

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การพยากรณ์ราคาทองคำโดยใช้
อิตเดนมาร์คอฟโมเดล

FORECASTING GOLD PRICE USING
HIDDEN MARKOV MODELS



T146476



กท.
๑๖๒๓๗
๒๕๕๘

เลขหมู่.....
เลขทะเบียน 146476
วัน,เดือน,ปี 23 พ.ค. 25๖0

b. 12842126
i.

รายงานนี้เป็นส่วนหนึ่งของวิชาการศึกษาระดับ 2

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**FORECASTING GOLD PRICE USING
HIDDEN MARKOV MODELS**



**A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS OF THE COURSE
INDEPENDENT STUDY 2
MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2 / 2015

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2016

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบรับรองการศึกษาระดับ 2 (INDEPENDENT STUDY 2)

เรื่อง

การพยากรณ์ราคาทองคำโดยใช้

อัลกอริทึมการถดถอยแบบ

FORECASTING GOLD PRICE USING

HIDDEN MARKOV MODELS

นายอนันต์ วงศ์กิตติพร

รหัสประจำตัว 56606028

ขอรับรองว่ารายงานฉบับนี้ ข้าพเจ้าไม่ได้คัดลอกมาจากที่ใด

รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของ

การศึกษาระดับปริญญาโท สาขาวิชาเศรษฐศาสตร์บัณฑิต (เทคโนโลยีสารสนเทศ)

ภาคเรียนที่ 2 ปีการศึกษา 2558

.....อาจารย์ที่ปรึกษา

(ผศ.ดร.ศุภกิจ นุตตะสฤต)

.....กรรมการสอบ

(ผศ.ดร.ศุภเมศ ไร่ระภาวัต)

.....กรรมการสอบ

(วศ.ดร.พรฤดี เนติโสภานกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	การพยากรณ์ราคาทองคำโดยใช้ฮิตเดนมาร์คอฟโมเดล
นักศึกษา	นายอานนท์ วงศ์กิตติพร
รหัสนักศึกษา	56606028
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	เทคโนโลยีระบบสารสนเทศ
ปีการศึกษา	2558
อาจารย์ที่ปรึกษา	ศศ.ดร.สุภกิจ นุตยะสกุล

บทคัดย่อ

ปัจจุบันการลงทุนในตลาดทองคำมีความนิยมมากขึ้น ถึงแม้ว่าการลงทุนจะมีความเสี่ยงสูง แต่การลงทุนที่มีความเสี่ยงสูงย่อมมีโอกาสที่จะได้ผลตอบแทนที่สูงเช่นเดียวกัน ในการลงทุนแต่ละครั้ง นักลงทุนควรศึกษาข้อมูลก่อนการลงทุน เพื่อลดความเสี่ยงลง จึงได้นำข้อมูลราคาทองคำย้อนหลังมาแปลงเป็นข้อมูลเข้า และทำการแบ่งข้อมูลด้วย Window Sliding Algorithm ก่อนที่จะส่งให้ฮิตเดนมาร์คอฟทำการทำนายผล โดยจากผลการทดลองได้มีทำการเปรียบเทียบค่าที่ทำนายได้กับค่าข้อมูลจริงด้วยกราฟเส้น พบว่า กราฟทั้งสองมีความใกล้เคียงกันอย่างมาก และเมื่อทดสอบค่าความคลาดเคลื่อนด้วยวิธี MAPE พบว่า ค่าของ MAPE คือ 1.0727% ด้วยสถานะของโมเดลที่ 40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Forecasting Gold Price using Hidden Markov Model
Student	Mr.Anon Wongkittiporn
Student ID	56606028
Degree	Master of Science
Program	Information Technology
Major	Information System Technology
Academic Year	2015
Advisor	Dr. Supakit Nootyaskool

ABSTRACT

Current investment in the gold market has become more popular although there is a high risk investment. Gold prices, which have been modified to not be predictable.

In this report, we design Hidden Markov Model (HMM) to predict the gold price by using gold history data being input to train the model. In addition, the input data is converted to observation and uses Windows Sliding Algorithm to split the data before training HMM. The result in this experiment we compare prediction data and actual data with Linear graph, they are closely. Moreover, result of MAPE is 1.0727% at 40 hidden states.

กิตติกรรมประกาศ

การศึกษานิพนธ์ในหัวข้อเรื่อง “การพยากรณ์ราคาทองคำโดยใช้ฮิตเดนมาร์คอฟโมเดล” จะสำเร็จลุล่วงด้วยดีไม่ได้หากไม่ได้รับความอนุเคราะห์ช่วยเหลือจาก ผศ.ดร. สุภกิจ นุตยะสกุล อาจารย์ที่ปรึกษาการค้นคว้าอิสระที่คอยให้คำปรึกษา และคำแนะนำ อีกทั้ง นายวุฒิเชาวน์ เจริญทอง ที่ช่วยแนะนำแนวทาง ขั้นตอนในการพัฒนาโปรแกรม

อานนท์ วงศ์กิตติพร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ III ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
หน้า.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1.....	1
บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ในการศึกษา.....	1
1.3 ขอบเขตของการศึกษา.....	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.5 ขั้นตอนการดำเนินงาน.....	2
บทที่ 2.....	3
ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 ฮิดเดนมาร์คอฟโมเดล (Hidden Markov Model).....	3
2.2 กราฟแท่งเทียน.....	10
2.3 ข้อมูลอนุกรมเวลา.....	17
2.4 งานวิจัยที่เกี่ยวข้อง.....	17
บทที่ 3.....	19
ออกแบบการทดลอง.....	19
3.1 การแปลงข้อมูลเป็นข้อมูลการสังเกต.....	19
3.2 กระบวนการของฮิดเดนมาร์คอฟโมเดลในการทดลอง.....	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ IV ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3.3 การออกแบบขั้นตอนในการรู้จำและการทำนายผลในการทดลอง.....	24
3.4 การเตรียมข้อมูลเพื่อแปลงค่าผลของการทำนาย	25
บทที่ 4.....	27
4.1 การเตรียมทรัพยากรของการทดลอง	27
4.2 ผลการทดลอง	27
บทที่ 5.....	35
สรุปผลการทดลอง.....	35
5.1 สรุปผลการทดลอง.....	35
5.2 ปัญหาและอุปสรรค.....	35
5.3 แนวทางการพัฒนาต่อ.....	36
บรรณานุกรม	37
ภาคผนวก	38
รายละเอียดซอร์สโค้ด.....	38
ก. Class Program.cs.....	38
ข. Class HMMCalculate.cs.....	45
ค. SettingVar.cs.....	55
ง. ProbTestData.cs.....	56

สารบัญตาราง

ตารางที่	หน้า
2.1 รูปแบบของกราฟแท่งเทียน Bullish Pattern	12
2.2 รูปแบบของกราฟแท่งเทียน Bearish Pattern.....	15
3.1 ตารางแปลงกราฟแท่งเทียนเป็นตัวเลขข้อมูลการสังเกตที่ อัตราส่วนแท่งเทียนเป็น 100%.....	19
3.2 ตารางแปลงแผนภูมิแท่งเทียนเป็นตัวเลขข้อมูลการสังเกตตามอัตราส่วน.....	20
4.1 รายละเอียดของทรัพยากรในการทดลอง.....	27



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ VL ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

ภาพที่	หน้า
2.1 ความน่าจะเป็นในการเปลี่ยนสถานะของมาร์คอฟโมเดล	3
2.2 แบบจำลองฟูลติคอนเนคโทโปโลยี	5
2.3 แบบจำลองเลฟ-ไฮ โทโปโลยี	5
2.4 ลักษณะการคำนวณ ด้วย Forward Procedure	7
2.5 ลักษณะการคำนวณ เพื่อหาค่า β_{ti}	8
2.6 ลำดับเหตุการณ์คำนวณร่วมกันระหว่างวิธี Forward และ Backward	9
2.7 ส่วนประกอบของกราฟแท่งเทียน	11
2.8 รูปแบบของกราฟแท่งเทียน	11
2.9 รูปแบบจำลองของลักษณะข้อมูลเวลา	17
3.1 ตัวอย่างข้อมูล	22
3.2 Flow จำลองการรู้จำของฮิตเดนมาร์คอฟโมเดล	23
3.3 Flow จำลองการทำนายผลของฮิตเดนมาร์คอฟโมเดล	24
3.4 Flow ขั้นตอนการ Train และ Testing	24
3.5 การจัดการข้อมูลการสังเกตด้วย Window Sliding Algorithm	25
3.6 ตัวอย่างข้อมูลการจัดเรียง observation ควบคู่กับราคาทองคำ	25
3.7 ตัวอย่างการจัดกลุ่มข้อมูลที่ $\text{observation} = 4$	26
3.8 ตัวอย่างค่าเฉลี่ยราคาแยกตาม observation	26
4.1 กราฟผลการทำนายราคาทองคำ ที่ $N = 50$	28
4.2 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 50$	28
4.3 กราฟผลการทำนายราคาทองคำ ที่ $N = 40$	29
4.4 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 40$	29
4.5 กราฟผลการทำนายราคาทองคำ ที่ $N = 30$	30
4.6 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 30$	30

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันการลงทุนในตลาดทองคำมีความนิยมมากขึ้น ถึงแม้ว่าการลงทุนจะมีความเสี่ยงสูง แต่การลงทุนที่มีความเสี่ยงสูงย่อมมีโอกาสที่จะได้ผลตอบแทนที่สูงเช่นเดียวกัน ในการลงทุนแต่ละครั้ง นักลงทุนควรศึกษาข้อมูลก่อนการลงทุน เพื่อลดความเสี่ยงลง

ในการที่นักลงทุนจะทำการลงทุนเข้าสู่ตลาดทองคำหรือทำการขายเพื่อเกร็งกำไรนั้น นักลงทุนไม่อาจทราบได้เลยว่าราคาทองคำในตอนนั้นถึงจุดต่ำสุดหรือสูงสุดแล้วหรือยัง ราคาจะเกิดการกลับตัวเมื่อใด เพราะถ้าหากตัดสินใจผิดพลาดจะส่งผลให้เกิดการขาดทุนหรือการที่เสียโอกาสที่จะได้ผลกำไรที่สูงๆได้

จากความรู้เรื่องฮิดเดนมาร์คอฟโมเดล เราสามารถนำมาประยุกต์ใช้ในการทำนายข้อมูลได้ โดยใช้ทฤษฎีฮิดเดนมาร์คอฟโมเดล ร่วมกับชุดข้อมูลอนุกรมเวลา(Time Series) ซึ่งเป็นข้อมูลในอดีตในช่วงเวลาใดเวลาหนึ่ง มาพยากรณ์ราคาที่จะเกิดขึ้นในอนาคตเพื่อที่จะสามารถนำข้อมูลเหล่านี้ไปวิเคราะห์ทำนายข้อมูลราคาทองคำที่จะเกิดขึ้นในอนาคต เพื่อลดความเสี่ยงของนักลงทุนลงได้ โดยผลการพยากรณ์จะเป็นส่วนหนึ่งในการช่วยการตัดสินใจ เพื่อลดความเสี่ยง และเพิ่มโอกาสที่จะลงทุนเพื่อให้เกิดกำไร

1.2 วัตถุประสงค์ในการศึกษา

1. ศึกษาทฤษฎีและการทำงานของฮิดเดนมาร์คอฟโมเดล
2. ศึกษาทฤษฎีกราฟแท่งเทียนเพื่อนำมาใช้ร่วมกับทฤษฎีฮิดเดนมาร์คอฟโมเดล
3. เพื่อนำฮิดเดนมาร์คอฟโมเดลมาประยุกต์ใช้งานให้เกิดประโยชน์
4. ทดลองนำฮิดเดนมาร์คอฟโมเดลมาใช้ในการพยากรณ์ราคาทองคำ

1.3 ขอบเขตของการศึกษา

ศึกษาทฤษฎีฮิดเดนมาร์คอฟโมเดล(Hidden Markov Model) และศึกษาทฤษฎีกราฟแท่งเทียน แล้วแปลงเป็นลำดับข้อมูลการสังเกต(Observation sequence)เพื่อนำเข้าสู่ระบบเพื่อทำการสอน(Training) จากนั้นระบบจะต้องสามารถทำนายราคาทองคำได้ใกล้เคียงหรือตรงกับข้อมูลจริง (Actual Data)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้เกี่ยวกับทฤษฎีฮิดเดนมาร์คอฟแบบต่างๆที่ใช้ในการพยากรณ์
2. สามารถใช้ผลลัพธ์จากการพยากรณ์นำมาเป็นแนวทางเพื่อให้นักธุรกิจใช้ประกอบการตัดสินใจก่อนลงทุน
3. สามารถใช้เป็นข้อมูลสำหรับผู้ที่ศึกษาต่อไปในอนาคต

1.5 ขั้นตอนการดำเนินงาน

ระบบจัดการคำสั่งซื้อบรรจุภัณฑ์พร้อมภาพเสมือนจริง มีขั้นตอนการดำเนินงาน มีดังต่อไปนี้

1. ศึกษาทฤษฎีฮิดเดนมาร์คอฟโมเดล(Hidden Markov Model)
2. ศึกษาทฤษฎีกราฟแท่งเทียบ
3. ศึกษาวิธีการแปลงข้อมูลค่าลำดับการสังเกต(Observation Sequence) เพื่อนำมาใช้ในการคำนวณของทฤษฎีฮิดเดนมาร์คอฟโมเดล(Hidden Markov Model)
4. นำข้อมูลที่แปลงเป็นข้อมูลค่าลำดับการสังเกต (Observation Sequence) นำเข้าสู่ระบบเพื่อทำการสอน(Training)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

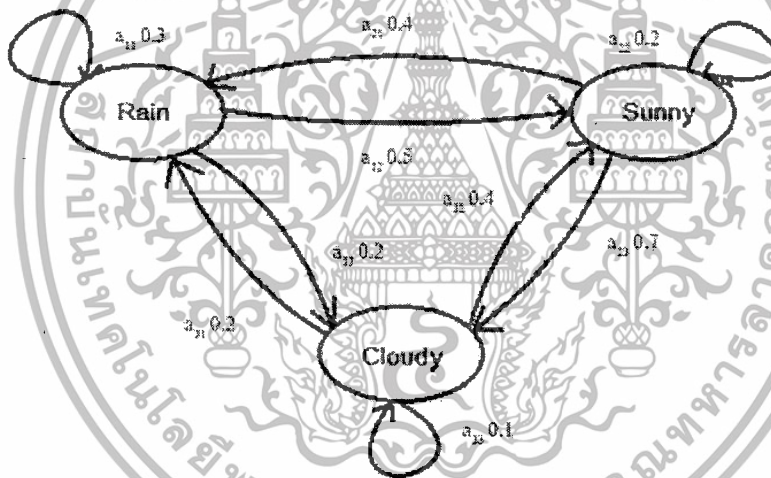
ทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องในการทำงานซึ่งประกอบไปด้วย ทฤษฎีฮิตเดน มาร์คอฟโมเดล(Hidden Markov Model) และทฤษฎีกราฟแท่งเทียน (Candlestick) โดยรายละเอียดของทฤษฎีที่จะนำเสนอมีดังนี้

2.1 ฮิตเดนมาร์คอฟโมเดล(Hidden Markov Model)

2.1.1 มาร์คอฟโมเดล(Markov Model)

มาร์คอฟโมเดลเป็นโมเดลที่ไม่สามารถรู้ถึงการเปลี่ยนแปลงสถานะที่แน่นอน จะรู้เพียงแต่ความน่าจะเป็นในการเปลี่ยนจากสถานะหนึ่งไปยังยังสถานะหนึ่งด้วยค่าความน่าจะเป็นเท่าไร



ภาพที่ 2.1 ความน่าจะเป็นในการเปลี่ยนสถานะของมาร์คอฟโมเดล

จากภาพที่ 2.1 จะเห็นว่าการเปลี่ยนสถานะจากสถานะหนึ่งนั้นทุกเส้นที่ออกจากสถานะนั้นจะต้องมีค่าความน่าจะเป็น รวมกันเป็น 1 เสมอ และสามารถเขียนให้อยู่ในรูปแบบของเมตริกซ์ A ได้โดย

$$A = \begin{matrix} & \begin{matrix} rain & sun & cloud \end{matrix} \\ \begin{matrix} rain \\ sun \\ cloud \end{matrix} & \begin{bmatrix} 0.5 & 0.3 & 0.2 \\ 0.4 & 0.4 & 0.2 \\ 0.7 & 0.2 & 0.1 \end{bmatrix} \end{matrix} \quad (2.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาค่าความน่าจะเป็นของลำดับ การเปลี่ยนสถานะของสิ่งที่เราสังเกต (Observation) O เมื่อ $O =$ แดด, ฝน, เมฆ ถ้าอากาศวันแรก ($t = 1$) คือ แดด โดยมีค่าเริ่มต้นของสถานะ $\pi_{\text{แดด}} = 1$ จะได้ว่า

$$P(O|Model) = 1(0.4)(0.4)(0.2) = 0.032 \quad (2.2)$$

2.1.2 อิตเดนมาร์คอฟโมเดล(Hidden Markov Model)

คือการเพิ่มข้อจำกัดของมาร์คอฟโมเดลคือการทำที่เราจะไม่สามารถสังเกตเห็นกระบวนการในการเปลี่ยนแปลงของสถานะได้โดยจะต้องสังเกตจากเหตุการณ์อื่นๆที่ส่งผลกระทบต่อกระบวนการเปลี่ยนแปลงของสถานะ ยกตัวอย่างเช่น สภาพอากาศที่เปลี่ยนแปลงไปในแต่ละวัน โดยสมมติให้สภาพอากาศมีการเปลี่ยนแปลงได้ 3 รูปแบบ คือ แดดออก(Sunny) เมฆมาก(Cloudy) และ ฝนตก(Rainy) ซึ่งสภาพอากาศไม่สามารถคาดการณ์ล่วงหน้าได้ว่าจะมีการเปลี่ยนแปลงไปอย่างไร เนื่องจากไม่สามารถคาดการณ์สภาพอากาศที่จะเปลี่ยนแปลงไปได้จึงต้องอาศัยการสังเกตจากปัจจัยอื่นที่ส่งผลกระทบต่อกระบวนการเปลี่ยนแปลงของสภาพอากาศ เช่นการสังเกตจากสภาพความชื้นของอากาศ

2.1.2.1 ส่วนประกอบของอิตเดนมาร์คอฟโมเดล

อิตเดนมาร์คอฟโมเดลมีพารามิเตอร์ที่เป็นส่วนประกอบในการสร้างลำดับข้อมูลการสังเกตได้ดังนี้

N คือ จำนวนของสถานะในโมเดล

M คือ จำนวนชนิดข้อมูลการสังเกต (Observation symbol) ต่อหนึ่งสถานะ

S คือ จำนวนสถานะต่างๆ โดยที่ $S = \{S_1, S_2, \dots, S_N\}$

Q คือ $\{q_t\}$ การที่สถานะอยู่ในช่วงเวลา t

T คือ จำนวนของลำดับข้อมูลการสังเกต(Observation Sequence)

V คือ จำนวนข้อมูลการสังเกต โดยที่ $V = \{V_1, V_2, \dots, V_M\}$

A คือ $\{a_{ij}\}$ การกระจายค่าความน่าจะเป็นในการเปลี่ยนสถานะโดย

$$a_{ij} = P[q_{t+1} = S_j | q_t = S_i], \quad 1 \leq i, j \leq N \quad (2.3)$$

B คือ $\{b_j(k)\}$ การกระจายความน่าจะเป็นของค่าข้อมูลการสังเกต โดยที่

$$b_j(k) = P[V_k \text{ ที่เวลา } t | q_t = S_j], \quad 1 \leq j \leq N, \quad 1 \leq k \leq M \quad (2.4)$$

π คือ $\{\pi_i\}$ การกระจายค่าเริ่มต้นของสถานะ โดยที่

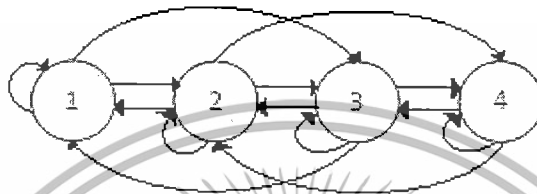
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\pi_i = P[q_1 = S_i], \quad 1 \leq i \leq N \quad (2.5)$$

λ คือ $\lambda = [A, B, \pi]$ การเขียนแบบย่อของค่าพารามิเตอร์ต่างที่เป็นส่วนประกอบโมเดล

2.1.2.2 ชนิดของอิตเดนมาร์คอฟโมเดล

- แบบจำลองฟูลลี คอนเนคโทโปโลยี (Fully Connect Topology)

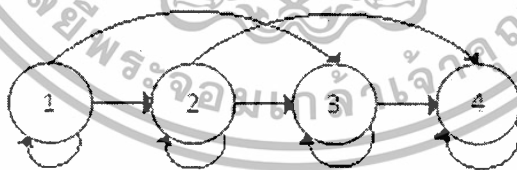


ภาพที่ 2.2 แบบจำลองฟูลลีคอนเนคโทโปโลยี

แบบจำลองฟูลลีคอนเนคโทโปโลยี คือแบบจำลองที่สถานะเริ่มต้นมีการเปลี่ยนไปเป็นสถานะอื่นๆ ได้ทุกสถานะในโมเดลรวมไปถึงตัวเองด้วย สามารถเขียนให้อยู่ในรูป เมทริกซ์ A ได้ดังนี้

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \quad (2.6)$$

- แบบจำลองเลฟ-ไล โทโปโลยี (Left-Right Topology)



ภาพที่ 2.3 แบบจำลองเลฟ-ไล โทโปโลยี

แบบจำลองเลฟ-ไล โทโปโลยี เป็นแบบจำลองที่มีโครงสร้างการเปลี่ยนสถานะจากซ้ายไปขวา หรือเปลี่ยนกลับมาเป็นสถานะเดิม มีสัมประสิทธิ์ในการย้ายสถานะดังนี้

$$a_{ij} = 0, \quad j < i \quad (2.7)$$

จะเห็นว่าแบบจำลองเลฟ-ไล โทโปโลยี ไม่มีการเปลี่ยนสถานะไปยังสถานะที่ต่ำกว่าสถานะปัจจุบัน และนอกจากนี้ความน่าจะเป็นของสถานะสถานะเริ่มต้นคือ

$$\pi_i = \begin{cases} 0, & i \neq 1 \\ 1, & i = 1 \end{cases} \quad (2.8)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับของสถานะจะต้องเริ่มที่ 1 เสมอ และแบบจำลอง เลข-ไค โทโปโลยี มีกฎข้อบังคับในการเปลี่ยนสถานะเพื่อไม่ให้มีการเปลี่ยนแปลงของดัชนีสถานะมากเกินไป คือ

$$a_{ij}=0, j < i + \Delta \quad (2.9)$$

จากภาพที่ 2.3 ถ้าค่า $\Delta=2$ หมายถึงว่าจะไม่มีการเปลี่ยนแปลงสถานะมากกว่า 2 สถานะ โดยสามารถเขียนให้อยู่ในรูปเมทริกซ์ A ได้ดังนี้

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.10)$$

โดยสัมประสิทธิ์ตัวสุดท้ายของโมเดลจะได้

$$a_{NN}=1 \quad (2.11)$$

2.1.2.3 ปัญหาพื้นฐานของฮิตเดนมาร์คอฟโมเดล

ฮิตเดนมาร์คอฟโมเดลมีปัญหาพื้นฐาน 3 ประการได้แก่

ปัญหาที่ 1 กำหนดให้ค่าลำดับข้อมูลการสังเกต (Observation sequence)

$O = \{O_1, O_2, \dots, O_t\}$ และแบบจำลอง $\lambda = \{A, B, \pi\}$ จะคำนวณหาค่าความน่าจะเป็นของค่าลำดับข้อมูลการสังเกต $P(O|\lambda)$ ได้อย่างไร

ปัญหาที่ 2 กำหนดให้ค่าลำดับข้อมูลการสังเกต (Observation sequence)

$O = \{O_1, O_2, O_3, \dots, O_t\}$ และแบบจำลอง $\lambda = \{A, B, \pi\}$ จะคำนวณเพื่อหาค่าลำดับของสถานะ $Q = \{Q_1, Q_2, \dots, Q_t\}$ ที่เหมาะสมได้อย่างไร

ปัญหาที่ 3 จะปรับค่าพารามิเตอร์ของแบบจำลอง $\lambda = \{A, B, \pi\}$ ที่จะทำให้

ได้ค่าความน่าจะเป็น $P(O|\lambda)$ สูงสุดได้อย่างไร

2.1.2.4 วิธีแก้ปัญหากับปัญหาพื้นฐานของฮิตเดนมาร์คอฟโมเดล

วิธีแก้ปัญหาที่ 1 เป็นการคำนวณหาว่าแบบจำลอง λ ใดๆ มีโอกาสจะให้ค่าลำดับเป็นไปตามลำดับของค่าปรากฏนั้น ด้วยค่าความน่าจะเป็นเท่าใด จะใช้วิธีแก้ปัญหา 2 วิธี ได้แก่ Forward Procedure และ Backward Procedure

Forward Procedure โดยพิจารณาตัวแปร $\alpha_t(i)$ กำหนดให้

$$\alpha_t(i) = P(O_1 O_2 \dots O_t, q_t = s_i | \lambda)$$

ซึ่งเป็นค่าความน่าจะเป็นของค่าลำดับข้อมูลการสังเกต (Observation sequence) $O_1 O_2 \dots O_t$ และ อยู่ในสถานะ (State) S_i ณ เวลา t โดยมีแบบจำลองคือ λ โดยการหาค่า $\alpha_t(i)$ จะมีวิธีการดังนี้

1. Initialization :

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N \quad (2.12)$$

ส่วนแรก (Initialization) คือการเริ่มคำนวณค่าลำดับข้อมูลการสังเกตลำดับแรก (O_1) ร่วมกับ ค่าสถานะเริ่มต้น (π_i) และเมื่อได้ผลลัพธ์จากการคำนวณแล้ว จะนำค่าไปใช้คำนวณต่อใน ส่วนของ Induction

2. Induction :

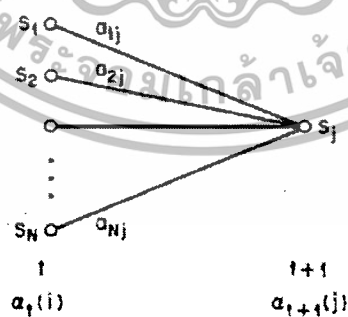
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1}), \quad 1 \leq t \leq T-1, \quad 1 \leq j \leq N \quad (2.13)$$

ส่วนที่สอง เป็นวิธีการคำนวณหา $\alpha_1(i)$ ของข้อมูลถัดไปจาก O_1 ทั้งหมดไปจนถึงตัวสุดท้าย เมื่อการคำนวณ ดำเนินไปจนถึง $t = T - 1$ ก็เสร็จสิ้นขั้นตอนของ Forward Procedure

3. Termination:

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i) \quad (2.14)$$

ส่วนสุดท้าย (Termination) คือ การคำนวณหาค่าความน่าจะเป็นของโมเดล $P(O|\lambda)$ จากผลรวมของตัวแปร $\alpha_T(i)$ ตัวสุดท้ายของทุกๆสถานะในโมเดลสามารถแสดงลักษณะการคำนวณได้ดังภาพที่ 2.2



ภาพที่ 2.4 ลักษณะการคำนวณ ด้วย Forward Procedure

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Backward Procedure โดยพิจารณาตัวแปร $\beta_t(i)$ กำหนดให้

$$\beta_t(i) = P(O_{t+1}O_{t+2} \dots O_T | q_t = S_i, \lambda)$$

ซึ่งเป็นค่าความน่าจะเป็นของค่าลำดับข้อมูลการสังเกตตั้งแต่ $t + 1$ จนถึง T และ อยู่ในสถานะ (State) S_i ณ เวลา t โดยมีแบบจำลองคือ λ โดยในการหาค่า $\beta_t(i)$ จะประกอบด้วย 2 ส่วนดังนี้

Initialization:

$$\beta_t(i) = 1, 1 \leq i \leq N \quad (2.15)$$

ส่วนแรก (Initialization) สมมุติให้ $\beta_t(i) = 1$ ทั้งหมด

Induction:

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j),$$

$$t = T-1, T-2, \dots, 1, 1 \leq i \leq N$$

(2.16)



ภาพที่ 2.5 ลักษณะการคำนวณ เพื่อหาค่า $\beta_t(i)$

ส่วนที่สอง (Induction) เมื่อดูจากภาพที่ 2.5 พบว่าเพื่อที่จะให้จากสถานะ S_j กลับมาที่สถานะ S_i ที่เวลา t โดยพิจารณาสถานะ S_j ที่เวลา $t + 1$ และคำนวณการเปลี่ยนสถานะตั้งแต่ S_i ถึง S_j เช่นเดียวกันกับค่าลำดับข้อมูลการสังเกตที่ O_{t+1}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีแก้ไขปัญหาที่ 2

คือการหาลำดับที่เหมาะสมที่สุดของสถานะที่สัมพันธ์กันกับค่าลำดับข้อมูลการสังเกตที่กำหนด โดยกำหนดตัวแปร

$$\gamma_t(i) = P(q_t = S_i | O, \lambda) \tag{2.17}$$

คือค่าความน่าจะเป็นเมื่ออยู่ในสถานะ S_i ที่เวลา t โดยค่าลำดับข้อมูลการสังเกต(Observation sequence) O และโมเดล λ จากสมการสามารถเขียนให้อยู่ในรูปแบบตัวแปร Forward-Backward ได้ดังนี้

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(O|\lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^N \alpha_t(i)\beta_t(i)} \tag{2.18}$$

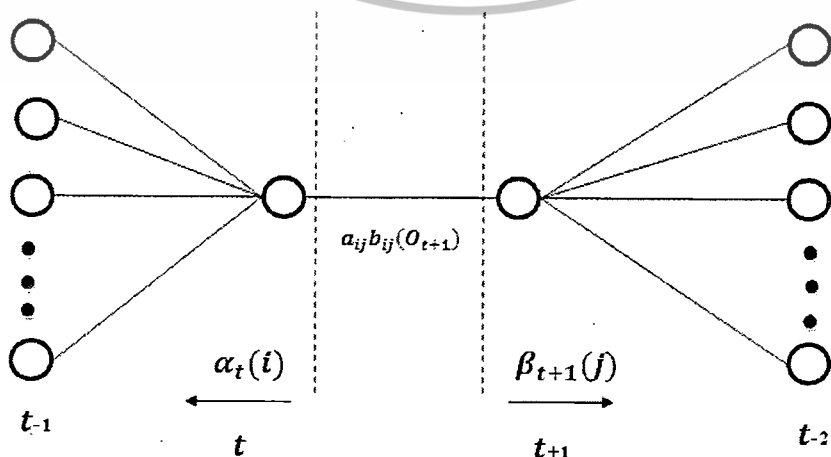
โดย $\alpha_t(i)$ คือการคำนวณจากค่าลำดับข้อมูลการสังเกต $O_1 O_2 \dots O_t$ ในขณะที่ $\beta_t(i)$ คือ การคำนวณจากค่าลำดับข้อมูลการสังเกต $O_{t+1} O_{t+2} \dots O_T$ โดยที่อยู่ในสถานะ(State) S_i ณ เวลา t

วิธีแก้ไขปัญหาที่ 3

คือการปรับค่าพารามิเตอร์ของแบบจำลองที่จะทำให้ได้ค่าความน่าจะเป็นของข้อมูลลำดับการสังเกต(Observation sequence) มีค่าสูงสุดโดยใช้วิธี บัมเวลช์(Baum-Welch) โดยกำหนดตัวแปร

$$\xi_t(i, j) = P(q_t = S_i, q_{t+1} = S_j | O, \lambda)$$

คือค่าความน่าจะเป็นเมื่ออยู่ในสถานะ S_i ที่เวลา t และ S_j ที่เวลา $t + 1$ โดยกำหนดให้โมเดลคือ λ และ ค่าข้อมูลการสังเกต O



ภาพที่ 2.6 ลำดับเหตุการณ์คำนวณร่วมกันระหว่างวิธี Forward และ Backward

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์เพื่อการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 2.6 คือ การแสดงลำดับเหตุการณ์ในการคำนวณค่าความน่าจะเป็นของลำดับข้อมูลการสังเกตพร้อมกันด้วยวิธี Forward และ Backward ที่สถานะ S_i ที่เวลา t และ สถานะ S_j ที่เวลา $t + 1$ จากตัวแปร Forward-Backward สามารถเขียนให้อยู่ในรูปแบบของ $\xi_t(i, j)$ ได้ดังนี้

$$\begin{aligned}\xi_t(i, j) &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{P(O|\lambda)} \\ &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}\end{aligned}\quad (2.19)$$

จากนั้นนำค่า $\xi_t(i, j)$ มาใช้ในการปรับค่าพารามิเตอร์ใหม่ของโมเดลที่ประกอบไปด้วย π_i, A, B ดังนี้

$$\bar{\pi}_i = \text{จำนวนครั้งที่คาดหวังที่อยู่ในสถานะ } S_i \text{ ที่เวลา } (t = 1) = \gamma_1(i) \quad (2.20)$$

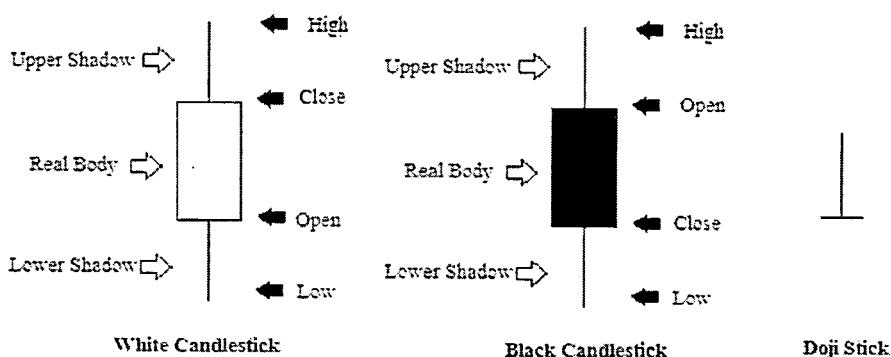
$$\begin{aligned}\bar{a}_{ij} &= \frac{\text{จำนวนครั้งที่คาดหวังจากเปลี่ยนสถานะ } S_i \text{ ไปยัง } S_j}{\text{จำนวนครั้งที่คาดหวังของการเปลี่ยนสถานะ } S_i} \\ &= \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}\end{aligned}\quad (2.21)$$

$$\begin{aligned}\bar{b}_{ij} &= \frac{\text{จำนวนครั้งที่คาดหวังในสถานะ } j \text{ และข้อมูลการสังเกต } V_k}{\text{จำนวนครั้งที่คาดหวังในสถานะ } j} \\ &= \frac{\sum_{t=1}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)}\end{aligned}\quad (2.22)$$

2.2 กราฟแท่งเทียน

กราฟแท่งเทียน คือ รูปแบบของแผนภูมิแท่งเทียน ซึ่งเป็นที่นิยมอย่างยิ่งในปัจจุบัน เป็นแผนภูมิที่ง่ายต่อการอ่านข้อมูลเพื่อทำการวิเคราะห์ โดยส่วนประกอบของกราฟแท่งเทียนประกอบไปด้วย 4 ส่วนหลัก คือ ราคาเปิด ราคาสูงสุด ราคาต่ำสุด และราคาปิดของวันนั้นๆ ดังภาพที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.7 ส่วนประกอบของกราฟแท่งเทียน

โครงสร้างของกราฟแท่งเทียนประกอบไปด้วย ราคาเปิด ราคาสูงสุด ราคาต่ำสุด และราคาปิด แท่งตรงกลางเรียกว่า แท่งเทียน (Real Body) ไม้เทียนด้านบนเรียกว่า เงาด้านบน(Upper Shadow) และไม้เทียนด้านล่างเรียกว่า เงาด้านล่าง (Lower Shadow)

ในกรณีที่ราคาเปิดสูงกว่าราคาปิด แท่งเทียนจะเป็นสีดำ เรียกว่า Black Candlestick แสดงถึงแนวโน้มที่ไม่ดี (ราคาปรับตัวลดลง) ในทางกลับกัน ในกรณีที่ ราคาเปิดสูงกว่าราคาเปิด เรียกว่า White Candlestick แสดงถึงแนวโน้มที่ดี (ราคาปรับตัวสูงขึ้น) และในกรณีที่ราคาเปิด เท่ากับราคาเปิด โดยราคาระนั้น จะเป็นจุดสูงสุด หรือต่ำสุดหรือ ไม่ก็ตาม แท่งเทียนจะมีลักษณะเป็นเส้นขีดขวาง เรียกว่า Doji stick แสดงลักษณะเป็นกลาง (มีแรงซื้อ แรงขายที่เท่ากัน)



ภาพที่ 2.8 รูปแบบของกราฟแท่งเทียน

รูปแบบทั้งหมดของกราฟแท่งเทียนจะมีด้วยกันทั้งหมด 12 รูปแบบ ดังภาพที่ 2.8 โดยที่ขนาดของ Real Body ,Upper Shadow และ Lower Shadow นั้นจะขึ้นอยู่กับราคาของตัวสินค้านั้นๆ ที่เกิดขึ้นในแต่ละวัน โดยได้มีการสร้างรูปแบบโดยเป็นการรวมกันของรูปแบบพื้นฐาน สามารถแบ่งออกได้เป็น 2 กลุ่มคือ Bullish Pattern แสดงถึงเทรนขาขึ้น (Uptrend) และ Bearish Pattern แสดงถึงเทรนขาลง (Downtrend) ดังตารางที่ 2.1 และตารางที่ 2.2 ตามลำดับ


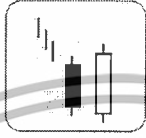
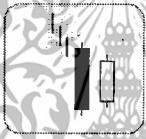
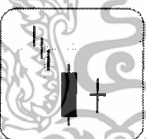
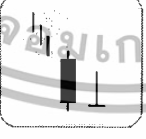

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 รูปแบบของกราฟแท่งเทียน Bullish Pattern

Bullish Pattern		
Name	Pattern	Suggestion
PIERCING PATTERN		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อน (downtrend) ในวันแรกจะพบ long black candle เกิดขึ้น หลังจากนั้นจะพบ long white candle ที่มีราคาเปิดต่ำกว่า long black candle ราคาปิดของ long white candle ทะลุเส้นกึ่งกลางแต่ไม่เลยราคาเปิดของ long black candle
Morning Star		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อนในวันแรก หลังจากนั้นจะพบแท่งเทียนที่มี body ขนาดเล็ก เปิดกระโดดลงจากแท่งที่หนึ่งแบบมี gap เป็นได้ทั้ง black และ white body สุดท้ายจะพบแท่งเทียน white candle ในวันที่สาม โดยมีราคาปิดขยับขึ้นไปในระดับที่อยู่ในส่วนของ body แท่งที่หนึ่ง
Morning Doji Star		รูปแบบที่จะเกิด Morning Doji Star นั้นเหมือนกันกับการเกิด Morning Star ต่างกันที่วันที่ 2 จะเป็น Doji
KICKING PATTERN		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อนในวันแรกเกิด long black candle และในวันถัดมาเกิด long white candle ที่มีราคาเปิดกระโดดเกิด gap จากราคาปิดของแท่งดำ
Three Inside Up		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อนจะพบ bullishharami ในสองวันแรกในวันที่สามจะเป็นแท่งเทียน white candle ที่มีราคาปิดสูงกว่าราคาเปิดของแท่งดำแท่งที่หนึ่ง
Dragonfly Doji		เป็นรูปแบบแท่งเทียนหนึ่งแท่ง ที่เกิดขึ้นที่จุดต่ำสุดหรืออยู่ระหว่าง downtrend จะเป็น doji ที่มี lower shadow ขนาดยาว และไม่มี upper shadow

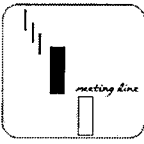
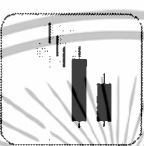

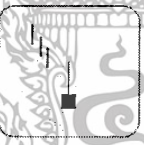
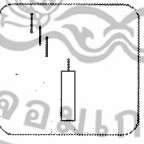
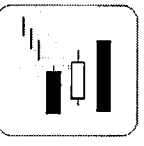
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 (ต่อ)

Bullish Pattern		
Name	Pattern	Suggestion
Long Legged Doji		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อน ราคาเปิดของ doji จะเปิดกระโดดลงแบบมี gap Upper shadow และ lower shadow จะมีขนาดยาว โดยที่ shadow ทั้งสองมีขนาดความยาวเท่ากัน หรือใกล้เคียงกัน
Bullish Engulfing		เป็นรูปแบบที่มีแท่งเทียนแท่งที่สองเป็น white candle ที่มีขนาด body ยาวกว่า body ของแท่งเทียนแท่งที่หนึ่งซึ่งเป็น black candle
Bullish Harami		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อน เป็นรูปแบบที่มีแท่งเทียนแท่งที่สองเป็น white candle โดยที่ small white candle จะอยู่ในส่วน body ของแท่งเทียน long black candle มีแนวโน้มที่จะกลับไปเป็นขาขึ้น
Bullish Harami Cross		เหมือนกับ Bullish Harami แต่ในวันที่สองแท่งเทียนจะเป็น Doji มีแนวโน้มที่จะกลับไปเป็นขาขึ้น โดยที่มีความเป็นไปได้มากกว่าตัวของ Bullish Harami
Bullish Gravestone Doji		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อน ในวันแรกจะพบ แท่งเทียน black candle แล้วในวันที่สองจะเป็น stone doji เป็น doji ที่ไม่มี lower shadow
Bullish Doji Star		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อน ในวันแรกจะพบ แท่งเทียน long black candle แล้วในวันถัดมาเกิด doji ที่กระโดดลงมาแบบที่ gap โดยที่มี upper และ lower shadow ขนาดสั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 (ต่อ)

Bullish Pattern		
Name	Pattern	Suggestion
Meeting Line		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อน เกิด long black candle ในวันแรก ในวันถัดมาจะเกิดเป็น long white candle ที่มีราคาเปิดต่ำกว่า black candle ในวันก่อนกระโดดลงไปมาก และมีราคาปิดที่กลับมาใกล้เคียงกับราคาปิดของ black candle ในวันก่อน
Matching Low		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาลงมาก่อน ประกอบไปด้วยแท่งเทียน black candle 2 แท่ง โดยแท่งเทียนแรกเป็น long black candle ส่วนแท่งเทียนแท่งที่สองเป็น black candle เช่นกัน แต่สั้นกว่า ทั้งนี้ทั้ง 2 แท่งจะมีราคาปิดเท่ากันหรือใกล้เคียงกันมาก
Hammer		Bullish Hammer จะมีหน้าตาคล้ายๆกับ Bullish Dragonfly Doji แต่จะต่างกันที่ body ของแท่งเทียนคือ hammer จะมี body สั้น แต่ dragonfly แทบจะไม่มี body เลย
Inverted Hammer		รูปแบบของ inverted hammer จะเป็นแท่งเทียนที่มี body สั้นแต่มี upper shadow ยาว ส่วน lower shadow จะสั้นมากหรือแทบไม่มีเลย
Belt Hold		เป็นรูปแบบที่ประกอบด้วยแท่งเทียน 1 แท่ง โดยจะเป็นแท่งเทียน white candle ที่มีราคาเปิดตกลงไปจากวันก่อนมาก และไม่มี lower shadow หลังจากนั้นราคาปิดดีดกลับขึ้นไปใกล้เคียงกับราคาสูงสุดของวัน
Stick Sandwich		เป็นรูปแบบที่ประกอบด้วยแท่งเทียน 3 แท่ง ในวันแรกเป็น black candle ส่วนวันที่สองเป็น white candle ที่มีราคาปิดสูงกว่าวันแรก ต่อมาเกิดแท่งเทียนในวันที่สามที่เป็น black candle ที่มีราคาเปิดสูงกว่าแท่งเทียนวันที่สอง แต่สุดท้ายราคาปิดของแท่งเทียนวันที่สามก็ลงมาปิดที่ระดับเดียวกับราคาเปิดของแท่งเทียนวันที่หนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 รูปแบบของกราฟแท่งเทียน Bearish Pattern

Bullish Pattern		
Name	Pattern	Suggestion
Dark Cloud Cover		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาขึ้นมาก่อน วันแรกเราจะเห็นกราฟแท่ง long white candle ในวันที่สองเราจะเห็นกราฟแท่ง long black candle ที่มีราคาเปิดสูงกว่าราคาปิดของวันแรก ราคาปิดของ long black candle ปิดต่ำกว่าเส้นกึ่งกลาง body ของ long white candle
Evening Star		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาขึ้นมาก่อน ประกอบด้วยแท่งเทียน 3 แท่ง โดยที่แท่งแรกจะเป็น long white candle และแท่งที่สองจะเป็น small body ที่จะเป็นแบบแท่งดำหรือแท่งขาวก็ได้ โดยมีราคาเปิดกระโดดแบบมี gap ขึ้นไปในทิศทางเดียวกับแท่งแรก แท่งที่สามจะเป็น black candle ที่มีราคาปิดเข้ามาอยู่ในพื้นที่ของแท่ง long white candle
Evening Doji Star		รูปแบบนี้เหมือนกับ evening star เพียงแต่แท่งที่สองเป็น doji แทนที่จะเป็น small body
Bearish Kicking Pattern		เป็นรูปแบบที่ประกอบด้วยแท่งเทียนสองแท่ง โดยแท่งแรกเป็น white marubozu candle (เป็นแท่งเทียนที่ไม่มี upper/lower shadow) และแท่งที่สองจะเป็น long black candle ที่มีราคาเปิดกระโดด gap ต่ำกว่าราคาเปิดของแท่งแรก
Three Inside Down		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาขึ้นมาก่อน โดยจะเกิด bearish harami ขึ้นมาก่อนแล้วถัดมาเป็นแท่งเทียน black candle ที่มีราคาปิดต่ำกว่าราคาเปิดของแท่งขาวแท่งที่หนึ่ง
Bearish Dragonfly Doji		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาขึ้นมาก่อน ที่ปลาย trend จะเห็นแท่งเทียน doji ที่มี lower shadow ขนาดยาว แต่ไม่มี upper shadow

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 (ต่อ)

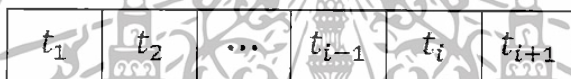
Bullish Pattern		
Name	Pattern	Suggestion
<u>Bearish Long Legged Doji</u>		ก่อนเกิดรูปแบบนี้จะเป็นเทรนขาขึ้นมาก่อน ราคาเปิดของ doji จะเปิดกระโดดขึ้น Upper shadow และ lower shadow จะมีขนาดยาว โดยที่ shadow ทั้งสองมีขนาดความยาวเท่ากัน
<u>Bearish Engulfing</u>		เป็นรูปแบบที่มีแท่งเทียนแท่งที่สองเป็น black candle ที่มีขนาด body ยาวกว่า body ของแท่งเทียนแท่งที่หนึ่งซึ่งเป็น white candle
<u>Bearish Harami</u>		รูปแบบนี้ประกอบด้วยแท่งเทียนสองแท่ง โดยแท่งแรกเป็น long white candle ส่วนแท่งที่สองจะเป็น black candle ที่มีขนาด body อยู่ในพื้นที่ของแท่งแรก
<u>Bearish Harami Cross</u>		รูปแบบนี้ประกอบด้วยแท่งเทียนสองแท่ง โดยแท่งแรกเป็น long white candle ส่วนแท่งที่สองจะเป็น doji ที่เกิด อยู่ในพื้นที่ของแท่งแรก
<u>Bearish Gravestone Doji</u>		เป็นรูปแบบที่มีราคาเปิดเท่ากับราคาปิดและเป็นระดับราคาที่ต่ำสุดของวัน ประกอบด้วยส่วนที่เป็น upper shadow และไม่มี lower shadow โคนที่ราคาปิดของ doji นั้นจะใกล้เคียงกับราคาปิดของวันก่อน
<u>Bearish Doji Star</u>		เป็นรูปแบบที่มี doji กระโดดขึ้นจากราคาปิดของแท่งเทียนขาวแบบมี gap มี Upper และ Lower shadow มีขนาดสั้น โดยที่วันแรกเกิด long white candle ขึ้นก่อน
<u>Shooting Star</u>		จะเป็นแท่งเทียนที่มี body สั้นแต่มี upper shadow ยาว ส่วน lower shadow จะสั้นมากหรือแทบไม่มีเลย
<u>Hanging Man</u>		Hanging Man จะมีหน้าตาคล้ายๆกับ Bearish Dragonfly Doji แต่จะต่างกันว่า body ของแท่งเทียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Body คือช่วงของราคาระหว่างราคาปิด (Close) กับราคาเปิด (Open) ตัว Body เป็นตัวที่สำคัญที่สุดใน Candlestick การเป็น White Candle หรือ Black Candle ประกอบกับขนาดของ Body แล้วสามารถแสดงให้เห็นให้เราเห็นได้ว่า ตลาดในขณะนั้น เป็น Bullish trend หรือ Bearish trend ถ้าเรามี White body ที่ยาว และ Black body ที่สั้น แสดงว่าตลาดอยู่ใน Bullish sentiment และในขณะที่เรามี White body ที่สั้น และ Black body ที่ยาว ก็แสดงว่าตลาดอยู่ใน Bearish sentiment

2.3 ข้อมูลอนุกรมเวลา

คือข้อมูลที่ถูกเก็บไว้ในช่วงระยะเวลาที่เท่ากันและต่อเนื่องกันไปเรื่อยๆเป็นระยะเวลา n โดย ข้อมูลอนุกรมเวลาสามารถนำมาประยุกต์ใช้วิเคราะห์ปัญหาต่างๆเช่น ประยุกต์ใช้ร่วมกับการวิเคราะห์ทางเศรษฐศาสตร์ เพื่อนำมาวางแผนหรือพยากรณ์สิ่งที่จะเกิดขึ้นในอนาคตโดยลักษณะของข้อมูลจะอยู่ในรูปแบบดังภาพที่ 2.9



ภาพที่ 2.9 รูปแบบจำลองของลักษณะข้อมูลเวลา

โดยจะเลือกข้อมูล ณ ช่วงเวลาใดช่วงหนึ่งจนถึงข้อมูลค่าสุดท้าย ณ เวลานั้น (t_i) เพื่อพยากรณ์หาข้อมูลที่จะเกิดขึ้นในอนาคต (t_{i+1})

2.4 งานวิจัยที่เกี่ยวข้อง

ทฤษฎีปัญญาประดิษฐ์ ถูกนำไปประยุกต์ใช้ในพยากรณ์ด้านต่างๆมากมาย ในส่วนของทฤษฎีฮิตเดนมาร์คอฟโมเดล นอกจาก Lawrence R Rabiner ที่นำไปประยุกต์ใช้กับการรู้จำเสียงของมนุษย์ (Speech Recognition) แล้วยังได้มีการนำมาประยุกต์ใช้ในด้านธุรกิจด้วย จึงได้ทำการยกตัวอย่างงานวิจัยที่เกี่ยวข้องที่นำทฤษฎีปัญญาประดิษฐ์ มาใช้ในด้านธุรกิจได้แก่

Md. Rafiul Hassan, BaikunthNath. ได้ทำการพยากรณ์ราคาหุ้นโดยใช้ฮิตเดนมาร์คอฟโมเดล โดยใช้แบบจำลอง Left-Right Topology กำหนดให้มี 4 สถานะ ($N = 4$) โดยมีเป้าหมายของการพยากรณ์คือ พยากรณ์ราคาปิดของวันถัดไป ใช้ข้อมูลอนุกรมเวลาของหุ้น โดยลักษณะข้อมูลประกอบด้วย ราคาปิด ราคาเปิด ราคาสูงสุด และราคาต่ำสุดในการใช้งานข้อมูลจะแบ่งข้อมูลเป็น 2 ชุดคือ ข้อมูลในการสอน (Training) และ ข้อมูลในการทดสอบ (Testing)

ในส่วนของกระบวนการสอน (Training) จะใช้ค่าความน่าจะเป็นในการคำนวณ เช่น ถ้าสมมุติว่าค่าความเป็นไปได้ของราคาในวันนี้คือ A จะทำการหาค่าความเป็นไปได้จากข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ย้อนหลังที่มีพฤติกรรมใกล้เคียงหรือเหมือนกับ A ถ้าสมมุติว่าราคาหุ้นวันถัดไปมีพฤติกรรมที่มีลักษณะคล้ายกับข้อมูลในอดีตก็จะทำการคำนวณเพื่อหาค่าความแตกต่างระหว่าง ราคาปิดของวันที่ (อดีต) ข้อมูลคล้ายกัน กับ ราคาปิดของวันถัดไป(อดีต)ที่ข้อมูลคล้ายกัน เมื่อได้ค่าแล้วก็จะนำมาพยากรณ์โดย นำราคาปิดของวันที่ล่าสุดบวกกับค่าความต่างที่คำนวณได้ โดยในส่วนของ การสอน จะใช้ข้อมูลของ Southwest Airlines ตั้งแต่วันที่ 18 ธันวาคม 2002 ถึง 29 กันยายน 2004 เพื่อพยากรณ์ราคาปิดในวันที่ 30 กันยายน ค.ศ. 2004 จากการทดลองพบว่า มีค่า Mean Absolute Percentage Error (MAPE) = 2.01 และ $R^2 = 0.87498$

มณีนี อรรถศิริปัญญา และ วิไลณี ประสิทธิ์ชัยกุล แนวโน้มหุ้นโดยการใช้อัตราเงินปันผล โดยแบ่งข้อมูลเป็น 2 ชุดคือ ข้อมูลในการสอน(Training) และ ข้อมูลในการทดสอบ(Testing) ใช้วิธีการแปลงข้อมูลนำเข้าด้วยรูปแบบของกราฟแท่งเทียนมีด้วยกันทั้งหมด 19 รูปแบบ

ในส่วนของกระบวนการสอน(Training) จะจัดชุดข้อมูลเป็น 8 และ 10 แท่งเทียน กำหนดให้มี 20 สถานะ(N = 20) โดยใช้ข้อมูลหุ้น TISCO , LPN , BIG-C, KTB , TRUE และ HQ โดยใช้ข้อมูลของวันที่ 3 มกราคม 2014 ถึง 13 กุมภาพันธ์ 2014 และได้ทำการพยากรณ์ผลของวันที่ 13 กุมภาพันธ์ 2014 ถึง 17 กุมภาพันธ์ 2014 โดยทำการเปรียบเทียบผลการพยากรณ์ระหว่างการใช้ อัลกอริทึม Baum-welch กับ Heuristic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

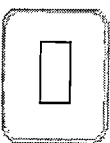

ออกแบบการทดลอง

ในบทนี้จะนำเสนอเรื่องของการออกแบบขั้นตอนและวิธีในการทำการทดลอง โดยจะเริ่มตั้งแต่ การนำข้อมูลปัจจัย เข้าสู่กระบวนการการจัดการข้อมูล จากนั้นนำข้อมูลเข้าสู่กระบวนการ สอนและการทำนายของอิตเดนมาร์คอฟโมเดล และการแปลงค่าข้อมูลในการทำนายผล โดย รายละเอียดของแต่ละขั้นตอนจะประกอบด้วยส่วนต่างๆดังนี้

3.1 การแปลงข้อมูลเป็นข้อมูลการสังเกต

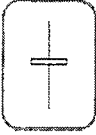

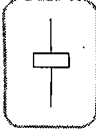

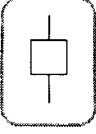

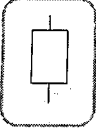

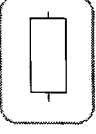

การแปลงข้อมูลการสังเกต(Observation) จะเริ่มด้วยการนำราคาเปิด ราคาสูงสุด ราคาต่ำสุด และราคาปิด มาคำนวณหาค่าของ เงาด้านบน (Upper Shadow) เงาด้านล่าง (Lower Shadow) และตัวแท่งเทียน (Real Body) ว่าแต่ละตัวมีค่าเป็นเท่าไร หลังจากนั้นจะจำแนกรูปแบบของข้อมูล การสังเกต ด้วยวิธีการคำนวณหาอัตราส่วนของตัวแท่งเทียน อัตราส่วนของขนาดเงาด้านบนและ อัตราส่วนของเงาทั้งหมด ยกเว้นแต่รูปแบบที่มีอัตราส่วนของตัวแท่งเทียน เป็น 100% จะใช้วิธีการ จำแนกด้วยช่วงของราคาที่เปลี่ยนแปลงของวัน โดยจะกำหนดรูปแบบดังตารางที่ 3.1 และ 3.2

ตารางที่ 3.1 ตารางแปลงกราฟแท่งเทียนเป็นตัวเลขข้อมูลการสังเกตที่ อัตราส่วนแท่งเทียนเป็น 100%

ตารางแปลงแผนภูมิแท่งเทียนเป็นตัวเลขข้อมูลการสังเกตที่ อัตราส่วนแท่งเทียนเป็น 100%					
Pattern	Range Data	Observation	Pattern	Range Data	Observation
	0 – 15.99	0		(-0.01) – (-11)	4
	16 – 31.99	1		(-11.01) – (-22)	5
	32 – 47.99	2		(-22.01) – (-33)	6
	48-64	3		(-33.01) – (-44)	7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 ตารางแปลงแผนภูมิแท่งเทียนเป็นตัวเลขข้อมูลการสังเกตตามอัตราส่วน

ตารางแปลงแผนภูมิแท่งเทียนเป็นตัวเลขข้อมูลการสังเกตตามอัตราส่วน							
Pattern	Body%	Shadow%	Observation	Pattern	Body%	Shadow%	Observation
	0 – 19.99%	Zone	8		0 – 19.99%	Zone	28
		Zone * 2	9			Zone * 2	29
		Zone * 3	10			Zone * 3	30
		Zone * 4	11			Zone * 4	31
	20 – 39.99%	Zone	12		20 – 39.99%	Zone	32
		Zone * 2	13			Zone * 2	33
		Zone * 3	14			Zone * 3	34
		Zone * 4	15			Zone * 4	35
	40 – 59.99%	Zone	16		40 – 59.99%	Zone	36
		Zone * 2	17			Zone * 2	37
		Zone * 3	18			Zone * 3	38
		Zone * 4	19			Zone * 4	39
	60 – 79.99%	Zone	20		60 – 79.99%	Zone	40
		Zone * 2	21			Zone * 2	41
		Zone * 3	22			Zone * 3	42
		Zone * 4	23			Zone * 4	43
	80 – 99.99%	Zone	24		80 – 99.99%	Zone	44
		Zone * 2	25			Zone * 2	45
		Zone * 3	26			Zone * 3	46
		Zone * 4	27			Zone * 4	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1. วิธีการคำนวณเพื่อแปลงค่าเป็นข้อมูลการสังเกต

การคำนวณมีส่วนประกอบหลักอยู่ด้วยกัน 4 ส่วน คือ ราคาเปิด(open) ราคาสูงสุด (high) ราคาต่ำสุด(low) และราคาปิด(close) โดยจะกำหนดค่าตัวแปร ดังนี้

- Body คือ ขนาดของตัวแท่งเทียน จะเท่ากับ $close - open$
ค่าของ body เป็นตัวบ่งบอกถึงสีของแท่งเทียน ถ้า $body > 0$ แท่งเทียนจะเป็นสีเขียว
ถ้า $body < 0$ แท่งเทียนจะเป็นสีดำ
- Full candle คือ ขนาดของแท่งเทียนทั้งหมด มีค่าเท่ากับ $high - low$
- Upper Shadow คือเงาด้านบน มีค่าเท่ากับ $high - close$ ในกรณีที่ $close > open$
และจะเท่ากับ $high - open$ ในกรณีที่ $open > close$
- Lower Shadow คือเงาด้านล่าง มีค่าเท่ากับ $open - low$ ในกรณีที่ $close > open$
และจะเท่ากับ $close - low$ ในกรณีที่ $open > close$

วิธีการคำนวณคือ เริ่มจากการหาอัตราส่วนของตัวแท่งเทียน อัตราส่วนของขนาดเงาด้านบน และอัตราส่วนของเงาทั้งหมด

$$body\% = \left(\frac{|body|}{Full\ candle} \right) \times 100 \quad (3.1)$$

$$Upper\ shadow\% = \left(\frac{Upper\ shadow}{Full\ candle} \right) \times 100 \quad (3.2)$$

$$Shadow\% = \left(\frac{Upper\ shadow + Lower\ shadow}{Full\ candle} \right) \times 100 \quad (3.3)$$

หลังจากนั้น จะแบ่ง Shadow% ออกเป็น 4 ส่วน จะได้ $zone = Shadow\% / 4$ แล้วจึงนำค่าที่ได้มาทำการเปรียบเทียบกับตารางที่ 3.2

ในกรณีที่ $body\% = 100$ จะทำการคำนวณหาค่าของของราคาที่เปลี่ยนแปลงแล้ว จึงนำไปเทียบกับตารางที่ 3.1

3.1.2. ตัวอย่างการคำนวณเพื่อแปลงค่าเป็นข้อมูลการสังเกต

CONMONTH	POPEN	PHIGH	PLOW	PCLOSE
2006-06-22	588	588	582.2	582.2
2006-06-23	579	584.8	577	584.8
2006-06-26	583	584	580	584.8
2006-06-27	589	589	589	581.6

ภาพที่ 3.1 ตัวอย่างข้อมูล

เริ่มต้นด้วยการคำนวณหาค่า $body\%$ $Upper\ shadow\%$ และ $Shadow\%$

จากภาพที่ 3.1 ใช้ข้อมูลของวันที่ 2006-06-22 $open = 588$, $high = 588$, $low = 582.2$, $close = 582.2$
จะได้ $body = 582.2 - 588 = -5.8$. $Full\ candle = 588 - 582.2 = 5.8$

$$body\% = \left(\frac{|-5.8|}{5.8} \right) \times 100 = 100\%$$

เนื่องจาก $body\% = 100\%$ และ $body = -5.8$ นำไปเทียบกับตารางที่ 3.1 โดยที่ -5.8 อยู่ในช่วง
ของ $(-0.01) - (-11)$ ข้อมูลของวันที่ 2006-06-22 จะแปลงเป็นข้อมูลการสังเกต (Observation) = 4

ข้อมูลของวันที่ 2006-06-23 $open = 579$, $high = 584.8$, $low = 577$, $close = 584.8$

จะได้ $body = 584.8 - 579 = 5.8$ $Full\ candle = 584.8 - 577 = 7.8$ $Upper\ shadow = 0$. $Lower\ shadow = 579 - 577 = 2$

$$body\% = \left(\frac{5.8}{7.8} \right) \times 100 = 74.36\%$$

$$Upper\ shadow\% = \left(\frac{0}{7.8} \right) \times 100 = 0\%$$

$$Shadow\% = \left(\frac{0+2}{7.8} \right) \times 100 = 25.64\%$$

$$zone = Shadow\% / 4 = 6.41$$

ได้ค่า $body\% = 74.36\%$, $Upper\ shadow\% = 0$ และ $zone = 6.41$ นำไปเทียบกับตารางที่ 3.2

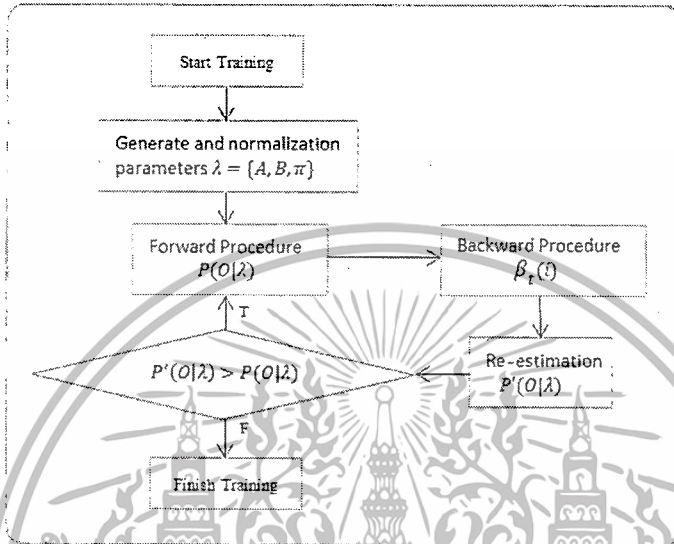
ข้อมูลของวันที่ 2006-06-23 จะแปลงเป็นข้อมูลการสังเกต (Observation) = 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 กระบวนการของฮิตเดนมาร์คอฟโมเดลในการทดลอง

ฮิตเดนมาร์คอฟโมเดลมีกระบวนการทำงานอยู่ทั้งหมด 2 ส่วน คือ การสอนเพื่อให้ฮิตเดนมาร์คอฟโมเดลรู้จำค่าของพารามิเตอร์(Training Phase) และการทำนายผล(Testing Phase)

3.2.1. กระบวนการสอนเพื่อให้ฮิตเดนมาร์คอฟโมเดลรู้จำ (Training Phase)



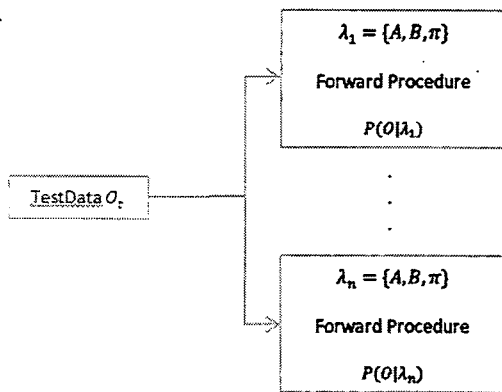
ภาพที่ 3.2 Flow จำลองการรู้จำของฮิตเดนมาร์คอฟโมเดล

จากภาพที่ 3.2 เริ่มต้นด้วยการสุ่มค่า A, B, π และทำการ Normalization แล้วจึงนำค่า O_T และ A, B, π เข้าสู่กระบวนการ Forward Procedure เพื่อหาค่า $\alpha_t(i)$ และค่าความน่าจะเป็น $P(O|\lambda)$ ต่อมาเข้าสู่กระบวนการ Backward Procedure เพื่อหาค่า $\beta_t(i)$ และทำการปรับค่าพารามิเตอร์ของโมเดลด้วยกระบวนการ Re-estimation จะได้ผลลัพธ์คือ $\bar{A}, \bar{B}, \bar{\pi}$ แล้วจึงนำค่าพารามิเตอร์นี้ไปคำนวณหาค่าความน่าจะเป็นของ $P'(O|\lambda)$ แล้วจึงนำค่าความน่าจะเป็น $P(O|\lambda)$ และ $P'(O|\lambda)$ มาเปรียบเทียบกัน ถ้าค่าความน่าจะเป็น $P(O|\lambda) < P'(O|\lambda)$ จะทำการแทนที่ข้อมูลพารามิเตอร์ A, B, π ด้วย $\bar{A}, \bar{B}, \bar{\pi}$ แล้วจะทำตามกระบวนการเดิมต่อไป ซึ่งจะทำเช่นนี้ไปจนกว่าค่าของ $P(O|\lambda) > P'(O|\lambda)$ จึงหยุดกระบวนการสอนใน O_T ชุดนั้น และนำค่า O_T ชุดใหม่เข้าไปสู่กระบวนการต่อไปจนกระทั่งหมดข้อมูลที่จะส่งเข้าไป

3.2.2. กระบวนการทำนายผลของฮิตเดนมาร์คอฟโมเดล (Testing Phase)

การทำนายผลคือการนำ O_T ที่ใช้ในการทดสอบ เข้าไปหาความน่าจะเป็นด้วยวิธีการ Forward-Procedure ทุกๆ โมเดลที่ผ่านการสอนมาแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

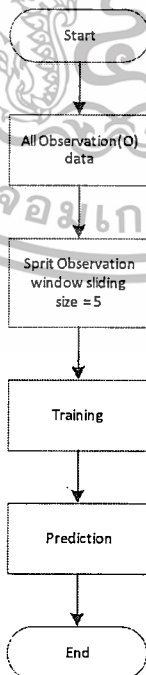


ภาพที่ 3.3 Flow จำลองการทำนายผลของฮิตเดนมาร์คอฟโมเดล

จากภาพที่ 3.3 เมื่อส่ง O_T ไปหาค่าความน่าจะเป็นจะพบว่าจะได้ค่าความน่าจะเป็น $P(O|\lambda)$ จำนวน n โมเดล (โดยที่ n คือจำนวนโมเดลที่ผ่านการสอนมาแล้ว) แล้วจึงนำค่าความน่าจะเป็น $P(O|\lambda_1), \dots, P(O|\lambda_n)$ มาหาค่าสูงสุดโดยถ้าพบว่าค่าของ $P(O|\lambda)$ ใหนให้ค่าความน่าจะเป็นสูงที่สุดจะหมายความว่า O_T ที่ส่งเข้าไปทดสอบมีลักษณะคล้ายหรือเหมือนกันกับ โมเดลนั้น และค่าของโมเดลนั้นคือคำตอบของการพยากรณ์

3.3 การออกแบบขั้นตอนในการรู้จำและการทำนายผลในการทดลอง

ในการทดลองนี้จะนำ Window Sliding Algorithm มาใช้ในการจัดแบ่งข้อมูลการสังเกต ออกเป็น window size เพื่อเป็นข้อมูลในการส่งเข้าไปใช้ในกระบวนการ ดังภาพที่ 3.4 และภาพที่ 3.5



ภาพที่ 3.4 Flow ขั้นตอนการ Train และ Testing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

O_1	O_2	O_3	O_4	O_5	O_6	\dots	O_t
O_1	O_2	O_3	O_4	O_5			
	O_2	O_3	O_4	O_5	O_6		
		O_3	O_4	O_5	O_6	\dots	

ภาพที่ 3.5 การจัดการข้อมูลการสังเกตด้วย Window Sliding Algorithm

จากภาพที่ 3.4 กระบวนการสำคัญจะเริ่มตั้งแต่การจัดการข้อมูล O_T ด้วย Window Sliding Algorithm โดยกำหนดให้ $size = 5$ (ดังภาพที่ 3.5) เพื่อเป็นข้อมูลเข้า (Input) ให้กับอิตเดนมาร์คอฟโมเดลโดยจะทำการตรวจสอบ Observation สุดท้ายของ Window เพื่อที่จะดูว่า ลำดับของข้อมูลที่เข้ามาอยู่ใน โมเดลใด แล้วจึงเริ่มทำการ Training จะทำอย่างนี้ไปเรื่อยๆจนกว่าข้อมูล O_T จะหมด

ผลลัพธ์จากการทำนายอิตเดนมาร์คอฟโมเดลจะทำนายออกมาเป็น ค่า O (ข้อมูลที่ถูกลดขนาด) ดังนั้นจึงต้องนำข้อมูลที่ทำนายได้มาแปลงเป็นค่าเงิน เพื่อนำมาเปรียบเทียบกับข้อมูลจริง (Actual Data)

3.4 การเตรียมข้อมูลเพื่อแปลงค่าผลของการทำนาย

จากกระบวนการ อิตเดนมาร์คอฟโมเดล ข้อมูลที่เข้า และ ออก(ผลการทำนาย) คือ Observation ดังนั้นจะต้องแปลงข้อมูลจาก Observation กลับมาเป็นราคาทองคำ โดย

นำข้อมูลราคาทองคำในแต่ละวันมา นำค่า close = open เพื่อหาค่าที่เปลี่ยนแปลงไปในแต่ละวัน และนำค่า open ของวัน ไปลบด้วย close ของวันก่อนหน้า เพื่อหาค่าส่วนต่างของราคาปิดวันก่อนกับราคาเปิดในอีกวัน

date	open	close	low	high	close	Observation	DiffClose	DiffOPEN
2006-01-27	558.7	558.7	558.7	558.7	558.7	0	0	-1
2006-01-30	559.2	565.8	557.6	565.8	565.8	20	6.6	0.5
2006-01-31	566.5	572.7	565.3	572.7	569.4	18	4.3	0.7
2006-02-01	571.6	571.6	563	569.4	569.4	32	-2.2	0.8
2006-02-02	569.8	575.3	568	572.5	572.5	14	2.7	0.4
2006-02-03	571.5	574.2	565	567.4	567.4	38	-4.1	-1
2006-02-06	568.4	574	567.5	570.2	570.2	15	1.8	1

ภาพที่ 3.6 ตัวอย่างข้อมูลการจัดเรียง observation ควบคู่กับราคาทองคำ

จากนั้นทำการจัดกลุ่มตามแต่ละ observation แล้วหาค่าเฉลี่ยของราคาที่เปลี่ยนแปลงไปในแต่ละวัน และส่วนต่างของราคาเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMMONH	POPEN	PHIGH	POLOW	POLOSE	PSERVATIO	DiffPrice	DiffOPEN
2006-04-27	636.5	636.5	636.5	633.3	4	-3.2	-2.3
2006-05-01	658	658	658	657.5	4	-0.5	6.2
2006-05-02	666	666	666	664.7	4	-1.3	8.5
2006-05-03	669.2	669.2	669.2	665.9	4	-3.3	4.5
2006-05-08	679	679	679	677.8	4	-1.2	-3.2
2006-05-11	723	723	723	719.8	4	-3.2	19.3
2006-06-22	588	588	582.2	582.2	4	-5.8	0.5

ภาพที่ 3.7 ตัวอย่างการจัดกลุ่มข้อมูลที่ observation = 4

เมื่อจัดกลุ่มแยกตาม observation แล้วหาค่า หาค่าเฉลี่ยของราคาที่ยื่นแปลง และค่าเฉลี่ยของ ส่วนต่างราคาเปิดในทุก observation ดังภาพที่ 3.8

OBSERVATION	AVG_PRICE	AVG_PRICEOPEN
0	1.804761905	-0.091808874
1	20.96363636	2.518181818
2	40.4	-1.9
3	63.7	6.4
4	-4.279166667	1.95
5	-13.75833333	-0.125
6	-25.9	4.833333333
7	-38	-1.533333333
8	1.203703704	0.240740741

ภาพที่ 3.8 ตัวอย่างค่าเฉลี่ยราคาแยกตาม observation

การแปลงข้อมูลหลังจากการพยากรณ์ผลเพื่อให้ได้ข้อมูลค่า ราคาทองคำโดยจากภาพที่ 3.8 สมมุติว่า ผลของการพยากรณ์ได้ข้อมูลออกมาเป็น 0 เมื่อต้องการแปลงค่ากลับเป็นราคาทองคำ จะทำโดยการ นำ 0 ไปเทียบกับตารางค่าเฉลี่ย ได้ค่า $AVG_PRICE = 1.805$ และ $AVG_PRICEOPEN = -0.092$ หลังจากนั้นจะต้องไปนำค่าราคาปิดของวันก่อนหน้าวันที่พยากรณ์ มาจะได้ว่า

$$Prediction\ data = old\ price + avg\ price + avg\ price\ open \tag{3.4}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลลัพธ์การทดลอง

4.1 การเตรียมทรัพยากรของการทดลอง

จะเป็นการเลือก ชนิดของฮิตเดนมาร์คอฟโมเดล การกำหนดค่าตัวแปรหลักๆที่เกี่ยวข้อง ในการคำนวณของกระบวนการการรู้จำและทำนายผล โดยมีรายละเอียดดังตารางที่ 4.1

ตารางที่ 4.1 รายละเอียดของทรัพยากรในการทดลอง

ชนิดของฮิตเดนมาร์คอฟโมเดล	ฟูลีคอนเนคโทโปโลยี (Fully connect topology) เลฟ-ไรท์ โทโปโลยี (Left-Right topology)
จำนวนสถานะ (N)	50, 40 และ 30
จำนวนชนิดข้อมูลการสังเกต (M)	เป็นรูปแบบของโมเดล คือรูปแบบของข้อมูลการสังเกต มีรูปแบบทั้งหมด 48 (หรือ $M = 48$) รูปแบบ
ตัวแปรหลักๆที่เกี่ยวข้อง	<ul style="list-style-type: none">• $A = (a_{ij})_{N \times N}$• $B = (b_{j,k})_{N \times M}$• $\pi = (\pi_i)_N$
จำนวนข้อมูลทั้งหมด	• 2018 วัน หรือ $T=2018$
จำนวนข้อมูลที่พยากรณ์	• 530 วัน
ขนาดของชุดข้อมูลเข้า (t)	อิงตาม ขนาดของ Window Sliding Algorithm โดยกำหนดให้ $size=5$

4.2 ผลการทดลอง

4.2.1 ผลการพยากรณ์ราคาทองคำ

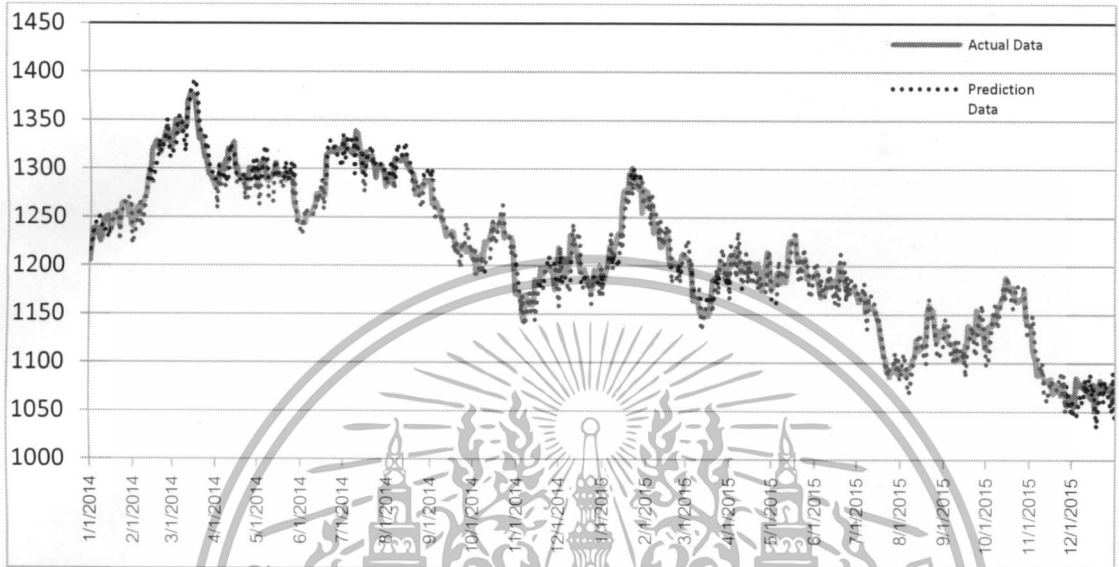
เมื่อผ่านการเตรียมทรัพยากรแล้วจึงได้ทำการทดลองตามที่ได้ออกแบบไว้ (จากหัวข้อที่ 3.3) และ เมื่อนำผลการทำนายที่ได้มาแปลงเป็นราคาทองคำ (ด้วยขั้นตอนหัวข้อที่ 3.4) โดยการทดลองจะทำการทดลองเป็น 3 กรณีตามการตั้งค่า จำนวนสถานะ (N) ซึ่งผลการทดลองมีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.1 ฟูลีคอนเนคโทโพลี (Fully connect topology)

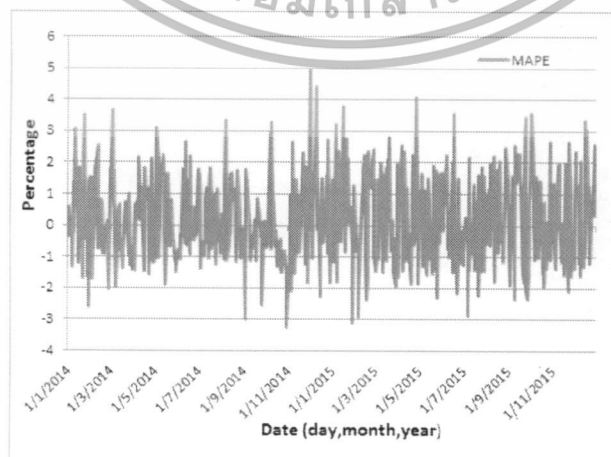
- ผลการทดลองในกรณีที่มีสถานะ $N = 50$

หลังจากที่นำ O_T เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 50$ และ ทำการแปลงค่าผลการกลับมาเป็นค่าเงินบาท/ดอลลาร์สหรัฐฯแล้วจึงนำข้อมูลมาพล็อตกราฟ ดังภาพที่ 4.1



ภาพที่ 4.1 กราฟผลการทำนายราคาทองคำแบบฟูลีคอนเนคโทโพลี ที่ $N = 50$

จากภาพที่ 4.1 คือกราฟที่เปรียบเทียบระหว่างราคาทองคำที่ได้จากการทำนายโดยฮิดเดน มาร์คอฟโมเดล(เส้นประ)และข้อมูลจริง(เส้นทึบสีเทา)พบว่า กราฟทั้ง 2 เส้นมีความใกล้เคียงกันมาก และเมื่อทำการวัดค่าความคลาดเคลื่อนระหว่างการทำนายและข้อมูลจริงใน ด้วยวิธี Mean Absolute Percentage Error (MAPE) พบว่าค่ากราฟ Error ในแต่ละวันจะแกว่งในลักษณะดังภาพที่ 4.2 และ ถ้าทำการคิดค่าแบบโดยรวมจะได้ค่า $MAPE = 1.1805\%$

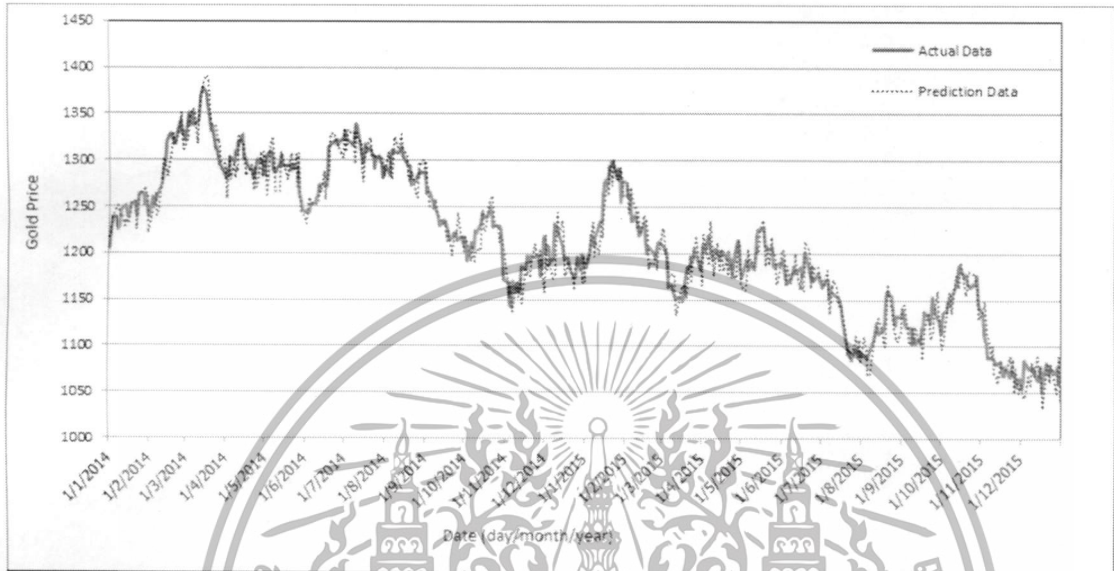


ภาพที่ 4.2 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 50$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

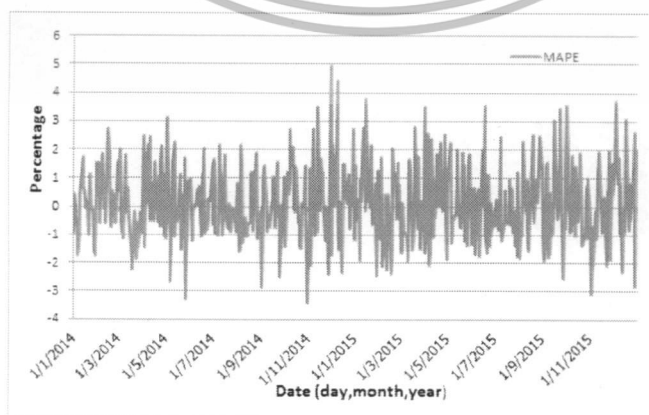
- ผลการทดลองในกรณีที่สถานะ $N = 40$

หลังจากที่นำ O_7 เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 40$ และ ทำการแปลงค่าผลการกลับมาเป็นค่าเงินบาท/ดอลลาร์สหรัฐฯแล้วจึงนำข้อมูลมาพล็อตกราฟ ดังภาพที่ 4.3



ภาพที่ 4.3 กราฟผลการทำนายราคาทองคำ แบบฟิวชันเนลเวทไปโลจี ที่ $N = 40$

จากภาพที่ 4.3 คือกราฟที่เปรียบเทียบระหว่างราคาทองคำที่ได้จากการทำนายโดยฮิดเดน มาร์คอฟโมเดล(เส้นประ)และข้อมูลจริง(เส้นทึบสีเทา)พบว่า กราฟทั้ง 2 เส้นมีความใกล้เคียงกันมาก และเมื่อทำการวัดค่าความคลาดเคลื่อนระหว่างการทำนายและข้อมูลจริงใน ด้วยวิธี Mean Absolute Percentage Error (MAPE) พบว่าค่ากราฟ Error ในแต่ละวันจะแกว่งในลักษณะดังภาพที่ 4.4 และ ถ้าทำการคิดค่าแบบโดยรวมจะได้ค่า $MAPE = 1.0727\%$

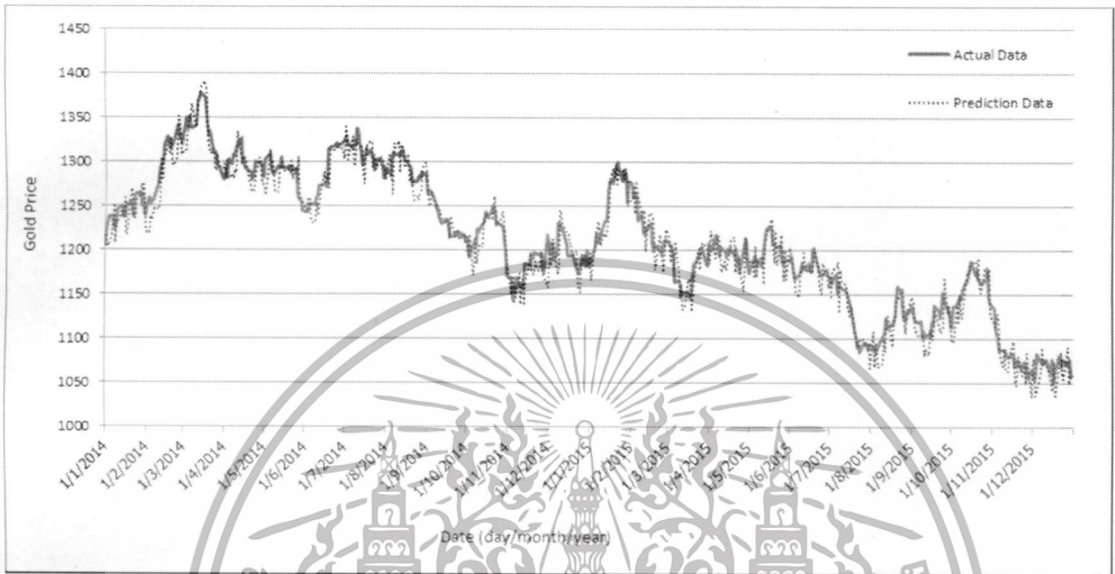


ภาพที่ 4.4 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 40$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

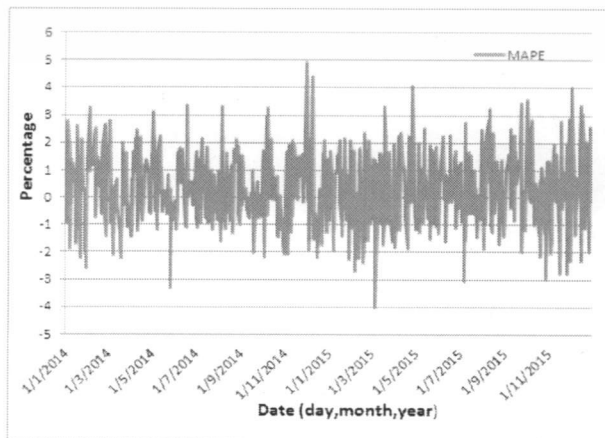
- ผลการทดลองในกรณีที่สถานะ $N = 30$

หลังจากที่นำ O_T เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 30$ และ ทำการแปลงค่าผลการกลับมาเป็นค่าเงินบาท/ดอลลาร์สหรัฐฯ แล้วจึงนำข้อมูลมาพล็อตกราฟ ดังภาพที่ 4.5



ภาพที่ 4.5 กราฟผลการทำนายราคาทองคำแบบฟูลคอนเนคทีฟโพลีโลยีที่ $N = 30$

จากภาพที่ 4.5 คือกราฟที่เปรียบเทียบระหว่างราคาทองคำที่ได้จากการทำนายโดยฮิดเดน มาร์คอฟโมเดล(เส้นประ)และข้อมูลจริง(เส้นที่บดสีเทา)พบว่า กราฟทั้ง 2 เส้นมีความใกล้เคียงกันมาก และเมื่อทำการวัดค่าความคลาดเคลื่อนระหว่างการทำนายและข้อมูลจริงใน ด้วยวิธี Mean Absolute Percentage Error (MAPE) พบว่าค่ากราฟ Error ในแต่ละวันจะแกว่งในลักษณะดังภาพที่ 4.6 และ ถ้าทำการคิดค่าแบบโดยรวมจะได้ค่า $MAPE = 1.2035\%$



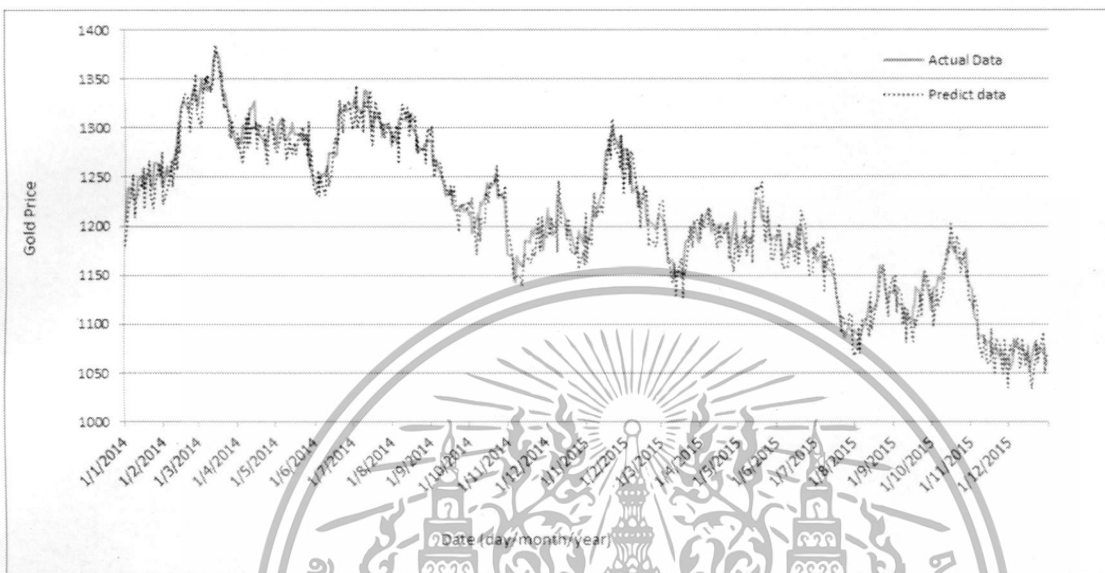
ภาพที่ 4.6 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 30$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.2 เลฟ-ไรท์ โทโปโลยี (Left-Right topology)

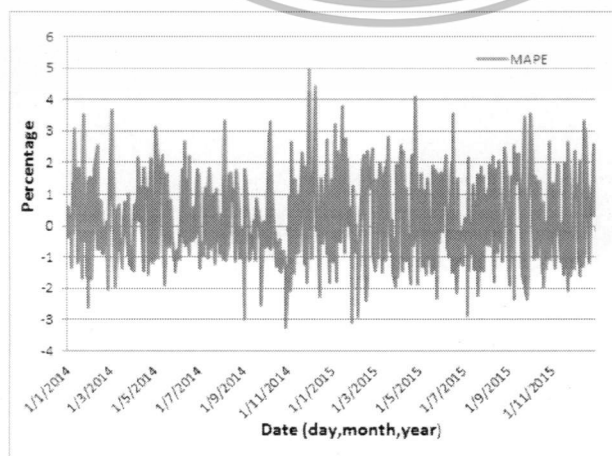
- ผลการทดลองในกรณีที่สถานะ $N = 50$

หลังจากที่นำ O_T เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 50$ และ ทำการแปลงค่าผลการกลับมาเป็นค่าเงินบาท/ดอลลาร์สหรัฐแล้วจึงนำข้อมูลมาพล็อตกราฟ ดังภาพที่ 4.1



ภาพที่ 4.7 กราฟผลการทำนายราคาทองคำ แบบเลฟ-ไรท์ โทโปโลยี ที่ $N = 50$

จากภาพที่ 4.1 คือกราฟที่เปรียบเทียบระหว่างราคาทองคำที่ได้จากการทำนายโดยอิตเดน มาร์คอฟโมเดล (เส้นประ) และข้อมูลจริง (เส้นทึบสีเทา) พบว่า กราฟทั้ง 2 เส้นมีความใกล้เคียงกันมาก และเมื่อทำการวัดค่าความคลาดเคลื่อนระหว่างการทำนายและข้อมูลจริงใน ด้วยวิธี Mean Absolute Percentage Error (MAPE) พบว่าค่ากราฟ Error ในแต่ละวันจะแกว่งในลักษณะดังภาพที่ 4.2 และ ถ้าทำการคิดค่าแบบโดยรวมจะได้ค่า $MAPE = 1.1839\%$



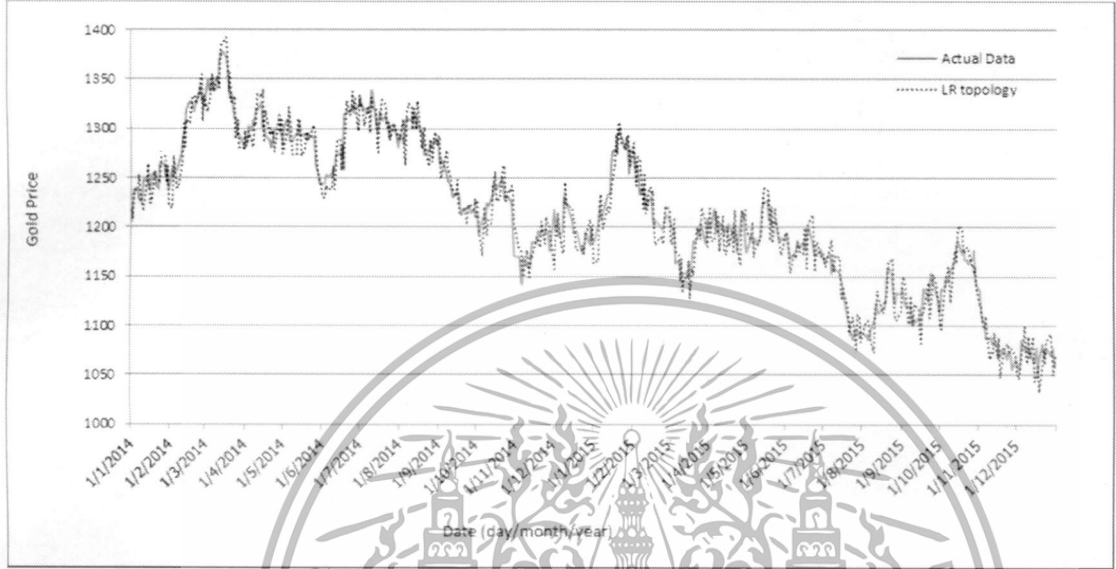
ภาพที่ 4.8 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 50$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

● ผลการทดลองในกรณีที่สถานะ $N = 40$

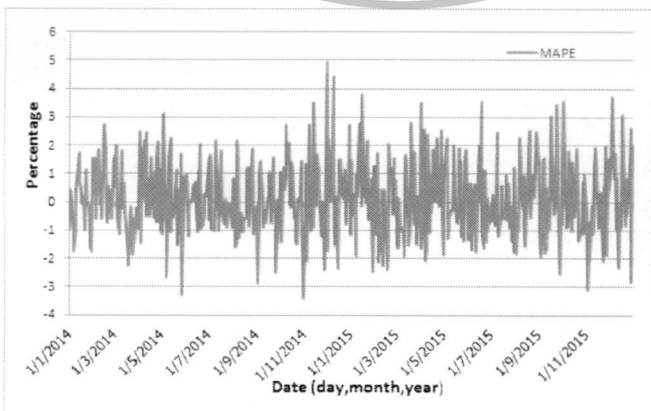
หลังจากที่นำ O_7 เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 40$

และ ทำการแปลงค่าผลการกลับมาเป็นค่าเงินบาท/ดอลลาร์สหรัฐฯแล้วจึงนำข้อมูลมาพล็อตกราฟ ดังภาพที่ 4.3



ภาพที่ 4.9 กราฟผลการทำนายราคาทองคำ แบบเลฟ-โล โทโปโลยี ที่ $N = 40$

จากภาพที่ 4.3 คือกราฟที่เปรียบเทียบระหว่างราคาทองคำที่ได้จากการทำนาย โดยฮิดเดน มาร์คอฟโมเดล(เส้นประและข้อมูลจริง(เส้นทึบสีเทา)พบว่า กราฟทั้ง 2 เส้นมีความใกล้เคียงกัน มาก และเมื่อทำการวัดค่าความคลาดเคลื่อนระหว่างการทำนายและข้อมูลจริงใน ด้วยวิธี Mean Absolute Percentage Error (MAPE) พบว่าค่ากราฟ Error ในแต่ละวันจะแกว่งในลักษณะดังภาพที่ 4.4 และ ถ้าทำการคิดค่าแบบโดยรวมจะได้คือ $MAPE = 1.1117\%$

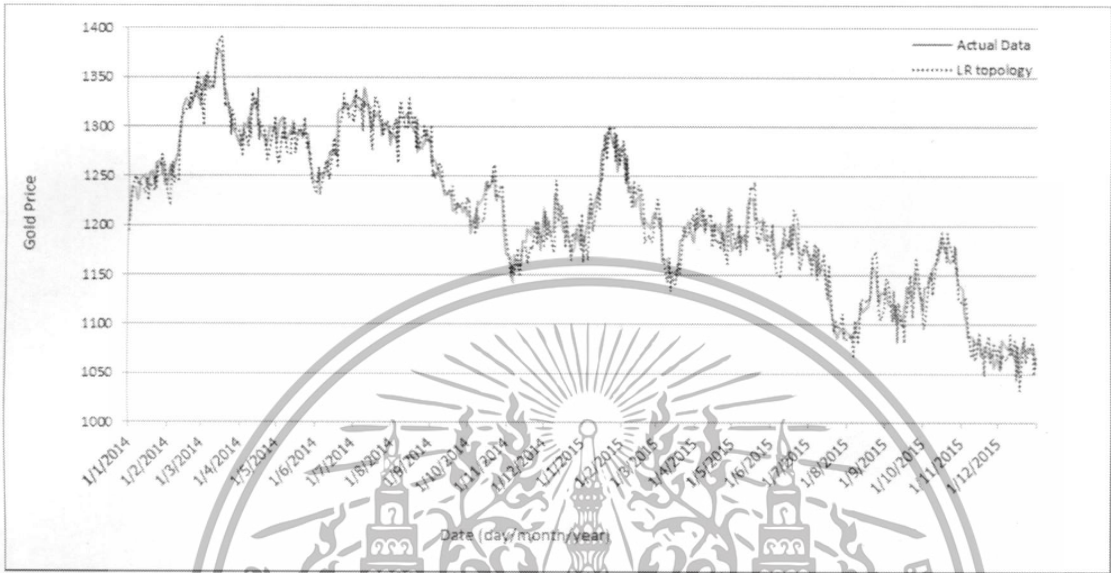


ภาพที่ 4.10 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 40$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

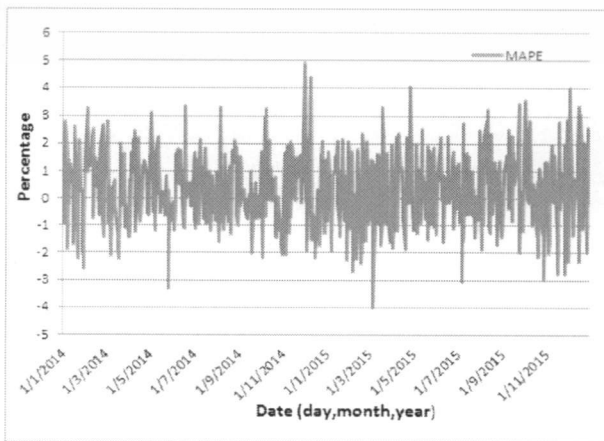
● ผลการทดลองในกรณีที่สถานะ $N = 30$

หลังจากที่นำ O_T เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 30$ และ ทำการแปลงค่าผลการกลับมาเป็นค่าเงินบาท/ดอลลาร์สหรัฐแล้วจึงนำข้อมูลมาพล็อตกราฟ ดังภาพที่ 4.5



ภาพที่ 4.11 กราฟผลการทำนายราคาทองคำ แบบเลฟ-โวล-โวล-โวล ที่ $N = 30$

จากภาพที่ 4.5 คือกราฟที่เปรียบเทียบระหว่างราคาทองคำที่ได้จากการทำนายโดยอิตเดน มาร์คอฟโมเดล(เส้นประ)และข้อมูลจริง(เส้นทึบสีเทา)พบว่า กราฟทั้ง 2 เส้นมีความใกล้เคียงกันมาก และเมื่อทำการวัดค่าความคลาดเคลื่อนระหว่างการทำนายและข้อมูลจริงใน ด้วยวิธี Mean Absolute Percentage Error (MAPE) พบว่าค่ากราฟ Error ในแต่ละวันจะแกว่งในลักษณะดังภาพที่ 4.6 และ ถ้าทำการคิดค่าแบบโดยรวมจะได้ค่า $MAPE = 1.0980\%$



ภาพที่ 4.12 กราฟแสดงค่า MAPE ในแต่ละวันที่ $N = 30$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ผลการพยากรณ์แนวโน้มราคาทองคำ

ผลพยากรณ์แนวโน้มของราคาทองคำ จะเป็นการนำ ข้อมูลจริง(actual data) ของวันที่ทำการพยากรณ์ ว่าราคาทองคำในวันนั้นมีราคาเพิ่มขึ้น หรือ ลดลง เปรียบเทียบกับผลการพยากรณ์ (prediction data) ว่าตรงกันหรือไม่

- ผลการทดลองในกรณีที่ สถานะ $N = 50$

หลังจากที่นำ O_T เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 50$ และทำการเปรียบเทียบผลแล้ว มีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 51.7% และเมื่อจำแนก ข้อมูลแนวโน้มออกเป็นแนวโน้มขาขึ้น และแนวโน้มขาลง จะได้ผลว่าในช่วงแนวโน้มขาขึ้น มีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 57.6% และแนวโน้มขาลงมีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 46.1%

- ผลการทดลองในกรณีที่ สถานะ $N = 40$

หลังจากที่นำ O_T เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 40$ และทำการเปรียบเทียบผลแล้ว มีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 52.9% และเมื่อจำแนก ข้อมูลแนวโน้มออกเป็นแนวโน้มขาขึ้น และแนวโน้มขาลง จะได้ผลว่าในช่วงแนวโน้มขาขึ้น มีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 50.9% และแนวโน้มขาลงมีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 55.3%

- ผลการทดลองในกรณีที่ สถานะ $N = 30$

หลังจากที่นำ O_T เข้าสู่กระบวนการ Training Testing ด้วยการตั้งค่าสถานะ ที่ $N = 30$ และทำการเปรียบเทียบผลแล้ว มีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 49.6% และเมื่อจำแนก ข้อมูลแนวโน้มออกเป็นแนวโน้มขาขึ้น และแนวโน้มขาลง จะได้ผลว่าในช่วงแนวโน้มขาขึ้น มีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 46.3% และแนวโน้มขาลงมีค่าความน่าจะเป็นในการทายผลที่ถูกต้องอยู่ที่ 52.8%

บทที่ 5

สรุปผลการทดลอง

5.1 สรุปผลการทดลอง

จากการทดลองทำนายราคาทองคำโดยใช้ฮิดเดนมาร์คอฟโมเดล โดยใช้วิธีการแปลงข้อมูลการสังเกตในรูปแบบของกราฟแท่งเทียน ที่แบ่งรูปแบบโดยการใช้อัตราส่วนของแท่งเทียนและอัตราส่วนของเงา พบว่า สามารถนำมาใช้เป็นข้อมูลเข้า (Observation data) ให้กับฮิดเดนมาร์คอฟโมเดลได้ อีกทั้ง ข้อมูลที่ถูกพยากรณ์ออกมาของในแต่ละการทดลองพบว่าค่อนข้างมีความแม่นยำโดยดูจากกราฟเปรียบเทียบ ซึ่งผลลัพธ์ไม่ต่างกันมากนักของแต่ละการทดลอง

จากผลของการทดลองในแต่ละกรณีของการปรับสถานะของฮิดเดนมาร์คอฟโมเดลพบว่า กราฟการเปรียบเทียบระหว่าง ผลลัพธ์ที่พยากรณ์ได้กับค่าข้อมูลจริงของทุกการทดลองมีความใกล้เคียงกันอย่างมาก โคนสถานะที่ $N = 40$ ให้ผลลัพธ์ที่ดีที่สุด อีกทั้งเมื่อวัดค่าความคลาดเคลื่อนด้วย MAPE พบว่าได้ค่าที่ต่ำมาก

5.2 ปัญหาและอุปสรรค

5.2.1 เนื่องจากฮิดเดนมาร์คอฟเป็นการคำนวณที่ซับซ้อนค่อนข้างมาก อีกทั้งยังใช้ข้อมูลที่มีปริมาณมาก ดังนั้นทำให้เกิดความล่าช้าในการคำนวณ

5.2.2 การแปลงผลของการทำนายใช้วิธีการหาค่าเฉลี่ยการเปลี่ยนแปลงราคาในแต่ละวันรวมกับค่าเฉลี่ย ของราคาเปิด ซึ่งยังให้ผลที่ไม่แม่นยำมากนัก

5.2.3 ข้อมูลปัจจัยที่ได้เลือกใช้ในการทดลองนี้ ใช้เพียงแต่ข้อมูลของราคาทองคำย้อนหลังเพียงตัวเดียว ดังนั้นยังมีปัจจัยอีกหลายปัจจัยที่ส่งผลกระทบต่อ การเปลี่ยนแปลงของราคาทองคำที่ยังไม่ได้ถูกนำมาใช้ในการทดลอง

5.3 แนวทางการพัฒนาต่อ

5.3.1 ศึกษาเพิ่มเติมเกี่ยวกับปัจจัยต่างๆที่ส่งผลกระทบต่อ การเปลี่ยนแปลงของราคาทอง เพื่อนำมาใช้เพิ่มเติมในการทดลอง

5.3.2 ศึกษาหาวิธีการแปลงผลของการทำนายที่ให้ผลลัพธ์ที่แม่นยำมากขึ้น

5.3.3 ศึกษาเพิ่มเติมเพื่อประยุกต์การทดลองนี้เข้ากับ ซอฟต์แวร์ประยุกต์(Application) เพื่อให้ง่ายต่อการใช้งานจริงๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ไฟโรจน์ วิเชนชาติ “วิเคราะห์เจาะลึกหุ้นด้วย Candlesticks” [online]

<http://www.richerstock.com/candlepattern/cdhome.htm>

มณสิณี อรรถศิริปัญญา, วิสิณี ประสิทธิ์ชัยกุล. 2556. “ระบบวิเคราะห์แนวโน้มหุ้นด้วยอิตเดนมาร์คอฟโมเดล” วิทยานิพนธ์วิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศคอมพิวเตอร์. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

อังกร เอี่ยมอาภรณ์. 2552. “การศึกษาอิตเดนมาร์คอฟโมเดล” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

Gold History Data. [Online] <http://www.investing.com/commodities/gold-historical-data>

JSW. Candlestick interpretations table. [Online].

<http://www.investorsforum.com/InvestorClub/TechnicalAnalysis/CandlestickInterpretationsTable/tabid/760/Default.aspx>

Lawrence R Rabiner. 1989. “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition.” *Proceedings of the IEEE*. 77(2): 257-286.

Md. RafiulHassan, BaikunthNath. 2005. “StockMarket Forecasting Using Hidden Markov Model: A New Approach.” *Proceedings of the 2005 5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

รายละเอียดซอร์สโค้ด

ก. Class Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data;
using System.IO;
using System.Runtime.Serialization.Formatters.Binary;
using System.Globalization;
using System.Threading;
namespace HMMTRAINING
{
    class Program
    {
        SettingVar setV = new SettingVar();
        Stream stream;
        BinaryFormatter bformatter;
        HMMCalculate hmmCal;
        static String SQL;
        static void Main(string[] args)
        {
            Thread.CurrentThread.CurrentCulture = new CultureInfo("en-US");
            Thread.CurrentThread.CurrentUICulture = new CultureInfo("en-US");
            int i, j, mp, k;
            SettingVar setVp = new SettingVar();
            Program pgm = new Program();
            int[,] ObsArrayPattern = new int[1000, 5];
            int[,] EachRowPatt = new int[400]; // Rows Number for each pattern
            int[,] DsPredic = new int[48, 5];
            int[] PatTrain = new int[48];
            int NumrowDt = 0;
            int NumColumnDt = 0;
            int sizedata, Psizeedata;
            Double[,] MAForTest = new Double[setVp._MaxN, setVp._MaxN];
            Double[,] MBForTest = new Double[setVp._MaxN, setVp._MaxO];
            Double[] PIForTest = new Double[setVp._MaxN];
            String chkFrPatt;
            DataTable EachPattData = new DataTable();
            AppProcess actionData = new AppProcess();
            SQL = "SELECT OBSERVATION FROM OB_CONVERTSET1 " +           #select OBS
                "ORDER BY CONMONTH ASC";
            EachPattData = actionData.SQLReadData(SQL);
            sizedata = EachPattData.Rows.Count;
            int[] Storedata = new int[sizedata];
            Console.WriteLine("=====");
            Console.WriteLine("====++START Experiment++====");
            Console.WriteLine("SET STATE : " + setVp._MaxN.ToString());
            Console.WriteLine("PATTERN : " + setVp._MaxO.ToString());
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Console.WriteLine(".....");
int counter = 0;
for (i = 0; i < Storedata.GetLength(0); i++)
{
    Storedata[i] = (int)EachPattData.Rows[i]["OBSERVATION"];
}

foreach (var window in GetWindows(Storedata, 5)) //save OBS into DBS
{
    chkFrPatt = window[4].ToString();
    StorePattern(window, chkFrPatt);
    ShowArray(window, counter);
}

for (k = 0; k < 48; k++)
{
    chkFrPatt = k.ToString();
    SQL = "SELECT * FROM OB_PATTERN" + chkFrPatt;
    EachPattData = actionData.SQLReadData(SQL);
    NumrowDt = EachPattData.Rows.Count; // จำนวนแถว ของ OBS ตามแต่ละ pattern
    NumColumnDt = EachPattData.Columns.Count; //window 5
    Array.Clear(ObsArrayPattern, 0, ObsArrayPattern.Length);
    for (i = 0; i < NumrowDt; i++)
    {
        for (j = 0; j < NumColumnDt; j++)
        {
            if (j == 0)
            {
                ObsArrayPattern[i, j] = (int)EachPattData.Rows[i]["PatData1"];
            }
            else if (j == 1)
            {
                ObsArrayPattern[i, j] = (int)EachPattData.Rows[i]["PatData2"];
            }
            else if (j == 2)
            {
                ObsArrayPattern[i, j] = (int)EachPattData.Rows[i]["PatData3"];
            }
            else if (j == 3)
            {
                ObsArrayPattern[i, j] = (int)EachPattData.Rows[i]["PatData4"];
            }
            else if (j == 4)
            {
                ObsArrayPattern[i, j] = (int)EachPattData.Rows[i]["PatData5"];
            }
        }
    }
    // End Loop Get Obs
    pgm.TrainModel(ObsArrayPattern, chkFrPatt, NumrowDt);
}

SQL = "SELECT OBSERVATION FROM OB_OBSPREDICT " +
    "ORDER BY CONMONTH ASC";
EachPattData = actionData.SQLReadData(SQL);
Pszedata = EachPattData.Rows.Count;
int[] PredicData = new int[Pszedata];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for (i = 0; i < PredicData.GetLength(0); i++)
{
    PredicData[i] = (int)EachPattData.Rows[i]["OBSERVATION"];
}

counter = 0; // เริ่ม predict
foreach (var window in GetWindows(PredicData, 5)) // train data
{
    counter++;
    chkFrPatt = window[4].ToString();
    NumrowDt = EachPattData.Rows.Count; // จำนวนแถว ของ OBS ตามแต่ละ pattern
    NumColumnDt = EachPattData.Columns.Count; //window 5
    Array.Clear(ObsArrayPattern, 0, ObsArrayPattern.Length);
    SQL = "select * from OB_OBSSTB WHERE TRAINED = 1";
    EachPattData.Clear();
    EachPattData = actionData.SQLReadData(SQL);
    Array.Clear(PatTrain, 0, PatTrain.Length);
    Console.WriteLine("Test Model");
    // Loop record model are trained
    for (int pat = 0; pat < EachPattData.Rows.Count; pat++)
    {
        PatTrain[pat] = Convert.ToInt32(EachPattData.Rows[pat]["OBSERVATION"]);
    }
    for (mp = 0; mp < 48; mp++) // 48 วันก่อน
    {
        DsPredic[mp, 0] = window[0];
        DsPredic[mp, 1] = window[1];
        DsPredic[mp, 2] = window[2];
        DsPredic[mp, 3] = window[3];
        DsPredic[mp, 4] = mp;
    }
    pgm.TestModel(DsPredic, PatTrain, chkFrPatt, EachPattData.Rows.Count, counter);
}
}

public double[,] GenerateMatrixA()
{
    int i, j;
    double SumRow;
    Double[,] MatxA = new Double[setV_MaxN, setV_MaxN];
    Random RanA = new Random();
    for (i = 0; i < setV_MaxN; i++)
    {
        for (j = 0; j < setV_MaxN; j++)
        {
            MatxA[i, j] = RanA.NextDouble();
        }
    }
    //Normalize MatrixA
    for (i = 0; i < setV_MaxN; i++)
    {
        SumRow = 0;
        for (j = 0; j < setV_MaxN; j++)
        {
            SumRow += MatxA[i, j];
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }

    for (j = 0; j < setV_MaxN; j++)
    {
        MatxA[i, j] = MatxA[i, j] / SumRow;
    }
}
return MatxA;
}

public double[,] GenerateMatrixB()
{
    int i, j;
    double SumRow;
    Double[,] MatxB = new Double[setV_MaxN, setV_MaxO];
    Random RanA = new Random();
    //Create MatrixB
    for (i = 0; i < setV_MaxN; i++)
    {
        for (j = 0; j < setV_MaxO; j++)
        {
            MatxB[i, j] = RanA.NextDouble();
        }
    }
    //Normalize MatrixB
    for (i = 0; i < setV_MaxN; i++)
    {
        SumRow = 0;
        for (j = 0; j < setV_MaxO; j++)
        {
            SumRow += MatxB[i, j];
        }

        for (j = 0; j < setV_MaxO; j++)
        {
            MatxB[i, j] = MatxB[i, j] / SumRow;
        }
    }
    return MatxB;
}

public double[] GeneratePI()
{
    int i;
    double SumRow;
    Double[] MatxPI = new Double[setV_MaxN];
    Random RanA = new Random();
    for (i = 0; i < setV_MaxN; i++)
    {
        MatxPI[i] = RanA.NextDouble();
    }
    //Normalize PI
    SumRow = 0;
    for (i = 0; i < setV_MaxN; i++)
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
for (i = 0; i < setV._MaxN; i++)
{
    MatxPI[i] /= SumRow;
}
return MatxPI;
}
private static IEnumerable<T[]> GetWindows<T>(T[] array, int windowHeight)
{
    for (var y = 0; y < array.GetLength(0) - windowHeight + 1; y++)
    {
        var slice = new T[windowHeight];
        CopyArray(array, y, slice, 0, windowHeight);
        yield return slice;
    }
}
private static void CopyArray<T>(T[] src, int srcY, T[] dst, int dstY, int height)
{
    for (var y = 0; y < height; y++)
    {
        dst[dstY + y] = src[srcY + y];
    }
}
private static void StorePattern<T>(T[] WinData, string PattName)
{
    int IndexPat;
    DataTable InPatt;
    AppProcess actionData = new AppProcess();
    SQL = "select TOP 1 PatIndex from OB_PATTERN" + PattName + " ORDER BY PatIndex DESC";
    InPatt = actionData.SQLReadData(SQL);
    if (InPatt.Rows.Count > 0)
    {
        IndexPat = Convert.ToInt16(InPatt.Rows[0]["PatIndex"]) + 1;
        SQL = "INSERT INTO OB_PATTERN" + PattName + "(PatIndex,PatData1,PatData2,PatData3,PatData4,PatData5) +
            "values (" + IndexPat +
            "." + WinData[0] +
            "." + WinData[1] +
            "." + WinData[2] +
            "." + WinData[3] +
            "." + WinData[4] +
            ")";
        actionData.SQLQueryAction(SQL);
    }
    else
    {
        SQL = "INSERT INTO OB_PATTERN" + PattName + "(PatIndex,PatData1,PatData2,PatData3,PatData4,PatData5) +
            "values (" + 1 +
            "." + WinData[0] +
            "." + WinData[1] +
            "." + WinData[2] +
            "." + WinData[3] +
            "." + WinData[4] +
            ")";
        actionData.SQLQueryAction(SQL);
    }
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

private static void ShowArray<T>(T[] array, int counter)
{
    StreamWriter file = new StreamWriter("C:\Users\HMM\Dropbox\MS-VYEAR2\CODE\Temp\ResultPatternTrain.txt", true);
    string lines = counter + " " + array[0] + " " + array[1] + " " + array[2] + " " + array[3] + " " + array[4];
    file.WriteLine(lines);
    file.Close();
}

public void TrainModel(int[,] Obs, String Pattern, int row)
{
    int turn;
    int turnNaN = 0;
    Object Trained;
    DataTable Chk = new DataTable();
    AppProcess actionData = new AppProcess();
    Program pg = new Program();

    SQL = "SELECT TRAINED FROM OB_OBSTB WHERE OBSERVATION=" + Pattern;
    Chk = actionData.SQLReadData(SQL);

    Trained = Chk.Rows[0]["TRAINED"];
    if (Trained == DBNull.Value)
    {
        SQL = "UPDATE OB_OBSTB SET TRAINED = 'true' WHERE OBSERVATION=" + Pattern;
        actionData.SQLQueryAction(SQL);
    }
ABC:
    turn = 0;
    HMMCalculate hmmCal = new HMMCalculate(pg.GenerateMatrixA(), pg.GenerateMatrixB(), pg.GeneratePi(), Convert.ToInt16(Pattern));
    for (; )
    {
        turn++;
        hmmCal.ForwardProcedure(Obs, row);
        hmmCal.BackwardProcedure(Obs, row);
        hmmCal.ReEstimation(Obs, row);
        if (hmmCal.ProbMBarPatt != hmmCal.ProbMBarPatt)
        {
            turnNaN++;
            Console.WriteLine("Number of Find NaN " + turnNaN);
            if (turnNaN < 5)
            {goto ABC;}
        }

        StreamWriter file = new StreamWriter("C:\Users\HMM\Dropbox\MS-VYEAR2\CODE\Temp\TxtFileHMMparameterModel" +
(Convert.ToInt16(Pattern) + 1) + ".txt", true);

        if (hmmCal.ProbMPatt < hmmCal.ProbMBarPatt)
        {
            # write to file
            string lines = "ProbPattern " + (Pattern + 1) + " : " + hmmCal.ProbMPatt + "%\n";
            file.WriteLine(lines);
            file.Close();
            hmmCal.CopyModelBarToMainModel();
            Console.WriteLine("Copy Model");
        }
    }
else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

# write to file
string lines = "ProbPattern " + (Pattern + 1) + " : " + hmmCal.ProbMPatt + ".txt";
file.WriteLine(lines);
file.Close();
#Serialize the object
stream = File.Open("C:\Users\HMM\Dropbox\MS-VYEAR2\CODE\Temp\HMMparameterModel" + (Convert.ToInt16(Pattern) + 1) + ".dat",
FileMode.Create, FileAccess.Write);
bformatter = new BinaryFormatter();
Console.WriteLine("Writing HMMparameterMedel " + (Convert.ToInt16(Pattern) + 1) + ".dat Information");
bformatter.Serialize(stream, hmmCal);
stream.Close();
break;
}
}
Console.WriteLine(" Pattern " + (Convert.ToInt16(Pattern) + 1) + " Exit For: " + turn);
Console.WriteLine(" ");
}
public void TestModel(int[,] Obs, int[] PatTrained, String Pattern, int RowPatTrained, int RowToPredicDate)
{
SettingVar setVp = new SettingVar();
Double[,] MAForTest = new Double[setVp._MaxN, setVp._MaxN];
Double[,] MBForTest = new Double[setVp._MaxN, setVp._MaxO];
Double[,] PIForTest = new Double[setVp._MaxN];
for (int i = 0; i < RowPatTrained; i++)
{
hmmCal = null;
stream = File.Open("C:\Users\HMM\Dropbox\MS-VYEAR2\CODE\Temp\HMMparameterModel" + (PatTrained[i] + 1) + ".dat", FileMode.Open);
bformatter = new BinaryFormatter();
hmmCal = (HMMCalculate)bformatter.Deserialize(stream);
stream.Close();
MAForTest = hmmCal.MA;
MBForTest = hmmCal.MB;
PIForTest = hmmCal.PI;
hmmCal = new HMMCalculate(MAForTest, MBForTest, PIForTest, 0);
hmmCal.ForwardProcedurObs(Obs, Obs.GetLength(0), PatTrained[i]);
}
HMMCalculate.FindMaxProb(Convert.ToInt16(Pattern), RowToPredicDate);
}
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

๗. Class HMMCalculate.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Runtime.Serialization;
using System.IO;
using System.Data;
using System.Threading;
using System.Globalization;
namespace HMMTRAINING
{
    [Serializable()]
    class HMMCalculate : ISerializable
    {
        int i, k, t, j, r, q;
        int Pattern;
        public double[,] MA;
        public double[,] MABar;
        public double[,] MB;
        public double[,] MBBar;
        public double[] PI;
        public double[] PBar;
        public int[,] ObsArrayPatt;
        public double[,] Alpha;
        public double[,] Beta;
        public double[] ObsProb;
        public double[] ObsProbBar;
        public double[] ObsProbTest;
        public double ProbMPatt;
        public double ProbMBarPatt = 0;
        public double[] AlphaBeta;
        SettingVar setV = new SettingVar();
        static List<ProbTestData> ProbList = new List<ProbTestData>();
        public HMMCalculate()
        {
        }
        public HMMCalculate(Double[,] MatrixA, Double[,] MatrixB, Double[] MatrixPI, int Patt)
        {
            MA = new double[setV._MaxN, setV._MaxN];
            MB = new double[setV._MaxN, setV._MaxO];
            PI = new double[setV._MaxN];
            Pattern = Patt;
            MA = MatrixA;
            MB = MatrixB;
            PI = MatrixPI;
            MABar = new double[setV._MaxN, setV._MaxN];
            MBBar = new double[setV._MaxN, setV._MaxO];
            PBar = new double[setV._MaxN];
            ObsArrayPatt = new int[setV._MaxRows, setV._LengthPat];
            Alpha = new double[setV._MaxRows, setV._MaxN, setV._LengthPat];

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์เพื่อการเรียนการสอนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Beta = new double[setV_MaxRows, setV_MaxN, setV_LengthPat];
ObsProb = new double[setV_MaxRows];
ObsProbBar = new double[setV_MaxRows];
ObsProbTest = new double[setV_MaxRows];
AlphaBeta = new double[setV_MaxN];
}

public HMMCalculate(SerializationInfo info, StreamingContext context)
{
    MA = (Double[,])info.GetValue("MatrixA", typeof(Double[,]));
    MB = (Double[,])info.GetValue("MatrixB", typeof(Double[,]));
    PI = (Double[])info.GetValue("PI", typeof(Double[]));
    ObsProb = (Double[])info.GetValue("ObsProb", typeof(Double[]));
    ProbMPatt = (Double)info.GetValue("ProbPattern", typeof(Double));
}

public void GetObjectData(SerializationInfo info, StreamingContext context)
{
    info.AddValue("MatrixA", MA);
    info.AddValue("MatrixB", MB);
    info.AddValue("PI", PI);
    info.AddValue("ObsProb", ObsProb);
    info.AddValue("ProbPattern", ProbMPatt);
}

public void ForwardProcedure(int[,] Obs, int row)
{
    int MaxT;
    MaxT = setV_LengthPat;
    double SumRowAlpha;
    //Fw Initialization
    t = 0;
    for (r = 0; r < row; r++)
    {
        for (i = 0; i < setV_MaxN; i++)
        {
            k = Obs[r, i];
            Alpha[r, i, t] = PI[i] * MB[i, k];
        }
    }
    //Fw Termination
    for (r = 0; r < row; r++)
    {
        for (t = 1; t < MaxT; t++)
        {
            k = Obs[r, t];
            for (i = 0; i < setV_MaxN; i++)
            {
                SumRowAlpha = 0;
                for (j = 0; j < setV_MaxN; j++)
                {
                    if (Alpha[r, j, t - 1] != 0 & MA[j, i] != 0)
                    {
                        SumRowAlpha += (Alpha[r, j, t - 1] * MA[j, i]);
                    }
                }
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
}
}
//induction
t = MaxT - 1; // set position last index array Alpha
for (r = 0; r < row; r++)
{
    SumRowAlpha = 0;
    StreamWriter file = new StreamWriter("C:\\Users\\HMM\\Dropbox\\WS-VYEAR2\\CODE\\Temp\\TxFile\\HMMparameterModel" + (Pattern + 1) + ".txt",
true);
    for (i = 0; i < setV_MaxN; i++)
    {
        SumRowAlpha += Alpha[r, i, t];
    }
    ObsProb[r] = SumRowAlpha;
    string lines = "ProbPattern " + (Pattern + 1) + " Row: " + r + " = " + ObsProb[r];
    file.WriteLine(lines);
    file.Close();
}
ComputeProbM(row);
}
private void ComputeProbM(int row)
{
    double SumRowProb = 0;
    for (i = 0; i < row; i++)
    {
        SumRowProb = Math.Log10(ObsProb[i]);
    }
    ProbMPatt = Math.Pow(10, (SumRowProb / row));
    Console.WriteLine("ProbPattern " + (Pattern + 1) + " : " + ProbMPatt);
}
public void BackwardProcedure(int[,] Obs, int row)
{
    int MaxT;
    double SumRowBeta;
    MaxT = setV_LengthPat - 1;
    //initialization
    t = MaxT;
    for (r = 0; r < row; r++)
    {
        for (i = 0; i < setV_MaxN; i++)
        {
            Beta[r, i, t] = 1;
        }
    }
}
//induction
for (r = 0; r < row; r++)
{
    for (t = MaxT - 1; t >= 0; t--)
    {
        k = Obs[r, t + 1];
        for (i = 0; i < setV_MaxN; i++)
        {
            SumRowBeta = 0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

    }
    for (q = 0; q < setV_MaxO; q++)
    {
        if (ProbMPatt > 0)
        {
            SumAlphaBetaO[q] += (AlphaBetaO[q] / ObsProb[r]);
        }
    }
} //end for t
} //end for row
for (q = 0; q < setV_MaxO; q++)
{
    MBBBar[j, q] = SumAlphaBetaO[q] / AlphaBeta[j];
}
}
NormalizeParameterBar();
ForwardProcedure_ModelBar(Obs, row);
}
private void NormalizeParameterBar()
{
    double SumRow;
    //Normalize Matrix ABar
    for (i = 0; i < setV_MaxN; i++)
    {
        SumRow = 0;
        for (j = 0; j < setV_MaxN; j++)
        {
            SumRow += MABar[i, j];
        }
        for (j = 0; j < setV_MaxN; j++)
        {
            MABar[i, j] = MABar[i, j] / SumRow;
        }
    }
    //Normalize Matrix BBar
    for (i = 0; i < setV_MaxN; i++)
    {
        SumRow = 0;
        for (j = 0; j < setV_MaxO; j++)
        {
            SumRow += MBBBar[i, j];
        }
        for (j = 0; j < setV_MaxO; j++)
        {
            MBBBar[i, j] = MBBBar[i, j] / SumRow;
        }
    }
    //Normalize PBar
    SumRow = 0;
    for (i = 0; i < setV_MaxN; i++)
    {
        SumRow += PBar[i];
    }
    for (i = 0; i < setV_MaxN; i++)
    {
        PBar[i] /= SumRow;
    }
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

private void ForwardProcedure_ModelBar(int[,] Obs, int row)
{
    int MaxT;
    MaxT = setV_LengthPat;
    double SumRowAlpha;
    //Fw Initialization
    t = 0;
    for (r = 0; r < row; r++)
    {
        for (i = 0; i < setV_MaxN; i++)
        {
            Alpha[r, i, t] = 0; // set Alpha = 0
            k = Obs[r, t];
            Alpha[r, i, t] = PBar[i] * MBar[i, k];
        }
    }
    //Fw Termination
    for (r = 0; r < row; r++)
    {
        for (t = 1; t < MaxT; t++)
        {
            k = Obs[r, t];
            for (i = 0; i < setV_MaxN; i++)
            {
                SumRowAlpha = 0;
                for (j = 0; j < setV_MaxN; j++)
                {
                    if (Alpha[r, j, t - 1] != 0 & MA[j, i] != 0)
                    {
                        SumRowAlpha += (Alpha[r, j, t - 1] * MBar[j, i]);
                    }
                }
                Alpha[r, i, t] = SumRowAlpha * MBar[i, k];
            }
        }
    }
    //induction
    t = MaxT - 1; // set position last index array Alpha
    SumRowAlpha = 0;
    for (r = 0; r < row; r++)
    {
        SumRowAlpha = 0;
        for (i = 0; i < setV_MaxN; i++)
        {
            SumRowAlpha += Alpha[r, i, t];
        }
        ObsProbBar[r] = SumRowAlpha;
        //Console.WriteLine("ProbPatternBar 1 Row: " + r + " = " + SumRowAlpha);
    }
    ComputeProbMBar(Obs, row);
}

private void ComputeProbMBar(int[,] Obs, int row)
{
    double SumRowProb = 0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for (i = 0; i < row; i++)
{
    SumRowProb = Math.Log10(ObsProbBar[i]);
}
ProbMBarPatt = Math.Pow(10, (SumRowProb / row));
Console.WriteLine("ProbPatternBar " + (Pattern + 1) + " : " + ProbMBarPatt);
}

```

```

public void CopyModelBarToMainModel()

```

```

{
    for (i = 0; i < setV._MaxN; i++)
    {
        for (j = 0; j < setV._MaxN; j++)
        {
            MA[i, j] = MABar[i, j];
        }
    }
}

```

```

for (i = 0; i < setV._MaxN; i++)
{
    for (j = 0; j < setV._MaxO; j++)
    {
        MB[i, j] = MBBar[i, j];
    }
}

```

```

for (i = 0; i < setV._MaxN; i++)
{
    PI[i] = PIBar[i];
}
}

```

```

public void ForwardProcedurObs(int[,] Obs, int row, int model)

```

```

{
    int MaxT;
    MaxT = setV._LengthPat;
    double SumRowAlpha;
    //Fw Initialization
    t = 0;
    for (r = 0; r < row; r++)
    {
        for (i = 0; i < setV._MaxN; i++)
        {
            k = Obs[r, i];

            Alpha[r, i, t] = PI[i] * MB[i, k];
        }
    }
}

```

```

//Fw induction

```

```

for (r = 0; r < row; r++)
{
    for (t = 1; t < MaxT; t++)
    {
        k = Obs[r, t];
        for (i = 0; i < setV._MaxN; i++)
        {
            SumRowAlpha = 0;

```

```

            for (j = 0; j < setV._MaxN; j++)

```

เอกสารนี้เป็นเอกสารที่ สงวนลิขสิทธิ์ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

SQL = "SELECT * FROM OB_OBSPREDICT WHERE CONMONTH = " + SearchDateAct + " ";
PredicDateDt = actionData.SQLReadData(SQL);
ActualHigh = Convert.ToDouble(PredicDateDt.Rows[0]["PHIGH"]);
ActualLow = Convert.ToDouble(PredicDateDt.Rows[0]["PLOW"]);
ActualColse = Convert.ToDouble(PredicDateDt.Rows[0]["PCLOSE"]);
ActualOPEN = Convert.ToDouble(PredicDateDt.Rows[0]["POPEN"]);
oldprice = Convert.ToDouble(PredicDateDt.Rows[0]["OLDPRICE"]);
if (ProbList.Count == 0)
{
    ptd.model = 48;
    return;
}
else
{
    //Find MaxProb for Predic
    ptd = ProbList.Max();
    //Find vale from MaxProb
    PredicDateDt.Clear();
    SQL = "SELECT AVG_PRICEOPEN,AVG_PRICE FROM OB_OBSDETAIL WHERE OBSERVATION = " + ptd.model.ToString();
    PredicDateDt = actionData.SQLReadData(SQL);
    PdAvgPrice = Convert.ToDouble(PredicDateDt.Rows[0]["AVG_PRICE"]);
    PdAvgOpen = Convert.ToDouble(PredicDateDt.Rows[0]["AVG_PRICEOPEN"]);
    PREDICTPRICE = oldprice + PdAvgOpen + PdAvgPrice;
    pdtprice = oldprice + PdAvgPrice;
    PredicDateDt.Clear();
}
Console.WriteLine("Date : " + Predicdate + " Maximum : " + ptd.model + " Actual Model: " + ObsChkCompareData);
StreamWriter file = new StreamWriter("C:\Users\HMM\Dropbox\MS-VYEAR\CODE\Temp\Result\Result\Predic.csv", true);
lines = Predicdate +
    "." + ptd.model +
    "." + ObsChkCompareData.ToString() +
    "." + ActualOPEN.ToString() +
    "." + ActualHigh.ToString() +
    "." + ActualLow.ToString() +
    "." + ActualColse.ToString() +
    "." + oldprice.ToString() +
    "." + PREDICTPRICE.ToString() +
    "." + pdtprice.ToString();
if (ptd.model == ObsChkCompareData)
{
    ResultP = true;
    lines += "." + ResultP.ToString();
}
else
{
    ResultP = false;
    lines += "." + ResultP.ToString();
}
file.WriteLine(lines);
file.Close();
ProbList.Clear();
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก. SettingVar.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace HMMTRAINING
{
    public class SettingVar
    {
        public static int MaxN = 40; //State in Matrix A
        private static int MaxO = 48; //Matrix b according to pattern
        private static int MaxPatt = 48; //Observation Pattern in Hash table
        private static int MaxRows = 500; // set rows at Obsarray(Observation Pattern) default is 100
        private static int LengthPat = 5; //set Length of each patternk

        public int _MaxN;
        public int _MaxO;
        public int _MaxPatt;
        public int _MaxRows;
        public int _LengthPat;

        public SettingVar()
        {
            _MaxN = MaxN;
            _MaxO = MaxO;
            _MaxPatt = MaxPatt;
            _MaxRows = MaxRows;
            _LengthPat = LengthPat;
        }
    }
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ProbTestData.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace HMMTRAINING
{
    class ProbTestData : IComparable<ProbTestData>
    {
        public int row;
        public int model;
        public double Prob;

        public ProbTestData(int _row, int _Model, double _Prob)
        {
            row = _row;
            model = _Model;
            Prob = _Prob;
        }

        public ProbTestData()
        {
            // TODO: Complete member initialization
        }

        public int CompareTo(ProbTestData other)
        {
            double OtSumprob = other.Prob + 0;
            double thisSumprob = this.Prob + 0;

            if (OtSumprob > thisSumprob)
                return -1;
            else if (OtSumprob == thisSumprob)
                return 0;
            else
                return 1;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จ. AppProcess.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data;
using System.Data.SqlClient;
using System.Data.OleDb;
namespace HMMTRAINING
{
    class AppProcess
    {
        String cnStrSQL = "Server=HMM-PC;Database=ProjectHMMFyear;User Id=sa;Password=Petorman236;";
        #String SQL;
        DataTable dt;
        SqlConnection SQLCon;
        SqlDataAdapter rsAdapter;
        SqlCommand SQLCmd;
        SqlDataReader rsReader;
        OleDbConnection EobjConn;
        OleDbDataAdapter ExAdapter;
        public DataTable ExcellReadData(string cnstrEX, string EXSQL)
        {
            EobjConn = new OleDbConnection(cnstrEX);
            if (EobjConn.State == ConnectionState.Open)
            {
                EobjConn.Close();
            }
            EobjConn.Open();
            dt = new DataTable();
            ExAdapter = new OleDbDataAdapter(EXSQL, EobjConn);
            ExAdapter.Fill(dt);
            ExAdapter = null;
            EobjConn.Close();
            return dt;
        }
        public void SQLqueryAction(string SQL)
        {
            SQLCon = new SqlConnection(cnStrSQL);
            if (SQLCon.State == ConnectionState.Open)
            {
                SQLCon.Close();
            }
            SQLCon.Open();
            SQLCmd = new SqlCommand(SQL, SQLCon);
            rsReader = SQLCmd.ExecuteReader();
            rsReader.Close();
            rsReader.Dispose();
            SQLCon.Close();
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
public DataTable SQLReadData(string SQL)
{
    SQLCon = new SqlConnection(cnStrSQL);

    if (SQLCon.State == ConnectionState.Open)
    {
        SQLCon.Close();
    }
    SQLCon.Open();
    dt = new DataTable();
    rsAdapter = new SqlDataAdapter(SQL, SQLCon);
    rsAdapter.Fill(dt);
    SQLCon.Close();
    rsAdapter.Dispose();
    return dt;
}
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้