจำลอง SVR

4.4. เปรียบเทียบผลการพยากรณ์

จากผลการทดลองสร้างแบบจำลองทั้ง 2 แบบ นั่นคือ แบบจำลอง ARIMAX และแบบจำลอง SVR โดยหาค่าความคลาดเคลื่อนของทั้ง 2 ตัวแบบมาสรุปเพื่อเปรียบเทียบประสิทธิภาพเพื่อหาแบบจำลองที่ดีที่สุดได้ดังตารางที่ 4.6

ตารางที่ 4.6 เปรียบเทียบค่าความคลาดเคลื่อนของแบบจำลอง ARIMAX และแบบจำลอง SVR

แบบจำลอง	RMSE	MAE	MAPE
ARIMA(0,1,1) (ARIMAX)	0.54501	0.41567	1.24 %
SVR	0.96868	0.65347	1.95 %

จากตารางที่ 4.6 เมื่อนำค่าของแบบจำลองมาเปรียบเทียบกันแล้ว พบว่า แบบจำลองที่ให้ค่าความคลาดเคลื่อนน้อยที่สุด นั่นคือ ARIMA(0,1,1) มีค่า MAPE เท่ากับ 1.24 %

จากนั้นนำผลมาเปรียบเทียบกับการศึกษาก่อนหน้านี้ที่สร้างแบบจำลองโดยไม่ใช้ตัวแปรภายนอกเสริมได้ผลดังตารางที่ 4.7

ตารางที่ 4.7 เปรียบเทียบค่าความคลาดเคลื่อนของแบบจำลอง ARIMA และแบบจำลอง ARIMAX

แบบจำลอง	RMSE	MAE	MAPE
ARIMA(2,1,2) (ARIMA)	1.27548	1.0748	3.26 %
ARIMA(0,1,1) (ARIMAX)	0.54501	0.41567	1.24 %

จากตารางที่ 4.7 เมื่อนำค่าของแบบจำลองมาเปรียบเทียบกันแล้ว พบว่า แบบจำลองที่ให้ค่าความคลาดเคลื่อนน้อยที่สุด นั่นคือ ARIMA(0,1,1) มีค่า MAPE เท่ากับ 1.24 %

บทที่ 5

ผลการศึกษา

จากการศึกษาการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐโดยใช้ปัจจัยนอกเสริม นั้นคือ อัตราดอกเบี้ยนโยบาย โดยศึกษาชุดข้อมูลอัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐจากธนาคารแห่งประเทศไทย (Bank of Thailand) ซึ่งเป็นข้อมูลทุดิถีภูมิและเป็นข้อมูลเชิงปริมาณ ระหว่างเดือนกุมภาพันธ์ พ.ศ. 2548 ถึงเดือนกุมภาพันธ์ พ.ศ.2567 จำนวนทั้งสิ้น 229 เดือน ซึ่งใช้แบบจำลองในการพยากรณ์ 2 ตัวแบบ นั่นคือ แบบจำลอง Autoregressive Integrated Moving Average with exogenous (ARIMAX) และแบบจำลอง Support Vector Regression (SVR) เพื่อเปรียบเทียบผลการพยากรณ์ล่วงหน้า 1 เดือน โดยใช้การเปรียบเทียบประสิทธิภาพแบบจำลองด้วยค่า RMSE, MAE และ MAPE

5.1 สรุปผลการศึกษา

ในการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐโดยใช้ปัจจัยนอกเสริม โดยใช้แบบจำลองในการพยากรณ์ 2 ตัวแบบ นั่นคือ แบบจำลอง ARIMAX และแบบจำลอง SVR จากนั้นพิจารณาค่า RMSE, MAE และ MAPE ของตัวแบบ

ตารางที่ 5.1 เปรียบเทียบค่าความคลาดเคลื่อนของแต่ละแบบจำลอง

ข้อมูล	แบบจำลอง	RMSE	MAE	MAPE
ไม่ใช้ปัจจัยภายนอกเสริม	ARIMA(2,1,2)	1.27548	1.0748	3.26 %
ใช้ปัจจัยภายนอกเสริม	ARIMA(0,1,1)	0.54501	0.41567	1.24 %
	SVR	0.65347	0.96868	1.95 %

เมื่อนำผลพยากรณ์ที่ได้มาเปรียบเทียบกัน พบว่า แบบจำลองที่ให้ค่าคลาดเคลื่อนน้อยที่สุด คือแบบจำลอง ARIMA(0,1,1) หรือ แบบจำลอง ARIMAX ซึ่งให้ค่า MAPE เท่ากับ **1.24 %** ในกรณีที่ใช้ตัวแปรภายนอกเสริม และเมื่อนำมาเปรียบเทียบกับการศึกษาก่อนหน้านี้พบว่าแบบจำลอง ARIMAX ให้ค่าการพยากรณ์มีค่าความคลาดเคลื่อนเปอร์เซ็นต์ต่ำที่สุด

5.2 อภิปรายผล

จากการพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อเงินดอลลาร์สหรัฐโดยใช้ปัจจัยนอกเสริม นั่นคือ อัตราดอกเบี้ยนโยบาย เมื่อพิจารณาค่า RMSE, MAE และ MAPE ของตัวแบบ จะพบว่าแบบจำลองที่ดีที่สุดคือ ARIMA(0,1,1) หรือ แบบจำลอง ARIMAX โดยมีค่า MAPE เท่ากับ 1.24 % แต่เมื่อเปรียบเทียบในกรณีที่ไมใช้ปัจจัยภายนอกเสริม พบว่าแบบจำลอง ARIMAX ให้ค่าการพยากรณ์มีความคลาดเคลื่อนเปอร์เซ็นต์ต่ำที่สุด

5.3 ข้อเสนอแนะ

จากผลการวิจัย ผู้วิจัยได้รวบรวมปัญหาและข้อเสนอแนะเพื่อเป็นแนวคิดในการปรับปรุงวิธีการวิจัยการพยากรณ์ให้ดียิ่งขึ้น ได้ดังนี้

5.3.1 เพิ่มปัจจัยภายนอกที่สามารถคาดการณ์ในอนาคตได้ เพื่อให้โมเดลมีการพยากรณ์ได้แม่นยำมากยิ่งขึ้น

เอกสารอ้างอิง

- [1] ธนศักดิ์ ท่อนโพธิ์, 2564, การพยากรณ์อัตราแลกเปลี่ยนเงินตราโดยใช้ SARIMAX กับ ANN, วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต, สาขาวิชาวิทยาการคอมพิวเตอร์, คณะวิทยาศาสตร์, มหาวิทยาลัยมหาสารคาม.
- [2] ธนาคารแห่งประเทศไทย, “มาทำความรู้จักกับคำว่า “อัตราแลกเปลี่ยน” กัน” [Online], Available : <https://www.bot.or.th/our-roles/financial-markets/Foreign-Exchange-Market/Fx-hedging/What.html#:~:text=%E0%B8%AD%E0%B8%B1%E0%B8%95%E0%B8%A3%E0%B8%B2%E0%B9%81%E0%B8%A5%E0%B8%81%E0%B9%80%E0%B8%9B%E0%B8%A5%E0%B8%B5>

%E0%B9%88%E0%B8%A2%E0%B8%99%E0%B9%80%E0%B8%9B%E0%B9%87%E0%B8%99%E0%B8%95%E0%B8%B1%E0%B8%A7%E0%B9%81%E0%B8%9B%E0%B8%A3%E0%B8%AA%E0%B8%B3%E0%B8%84%E0%B8%B1%E0%B8%8D,%E0%B9%80%E0%B8%9E%E0%B8%B4%E0%B9%88%E0%B8%A1%E0%B8%82%E0%B8%B6%E0%B9%89%E0%B8%99%20%E0%B8%AB%E0%B8%A3%E0%B8%B7%E0%B8%AD%E0%B8%82%E0%B8%B2%E0%B8%94%E0%B8%97%E0%B8%B8%E0%B8%99%E0%B8%81%E0%B9%87%E0%B9%84%E0%B8%94%E0%B9%89, [10 สิงหาคม 2566].

[3] ชีร์วัช แก้ววิจิตร, 2559, **การเพิ่มประสิทธิภาพซอฟต์แวร์เรกเตอร์รีเกรสชันในการพยากรณ์อนุกรมเวลา**, วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต, สาขาวิชาวิศวกรรมคอมพิวเตอร์, คณะวิศวกรรม, มหาวิทยาลัยเทคโนโลยีสุรนารี.

[4] ประวีณา ศาลิคุปต์ และกิตติพันธ์ คงสวัสดิ์เกียรติ, 2556, “ปัจจัยที่กำหนดอัตราแลกเปลี่ยนระหว่างเงินบาทกับดอลลาร์สหรัฐอเมริกา”, **วารสารการเงิน การลงทุน การตลาด และการบริหารธุรกิจ**, เมษายน - มิถุนายน 2556, ปีที่ 3, ฉบับที่ 2.

[5] รณชัย ชื่นวัช, กิตติศักดิ์ เกิดประสพ และนิตยา เกิดประสพ, 2560, “การพยากรณ์ความต้องการใช้งานหน่วยจำหน่ายไฟฟ้าด้วยซอฟต์แวร์เรกเตอร์รีเกรสชันแบบตรวจสอบสลับ 3 ส่วน”, **วารสารวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยอุบลราชธานี**, มกราคม – เมษายน 2560, ปีที่ 19, ฉบับที่ 1.

[6] วัลลภ คุ่มประดิษฐ์, 2561, “การพยากรณ์อัตราแลกเปลี่ยนเงินบาทต่อดอลลาร์สหรัฐด้วยเทคนิคซอฟต์แวร์เรกเตอร์แมชชีน”, **การประชุมวิชาการระดับชาตินักเศรษฐศาสตร์**, ครั้งที่ 12, 13 กรกฎาคม 2561, หน้า 214-231.

[7] ศศิวุฒิ ไชยะเดชะ, 2563, “ARIMA Model ตอนที่ 2: หา Integrated (d) และ Stationary analysis” [Online], Available : <https://lengyi.medium.com/arima-model-%E0%B8%95%E0%B8%AD%E0%B8%99%E0%B8%97%E0%B8%B5%E0%B9%88-2-%E0%B8%AB%E0%B8%B2-integrated-d-%E0%B9%81%E0%B8%A5%E0%B8%B0-stationary-analysis-38df96394207>, [14 เมษายน 2563].

[8] ศศิวุฒิ ไชยะเดชะ, 2563, “Forecast ราคาน้ำมัน จากข้อมูลย้อนหลังด้วย SARIMA Model” [Online], Available : <https://lengyi.medium.com/sarima-oil-price-forecast-7f6006562214>, [18 กันยายน 2563].

- [9] ศศิวิทย์ ไชยะเดชะ, 2563, “ใช้ Support Vector Regression (SVR) ทำ Forward-looking model” [Online], Available : <https://lengyi.medium.com/support-vector-regression-forward-looking-model-9e9a0c8572e1>, [9 มิถุนายน 2563].
- [10] สถาบันนวัตกรรมและกรรมาภิบาลข้อมูล, 2566, “TimeSeries Data หรือ ข้อมูลอนุกรมเวลา คืออะไร” [Online], Available : <https://digi.data.go.th/blog/what-is-time-series-data/>, [16 สิงหาคม 2566].
- [11] อนุธิดา อนันต์ทรัพย์สุข, 2560, การเปรียบเทียบตัวแบบอนุกรมเวลาแบบผสมสำหรับการพยากรณ์ข้อมูลอนุกรมเวลาที่มีปัจจัยเชิงฤดูกาล, วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต, สาขาวิชาสถิติ, ภาควิชาสถิติ, คณะพาณิชยศาสตร์และการบัญชี, จุฬาลงกรณ์มหาวิทยาลัย.
- [12] Amazon, “การพยากรณ์คืออะไร” [Online], Available : <https://aws.amazon.com/th/what-is/forecast/>, [16 สิงหาคม 2566].
- [13] Appiah, S.T. and Adetunde, I.A., 2011, “Forecasting Exchange Rate Between the Ghana Cedi and the US Dollar using Time Series Analysis”, **Current Research Journal of Economic Theory**, August 2011, Vol. 3, No. 2.
- [14] Yasir, M. et al., 2019, “An Intelligent Event-Sentiment-Based Daily Foreign Exchange Rate Forecasting System”, **Applied Sciences**, July 2019, Vol.9, No. 15.

ภาคผนวก

Script code ARIMA & SARIMAX

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import os

import seaborn as sns
```

```
%matplotlib inline

import statsmodels.api as sm

from statsmodels.tsa.seasonal import seasonal_decompose

from statsmodels.tsa.arima.model import ARIMA

from statsmodels.tsa.statespace.sarimax import SARIMAX

from pmdarima import auto_arima

from statsmodels.graphics.tsaplots import plot_acf, plot_pacf

from pandas.plotting import autocorrelation_plot

from statsmodels.tsa.stattools import adfuller

from math import sqrt

from sklearn import preprocessing

from sklearn.metrics import r2_score, mean_absolute_error,
mean_absolute_percentage_error, mean_squared_error

import pickle

import warnings

warnings.filterwarnings('ignore')

os.chdir('C:\\Users\\...\\Final_PJ\\Data')

Forex = pd.read_csv('USDTHB_N.csv')

Forex.head()

Forex.shape

Forex.isnull().sum()

Forex.duplicated().sum()

Forex.dtypes
```

```
Forex.describe()

Forex['Date'] = pd.to_datetime(Forex['Date'])

Forex.set_index('Date',inplace = True)

Forex.plot(figsize = (10,5))

plt.title('Foreign Exchange Rate - THB to USD')

plt.savefig('Foreign Exchange Rate - THB to USD.png')

plt.show()

Forex_week = Forex.resample('W').mean()

print('Count of The Weekly Data Frame : ',Forex_week.shape[0])

Forex_week.head()

Forex_week.plot(figsize = (10,5))

plt.title('Foregin Exchange Rate (weekly) - THB to USD')

plt.savefig('Foregin Exchange Rate (weekly) - THB to USD.png')

plt.show()

Forex_month = Forex.resample('M').mean()

print('Count of The Monthly Data Frame : ',Forex_month.shape[0])

Forex_month.head()

Forex_month.plot(figsize = (10,5))

plt.title('Foregin Exchange Rate (Monthly) - THB to USD')

plt.savefig('Foregin Exchange Rate (Monthly) - THB to USD')

plt.show()

Forex_year = Forex.resample('Y').mean()
```

```

print('Count of The Yearly Data Frame : ',Forex_year.shape[0])

Forex_year.head()

Forex_year.plot(figsize = (10,5))

plt.title('Foregin Exchange Rate (Yearly) - THB to USD')

plt.savefig('Foregin Exchange Rate (Yearly) - THB to USD.png')

plt.show()

plt.rcParams['figure.figsize'] = (15,7)

sns.scatterplot(x = Forex_month.index , y = Forex_month.Value , color = 'black')

plt.title('Foreign Exchange Rate (weekly) - THB to USD [Scatter Plot]')

plt.savefig('Foreign Exchange Rate (weekly) - THB to USD [Scatter Plot].png')

plt.show()

sns.barplot(data = Forex_month,x = Forex_month.index , y = Forex_month.Value)

plt.title('Data Spread of Foreign Exchange Rate through Bar Plot')

plt.savefig('Data Spread of Foreign Exchange Rate through Bar Plot.png')

plt.show()

sns.distplot(Forex_month)

plt.title('Distribution of data in Foreign Exchange Rate (Weekly) - THB to USD')

plt.savefig('Distribution of data in Foreign Exchange Rate (Weekly) - THB to USD.png')

plt.show()

fig , (ax1,ax2) = plt.subplots(nrows = 2 ,ncols = 1,sharex = False , sharey = False , figsize =
(10,5))

Forex_month.hist(ax = ax1)

Forex_month.plot(kind = 'kde' , ax = ax2,c = 'r')

```

```

plt.title('Data Distribution of Foreign Exchange Rate')

plt.savefig('Data Distribution of Foreign Exchange Rate.png')

plt.show()

plt.rcParams['figure.figsize']=(12,6)

decomposition = seasonal_decompose(Forex_month , period = 12 , model = 'additive')

decomposition.plot()

plt.savefig('Discription , trend , seasonal , residuals.png')

plt.show()

fig , (ax1,ax2) = plt.subplots(nrows = 2 ,ncols = 1,sharex = False , sharey = False , figsize =
(10,5))

ax1 = plot_pacf(Forex_month , lags = 5 , ax = ax1)

ax2 = plot_acf(Forex_month , lags = 5 , ax = ax2)

plt.savefig('Partial Autocorrelation and Autocorrelation.png')

plt.show()

def adf_check(time_series):

    result = adfuller(time_series , autolag = 'AIC')

    label = pd.Series(result[0:4], index=["Test Statistic",'p-value','Number of Lags Used','Number
of Observations Used'])

    for key,value in result[4].items():

        label['Critical Value (%s)%key] = value

    print(label)

    if result[1] <= 0.05:

```

```
print('Strong evidence against the null hypothesis, hence REJECT null hypothesis and
the series is Stationary')
```

```
else:
```

```
print ('Weak evidence against the null hypothesis, hence ACCEPT null hypothesis and
the series is Not Stationary ')
```

```
adf_check(Forex_month)
```

```
Forex1_month = Forex_month.diff().dropna()
```

```
print('Count of monthlyly First Difference',Forex1_month.shape[0])
```

```
Forex1_month.head()
```

```
adf_check(Forex1_month)
```

```
Forex1_month.plot(figsize = (10,5))
```

```
plt.title('Foreign Exchange Rate Weekly (First Difference) - THB to USD')
```

```
plt.savefig('Foreign Exchange Rate Weekly(First Difference) - THB to USD.png')
```

```
plt.show()
```

```
fig , (ax1,ax2) = plt.subplots(nrows = 2 ,ncols = 1,sharex = False , sharey = False , figsize =
(10,5))
```

```
ax1 = autocorrelation_plot(Forex1_month , ax = ax1)
```

```
ax1.set_title('Non-Stationary Data')
```

```
ax2 = autocorrelation_plot(Forex1_month , ax = ax2)
```

```
ax2.set_title('Stationary Data')
```

```
plt.subplots_adjust(hspace = 0.5)
```

```
plt.savefig('Stationary data and Non-Stationary data.png')
```

```
plt.show()
```



```
model = auto_arma(Forex_month , m = 12, d = 1 ,seasonal = False , max_order = 8 , test =
'adf' , trace = True)
```

```
model.summary()
```

```
model = ARIMA(Forex_month , order = (0,1,1))
```

```
result = model.fit()
```

```
result.summary()
```

```
result.plot_diagnostics(figsize = (10,5))
```

```
plt.subplots_adjust(hspace = 0.5)
```

```
plt.savefig('Diagnostic plot of best model.png')
```

```
plt.show()
```

```
predictions = result.predict(typ = 'levels')
```

```
print('Evaluation Result for whole data : ','\n')
```

```
print('R2 Score for whole data :
```

```
{0:.2f} %'.format(100*r2_score(Forex_month['Value'],predictions)),'\n')
```

```
print('Mean Squared Error : ',mean_squared_error(Forex_month['Value'],predictions),'\n')
```

```
print('Mean Absolute Error : ',mean_absolute_error(Forex_month['Value'],predictions),'\n')
```

```
print('Root Mean Squared Error :
```

```
',sqrt(mean_squared_error(Forex_month['Value'],predictions)),'\n')
```

```
print('Mean Absolute Percentage Error :
```

```
{0:.2f} %'.format(100*mean_absolute_percentage_error(Forex_month['Value'],predictions)))
```

```
Final_data = pd.concat([Forex_month,Forex1_month,predictions],axis=1)
```

```
Final_data.columns = ['Foreign Exchange Rate (monthly)','Monthly First Difference','Predicted
Exchange Rate']
```

```
#Final_data.to_csv('Foreign Exchange Rate with Prediction (THB To USD).csv')
```

```

Final_data.head()

size = int(len(Forex_month)*0.80)

train , test = Forex_month[0:size]['Value'] , Forex_month[size:(len(Forex_month))]['Value']

print('Counts of Train Data : ',train.shape[0])

print('Counts of Test Data : ',test.shape[0])

train_values = [x for x in train]

prediction = []

print('Printing Predictied vs Expected Values....')

print('\n')

for t in range(len(test)):

    model = ARIMA(train_values , order = (0,1,1))

    model_fit = model.fit()

    output = model_fit.forecast()

    pred_out = output[0]

    prediction.append(float(pred_out))

    test_in = test[t]

    train_values.append(test_in)

    print('Predicted = %f , Actual = %f' % (pred_out , test_in))

print('Evaluation Result for Test data : ', '\n')

print('R2 Score for Test data : {0:.2f} %'.format(100*r2_score(test,prediction)), '\n')

print('Mean Squared Error : ',mean_squared_error(test,prediction), '\n')

print('Mean Absolute Error : ',mean_absolute_error(test,prediction), '\n')

```

```

print('Root Mean Squared Error : ',sqrt(mean_squared_error(test,prediction)),'\n')

print('Mean Absolute Percentage Error :
{0:.2f} %'.format(100*mean_absolute_percentage_error(test,prediction)))

predictions_df = pd.Series(prediction, index = test.index)

plt.rcParams['figure.figsize'] = (12,6)

fig, ax = plt.subplots()

ax.set(title='Foreign Exchange Rate Prediction, THB to USD', xlabel='Date', ylabel='Foreign
Exchange Rate')

ax.plot(Forex_month, 'o', label='Actual')

ax.plot(predictions_df, 'r', label='forecast')

legend = ax.legend(loc='upper left')

legend.get_frame().set_facecolor('w')

plt.savefig('Foreign Exchange Rate Prediction - THB to USD.png')

Pr = pd.read_csv('Policy_rate_data.csv')

Pr.head()

Pr.shape

Pr.isnull().sum()

Pr.duplicated().sum()

Pr.dtypes

Pr.describe()

Pr['Date'] = pd.to_datetime(Pr['Date'])

Pr.set_index('Date',inplace = True)

Pr.plot(figsize = (10,5))

```

```

plt.title('Policy Rate - THB')

plt.savefig('Policy Rate - THB to USD.png')

plt.show()

Pr_month = Pr.resample('M').mean()

print('Count of The Monthly Data Frame : ',Pr_month.shape[0])

Pr_month.head()

Pr_month.plot(figsize = (10,5))

plt.title('Policy Rate (Monthly) - THB')

plt.savefig('Policy Rate (Monthly) - THB to USD')

plt.show()

Pr_year = Pr.resample('Y').mean()

print('Count of The Yearly Data Frame : ',Pr_year.shape[0])

Pr_year.head()

Pr_year.plot(figsize = (10,5))

plt.title('Policy Rate (Yearly) - THB')

plt.savefig('Policy Rate (Yearly) - THB to USD.png')

plt.show()

plt.rcParams['figure.figsize'] = (15,7)

sns.scatterplot(x = Pr_month.index.to_numpy().ravel() , y = Pr_month.values.ravel() , color =
'black')

plt.title('Policy Rate (monthly) - THB [Scatter Plot]')

plt.savefig('Policy Rate (monthly) - THB to USD [Scatter Plot].png')

plt.show()

```

```

sns.barplot(data = Pr_month,x = Pr_month.index , y = Pr_month.values.ravel())

plt.title('Data Spread of Policy Rate through Bar Plot')

plt.savefig('Data Spread of Policy Rate through Bar Plot.png')

plt.show()

sns.distplot(Pr_month)

plt.title('Distribution of data in Policy Rate (Monthly) - THB')

plt.savefig('Distribution of data in Policy Rate (Monthly) - THB to USD.png')

plt.show()

fig , (ax1,ax2) = plt.subplots(nrows = 2 ,ncols = 1,sharex = False , sharey = False , figsize =
(10,5))

Pr_month.hist(ax = ax1)

Pr_month.plot(kind = 'kde' , ax = ax2,c = 'r')

plt.title('Data Distribution of Policy Rate')

plt.savefig('Data Distribution of Policy Rate.png')

plt.show()

plt.rcParams['figure.figsize']=(12,6)

decomposition = seasonal_decompose(Pr_month , period = 12 , model = 'additive')

decomposition.plot()

plt.savefig('Discription , trend , seasonal , residuals.png')

plt.show()

fig , (ax1,ax2) = plt.subplots(nrows = 2 ,ncols = 1,sharex = False , sharey = False , figsize =
(10,5))

ax1 = plot_pacf(Pr_month , lags = 5 , ax = ax1)

```

```

ax2 = plot_acf(Pr_month , lags = 5 , ax = ax2)

plt.savefig('Partial Autocorrelation and Autocorrelation.png')

plt.show()

def adf_check(time_series):

    result = adfuller(time_series , autolag = 'AIC')

    label = pd.Series(result[0:4], index=['Test Statistic','p-value','Number of Lags Used','Number
of Observations Used'])

    for key,value in result[4].items():

        label['Critical Value (%s)'%key] = value

    print(label)

    if result[1] <= 0.05:

        print('Strong evidence against the null hypothesis, hence REJECT null hypothesis and
the series is Stationary')

    else:

        print ('Weak evidence against the null hypothesis, hence ACCEPT null hypothesis and
the series is Not Stationary ')

adf_check(Pr_month)

Pr1_month = Pr_month.diff().dropna()

print('Count of monthly Frist Diference', Pr1_month.shape[0])

Pr1_month.head()

adf_check(Pr1_month)

Pr1_month.plot(figsize = (10,5))

plt.title('Policy Rate Monthly (Frist Diference) - THB')

```

```

plt.savefig('Policy Rate Monthly (Frist Diference) - THB.png')

plt.show()

fig , (ax1,ax2) = plt.subplots(nrows = 2 ,ncols = 1,sharex = False , sharey = False , figsize =
(10,5))

ax1 = autocorrelation_plot(Pr1_month , ax = ax1)

ax1.set_title('Non-Stationary Data')

ax2 = autocorrelation_plot(Pr1_month , ax = ax2)

ax2.set_title('Stationary Data')

plt.subplots_adjust(hspace = 0.5)

plt.savefig('Stationary data and Non-Stationary data.png')

plt.show()

model = auto_arma(Pr_month, m = 12, d = 1, seasonal = False, max_order = 8, test = 'adf',
trace = True)

model.summary()

model = ARIMA(Pr_month, order = (1,1,1))

result = model.fit()

result.summary()

result.plot_diagnostics(figsize = (10,5))

plt.subplots_adjust(hspace = 0.5)

plt.savefig('Diagnostic Policy Rate plot of best model.png')

plt.show()

predictions = result.predict(typ = 'levels')

print('Evaluation Result for whole data : ', '\n')

```

```

print('R2 Score for whole data : {0:.2f} %'.format(100*r2_score(Pr_month['Policy
rate'],predictions)),'\n')

print('Mean Squared Error : ',mean_squared_error(Pr_month['Policy rate'],predictions),'\n')

print('Mean Absolute Error : ',mean_absolute_error(Pr_month['Policy rate'],predictions),'\n')

print('Root Mean Squared Error : ',sqrt(mean_squared_error(Pr_month['Policy
rate'],predictions)),'\n')

print('Mean Absolute Percentage Error :
{0:.2f} %'.format(100*mean_absolute_percentage_error(Pr_month['Policy rate'],predictions)))

Final_data = pd.concat([Pr_month,Pr1_month,predictions],axis=1)

Final_data.columns = ['Policy Rate (monthly)','Monthly First Difference','Predicted Policy
Rate']

Final_data.to_csv('FPolicy Rate with Prediction (THB To USD).csv')

Final_data.head()

size = int(len(Pr_month)*0.80)

train , test = Pr_month[0:size]['Policy rate'] , Pr_month[size:(len(Pr_month))]['Policy rate']

print('Counts of Train Data : ',train.shape[0])

print('Counts of Train Data : ',test.shape[0])

train_values = [x for x in train]

prediction = []

print('Printing Predictied vs Expected Values....')

print('\n')

for t in range(len(test)):

    model = ARIMA(train_values , order = (1,1,1))

```



```

model_fit = model.fit()

output = model_fit.forecast()

pred_out = output[0]

prediction.append(float(pred_out))

test_in = test[t]

train_values.append(test_in)

print('Predicted = %f , Actual = %f' % (pred_out , test_in))

print('Evaluation Result for Test data : ', '\n')

print('R2 Score for Test data : {0:.2f} %'.format(100*r2_score(test,prediction)), '\n')

print('Mean Squared Error : ', mean_squared_error(test,prediction), '\n')

print('Mean Absolute Error : ', mean_absolute_error(test,prediction), '\n')

print('Root Mean Squared Error : ', sqrt(mean_squared_error(test,prediction)), '\n')

print('Mean Absolute Percentage Error :
{0:.2f} %'.format(100*mean_absolute_percentage_error(test,prediction)))

predictions_df = pd.Series(prediction, index = test.index)

plt.rcParams['figure.figsize'] = (12,6)

fig, ax = plt.subplots()

ax.set(title='Policy Rate Prediction, THB', xlabel='Date', ylabel='Policy Rate')

ax.plot(Pr_month, 'o', label='Actual')

ax.plot(predictions_df, 'r', label='forecast')

legend = ax.legend(loc='upper left')

legend.get_frame().set_facecolor('w')

plt.savefig('Policy Rate Prediction - THB to USD.png')

```

```
print(Forex.columns)

print(Pr.columns)

merged_df = Forex_month.merge(Pr_month, how='inner', on='Date')

merged_df.head()

merged_df.shape

merged_df.isnull().sum()

merged_df.duplicated().sum()

merged_df.dtypes

merged_df.describe()

merged_df.plot(figsize = (10,5))

plt.title('Foreign Exchange Rate & Policy Rate - THB to USD')

plt.savefig('Foreign Exchange Rate & Policy Rate - THB to USD')

plt.show()

plt.rcParams['figure.figsize'] = (15,7)

sns.scatterplot(x = merged_df.index , y = merged_df.Value , color = 'black')

plt.title('Foreign Exchange Rate & Policy Rate - THB to USD [Scatter Plot]')

plt.savefig('Foreign Exchange Rate & Policy Rate - THB to USD [Scatter Plot].png')

plt.show()

sns.barplot(data = merged_df, x = merged_df.index , y = merged_df.Value)

plt.title('Data Spread of Foreign Exchange Rate & Policy Rate through Bar Plot')

plt.savefig('Data Spread of Foreign Exchange Rate & Policy Rate through Bar Plot.png')

plt.show()
```

```

print(merged_df.Value)

sns.distplot(merged_df)

plt.title('Distribution of data in Foreign Exchange Rate & Policy Rate - THB to USD')

plt.savefig('Distribution of data in Foreign Exchange Rate & Policy Rate - THB to USD.png')

plt.show()

fig , (ax1,ax2) = plt.subplots(nrows = 2, ncols = 1, sharex = False, sharey = False, figsize =
(10,7))

merged_df.Value.hist(ax = ax1)

merged_df.Value.plot(kind = 'kde' , ax = ax2,c = 'r')

plt.title('Data Distribution of Foreign Exchange Rate & Policy Rate')

plt.savefig('Data Distribution of Foreign Exchange Rate & Policy Rate.png')

plt.show()

plt.rcParams['figure.figsize']=(12,6)

decomposition = seasonal_decompose(merged_df.Value , period = 12 , model = 'additive')

decomposition.plot()

plt.savefig('Discription , trend , seasonal , residuals.png')

plt.show()

fig , (ax1,ax2) = plt.subplots(nrows = 2 ,ncols = 1,sharex = False , sharey = False , figsize =
(10,5))

ax1 = plot_pacf(merged_df.Value , lags = 5 , ax = ax1)

ax2 = plot_acf(merged_df.Value , lags = 5 , ax = ax2)

plt.savefig('Partial Autocorrelation and Autocorrelation.png')

plt.show()

```

```

def adf_check(time_series):

    result = adfuller(time_series , autolag = 'AIC')

    label = pd.Series(result[0:4], index=["Test Statistic",'p-value','Number of Lags Used','Number
of Observations Used'])

    for key,value in result[4].items():

        label['Critical Value (%s)%key] = value

    print(label)

    if result[1] <= 0.05:

        print('Strong evidence against the null hypothesis, hence REJECT null hypothesis and
the series is Stationary')

    else:

        print ('Weak evidence against the null hypothesis, hence ACCEPT null hypothesis and
the series is Not Stationary ')

adf_check(merged_df.Value)

merged_df1 = merged_df.diff().dropna()

print('Count of merged policy rate', merged_df1.shape[0])

merged_df1.head()

adf_check(merged_df1['Value'])

adf_check(merged_df1['Policy rate'])

merged_df1.plot(figsize = (10,5))

plt.title('Foreign Exchange Rate with Policy rate')

plt.show()

```

```
fig, (ax1,ax2) = plt.subplots(nrows = 2, ncols = 1, sharex = False, sharey = False, figsize =
(10,5))
```

```
ax1 = autocorrelation_plot(merged_df1, ax = ax1)
```

```
ax1.set_title('Non-Stationary Data')
```

```
ax2 = autocorrelation_plot(merged_df1 , ax = ax2)
```

```
ax2.set_title('Stationary Data')
```

```
plt.subplots_adjust(hspace = 0.5)
```

```
plt.savefig('Stationary data and Non-Stationary data.png')
```

```
plt.show()
```

```
model_sarimax = auto_arima(merged_df['Value'],
                             exog = merged_df['Policy rate'],
                             m = 12, d = 1,
                             seasonal = True,
                             max_order = 8,
                             test = 'adf',
                             trace = True)
```

```
model sarimax.summary()
```

```
model_sarimax1 = SARIMAX(merged_df['Value'],  
                           order = (0, 1, 1),  
                           seasonal_order = (1, 0, 1, 12),  
                           exog = merged_df['Policy rate'],  
                           freq = 'M',  
                           enforce_stationarity=False,
```

```

        enforce_invertibility=False)

result_SARIMAX = model_sarimax1.fit(dis = False)

result_SARIMAX.summary()

result_SARIMAX.plot_diagnostics(figsize = (10,5))

plt.subplots_adjust(hspace = 0.5)

plt.savefig('Diagnostic plot of best SARIMAX model.png')

plt.show()

predictions_sarimax = result_SARIMAX.predict(typ = 'levels')

print('Evaluation Result for whole data : ', '\n')

print('R2 Score for whole data :
{0:.2f} %'.format(100*r2_score(merged_df['Value'],predictions_sarimax)), '\n')

print('Mean Squared Error : ', mean_squared_error(merged_df['Value'],predictions_sarimax), '\n')

print('Mean Absolute Error :
', mean_absolute_error(merged_df['Value'],predictions_sarimax), '\n')

print('Root Mean Squared Error :
', sqrt(mean_squared_error(merged_df['Value'],predictions_sarimax)), '\n')

print('Mean Absolute Percentage Error :
{0:.2f} %'.format(100*mean_absolute_percentage_error(merged_df['Value'],predictions_sarimax)))

Final_data = pd.concat([merged_df, merged_df1,

                        predictions_sarimax], axis=1)

Final_data.columns = ['Foreign Exchange Rate (monthly)',

                      'Polici Rate (monthly)',

                      'Monthly First Difference',

```

```

        'Predicted Policy Rate',
        'Predicted Exchange Rate']

Final_data.to_csv('Foreign Exchange Rate with Prediction (THB To USD).csv')

Final_data.head()

train_size = int(0.8 * len(merged_df))

test_size = len(merged_df) - train_size

train_set = merged_df[:train_size]

test_set = merged_df[train_size:]

print('Counts of Train Data : ',train.shape[0])

print('Counts of Test Data : ',test.shape[0])

print(train_set)

print(test_set)

train_values_Poli_add_Date = train_set.loc[train_set.index]

print(train_values_Poli_add_Date)

train_values = train_set['Value']

train_values

train_values_Poli = [y for y in train_set['Policy rate']]

train_values_Poli_copy = train_set.copy()

train_values_Poli = train_set['Policy rate'].to_frame()

prediction = []

print('Printing Predictied vs Expected Values....')

print('\n')

```

for t, value in enumerate(test):

 model = SARIMAX(endog = train_values,

 order = (0, 1, 1),

 seasonal_order = (1, 0, 1, 12),

 exog = train_values_Poli,

 freq = 'M',

 enforce_stationarity=False,

 enforce_invertibility=False)

model_fit = model.fit()

policy_model_arima = ARIMA(train_values_Poli['Policy rate'],

 order = (1,1,1))

policy_model_arima_fit = policy_model_arima.fit()

future_policy_rates_arima = policy_model_arima_fit.forecast()

output = model_fit.forecast(exog = future_policy_rates_arima)

pred_out = output[0]

prediction.append(float(pred_out))

train_values = []

train_values.append(test[t])

print('Predicted = %f, Actual = %f' % (pred_out, train_values[-1]))

Script code SVR Model

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import os

import seaborn as sns

%matplotlib inline

import statsmodels.api as sm

from statsmodels.tsa.seasonal import seasonal_decompose
```

```
from statsmodels.tsa.arima.model import ARIMA

from statsmodels.tsa.statespace.sarimax import SARIMAX

from pmdarima import auto_arima

from statsmodels.graphics.tsaplots import plot_acf , plot_pacf

from pandas.plotting import autocorrelation_plot

from statsmodels.tsa.stattools import adfuller

from math import sqrt

from sklearn import preprocessing

from sklearn.metrics import r2_score , mean_absolute_error ,
mean_absolute_percentage_error , mean_squared_error

from sklearn.preprocessing import StandardScaler

from sklearn.svm import SVR

from sklearn.model_selection import train_test_split

import pickle

import warnings

warnings.filterwarnings('ignore')

import os

os.chdir('C:\\Users\\...\\Final_PJ\\Data')

usdthb = pd.read_csv('Merged Data.csv')

usdthb.head()

usdthb['Date'] = pd.to_datetime(usdthb['Date'])

usdthb.set_index('Date',inplace = True)

model = sm.OLS(usdthb['Value'], sm.add_constant(usdthb['Policy rate']))
```

```

results = model.fit()

print('R_Square:', format(results.rsquared))

print('MSE:', format(results.mse_resid))

print(results.summary())

X = usdthb['Policy rate']

y = usdthb['Value']

print(y)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

X_train = pd.DataFrame(X_train)

y_train = pd.DataFrame(y_train)

model = SVR()

model.set_params(C=1.0, epsilon=0.1)

model.fit(X_train, y_train)

X_test = pd.DataFrame(X_test)

y_pred = model.predict(X_test)

print('Evaluation Result for whole data : ', '\n')

print('R2 Score for whole data : {0:.2f} %'.format(100*r2_score(y_test, y_pred)), '\n')

print('Mean Squared Error : ', mean_squared_error(y_test, y_pred), '\n')

print('Mean Absolute Error : ', mean_absolute_error(y_test, y_pred), '\n')

print('Root Mean Squared Error : ', sqrt(mean_squared_error(y_test, y_pred)), '\n')

print('Mean Absolute Percentage Error :
{0:.2f} %'.format(100*mean_absolute_percentage_error(y_test, y_pred)))

predictions_df = pd.Series(y_pred, index = X_test.index)

```

```
print(predictions_df)

plt.rcParams['figure.figsize'] = (12,6)

x_ax = range(len(X_test))

plt.plot(x_ax, y_test, label = 'Actual', color = 'k', linestyle = '--')

plt.plot(x_ax, y_pred, label = 'SVR', color = 'b', linestyle = '-')

plt.ylabel('Foreign Exchange Rate')

plt.xlabel('Date')

plt.savefig('Foreign Exchange Rate SVR - THB to USD.png')
```