สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เทคนิคการพยากรณ์ข้อมูลโดยใช้ไฮบริดจ์อัลกอริทึม

DATA FORECASTING TECHNIQUE BY HYBRID ALGORITHM

T139308

โดย

พิชญา ประทุมศิริ

PICHAYA PRATOOMSIRI

อาจารย์ที่ปรึกษา

รศ.ดร. อาริต ธรรมโน

ว.
ว163ท
2554

.b.....
.i.....

เลขหมู่.....
เลขทะเบียน 139308
วัน.เดือน.ปี 30 ต.ค. 2558

b 12719857

รายงานนี้เป็นส่วนหนึ่งของวิชาการศึกษาอิสระ
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่ 1 ปีการศึกษา 2554

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA FORECASTING TECHNIQUE BY HYBRID ALGORITHM

A REPORT SUMMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS OF THE COURSE

INDEPENDENT STUDY

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

1/2011

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2011

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	เทคนิคการพยากรณ์ข้อมูลโดยใช้ไฮบริดจ์อัลกอลิทึม
นักศึกษา	นายพิชญา ประทุมศิริ
รหัสนักศึกษา	51066420
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2554
อาจารย์ที่ปรึกษา	รศ. ดร. อาริต ธรรมโน

บทคัดย่อ

รายงานฉบับนี้ได้นำเสนอเกี่ยวกับวิธีการและเทคนิคในการพยากรณ์ข้อมูลโดยใช้หลักการของไฮบริดจ์อัลกอริทึม (Hybrid Algorithm) ซึ่งจะเป็นการประยุกต์และการผสมผสานทฤษฎีของฟัซซีลอจิก (Fuzzy Logic), นูรอลเนตเวิร์ค (Neural Network) และ เจเนติกอัลกอริทึม (Genetic Algorithm) เพื่อให้ได้ประสิทธิภาพออกมาดีที่สุด ซึ่งได้ทำการทดลองกับข้อมูลสถิติของตลาดหุ้นของประเทศไทย รายงานนี้จะใช้ขั้นตอนการทำงานของไฮบริดจ์ที่ได้ออกแบบไว้เพื่อการพยากรณ์ข้อมูล รวมถึงยังแสดงการเปรียบเทียบของรูปแบบต่างๆในการพยากรณ์ เพื่อให้เห็นถึงความแตกต่างของการพยากรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Data forecasting technique by Hybrid Algorithm
Student	MR. Pichaya Pratoomsiri
Student ID	51066420
Degree	Master of Science
Program	Information Technology
Major	Information Science
Academic Year	2554
Advisor	Assoc. Prof. Dr. Arit Tummano

ABSTRACT

This thesis was presented about the methods and technique in forecasting using the principles of the hybrid algorithm, which will be applied and integrated theory of fuzzy logic, neural network and genetic algorithm to get the best performance. The experiments were performed with the statistical data of the stock market in Thailand. This thesis will apply the hybrid system is designed to forecast data. The model also shows a comparison of various forecasting. To see the difference of the forecasts.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานพัฒนาระบบงานนี้เกิดขึ้น และสำเร็จลุล่วงไปได้ด้วยดี ผู้จัดทำโครงงานขอกราบขอบพระคุณ รศ.ดร. อาริต ธรรมโน ซึ่งเป็นอาจารย์ที่ปรึกษาโครงงานที่ได้ให้โอกาสริเริ่มแนะนำโครงงานนี้และให้คำปรึกษาที่เป็นประโยชน์ในการทำโครงงาน ให้ความช่วยเหลือด้านต่างๆ ตลอดจนปรับปรุงแก้ไขเรียบเรียงเอกสารจนเอกสารมีความสมบูรณ์ ผู้จัดทำมีความซาบซึ้งในความกรุณาเป็นอย่างยิ่ง จึงขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอขอบคุณ นายพงศ์เลิศ สังกะเพศ ที่ช่วยแนะนำทางด้านการพัฒนาโปรแกรม จึงทำให้โปรแกรมที่พัฒนาขึ้นมานั้นสำเร็จลุล่วงด้วยดีขอขอบคุณเพื่อนๆทุกคนที่คอยรับฟังปัญหา ให้คำแนะนำรวมทั้ง ให้กำลังใจเสมอมา

สุดท้ายนี้ผู้จัดทำขอกราบครอบครัวที่เป็นกำลังใจ เป็นแรงผลักดัน และให้การสนับสนุนแก่ข้าพเจ้าในทุกด้าน คุณค่าและประโยชน์อันพึงมาจากโครงงานฉบับนี้ ขอมอบแด่ผู้มีพระคุณทุกท่าน

พิชญา ประทุมศิริ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ..... II

กิตติกรรมประกาศ..... III

สารบัญ.....IV

สารบัญรูป.....VII

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....1

1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการศึกษา.....2

1.4 ขอบเขตการดำเนินงาน.....2

1.5 ขั้นตอนและวิธีการดำเนินงาน.....2

1.6 ประโยชน์ที่คาดว่าจะได้รับ.....3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง..... 4

2.1 ทฤษฎี Neural Network 4

2.1.1 หลักการและการทำงาน..... 4

2.1.2 การเรียนรู้สำหรับ Neural Network 6

2.2 ทฤษฎี Fuzzy Logic..... 7

2.2.1 พื้นฐานแนวคิดแบบฟัซซี..... 8

2.2.2 โครงข่ายประสาทเทียมแบบฟัซซี.....8

2.3 Genetic Algorithm 9

2.3.1 รูปแบบการนำเสนอ.....9

2.3.2 การวัดค่าความเหมาะสม..... 10

2.3.3 การคัดเลือก..... 10

2.3.4 การครอสโอเวอร์.....13

2.3.5 การมิวเตชัน..... 14

2.4 การประยุกต์การใช้งานในการสร้างแบบจำลอง Hybrid..... 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.4.1 โครงสร้างระบบนิวโร-ฟัซซี (Neuro-fuzzy)..... 16
2.4.2 การเรียนรู้ของระบบ Neuro-Fuzzy..... 20

บทที่ 3 ขั้นตอนการทำงาน..... 25
3.1 การออกแบบระบบและกระบวนการทำงาน..... 25
3.2 เครื่องมือที่ใช้ในการพัฒนาระบบการพยากรณ์..... 25
3.3 การทำงานของระบบการพยากรณ์ราคาหุ้น..... 44
3.3.1 การนำข้อมูลเข้า..... 44
3.3.2 การเรียนรู้ของเครือข่าย..... 45
3.3.3 การพยากรณ์ราคาหุ้น..... 45
3.4 รายละเอียดหน้าจอของระบบ..... 45
3.4.1 หน้าจอแสดงข้อมูลตัวอย่าง..... 46
3.4.2 หน้าจอแสดงเลเยอร์ฟัซซี..... 46
3.4.3 หน้าจอแสดงเลเยอร์กฎ..... 47
3.4.4 หน้าจอแสดงการเรียนรู้ข้อมูล..... 48
3.4.5 หน้าจอแสดงผลลัพธ์..... 50

บทที่ 4 ผลการทดลอง..... 51
4.1 ข้อมูลที่ใช้ในการเรียนรู้และการพยากรณ์หุ้น..... 51
4.2 การกำหนดค่าเริ่มต้นสำหรับการฝึกสอน..... 53
4.3 การแสดงหน้าจอของโปรแกรมสำหรับการเรียนรู้และการพยากรณ์..... 53
4.4 ผลการทดสอบระบบ..... 58

บทที่ 5 สรุปและข้อเสนอแนะ..... 60
5.1 เปรียบเทียบผลการทดลอง..... 60
5.1.1 เปรียบเทียบโดยใช้จำนวนอินพุตและจำนวนฟังก์ชันสมาชิกเท่ากัน..... 60
5.1.2 เปรียบเทียบโดยใช้จำนวนอินพุตต่างกันฟังก์ชันสมาชิกเท่ากัน..... 62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

5.1.3 เปรียบเทียบโดยใช้จำนวนอินพุตเท่ากันฟังก์ชันสมาชิกต่างกัน.....65
5.2 สรุป.....68
5.3 ข้อเสนอแนะ.....68
บรรณานุกรม.....69
ประวัติผู้เขียน.....70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 แสดง Model ของ Neuron ในสมองมนุษย์ 4

2.2 แสดง Model ของ Neuron ในคอมพิวเตอร์ 4

2.3 แสดงการแยกแยะระหว่างสี่เหลี่ยมและสามเหลี่ยม 5

2.4 แสดงโครงสร้างวงจร Neural Network 5

2.5 แสดงการเรียนรู้แบบมีการสอน (Supervised Learning) 6

2.6 แสดงการเรียนรู้แบบไม่มีการสอน Unsupervised Learning 6

2.7 การกำหนดค่าความเป็นสมาชิกของเซตทวินัยและเซตแบบฟัซซี 7

2.8 ตรรกะแบบเท็จจริงและตรรกะแบบฟัซซี 8

2.9 วงล้อรูเล็ทจากค่าความเหมาะสมของแต่ละโครโมโซม 11

2.10 วงล้อรูเล็ทจากวิธีการจัดอันดับ 12

2.11 แสดงครอสโอเวอร์แบบจุดเดียว 13

2.12 แสดงครอสโอเวอร์แบบหลายจุด 14

2.13 การทำงานของ Hybrid Algorithm 15

2.14 ระบบอนุมานฟัซซีแบบ Mamdani 17

2.15 การรวมกฎของฟัซซี 17

2.16 neuro-fuzzy systems 18

2.17 นิวรอนการทำฟัซซี 19

2.18 ผลของพารามิเตอร์ a, b และ c ต่อรูปร่างของฟังก์ชันสมาชิก 19

2.19 ตัวอย่างข้อมูลรูปแบบของ XOR 20

2.20 เครือข่ายประสาทเทียมแบบ 3 ชั้น 21

2.21 ค่าโครโมโซมเริ่มต้น 21

2.22 การเข้ารหัสโดยตรง 22

2.23 โครโมโซมการเชื่อมต่อระหว่างนิวรอน 23

2.24 เปรียบเทียบแบบเมตริก 24

3.1 Class Membership Function 25

3.2 Class Trapmf 26

3.3 Class Tripmf 27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.4 Class LingVariable 27

3.5 Class LVInput Layer 28

3.6 Class Rule Layer 29

3.7 Class Create Rule 30

3.8 Class Fuzzy Output Layer 31

3.9 Class Output Layer 32

3.10 Class Variable 34

3.11 Class Save Data 36

3.12 Class Genetic 37

3.13 Class Train Data 38

3.14 Class Test Data 40

3.15 Class ของระบบ 40

3.16 รูปแสดงไฟล์ Text ของตลาดหุ้น 44

3.17 หน้าจอแสดงข้อมูลตัวอย่าง 46

3.18 หน้าจอแสดงเลเยอร์ฟัซซี 47

3.19 หน้าจอแสดงเลเยอร์กฎ 48

3.20 แสดงส่วนของการเรียนรู้ 48

3.21 แสดงหน้าจอการตั้งค่าพารามิเตอร์ 49

3.22 แสดงหน้าจอผลลัพธ์ของการเรียนรู้ 49

3.23 แสดงหน้าจอผลลัพธ์ 50

4.1 แสดงหน้าจอข้อมูลที่จะใช้ในการพยากรณ์ 51

4.2 กราฟแสดงราคาหุ้นที่ใช้ในการฝึกสอน 52

4.3 กราฟแสดงราคาหุ้นที่ใช้ในการพยากรณ์ 52

4.4 แสดงข้อมูลนำเข้าและจำนวนอินพุตของระบบ 54

4.5 แสดงเลเยอร์ฟัซซี 54

4.6 แสดงเลเยอร์กฎ 55

4.7 แสดงการเรียนรู้ข้อมูล 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.8 แสดงส่วนของค่า Mean square error 56
4.9 แสดงส่วนของการตั้งค่าการเรียนรู้ 57
4.10 แสดงส่วนผลลัพธ์จากการเรียนรู้ 57
4.11 แสดงผลลัพธ์ของการเรียนรู้ข้อมูลเทียบกับข้อมูลจริง 58
4.12 แสดงผลลัพธ์ของการเรียนรู้ข้อมูลทดสอบเทียบกับข้อมูลจริง 59
5.1 แสดงการเรียนรู้โดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม 60
5.2 แสดงผลการทดสอบโดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม 60
5.3 แสดงการเรียนรู้โดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบคางหมู 61
5.4 แสดงผลการทดสอบโดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบคางหมู 61
5.5 แสดงการเรียนรู้โดยใช้ 3 อินพุต 7 ฟังก์ชัน แบบสามเหลี่ยม 61
5.6 แสดงผลการทดสอบโดยใช้ 3 อินพุต 7 ฟังก์ชัน แบบสามเหลี่ยม 61
5.7 แสดงการเรียนรู้โดยใช้ 3 อินพุต 7 ฟังก์ชัน แบบคางหมู 62
5.8 แสดงผลการทดสอบโดยใช้ 3 อินพุต 7 ฟังก์ชัน แบบคางหมู 62
5.9 แสดงการเรียนรู้โดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม 62
5.10 แสดงผลการทดสอบโดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม 63
5.11 แสดงการเรียนรู้โดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม 63
5.12 แสดงผลการทดสอบโดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม 63
5.13 แสดงการเรียนรู้โดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบคางหมู 64
5.14 แสดงผลการทดสอบโดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบคางหมู 64
5.15 แสดงการเรียนรู้โดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบคางหมู 64
5.16 แสดงผลการทดสอบโดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบคางหมู 65
5.17 แสดงการเรียนรู้โดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม 65
5.18 แสดงผลการทดสอบโดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม 66
5.19 แสดงการเรียนรู้โดยใช้ 4 อินพุต 7 ฟังก์ชัน แบบสามเหลี่ยม 66
5.20 แสดงผลการทดสอบโดยใช้ 4 อินพุต 7 ฟังก์ชัน แบบสามเหลี่ยม 66
5.21 แสดงการเรียนรู้โดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบคางหมู 67
5.22 แสดงผลการทดสอบโดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบคางหมู 67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

5.23 แสดงการเรียนรู้โดยใช้ 4 อินพุต 7 ฟังก์ชัน แบบคางหมู 67

5.24 แสดงผลการทดสอบโดยใช้ 4 อินพุต 7 ฟังก์ชัน แบบคางหมู 68

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในช่วงเวลาที่ผ่านมา จะเห็นได้ชัดว่า การลงทุนในลักษณะต่างๆ เริ่มเป็นที่สนใจของประชาชนทั่วไป ไม่ว่าจะเป็นตลาดหลักทรัพย์ ทอง หรือ น้ำมันดิบ เป็นต้น อย่างไรก็ตาม การลงทุนที่มีผลตอบแทนสูงย่อมมีความเสี่ยงสูงตามมาด้วย ดังนั้นผู้ที่ต้องการจะลงทุนจึงต้องศึกษาและพิจารณาถึงปัจจัยต่างๆที่มีผลกระทบทั้งข้อมูล ข่าวสาร รวมถึงประสบการณ์มาช่วยในการวิเคราะห์การตัดสินใจในการลงทุนด้วย โดยการวิเคราะห์ประกอบด้วย ขั้นตอนในการรวบรวมข้อมูล การศึกษาและการวิเคราะห์ข้อมูล รวมถึงการตีความของผลลัพธ์ที่ได้จากข้อมูลที่วิเคราะห์ออกมาเพื่อนำมากำหนดเป็นแนวทางในการลงทุน ซึ่งการวิเคราะห์นั้นสามารถทำได้หลายวิธีวิเคราะห์โดยใช้ปัจจัยพื้นฐาน และการวิเคราะห์ทางเทคนิค โดยในโครงการนี้จะใช้ Hybrid Algorithm ในการวิเคราะห์

Hybrid Algorithm เป็นการใช้หลักการของ fuzzy logic, neural network และ genetic Algorithm มาผสมผสานกันเพื่อให้ได้แบบจำลองที่มีความถูกต้องแม่นยำมากขึ้น โดยประยุกต์ใช้หลักการคำนวณแบบ soft computing (SC) โดย Soft computing (SC) นั้นสามารถทำงานได้ดีกับข้อมูลที่เราไม่แน่ใจ คลุมเครือ และ ไม่สมบูรณ์ได้ โดยใช้รูปแบบวิธีการในการไตร่ตรองเลียนแบบมนุษย์ ในชีวิตจริงนั้นในการอธิบายต่างๆ มนุษย์จะใช้ข้อมูลที่มีความยืดหยุ่นโดยใช้เป็นคำพูดมากกว่าที่จะใช้ตัวเลขในการอธิบาย เราจึงคิดที่จะใช้เหตุผลในการตัดสินใจโดยไม่ใช้ตัวเลข

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

โครงการพัฒนาโปรแกรม Data forecasting technique by Hybrid Algorithm เพื่อใช้ในการพยากรณ์ข้อมูลมีวัตถุประสงค์ดังนี้

1.2.1 เพื่อศึกษาถึงกระบวนการ Hybrid Algorithm การเลือกอัลกอริทึมมาใช้ คุณสมบัติโดยทั่วไปของอัลกอริทึมแบบต่างๆที่สามารถนำมาประยุกต์ใช้ในการพยากรณ์ข้อมูลได้

1.2.2 เพื่อพัฒนาโปรแกรมในการพยากรณ์ข้อมูลให้มีประสิทธิภาพและมีความแม่นยำเพื่อที่จะนำไปใช้ในการประยุกต์การใช้งานจริงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการศึกษา

ศึกษาทฤษฎีที่ใช้พัฒนาโครงการ มีดังนี้

- ศึกษาทฤษฎีของ Neural Network
- ศึกษาทฤษฎีของ Fuzzy Logic
- ศึกษาทฤษฎีของ Genetic Algorithm
- ศึกษาการนำทฤษฎีมาประยุกต์ในการใช้งานร่วมกันเพื่อให้ได้ประสิทธิภาพมากที่สุด

1.4 ขอบเขตการดำเนินงาน

การศึกษาโครงการนี้ได้กำหนดขอบเขตในการศึกษาเป็นการพัฒนาโปรแกรมการพยากรณ์ข้อมูลโดยใช้ Hybrid Algorithm โดยใช้ทฤษฎีของ Neural Network , Fuzzy Logic และ Genetic Algorithm มาประยุกต์ในการใช้งาน เพื่อเพิ่มความสามารถในการตัดสินใจเกี่ยวกับการลงทุนในประเภทต่างๆ

ขอบเขตในการศึกษามีดังต่อไปนี้

- การออกแบบโปรแกรมเพื่อใช้ในการพยากรณ์ข้อมูลเพื่อช่วยในการตัดสินใจโดยใช้หลักการของ Hybrid Algorithm
- การพัฒนาโปรแกรม Hybrid Algorithm เพื่อให้สามารถนำมาใช้งานได้จริงและมีความถูกต้องแม่นยำ

1.5 ขั้นตอนและวิธีการดำเนินงาน

เพื่อให้การศึกษาเป็นไปตามวัตถุประสงค์ และขอบเขตที่กำหนด จึงได้กำหนดขั้นตอนในการดำเนินงานไว้ ดังนี้

- ศึกษาทฤษฎีของ Neural Network
- ศึกษาทฤษฎีของ Fuzzy Logic
- ศึกษาทฤษฎีของ Genetic Algorithm
- ศึกษาการนำทฤษฎีมาประยุกต์ในการใช้งานร่วมกันเพื่อให้ได้ประสิทธิภาพมากที่สุด
- ศึกษาทฤษฎีในการพยากรณ์ข้อมูลแบบจำลองต่างๆเพื่อนำไปปรับในการใช้งาน
- วิเคราะห์และออกแบบโครงการ
- พัฒนาโปรแกรม
- ทดสอบการทำงานของโปรแกรม
- ประเมินประสิทธิภาพการทำงานของโปรแกรมและให้ข้อคิดเห็นและเสนอแนะ
- สรุปผลและจัดทำเอกสารประกอบโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1.6.1 สามารถนำโปรแกรมที่พัฒนาขึ้น ไปใช้ในการพยากรณ์ข้อมูลได้ เพื่อช่วยในการตัดสินใจในการลงทุนประเภทต่างๆ

1.6.2 สามารถประยุกต์ใช้ทฤษฎีต่างๆที่ได้ศึกษามากับงานประเภทอื่นๆที่เกี่ยวข้องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

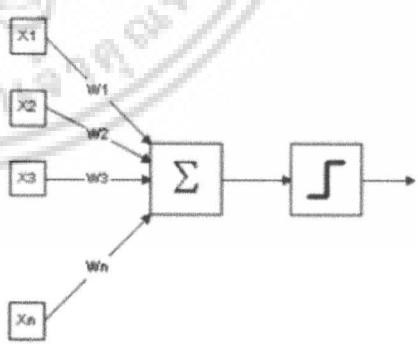
บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ทฤษฎี Neural Network

โครงข่ายประสาทเทียม (Artificial neural network) คือโมเดลทางคณิตศาสตร์สำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบคอนเนคชันนิสต์ (connectionist) เพื่อจำลองการทำงานของเครือข่ายประสาทในสมองมนุษย์ ด้วยวัตถุประสงค์ที่จะสร้างเครื่องมือซึ่งมีความสามารถในการเรียนรู้การจดจำรูปแบบ (Pattern Recognition) และการอุปมานความรู้ (Knowledge deduction) เช่นเดียวกับความสามารถที่มีในสมองมนุษย์

แนวคิดเริ่มต้นของเทคนิคนี้ได้มาจากการศึกษาข่ายงานไฟฟ้าชีวภาพ (bioelectric network) ในสมอง ซึ่งประกอบด้วย เซลล์ประสาท หรือ “นิวรอน” (neurons) และ จุดประสานประสาท (synapses) แต่ละเซลล์ประสาทประกอบด้วยปลายในการรับกระแสประสาทเรียกว่า "เดนไดรท์" (Dendrite) ซึ่งเป็น input และปลายในการส่งกระแสประสาทเรียกว่า "แอคซอน" (Axon) ซึ่งเป็นเหมือน output ของเซลล์ เซลล์เหล่านี้ทำงานด้วยปฏิกิริยาไฟฟ้าเคมี เมื่อมีการกระตุ้นด้วยสิ่งเร้าภายนอกหรือกระตุ้นด้วยเซลล์ด้วยกัน กระแสประสาทจะวิ่งผ่านเดนไดรท์เข้าสู่นิวเคลียสซึ่งจะเป็นตัวตัดสินว่าต้องกระตุ้นเซลล์อื่นๆ ต่อหรือไม่ ถ้ากระแสประสาทแรงพอนิวเคลียสก็จะกระตุ้นเซลล์อื่นๆ ต่อไปผ่านทางแอคซอนของมันโมเดลโครงข่ายประสาทเทียมเกิดจากการเชื่อมต่อระหว่างเซลล์ประสาท จนเป็นเครือข่ายที่ทำงานร่วมกัน

รูปที่ 2.1 แสดง Model ของ Neuron ในสมองมนุษย์



รูปที่ 2.2 แสดง Model ของ Neuron

2.1.1 หลักการและการทำงาน

สำหรับในคอมพิวเตอร์ Neurons ประกอบด้วย input และ output เหมือนกัน โดยทำการจำลองให้ input แต่ละอันมี weight (w) เป็นตัวกำหนดน้ำหนักของ input แต่ละอัน โดย neuron แต่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

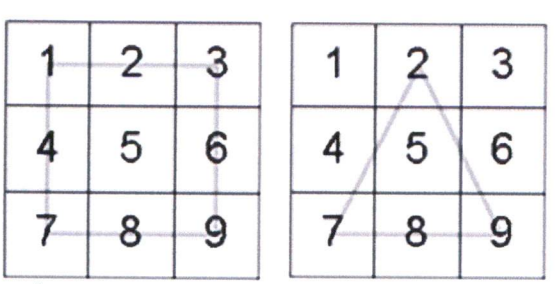
ละหน่วยจะมีค่า threshold (Σ) เป็นตัวกำหนดว่าน้ำหนักรวมของ input ต้องมากขนาดไหนจึงจะสามารถส่ง output ไปยัง neurons ตัวอื่นได้ เมื่อนำ neuron แต่ละหน่วยมาต่อกันให้ทำงานร่วมกัน การทำงานนี้ในทางตรรกะแล้วก็จะเหมือนกับปฏิกิริยาเคมีที่เกิดในสมอง เพียงแต่ในคอมพิวเตอร์ทุกอย่างเป็นตัวเลขเท่านั้นเอง

การทำงานของ Neural networks คือเมื่อมี input เข้ามายัง network ก็เอา input มาคูณกับ weight ของแต่ละขา ผลที่ได้จาก input ทุกขาของ neuron จะเอามารวมกันแล้วก็เอามาเทียบกับ threshold ที่กำหนดไว้ ถ้าผลรวมมีค่ามากกว่า threshold แล้ว neuron ก็จะส่ง output ออกไป output นี้ก็จะถูกส่งไปยัง input ของ neuron อื่น ๆ ที่ เชื่อมกันใน network ถ้าค่าน้อยกว่า threshold ก็จะไม่เกิด output การทำงานของ Neural สามารถเขียนออกมาได้ดังนี้

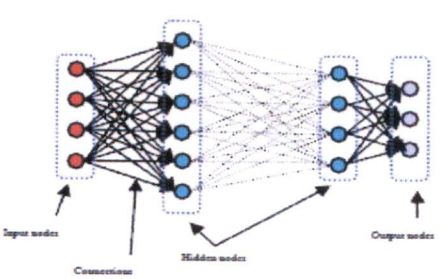
If (sum (input * weight) > threshold) then output

สิ่งสำคัญคือเราต้องทราบค่า weight และ threshold สำหรับสิ่งที่เราต้องการเพื่อให้คอมพิวเตอร์รู้จำ ซึ่งเป็นค่าที่ไม่แน่นอน แต่สามารถกำหนดให้คอมพิวเตอร์ปรับค่าเหล่านั้นได้โดยการสอนให้คอมพิวเตอร์รู้จัก pattern ของสิ่งที่เราต้องการให้คอมพิวเตอร์รู้จำ เรียกว่า "back propagation" ซึ่งเป็นกระบวนการย้อนกลับของการรู้จำในการฝึก feed-forward neural networks จะมีการใช้อัลกอริทึมแบบ back-propagation เพื่อใช้ในการปรับปรุงน้ำหนักคะแนนของเครือข่าย (network weight) หลังจากใส่รูปแบบข้อมูลสำหรับฝึกให้แก่เครือข่ายในแต่ละครั้งแล้ว ค่าที่ได้รับ (output) จากเครือข่ายจะถูกนำไปเปรียบเทียบกับผลที่คาดหวังแล้วทำการคำนวณหาค่าความผิดพลาด ซึ่งค่าความผิดพลาดนี้จะถูกส่งกลับเข้าสู่เครือข่ายเพื่อใช้แก้ไขค่าน้ำหนักคะแนนต่อไป

ตัวอย่างเช่น จะรู้จำรูปสามเหลี่ยม กับรูปสี่เหลี่ยม เราอาจแบ่ง input เป็น 9 ตัวคือเป็นตาราง 3x3 ถ้าวาดรูปสี่เหลี่ยมหรือสามเหลี่ยมให้เต็มกรอบ 3x3 พอดี สี่เหลี่ยมจะมีส่วนของขอบอยู่ในช่อง 1,2,3,4,6,7,8,9 ก็สมมติให้น้ำหนักตรงช่องเหล่านี้มีค่ามาก ๆ ถ้ามีเส้นขีดผ่านก็เอามาคูณกับน้ำหนักแล้วก็เอามารวมกัน ตั้งค่าให้พอเหมาะก็จะสามารถแยกแยะระหว่างสี่เหลี่ยมกับสามเหลี่ยมได้ ซึ่งนี่คือหลักการของ neural network



รูปที่ 2.3 การแยกแยะระหว่างสี่เหลี่ยมและสามเหลี่ยม



รูปที่ 2.4 โครงสร้าง Neural Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Output ของแต่ละ Node

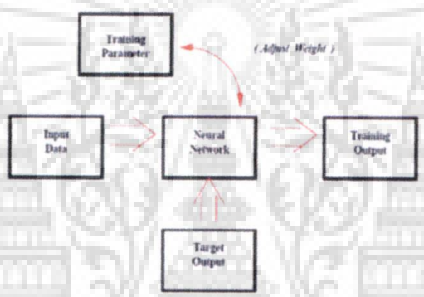
$$y_i = f(w_i^1 x_1 + w_i^2 x_2 + w_i^3 x_3 + ... + w_i^m x_m) = f(\sum_j w_i^j x_j)$$

เมื่อ X_i = input จากโหนดอื่นๆ

W_{ij} = น้ำหนัก (weight) ของแต่ละแขน (connection)

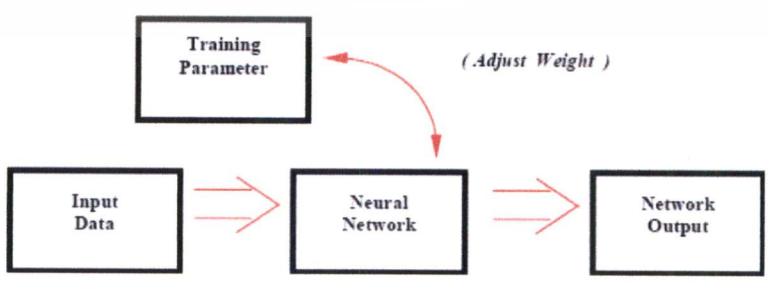
2.1.2 การเรียนรู้สำหรับ Neural Network

1. Supervised Learning การเรียนแบบมีการสอน เป็นการเรียนแบบที่มีการตรวจคำตอบเพื่อให้ Neural Network ปรับตัว ชุดข้อมูลที่ใช้สอน Neural Network จะมีคำตอบไว้คอยตรวจดูว่า Neural Network ให้คำตอบที่ถูกหรือไม่ ถ้าตอบไม่ถูก วงจรข่ายก็จะปรับตัวเองเพื่อให้ได้คำตอบที่ดีขึ้น (เปรียบเทียบกับคน เหมือนกับการสอนนักเรียนโดยมีครูผู้สอนคอยแนะนำ)



รูปที่ 2.5 แสดงการเรียนรู้แบบมีการสอน (Supervised Learning)

2. Unsupervised Learning การเรียนแบบไม่มีการสอน เป็นการเรียนแบบไม่มีผู้แนะนำ ไม่มีการตรวจคำตอบว่าถูกหรือผิด วงจรข่ายจะจัดเรียงโครงสร้างด้วยตัวเองตามลักษณะของข้อมูล ผลลัพธ์ที่ได้ วงจรข่ายจะสามารถจัดหมวดหมู่ของข้อมูลได้ (เปรียบเทียบกับคน เช่น การที่เราสามารถแยกแยะพันธุ์พืช พันธุ์สัตว์ตามลักษณะรูปร่างของมันได้เองโดยไม่มีใครสอน)



รูปที่ 2.6 แสดงการเรียนรู้แบบไม่มีการสอน Unsupervised Learning

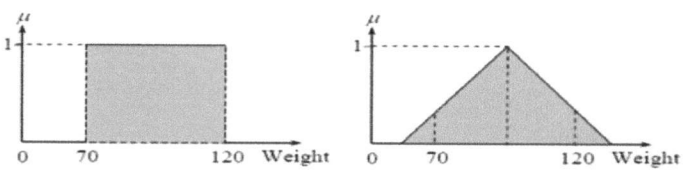
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ทฤษฎี Fuzzy Logic

เป็นระบบด้านคอมพิวเตอร์ที่ทำงานโดยอาศัยฟัซซีลอจิกที่คิดค้นโดย L. A. Zadeh ในปี ค.ศ. 1965 ซึ่งเป็นผลงานวิทยานิพนธ์ระดับปริญญาเอก ฟัซซีลอจิกเป็นตรรกะที่อยู่บนพื้นฐานความเป็นจริงที่ว่า ทุกสิ่งบนโลกแห่งความเป็นจริงไม่ใช่มีเฉพาะสิ่งที่มีความแน่นอนเท่านั้น แต่มีหลายสิ่งหลายเหตุการณ์ที่เกิดขึ้นอย่างไม่เที่ยงและไม่แน่นอน อาจเป็นสิ่งที่คลุมเครือ (fuzzy) ไม่ใช่ชัดเจน ยกตัวอย่างเช่น เซตของอายุคนอาจแบ่งเป็น วัยทารก วัยเด็ก วัยรุ่น วัยกลางคน และวัยชรา จะเห็นได้ว่าในแต่ละช่วงอายุคนไม่สามารถระบุได้แน่ชัดว่าวัยทารกกับวัยเด็กแยกจากกันแน่ชัดช่วงใด วัยทารกอาจถูกตีความว่าเป็นอายุระหว่าง 0 ถึง 1 ปี บางคนอาจตีความว่าวัยทารกอยู่ในช่วงอายุ 0 ถึง 2 ปี ในทำนองเดียวกัน วัยเด็กและวัยรุ่นก็ไม่สามารถระบุได้ชัดเจนว่าช่วงต่อของอายุควรจะอยู่ในช่วงใด อาจตีความว่าวัยเด็กมีอายุอยู่ในช่วง 1 ถึง 12 ปี หรืออาจเป็น 2 ถึง 10 ปี เป็นต้น สิ่งเหล่าเป็นตัวอย่างของความไม่แน่นอน ซึ่งเป็นลักษณะทางธรรมชาติที่เกิดขึ้นทั่วไป เซตของเหตุการณ์ที่ไม่แน่นอนเช่นนี้เรียกว่าฟัซซีเซต (Fuzzy set)

ฟัซซีเซต (Fuzzy Set) เป็นเซตที่มีขอบเขตที่ราบเรียบ ทฤษฎีฟัซซีเซตจะครอบคลุมทฤษฎีเซตแบบฉบับโดยเซตแบบฉบับคือ (classical set) หรือเซตทวินัย (crisp set) เป็นเซตที่มีค่าความเป็นสมาชิกเป็น 0 หรือ 1 {0, 1} เท่านั้น เซตในทฤษฎีเซตแบบฉบับจะมีขอบเขตแบบแข็ง (sharp boundary) ซึ่งเป็นขอบเขตที่ตัดขาดจากกันแบบทันทีทันใด เซตแบบฉบับมีการกำหนดค่าความเป็นสมาชิกตามแนวคิดเลขฐานสอง โดยที่ตัวแปรหนึ่ง ๆ จะมีค่าความเป็นสมาชิกเพียงสองค่า คือ 0 ไม่เป็นสมาชิก และ 1 เป็นสมาชิก ตัวอย่างเช่น เซตของคู่แต่งงาน โดยฟัซซีเซตยอมให้มีค่าความเป็นสมาชิกของเซตระหว่าง 0 และ 1 ในโลกแห่งความเป็นจริงเซตไม่ใช่มีเฉพาะเซตแบบฉบับเท่านั้นจะมีเซตแบบฟัซซีด้วย ฟัซซีเซตจะมีขอบเขตแบบฟัซซีไม่ใช่เปลี่ยนแปลงทันทีทันใดจากขาวเป็นดำ ตัวอย่างเช่น เซตของคู่แต่งงานที่มีความสุข จะเห็นได้ว่าสมาชิกในเซตนี้จะไม่มีเฉพาะคู่แต่งงานที่มีความสุขระดับเดียวกันหมด บางคู่จะมีความสุขมาก บางคู่มีความสุขน้อย แตกต่างกันไป การใช้เซตแบบดั้งเดิมจึงไม่เหมาะสม

ยกตัวอย่างเกี่ยวกับความอ้วน นิยามคำว่าคนอ้วนในเซตทวินัยอาจกำหนดเป็นคนที่มีน้ำหนักตั้งแต่ 70 ถึง 120 กิโลกรัม โดยนิยามแบบฟัซซีเซตอาจกำหนดเป็นคนที่มีความอ้วนประมาณ 80 กิโลกรัม ซึ่งเป็นการให้นิยามที่ไม่แสดงถึงขอบเขตที่แน่นอน

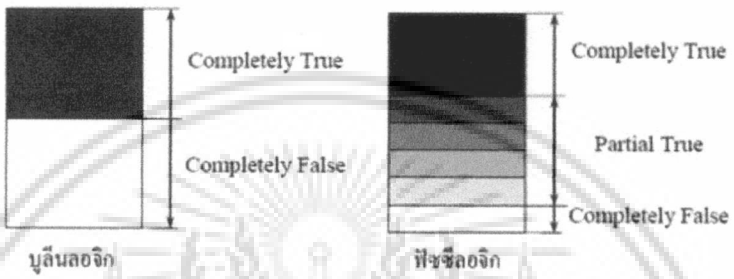


รูปที่ 2.7 การกำหนดค่าความเป็นสมาชิกของเซตทวินัยและเซตแบบฟัซซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 พื้นฐานแนวคิดแบบฟัซซี

ตรรกะแบบฟัซซี (Fuzzy logic) เป็นเครื่องมือที่ช่วยในการตัดสินใจภายใน ใต้ความไม่แน่นอนของข้อมูลโดยยอมให้มีความยืดหยุ่นได้ ใช้หลักเหตุผลที่คล้ายการเลียนแบบวิธีความคิดที่ซับซ้อนของมนุษย์ ฟัซซีลอจิกมีลักษณะที่พิเศษกว่าตรรกะแบบจริงเท็จ (Boolean logic) เป็นแนวคิดที่มีการต่อขยายในส่วนของความจริง (partial true) โดยค่าความจริงจะอยู่ในช่วงระหว่างจริง (completely true) กับเท็จ (completely false) ส่วนตรรกศาสตร์เดิมจะมีค่าเป็นจริงกับเท็จเท่านั้น



รูปที่ 2.8 ตรรกะแบบเท็จจริงและตรรกะแบบฟัซซี

2.2.2 โครงข่ายประสาทเทียมแบบฟัซซี

ฟัซซีลอจิกและโครงข่ายประสาทเทียมต่างก็มีข้อดีและข้อเสียที่แตกต่างกัน ฟัซซีลอจิกมีข้อดีในเรื่องการมีเหตุผลเชิงตรรกะ โครงสร้างของระบบฟัซซีสามารถเข้าใจได้เนื่องจากสามารถตีความให้ในรูป If-Then ซึ่งสอดคล้องกับตรรกะความคิดของมนุษย์ และนอกจากนั้นฟัซซีลอจิกยังช่วยในการตัดสินใจที่คลุมเครือที่ยอมให้การตัดสินใจเป็นแบบส่วน ไม่ใช่ผิดหรือถูกเพียงสองสถานะ แต่จะเป็นดีกรีของความถูกหรือผิด ซึ่งเป็นเหตุการณ์ที่เกิดขึ้นในธรรมชาติอยู่แล้ว

สำหรับข้อเสียของระบบฟัซซีก็คือ ไม่มีกระบวนการเรียนรู้ในการปรับแต่งโครงสร้างซึ่งกฎและตัวแปรต่างๆ ในตัวระบบเอง โครงสร้างของระบบจะถูกกำหนดโดยผู้เชี่ยวชาญในโดเมนที่กำลังพิจารณาร่วมกับนักเขียนโปรแกรมคอมพิวเตอร์ เช่น ถ้าหากต้องการสร้างระบบเพื่อการวิเคราะห์โรคมะเร็ง แพทย์ผู้เชี่ยวชาญด้านโรคมะเร็งจะต้องเป็นผู้กำหนดกฎและตัวแปรต่างๆ ของระบบ และนอกจากนั้นแพทย์ผู้เชี่ยวชาญต้องตรวจสอบประเมินความถูกต้องของระบบ ซึ่งบ่อยครั้งในการสร้างระบบฟัซซีอาจไม่มีผู้เชี่ยวชาญในโดเมนดังกล่าว การสร้างระบบจึงอาจไม่สัมฤทธิ์ผล การที่ระบบฟัซซีไม่มีกระบวนการเรียนรู้ด้วยตนเองจึงถือเป็นข้อด้อย แต่อย่างไรก็ตามปัจจุบันนักวิจัยได้มีการใส่กระบวนการเรียนรู้เข้าไปในระบบฟัซซีโดยอาศัยทฤษฎีการเรียนรู้ของโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมมีจุดเด่นด้านการเรียนรู้จากข้อมูล โครงข่ายประสาทเทียมมีการปรับแต่งความรู้ที่ซ่อนอยู่ภายในเครือข่ายที่มีการต่อเชื่อมโยงกันอย่างหนาแน่น มีการส่งผ่านข้อมูลที่จะประมวลผลจากอินพุตไปยังเอาต์พุตแบบขนาน การประมวลผลของโครงข่ายประสาทเทียมจึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นไปอย่างรวดเร็ว แต่ถึงอย่างไรก็ตาม โครงข่ายประสาทเทียมก็มีจุดด้อยในด้านการตีความหาเหตุผล โครงข่ายประสาทเทียมไม่สามารถให้เหตุผลได้ว่าเพราะเหตุใดจึงมีข้อสรุปออกมาดังที่ปรากฏที่เอาต์พุตของโครงข่าย จุดด้อยข้อนี้เป็นที่รู้จักกันดีในนาม "Black box" หรือกล่องดำ

จากข้อดีของฟัซซี ในด้านการให้เหตุผลเยี่ยงมนุษย์และข้อดีโครงข่ายประสาทเทียมด้านการเรียนรู้จากข้อมูล เมื่อนำสองศาสตร์นี้มารวมกันจะกลายเป็นโครงข่ายประสาทเทียมแบบฟัซซี ซึ่งเป็นระบบที่กระบวนการเรียนรู้ในตัวเอง และโครงสร้างของระบบสามารถตีความหมายและให้เหตุผลได้ เช่นเดียวกับระบบฟัซซี

2.3 Genetic Algorithm

ขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithms: GA) เป็นวิธีการค้นหาที่เลียนแบบการวิวัฒนาการของสิ่งมีชีวิตในธรรมชาติ GA จะค้นหาโดยใช้กลุ่มหรือประชากรของคำตอบ การค้นหาดำเนินแบบสุ่มโดยมีค่าความเหมาะสมของคำตอบเป็นตัวกำหนดทิศทาง คำตอบแต่ละตัวจะถูกเข้ารหัส (encode) อยู่ในรูปของโครโมโซม (chromosome) ที่เป็นเลขฐานสอง โครโมโซมแต่ละตัวในกลุ่มประชากรเดียวกันจะมีจำนวนบิตเท่ากัน GA มักจะถูกใช้ในการหาคำตอบของปัญหาที่มีปริภูมิการค้นหาที่ ใหญ่เกินกว่าจะหาโดยการค้นทุกรูปแบบ (exhaustive search) ขนาดของปริภูมิการค้นหาเป็นปัจจัยหนึ่งที่กำหนดความยากง่ายของปัญหา ปัญหาประเภทเดียวกันที่มีคำตอบที่ต้องใช้จำนวนบิตมากกว่าจะใช้เวลาในการค้นหามากกว่า ทั้งนี้เพราะการค้นหาจะต้องทำในปริภูมิการค้นหา (search space) ที่มีขนาดใหญ่กว่าจำนวนบิตที่เพิ่มขึ้นของโครโมโซมนั้นทำให้ปริภูมิการค้นหามีขนาดใหญ่ขึ้น

Genetic Algorithm : มีองค์ประกอบที่สำคัญ 5 องค์ประกอบ คือ

1) รูปแบบโครโมโซมที่ใช้ในการนำเสนอทางเลือกที่สามารถจะเป็นได้ของแต่ละปัญหา

2) วิธีสร้างประชากรต้นกำเนิด (initial population) ของทางเลือกที่สามารถจะเป็นไปได้

3) ฟังก์ชันสำหรับประเมินค่าความเหมาะสม (fitness) เพื่อให้คะแนนแต่ละทางเลือก

4) จีเนติกโอเปอเรเตอร์ (Genetic Operator) ซึ่งใช้ในการปรับเปลี่ยนองค์ประกอบของข้อมูลตลอดกระบวนการ ได้แก่การคัดเลือก การครอสโอเวอร์ และการมิวเตชัน

5) ค่าพารามิเตอร์ต่างๆ ซึ่งต้องใช้สำหรับ จีเนติกอัลกอริทึม เช่น ขนาดของประชากร, ความน่าจะเป็นของการใช้จีเนติกโอเปอเรเตอร์ และจำนวนรุ่น เป็นต้น

2.3.1 รูปแบบการนำเสนอ

ระบบค้นคืนสารสนเทศออนไลน์โดยใช้เจเนติกอัลกอริทึมนี้ จัดทำขึ้นภายใต้เวคเตอร์สเปซโมเดล โดยแต่ละเอกสารแทนด้วยเวคเตอร์ของคำสำคัญและคิวรีแทนด้วยเวคเตอร์ของคำที่ใช้ในคิวรี (query terms) การใช้ เวคเตอร์คิวรี ในการค้นคืนทำได้โดยการจับคู่ ระหว่างเอกสารกับคิวรี แล้วทำการคำนวณหาค่าความคล้ายคลึง โดยถ้าหากปรากฏคำนวณตำแหน่งนั้นในเอกสารหรือคิวรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะให้ค่าเป็น "1" หากไม่ปรากฏให้ค่าเป็น "0" ค่าที่คำนวณได้เป็นการแสดงว่าเอกสารนั้น ๆ ตรงกับดิวรีเพียงใด ซึ่งค่าความคล้ายคลึงที่วัดได้นี้จะถูกนำไปใช้ในขั้นตอนการคัดเลือกของกระบวนการเจเนติก

2.3.2 การวัดค่าความเหมาะสม

เนื่องจากโครโมโซมเป็นตัวเลขฐานสองที่ใช้นำเสนอคำตอบของปัญหาต่างๆกัน เราจึงต้องนำโครโมโซมมาถอดรหัส (decode) เสียก่อน การถอดรหัสคือการแปลงเลขฐานสองให้อยู่ในรูปแบบที่เราสามารถนำไปใช้งานอย่างอื่นต่อได้ เช่น อาจจะแปลงเลขฐานสองที่ต้อเรียงกันให้เป็นเลขจำนวนเต็ม เลขทศนิยม หรือโปรแกรมหุ่นยนต์ เป็นต้น

2.3.3 การคัดเลือก

การคัดเลือกสายพันธุ์เป็นขั้นตอนในการคัดเลือกโครโมโซม ที่ดีที่สุดจากภายในกลุ่มประชากรทั้งหมด ซึ่งโครโมโซมที่ได้จะถูกนำไปใช้เป็นต้นกำเนิดสายพันธุ์หรือ 'พ่อแม่' เพื่อใช้ในการให้กำเนิดลูกหลานในรุ่นถัดไป โดยปกติแล้วเพื่อให้ได้สายพันธุ์ที่ดีต้นกำเนิดของสายพันธุ์จะต้องดีด้วย จึงกลายเป็นปัญหาว่าจะทำการคัดเลือกต้นกำเนิดสายพันธุ์ที่ดีได้อย่างไร การคัดเลือกสายพันธุ์เป็นการจำลองการคัดเลือกโครโมโซมที่จะสามารถอยู่รอดได้ในแต่ละรุ่น สำหรับ GA นั้นจะทำการคัดเลือกโครโมโซมโดยการพิจารณาที่ค่าความเหมาะสมของโครโมโซมนั้นๆ ดังนั้นโครโมโซมไหนมีค่าความเหมาะสมที่ดีย่อมหมายถึงการเป็นโครโมโซมที่ดีและควรมีโอกาสที่จะให้ลูกหลาน (off-spring) ในจำนวนที่มากกว่าได้ซึ่งย่อมเป็นการบ่งบอกว่าโอกาสในการอยู่รอดในรุ่นถัดไปก็จะมีเพิ่มมากไปยิ่งขึ้นด้วย ได้มีการนำเสนอเทคนิควิธีของการคัดเลือกสายพันธุ์อย่างหลากหลาย ซึ่งทั้งหมดสามารถแบ่งออกได้เป็นสองกลุ่มหลักคือ

1. **วิธีที่ใช้ค่าความเหมาะสมโดยตรง** วิธีนี้จะใช้ค่าความเหมาะสม (fitness value) ของแต่ละโครโมโซมสำหรับการคัดเลือกสายพันธุ์
2. **วิธีใช้ค่าความเหมาะสมโดยอ้อม** วิธีนี้จะมีการแปลงค่าความเหมาะสมให้อยู่ในช่วงที่ต้องการ เช่นการทำให้เป็นบรรทัดฐานอยู่ในช่วง [0, 1] แล้วใช้สำหรับการคัดเลือกสายพันธุ์ต่อไป

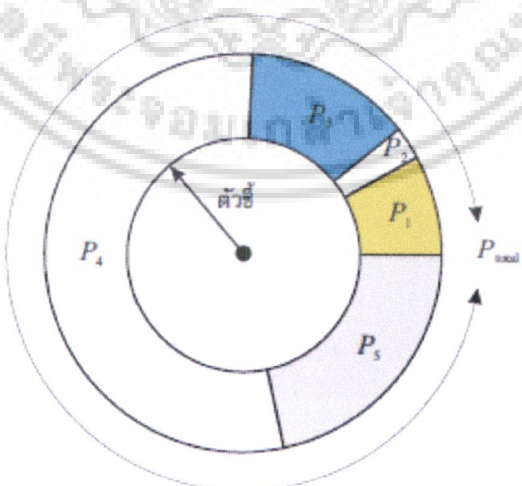
การกำหนดค่าโอกาสในการถูกคัดเลือก หลักการโดยทั่วไปของกำหนดโอกาส P คือการใช้ค่าความเหมาะสมของแต่ละโครโมโซม เพื่อช่วยในการคัดเลือกตัวอย่างวิธีการกำหนดค่าโอกาสมีดังนี้

- การคัดเลือกด้วยการแบ่งเป็นสัดส่วน
- การคัดเลือกแบบโบลต์ซมันน์
- การคัดเลือกแบบจัดอันดับ
- การคัดเลือกแบบจัดการแข่งขัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

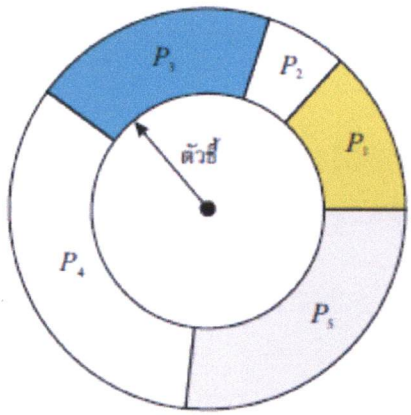
หลังจากที่ได้ทำการกำหนดค่าโอกาส P ในการถูกคัดเลือกให้กับแต่ละโครโมโซมจนหมดแล้ว ขั้นตอนต่อไปก็คือการชักตัวอย่าง ซึ่งเป็นการนำเอาค่าโอกาสนั้นไปทำการแปลงให้เป็นค่าตัวเลข ตัวเลขดังกล่าวจะแสดงถึงจำนวนของลูกหลานที่โครโมโซมนั้นๆ จะสามารถให้กำเนิดในขั้นต่อไป ได้วิธีการแปลงค่าโอกาสให้เป็นจำนวนโครโมโซมลูกหลานที่นิยมใช้มีดังต่อไปนี้

วิธีการชักตัวอย่างแบบวงล้อรูเล็ท (Roulette Wheel Sampling) ในขั้นตอนแรกจะทำการสร้างวงล้อรูเล็ทขึ้นมาก่อน โดยกำหนดให้ P_{total} คือผลรวมของค่าโอกาสในการถูกคัดเลือกของโครโมโซมในประชากรทั้งกลุ่ม ค่านี้จะมีค่าเทียบเท่ากับเส้นรอบวงของวงล้อรูเล็ท หลังจากนั้นค่า P ของโครโมโซมแต่ละตัวจะถูกแปลงไปยังบนวงล้อรูเล็ทภายในช่วง $[0, P_{total}]$ โดยที่ขนาดบนวงล้อรูเล็ทสำหรับแต่ละโครโมโซมจะสัมพันธ์กับค่า P ของโครโมโซมนั้นๆ รูปที่ 2.9 แสดงตัวอย่างของวงล้อรูเล็ทสำหรับกลุ่มประชากร S ที่มีค่าความเหมาะสม E = {3, 1, 5, 20, 8} ค่า Pi คือค่าโอกาสในการถูกคัดเลือกของโครโมโซมตัวที่ i ซึ่งได้มาจากวิธีการแบ่งเป็นสัดส่วน สังเกตว่าค่า P ของโครโมโซมจะสัมพันธ์โดยตรงกับค่าความเหมาะสมของโครโมโซม จากรูปดังกล่าว โครโมโซม S_4 ซึ่งมีค่าความเหมาะสมเท่ากับ 20 เป็นค่าสูงที่สุดในกลุ่มจะมีขนาดบนวงล้อรูเล็ทมากที่สุดและมีโอกาสในการถูกคัดเลือกสูงที่สุด (ด้วยค่า P_4) ในขณะที่โครโมโซม S_2 ซึ่งมีค่าความเหมาะสมเท่ากับ 1 เป็นค่าต่ำที่สุดในกลุ่มจะมีขนาดบนวงล้อรูเล็ทเล็กที่สุดและมีโอกาสในการถูกคัดเลือกต่ำที่สุด (ด้วยค่า P_2) ขั้นตอนในการคัดเลือกโครโมโซมจะเริ่มจากการสุ่มค่าตัวชี้ซึ่งเป็นตัวเลขระหว่าง 0 ถึง P_{total} และถ้าตัวเลขดังกล่าวตรงกับโครโมโซมใดบนวงล้อรูเล็ท โครโมโซมนั้นจะถูกเลือกกระบวนการนี้เปรียบได้กับการหมุนวงล้อในเกมรูเล็ทนั่นเอง



รูปที่ 2.9 วงล้อรูเล็ทจากค่าความเหมาะสมของแต่ละโครโมโซม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 วงล้อรูเล็ทจากวิธีการจัดอันดับ

ในแต่ละครั้งของการหมุนวงล้อก็จะได้โครโมโซมที่จะเป็นต้นกำเนิดสายพันธุ์มาหนึ่งตัว การสุ่มตัวเลขเพื่อคัดเลือกโครโมโซมจะดำเนินไปเรื่อยๆ จนกระทั่งได้ต้นกำเนิดสายพันธุ์เท่ากับจำนวนที่ต้องการ ดังนั้นสำหรับการคัดเลือกต้นกำเนิดสายพันธุ์จำนวน N โครโมโซมจะต้องทำการหมุนวงล้อรูเล็ททั้งหมด N ครั้ง เราจะเห็นได้ชัดเจนว่าโครโมโซมที่มีค่าความเหมาะสมที่สูง จะมีโอกาสถูกคัดเลือกมากกว่าโครโมโซมที่มีค่าความเหมาะสมที่ต่ำกว่า ซึ่งเป็นปรากฏการณ์ปกติในธรรมชาติทั่วๆ ไปค่า P ที่ใช้จากตัวอย่างในรูปที่ 2.9 ได้มาจากวิธีการแบ่งเป็นสัดส่วน เราจะเห็นได้ชัดเจนว่าโอกาสของโครโมโซมตัวที่สองที่จะถูกเลือกนั้น อาจมีค่าน้อยมากอันจะทำให้เกิดความเหลื่อมล้ำในการคัดเลือกขึ้น วิธีอื่นๆ ที่มีประสิทธิภาพดีกว่าจึงเป็นที่นิยมใช้มากกว่า ยกตัวอย่างเช่น วิธีการจัดอันดับ ซึ่งจะให้ค่าของโอกาสในการถูกคัดเลือกที่ไม่แปรผันไปตามขนาดของค่าความเหมาะสม ดังแสดงในรูปที่ 2.10 เราจะเห็นได้ว่าโอกาสการถูกคัดเลือกของ P_2 เพิ่มขึ้นเป็นสัดส่วน

ที่เหมาะสม (ถูกจัดเป็นอันดับสุดท้าย) ในขณะที่ P_4 และ P_5 มีค่าโอกาสในการถูกคัดเลือกที่ใกล้เคียงกันอันเนื่องมาจากมีอันดับที่ติดกัน วิธีวงล้อรูเล็ทเป็นขบวนการสุ่มคัดเลือกโครโมโซม ในบางโอกาสอาจจะมีโครโมโซมใดโครโมโซมหนึ่งเท่านั้นที่บังเอิญถูกสุ่มในการคัดเลือกทุกครั้ง ถ้าเกิดกรณีดังกล่าวขึ้น ประชากรในรุ่นถัดไปจะประกอบไปด้วยโครโมโซมชนิดเดียวกันหมดทั้งกลุ่ม ซึ่งไม่เป็นประโยชน์ต่อการค้นหาคำตอบแต่อย่างใด วิธีวงล้อรูเล็ทสามารถปรับปรุงได้หลายวิธี วิธีหนึ่งที่ง่ายและมีประสิทธิภาพคือ กำหนดให้ในแต่ละครั้งที่โครโมโซมถูกคัดเลือก ขนาดของโครโมโซมนั้นๆ บนวงล้อรูเล็ทจะมีค่าลดลงจนกระทั่งมีขนาดเป็นศูนย์วิธีนี้ทำให้แต่ละโครโมโซมที่เหลืออยู่มีขอบเขตในการถูกเลือกที่เท่าเทียมกัน ในขณะเดียวกัน ก็ช่วยลดโอกาสความเป็นใหญ่ในหมู่ประชากรของโครโมโซมตัวใดตัวหนึ่งได้

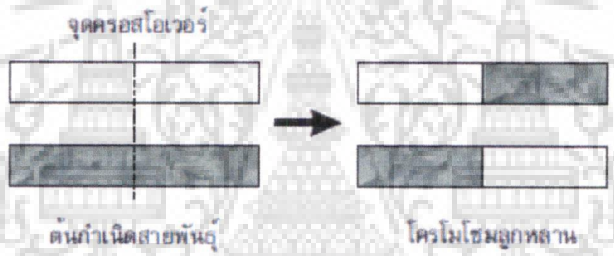
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4 การครอสโอเวอร์

การครอสโอเวอร์เป็นวิธีการรวมตัวใหม่ของโครโมโซม (Recombination operator) โดยทำการรวมส่วนย่อยระหว่างโครโมโซมต้นกำเนิดสายพันธุ์ตั้งแต่สองโครโมโซมขึ้นไป เพื่อให้กลายเป็นโครโมโซมลูกหลาน โครโมโซมลูกหลานที่ได้จากการครอสโอเวอร์นี้จะมีพันธุกรรมจากต้นกำเนิดสายพันธุ์อยู่ในตัว โดยปกติทั่วไปแล้วจะมีการกำหนดอัตราการทำครอสโอเวอร์เอาไว้ซึ่งส่วนใหญ่จะใช้ความน่าจะเป็น (P_C) เป็นตัวกำหนดอัตราดังกล่าว วิธีการทำครอสโอเวอร์มีได้หลายแบบดังรายละเอียดในตัวอย่างต่อไปนี้

1. **การทำครอสโอเวอร์แบบจุดเดียว** (Single-Point Crossover)

   การทำครอสโอเวอร์แบบจุดเดียวนี้ โครโมโซมลูกหลานจะมีสายพันธุ์ของแต่ละต้นกำเนิดอยู่อย่างละหนึ่งส่วน จุดตัดในการทำครอสโอเวอร์นั้นโดยปกติจะได้มาจากการสุ่มเลือก ตัวอย่างของการทำครอสโอเวอร์แบบจุดเดียวแสดงอยู่ในรูปที่ 2.11

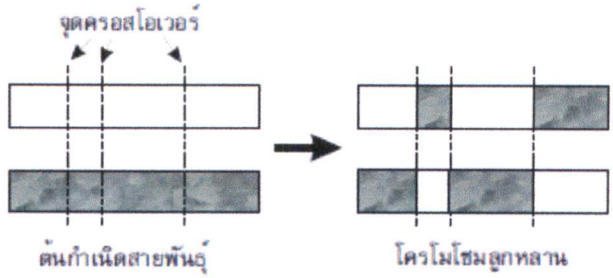


รูปที่ 2.11 แสดงครอสโอเวอร์แบบจุดเดียว

2. **การทำครอสโอเวอร แบบหลายจุด** (Multiple-Point Crossover)

   พิจารณาตัวอย่างการทำครอสโอเวอร์แบบหลายจุด ดังแสดงในรูปที่ 2.12 ที่ซึ่งมีการใช้จุดตัดทั้งหมด 3 จุด ดังนั้นโครโมโซมลูกหลานจะมีสายพันธุ์ของต้นกำเนิดอยู่มากกว่าหนึ่งส่วน หลักการเลือกจุดของครอสโอเวอร์นั้นมีอยู่หลายแบบ แต่ละแบบจะให้ผลต่อการเปลี่ยนแปลงของสายพันธุ์ในโครโมโซมลูกหลานที่แตกต่างกันออกไปด้วย วิธีที่ง่ายและเป็นที่นิยมใช้ทั่วไปคือการสุ่มเลือกจุดครอสโอเวอร์ การทำครอสโอเวอร์แบบหลายจุดจะให้ผลของลูกหลานที่มีความหลากหลายกว่าการทำครอสโอเวอร์แบบจุดเดียว อันจะมีผลให้การลู่เข้าสู่คำตอบของระบบสามารถครอบคลุมพื้นที่ของคำตอบได้มากยิ่งขึ้น อย่างไรก็ตามการทำครอสโอเวอร์แบบหลายจุด ซึ่งทำให้เกิดการเปลี่ยนแปลงของโครโมโซมลูกหลานได้มากกว่าการทำครอสโอเวอร์แบบจุดเดียวนั้นอาจจะทำให้มีโอกาสเบี่ยงเบนของคำตอบที่มีอยู่ในโครโมโซมลูกหลานได้ในอัตราที่สูงกว่าเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 แสดงครอสโอเวอร์แบบหลายจุด

2.3.5 การมิวเตชัน

การมิวเตชัน (Mutation) เป็นลักษณะของการผ่าเหล่าคือการนำโครโมโซมเก่ามาสุ่มแก้ไขบางส่วนของโครโมโซม เช่น บิทบางบิทให้เปลี่ยนไป ทำให้ได้โครโมโซมใหม่ที่มีสายพันธุ์ต่างจากเดิม ซึ่งมีโอกาสที่ จะเป็นโครโมโซมที่ดีขึ้นหรือเลวลงก็ได้ หากโครโมโซมที่ ได้ใหม่นี้เป็นโครโมโซมที่เลวลง โครโมโซมที่ได้นี้ก็จะถูกคัดออกไปในขั้นตอนการคัดเลือกเอง วัตถุประสงค์ของการมิวเตชันคือเพื่อประกันการสูญหายของข้อมูลและเพื่อความหลากหลายของข้อมูล ตัวอย่างของการทำมิวเตชัน เช่น สุ่มเลือกเปลี่ยนโครโมโซมตำแหน่งที่ 10

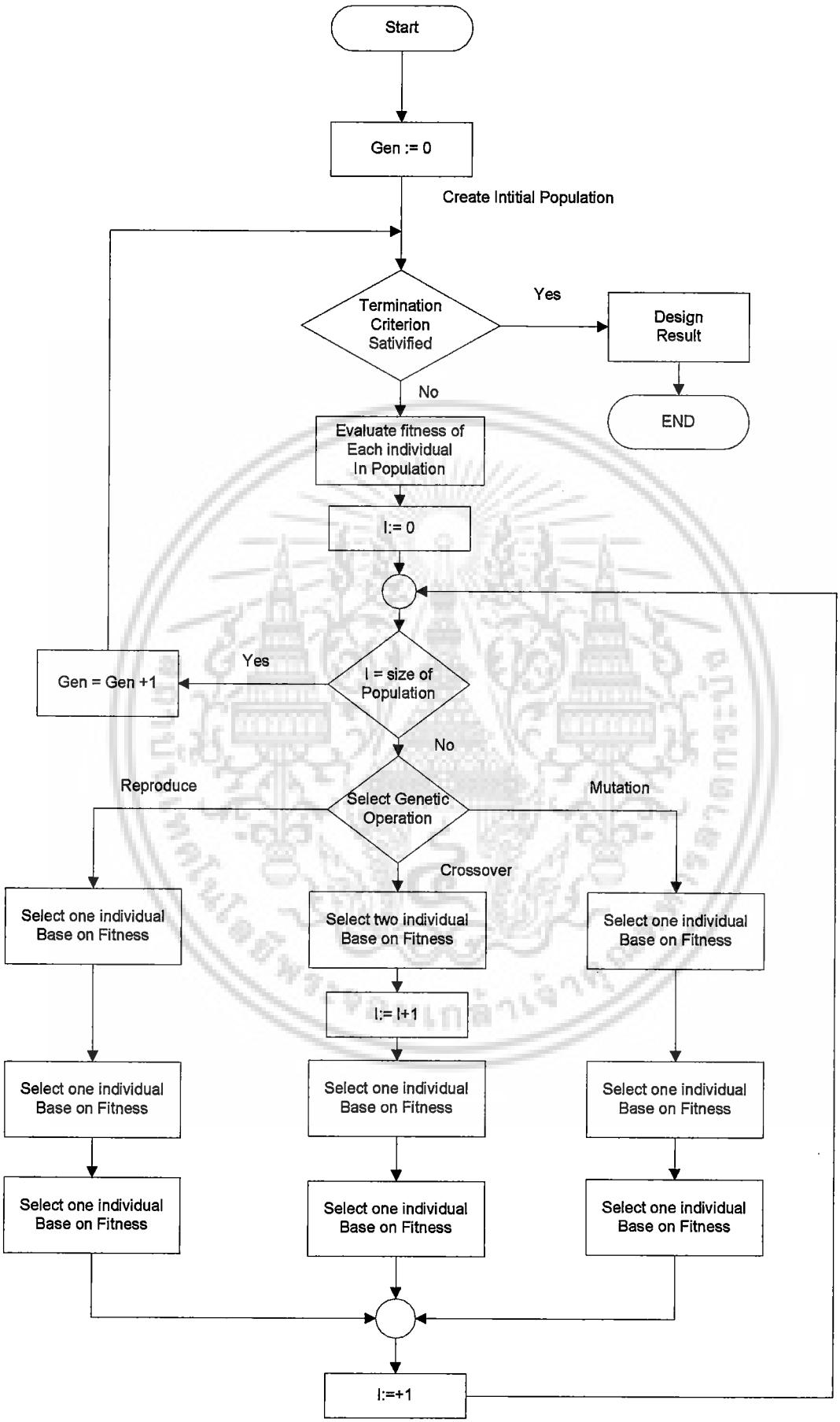
10111111**0**11101

ผลที่ได้คือ 10111111**1**11101

เจเนติกอัลกอริทึม จะทำเป็นวัฏจักรหมุนเวียนอยู่ เช่นนี้ จนกระทั่งถึงจุดหนึ่งตามเงื่อนไข โดยอาจสิ้นสุดเมื่อถึงรุ่น (Generation) ตามที่กำหนดหรือสิ้นสุดเมื่อพบคำตอบที่ดีที่สุดแล้วหรือถึง (threshold) ตามที่ได้กำหนดไว้ล่วงหน้าแล้วนั้น

หลักการทำงานของจีเนติกอัลกอริทึมคือการกำหนดฟังก์ชันความเหมาะสมรวมทั้งรูปแบบโครโมโซมเสียก่อน จากนั้นจึงเริ่มสร้างประชากรต้นกำเนิดตามรูปแบบโครโมโซมที่ได้กำหนดไว้ เมื่อได้ประชากรต้นกำเนิดแล้วก็ทำการวัดค่าความเหมาะสม (Fitness) ของแต่ละโครโมโซม เพื่อคัดเลือกเข้าสู่กระบวนการจีเนติกโอเปอเรเตอร์ โดยทำการคัดเลือกเอาเฉพาะโครโมโซมที่ มีค่าความเหมาะสมเป็นที่น่าพอใจชุดหนึ่งเก็บไว้ โครโมโซมที่คัดเลือกไว้นั้นจะถูกนำมาทำการครอสโอเวอร์และมิวเตชันได้เป็นโครโมโซมชุดใหม่ ซึ่งเราจะนำโครโมโซมชุดใหม่นี้มาวัดค่าความเหมาะสมเพื่อทำการคัดเลือกและทำการดำเนินการต่อไปจนสิ้นสุดตามเงื่อนไขที่ได้กำหนดไว้ ก็จะได้โครโมโซมที่มีค่าความเหมาะสมเป็นที่น่าพอใจ หรือได้คำตอบของปัญหาที่ต้องการ ดังแสดงในรูปที่ 2.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 การทำงานของ Hybrid Algorithm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การประยุกต์การใช้งานในการสร้างแบบจำลอง Hybrid

จากการศึกษาทฤษฎีข้างต้นแบบจำลอง Hybrid ที่จะสร้างขึ้นมานี้จะใช้หลักการของ Fuzzy logic , Neural network และ Genetic Algorithm ซึ่งจะใช้หลักการทำงานของ Neuro – Fuzzy ในการทำงานแต่จะมีการประยุกต์การใช้งานโดยใช้หลักการของ Genetic Algorithm เข้ามาช่วยในเรื่องการปรับค่าน้ำหนักแทนวิธีการ Back-propagation

ฟัซซีลอจิกและเครือข่ายประสาทเทียมถือเป็นเครื่องมือสำหรับระบบชาญฉลาดที่เป็นส่วนเติมเต็มของกันและกันได้อย่างเป็นธรรมชาติเครือข่ายประสาทเทียมมีโครงสร้างการคำนวณในระดับล่างที่มีความสามารถในการทำงานกับข้อมูลเชิงตัวเลขโดยตรง ในขณะที่ฟัซซีลอจิกมีความสามารถในการทำงานอย่างเป็นเหตุเป็นผลในระดับสูงโดยใช้ข้อมูลเชิงภาษาที่ซึ่งออกแบบโดยผู้เชี่ยวชาญที่เป็นมนุษย์ อย่างไรก็ตามฟัซซีลอจิกไม่มีความสามารถในการเรียนรู้ และไม่สามารถปรับตัวเองสำหรับสภาวะแวดล้อมที่เปลี่ยนแปลงไปได้ ในทางตรงกันข้าม เครือข่ายประสาทเทียมไม่ได้มีโครงสร้างเชิงภาษาที่ทำให้ผู้ใช้เข้าใจขั้นตอนการทำงานของระบบแต่อย่างใด ทำให้การปรับแก้โครงสร้างของเครือข่ายประสาทเทียมจะต้องทำผ่านขบวนการเรียนรู้ของเครือข่ายเท่านั้น การนำเอาฟัซซีลอจิกมารวมกับเครือข่ายประสาทเทียมจึงเป็นการนำเอาส่วนที่ถือว่าเป็นจุดเด่นที่สุดของทั้งสองระบบมาอยู่ด้วยกันทำให้ได้ระบบชาญฉลาดที่มีการประมวลผลแบบขนานและความสามารถในการปรับตัวและเรียนรู้พร้อมทั้งการใช้งานข้อมูลในรูปฐานข้อมูลความรู้ของมนุษย์และความกระจ่างของขั้นตอนการทำงานเชิงภาษาแก่ผู้ใช้ระบบ

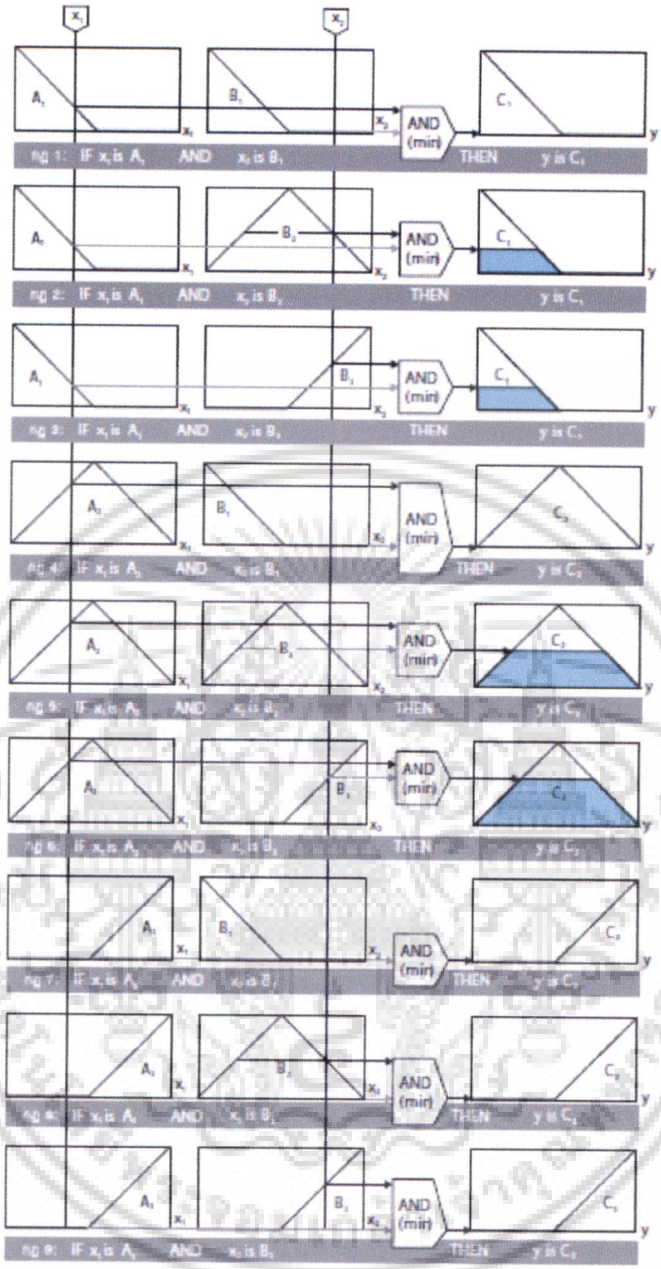
ระบบที่มีการนำเอาเครือข่ายประสาทเทียมมาใช้แทนส่วนการอนุมานฟัซซีเดิมของฟัซซีลอจิกเรียกว่าระบบนิวโร-ฟัซซี (neuro-fuzzy) ระบบดังกล่าวสามารถถูกฝึกสอนให้ทำการเรียนรู้กฎ IF-THEN ของฟัซซีลอจิก รวมไปถึงการปรับแก้พารามิเตอร์ของฟังก์ชันสมาชิกของตัวแปรอินพุตและเอาต์พุต ผู้เชี่ยวชาญที่เป็นมนุษย์สามารถแก้ไขเพิ่มเติมโครงสร้างกฎของระบบนิวโร-ฟัซซีได้อย่างมีประสิทธิภาพ รายละเอียดต่างๆ ของระบบนิวโร-ฟัซซีมีดังต่อไปนี้

2.4.1 โครงสร้างระบบนิวโร-ฟัซซี (neuro-fuzzy)

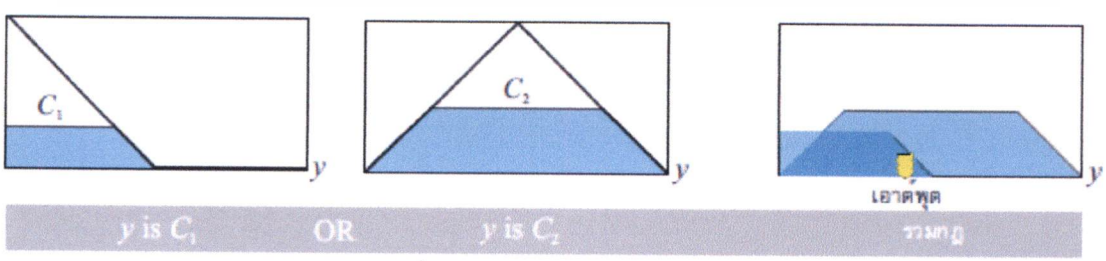
โครงสร้างของระบบนิวโร-ฟัซซีมีความคล้ายคลึงกับเครือข่ายประสาทเทียมแบบหลายชั้น โดยปกติแล้วระบบนิวโร-ฟัซซีจะมีชั้นอินพุตและชั้นเอาต์พุต และชั้นซ่อนเร้นที่ซึ่งใช้แทนฟังก์ชันสมาชิกและกฎของฟัซซีพิจารณาตัว-อย่างการอนุมานฟัซซีแบบ Mamdani ในรูปที่ 2.10 และการรวมกฎในรูปที่ 2.11 ระบบตัวอย่างมี 2 อินพุตคือ x_1 และ x_2 และ 1 เอาต์พุตคือ y อินพุต x_1 มีค่าเชิงภาษาคือ A_1 A_2 และ A_3 อินพุต x_2 มีค่าเชิงภาษาคือ B_1 B_2 และ B_3 และเอาต์พุต y มีค่าเชิงภาษาคือ C_1 C_2 และ C_3 ตัวอย่างเครือข่ายประสาทเทียมของระบบฟัซซีนี้แสดงในรูปที่ 2.12 สังเกตว่าแต่ละชั้นของเครือข่ายประสาทเทียมจะสัมพันธ์กับขั้นตอนการอนุมานในระบบฟัซซีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



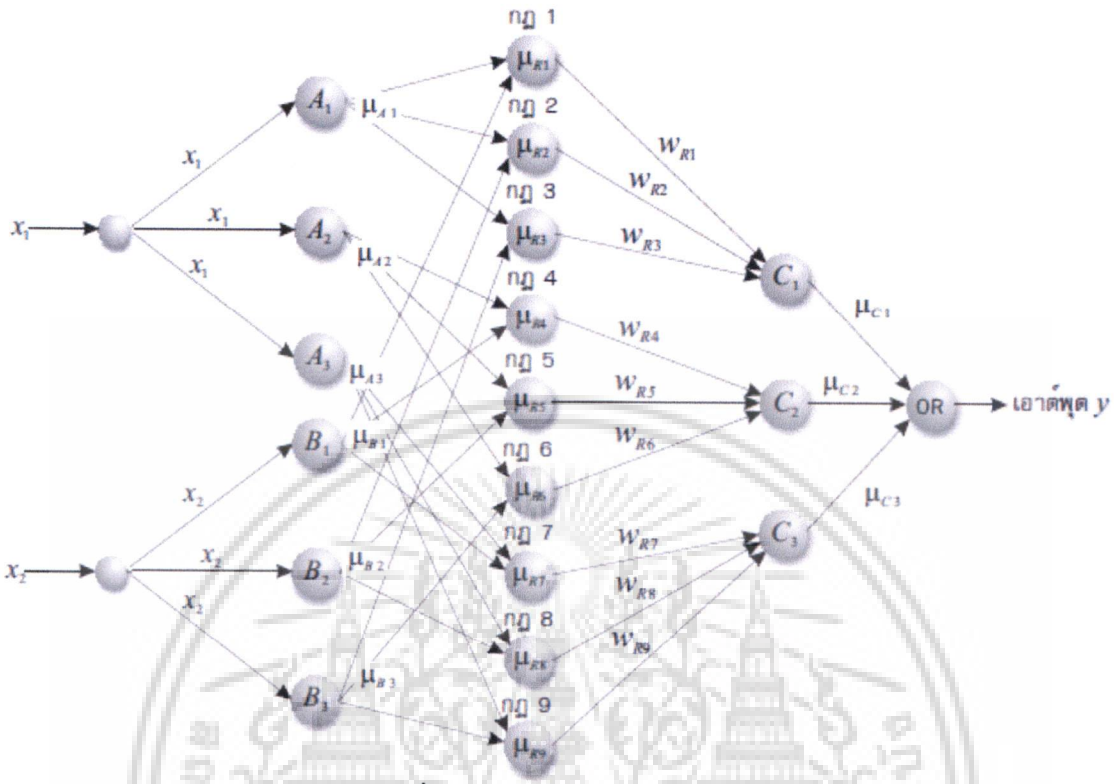
รูปที่ 2.14 ระบบอนุมานฟัซซีแบบ Mamdani



รูปที่ 2.15 การรวมกฎของฟัซซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

139308



รูปที่ 2.16 neuro-fuzzy systems

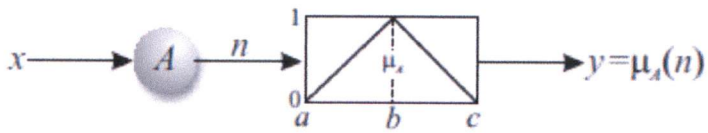
ชั้นอินพุต – ทำหน้าที่รับค่าอินพุต x_1 และ x_2 จากภายนอกระบบเข้าสู่ชั้นต่อไปของเครือข่าย

ชั้นการทำฟัซซีของอินพุต – ทำหน้าที่ในการแปลงค่าอินพุตให้อยู่ในค่าของฟังก์ชันสมาชิกนั่นเอง แต่ละนิวรอนในชั้นนี้จะเป็นส่วนเงื่อนไข IF จากกฎของฟัซซีและทำหน้าที่คำนวณค่าระดับความเป็นสมาชิกของอินพุตที่เข้ามาสู่นิวรอน ดังนั้นฟังก์ชันถ่ายโอนของนิวรอนในชั้นการทำฟัซซีนี้จะสัมพันธ์กับฟังก์ชันสมาชิกของตัวแปรอินพุตนั้นๆ ยกตัวอย่างเช่น ถ้าฟังก์ชันสมาชิกเป็นรูปสามเหลี่ยม ฟังก์ชันถ่ายโอนของนิวรอนนี้จะเป็นรูปสามเหลี่ยมด้วย ตัวอย่างฟังก์ชันสมาชิกสามเหลี่ยมที่มีพารามิเตอร์{0/a, 1/b, 0/c} มีความสัมพันธ์ดังต่อไปนี้ (ดูรูปที่2.14 ประกอบ)

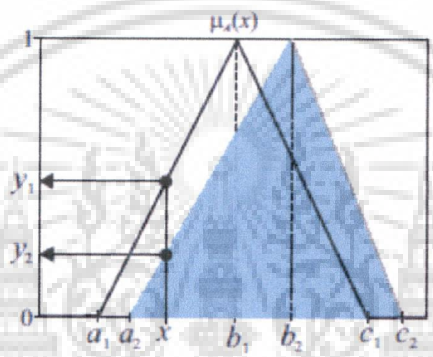
$$y = \begin{cases} 0; x \le a \\ \dfrac{x-a}{b-a}; a < x \le b \\ \dfrac{x-c}{b-c}; b < x < c \\ 0; x \ge c \end{cases}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พารามิเตอร์ a , b และ c มีผลโดยตรงต่อรูปร่างของฟังก์ชันสมาชิก ซึ่งจากรูปที่ 2.14 จะเห็นได้ว่านอกจากค่าอินพุต x แล้ว (มาจากภายนอกระบบ) ค่าเอาต์พุตของนิวรอนการทำฟัซซียังขึ้นอยู่กับพารามิเตอร์เหล่านี้ด้วย ดังนั้นค่าพารามิเตอร์เหล่านี้สามารถเปรียบเทียบได้กับค่าน้ำหนักประสาทในเครือข่ายประสาทเทียมนั่นเอง



รูปที่ 2.17 นิวรอนการทำฟัซซี



รูปที่ 2.18 ผลของพารามิเตอร์ a, b และ c ต่อรูปร่างของฟังก์ชันสมาชิก

ชั้นกฎของฟัซซี – ประกอบไปด้วยนิวรอนที่ใช้แทนกฎแต่ละข้อ นิวรอนกฎนี้จะรับอินพุต (ฟังก์ชันสมาชิก) มาจากชั้นการทำฟัซซีโดยรับจากนิวรอนที่สัมพันธ์กับเงื่อนไข IF จากแต่ละกฎ ยกตัวอย่างเช่น นิวรอนกฎข้อ 1 จะรับอินพุตจาก A1 และ B1 เป็นต้น

ในกรณีของนิวรอนกฎที่รับอินพุตมากกว่า 1 เงื่อนไข เครือข่ายจะต้องใช้ตัวปฏิบัติการทางเซตในการรวมเงื่อนไข ยกตัวอย่างเช่นการรวมเงื่อนไขด้วยตัวกระทำแบบ AND เราสามารถใช้การคูณค่าระดับความเป็นสมาชิกของแต่ละเงื่อนไข กล่าวคือสำหรับการรวม K เงื่อนไขจะได้ว่า

$$Y = X_1 \times X_2 \times X_3 \times \ldots X_k$$

โดยที่ $X_1, X_2, \ldots, X_k$ คืออินพุตจากส่วนเงื่อนไข IF y คือเอาต์พุตของนิวรอนกฎ ยกตัวอย่างเช่นเอาต์พุตนิวรอนกฎข้อที่1 จะมีค่าเท่ากับ $\mu A_1 \times \mu B_1$ ค่าเอาต์พุตของกฎข้อที่ n คือ μR_n ค่าเอาต์พุตของนิวรอนกฎนี้จะถูกปรับตามค่าน้ำหนักประสาท wR_n ในระหว่างขั้นตอนของการเรียนรู้

ชั้นการทำฟัซซีของเอาต์พุต – คล้ายคลึงกับชั้นการทำฟัซซีของอินพุต ชั้นการทำฟัซซีของเอาต์พุตทำการแปลงอินพุตจากกฎที่ผ่านเงื่อนไข IF มาจากในชั้นกฎของฟัซซีนั่นคือนิวรอนในชั้นนี้จะมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนเท่ากับค่าเชิงภาษาของตัวแปรเอาต์พุตของระบบฟัซซีนั่นเอง หลังจากนั้นแล้วจะทำการรวมกฎเข้าด้วยกันโดยใช้ตัวกระทำ OR ดังนี้

$$y = x_1 \oplus x_2 \oplus x_3 ... \oplus x_R$$

โดยที่ R คือจำนวนนิวรอนกฎทั้งหมดและ x คือเอาต์พุตจากนิวรอนกฎ ยกตัวอย่างเช่นเอาต์พุตนิวรอน C_1 จะรับอินพุตจากนิวรอนกฎข้อ 1 2 และ 3 โดยที่เอาต์พุตจากนิวรอนกฎข้อ 2 และ 3 เท่านั้นที่มีค่ามากกว่าศูนย์ (อย่างไรก็ดีเอาต์พุตจากทั้ง 3 นิวรอนกฎข้อ 1 2 และ 3 จะถูกประมวลผลด้วยตัวกระทำ OR) เอาต์พุตของนิวรอนในชั้นนี้คือ μ_{C1} μ_{C2} และ μ_{C3} (ตามค่าเชิงภาษาของตัวแปรเอาต์พุต) เอาต์พุตแต่ละตัวของนิวรอนในชั้นนี้จะเป็นค่าในการนำไปตัดยอดฟังก์ชันสมาชิกของตัวแปรเอาต์พุต แล้วนำผลของค่าเอาต์พุตทั้งหมดไปทำการดีฟัซซีในชั้นต่อไป

รูปที่ 2.19 ตัวอย่างข้อมูลรูปแบบของ XOR

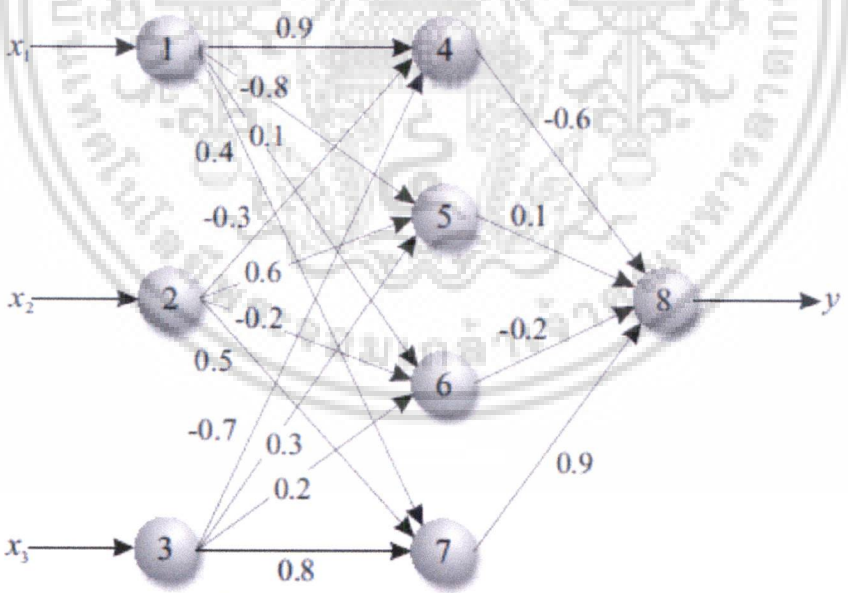
ชั้นการทำดีฟัซซี - เอาต์พุตของนิวรอนในชั้นนี้เป็นค่าเอาต์พุตที่นำออกไปใช้ภายนอกระบบ (เช่นนำไปควบคุมระบบ) การทำดีฟัซซีของชั้นนี้ใช้ปฏิบัติการดีฟัซซีปกติในระบบฟัซซีลอจิก เช่น วิธีหาค่าจุดศูนย์ถ่วง เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การเรียนรู้ของระบบ Neuro-Fuzzy

ถึงแม้ว่าเครือข่ายประสาทเทียมจะถูกใช้ในการแก้ไขปัญหาต่างๆ ได้อย่างมากมาย ตัวเครือข่ายประสาทเทียมเองยังมีข้อจำกัดอยู่บางประการ ข้อจำกัดที่สำคัญอย่างหนึ่งคือกรรมวิธีการเรียนรู้ยกตัวอย่างเช่น การเรียนรู้แบบแพร่กลับ (Back-propagation) ซึ่งเป็นอัลกอริทึมการเรียนรู้ที่นิยมใช้มากที่สุดวิธีหนึ่ง การเรียนรู้แบบแพร่กลับมีข้อจำกัดที่เด่นชัดอยู่อย่างหนึ่งคือไม่สามารถรับประกันการลู่เข้าสู่คำตอบที่เหมาะสมที่สุดได้ ในการประยุกต์ใช้งานจริง อัลกอริทึมแพร่กลับอาจจะลู่เข้าสู่คำตอบที่เป็นวงแคบเฉพาะถิ่นทำให้ไม่สามารถให้คำตอบที่ต้องการจริงๆได้ ปัญหาอีกอย่างหนึ่งของการออกแบบใช้งานเครือข่ายประสาทเทียมคือการเลือกพารามิเตอร์ของเครือข่าย ไม่ว่าจะเป็นจำนวนชั้น จำนวนนิวรอนหรือลักษณะการเชื่อมต่อระหว่างนิวรอน ซึ่งในทางทฤษฎีแล้วไม่มีหลักการที่แน่นอนในการเลือกพารามิเตอร์ของเครือข่ายให้เหมาะสมกับงาน

Genetic Algorithm หรือ GA ถือเป็นเทคนิคการหาค่าที่เหมาะสมที่สุดที่มีประสิทธิภาพมากวิธีหนึ่ง การนำเอา GA มาใช้งานร่วมกับเครือข่ายประสาทเทียมจึงเป็นที่คาดหวังในการช่วยเพิ่มประสิทธิภาพ ในการหาค่าน้ำหนักประสาทที่เหมาะสมที่สุดของเครือข่าย รวมไปถึงการค้นหาพารามิเตอร์ของเครือข่ายที่เหมาะสมกับงานด้วย ระบบดังกล่าวรวมเรียกว่า เครือข่ายประสาทเทียมเชิงวิวัฒนาการ (evolutionary neural network)



รูปที่ 2.20 เครือข่ายประสาทเทียมแบบ 3 ชั้น

0.9	-0.3	-0.7	-0.8	0.6	0.3	0.1	-0.2	0.2	0.4	0.5	0.8	-0.6	0.1	-0.2	0.9

รูปที่ 2.21 ค่าโครโมโซมเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.1 โครงสร้างเครือข่ายประสาทเทียมเชิงวิวัฒนาการ (evolutionary neural network)

2.4.2.1.1 โครโมโซม

จุดประสงค์ในการนำเอา GA มาใช้ในที่นี้คือการค้นหาค่าน้ำหนักประสาทของเครือข่ายประสาทเทียม เราจึงต้องออกแบบค่าน้ำหนักประสาทเทียมดังกล่าวในรูปของโครโมโซมที่ซึ่งใช้งานโดย GA พิจารณาการคำนวณหาค่าน้ำหนักประสาทของเครือข่ายไปข้างหน้าแบบ 3 ชั้นในรูปที่ 2.16 เครือข่ายประกอบไปด้วย 3 นิวรอนในชั้นอินพุต 4 นิวรอนในชั้นซ่อนเร้นและ 1 นิวรอนในชั้นเอาต์พุต ในรูปแสดงค่าเริ่มต้นของน้ำหนักประสาทเทียม

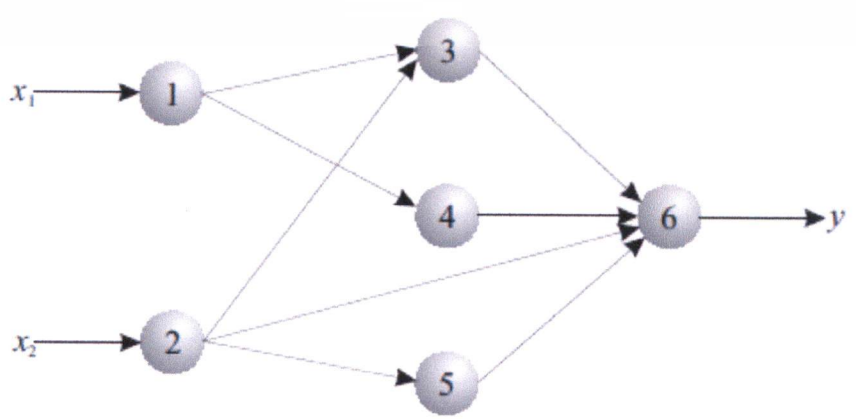
พิจารณาจำนวนการเชื่อมต่อด้วยค่าน้ำหนักประสาทในเครือข่ายทั้งหมด 16 ค่า เมื่อทำการนำเอาค่าน้ำหนักประสาทไปเข้ารหัสในเป็นโครโมโซมจะได้โครโมโซมขนาด 16 ตัวแปร อย่างไรก็ดีในการออกแบบโครโมโซมจะต้องพิจารณาการวางตำแหน่งของตัวแปรค่าน้ำหนักประสาทเหล่านี้เราสามารถวางค่าน้ำหนักประสาทเรียงจากนิวรอนในชั้นที่ 1 ซึ่งจะได้โครโมโซมเริ่มต้นคือ

0.9	-0.8	0.1	0.4	-0.3	0.6	-0.2	0.5	-0.7	0.3	0.2	0.8	-0.6	0.1	-0.2	0.9

เราสามารถวางเรียงค่าน้ำหนักประสาทในรูปแบบอื่นๆ ซึ่งมีผลต่อการทำงานของ GA ที่แตกต่างกันออกไป ยกตัวอย่างเช่น โครโมโซมในรูปที่ 2.17 แสดงการวางเรียงค่าน้ำหนักประสาทโดยพิจารณาตามลำดับค่าอินพุตที่ป้อนเข้าสู่แต่ละนิวรอน ดังนั้นจะได้ค่าโครโมโซมเริ่มต้นคือ

0.9	-0.3	-0.7	-0.8	0.6	0.3	0.1	-0.2	0.2	0.4	0.5	0.8	-0.6	0.1	-0.2	0.9

การวางเรียงค่าน้ำหนักประสาทในกรณีหลังนี้ทำให้เราสามารถพิจารณากลุ่มของยีนในโครโมโซม ที่สัมพันธ์กับนิวรอนที่มีค่าน้ำหนักประสาทในกลุ่มของยีนนั้นๆ ได้อย่างสะดวก



รูปที่ 2.22 การเข้ารหัสโดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.1.2 ฟังก์ชันค่าความเหมาะสม

ขั้นตอนต่อไปของการใช้ GA คือการออกแบบฟังก์ชันสำหรับกำหนดค่าความเหมาะสม ที่ซึ่งทำหน้าที่ในการประเมินค่าความเหมาะสมของโครโมโซมต่างๆ โดยปกติแล้วฟังก์ชันกำหนดค่าความเหมาะสมใช้ค่าวัตถุประสงค์ในการประเมิน ค่าวัตถุประสงค์ในกรณีนี้คือค่าความผิดพลาดแบบผลรวมกำลังสอง (Sum-squared error หรือ SSE) ของเครือข่าย จุดประสงค์ของ GA ในที่นี้ก็คือการหาค่าน้ำหนักประสาทของเครือข่ายที่ทำให้ค่าความผิดพลาดแบบผลรวมกำลังสองมีค่าน้อยที่สุด

2.4.2.1.3 ปฏิบัติการทางสายพันธุ์

ปฏิบัติการทางสายพันธุ์ประกอบไปด้วยการทำครอสโอเวอร์และการทำมิวเทชัน เครือข่ายลูกซึ่งมีโครงสร้างเดียวกันกับเครือข่ายพ่อแม่จะได้ค่าน้ำหนักประสาทที่เป็นผลมาจากการทำปฏิบัติการทางสายพันธุ์นี้พารามิเตอร์ของปฏิบัติการทางสายพันธุ์ เช่น ค่าความน่าจะเป็นของการทำครอสโอเวอร์และมิวเทชัน จะให้ผลลัพธ์ที่แตกต่างกันออกไปตามหลักการของ GA

2.4.2.1.4 พารามิเตอร์ อื่นๆ ของ GA

พารามิเตอร์อื่นๆ ที่เหลือของ GA เป็นส่วนที่เกี่ยวข้องกับการใช้งาน GA เท่านั้น ไม่ว่าจะเป็นวิธีการคัดเลือก วิธีการแทนที่ จำนวนประชากร (โครโมโซม) เริ่มต้นซึ่งหมายถึงจำนวนเครือข่ายที่มีค่าน้ำหนักประสาทที่แตกต่างกันตามค่าที่อยู่ในโครโมโซม หรือจำนวนรุ่นทั้งหมดที่จะทำการวนรอบหาคำตอบ

2.4.2.2 การประยุกต์ใช้ GA ในการค้นหาพารามิเตอร์ของเครือข่ายประสาทเทียม

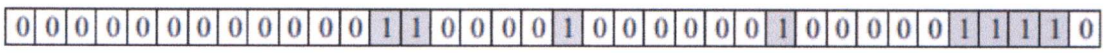
รายละเอียดการนำเอา GA มาใช้ข้างต้น เป็นการค้นหาค่าน้ำหนักประสาทที่เหมาะสมของเครือข่ายประสาทเทียมที่ซึ่งมีโครงสร้างคงที่ อย่างไรก็ดีเราสามารถนำเอา GA มาค้นหาโครงสร้างเครือข่ายประสาทเทียมที่เหมาะสมที่สุดได้ นั่นคือคำตอบที่ GA ต้องทำการค้นหา เช่นจำนวนนิวรอนและการเชื่อมต่อระหว่างนิวรอน เป็นต้น โดยทั่วไปแล้วค่าพารามิเตอร์ของเครือข่ายนี้จะใช้การสุ่มโดยผู้ออกแบบเครือข่าย เนื่องจากไม่มีทฤษฎีที่แน่นอนใดๆ ในการกำหนดพารามิเตอร์ดังกล่าวสำหรับงานหนึ่งๆ

ในการนำเอา GA มาใช้ในการค้นหาพารามิเตอร์ของเครือข่ายประสาทเทียม สิ่งแรกที่ต้องพิจารณาคือการออกแบบโครโมโซมซึ่งมีหลายวิธีในการเข้ารหัสโครโมโซม ขึ้นอยู่กับจำนวนของตัวแปรของเครือข่ายที่ต้องการจะค้นหา พิจารณาตัวอย่างการใช้วิธีการเข้ารหัสโดยตรง ถึงแม้ว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการนี้จะมีข้อจำกัดคือใช้กับเครือข่ายที่มีจำนวนนิวรอนคงที่แต่วิธีการดังกล่าวสามารถใช้ศึกษาการค้นหาการเชื่อมต่อของนิวรอนต่างๆ ภายในเครือข่ายได้ดังรายละเอียดต่อไปนี้

การเชื่อมต่อระหว่างนิวรอนภายในเครือข่ายสามารถเขียนในรูปของเมตริกซ์ได้ ยกตัวอย่าง เช่น เครือข่าย ในรูปที่ 2.18 สามารถเขียนความสัมพันธ์ ของการเชื่อมต่อระหว่างนิวรอนภายในเครือข่ายได้ดังนี้



รูปที่ 2.23 โครโมโซมการเชื่อมต่อระหว่างนิวรอน

มาจากนิวรอน:	1	2	3	4	5	6
ไปยังนิวรอน: 1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	1	1	0	0	0	0
4	1	0	0	0	0	0
5	0	1	0	0	0	0
6	0	1	1	1	1	0

รูปที่ 2.24 เปรียบเทียบแบบเมตริก

ค่าองค์ประกอบของเมตริกซ์ข้างต้นแสดงถึงการเชื่อมต่อจากนิวรอนหนึ่งไปยังอีกนิวรอนหนึ่งตามกำหนดไว้ด้วยค่า 0 (ไม่เชื่อมต่อ) หรือ 1 (เชื่อมต่อ) การเชื่อมต่อที่มีค่า 1 จะมีผลต่อการเรียนรู้ที่ซึ่งอนุญาติให้มีการปรับค่าน้ำหนักประสาทระหว่างนิวรอนคู่นั้นๆ เราสามารถทำการเข้ารหัสเมตริกซ์ข้างต้นได้ดังรูปที่ 2.19

นอกจากการค้นหาค่าน้ำหนักประสาทและการเชื่อมต่อของนิวรอน GA ยังสามารถนำมาประยุกต์ใช้ในการหาค่าพารามิเตอร์ของฟังก์ชันถ่ายโอนหรือพารามิเตอร์อื่นๆ เช่น จำนวนอินพุตที่จำเป็นต่อเครือข่าย การนำเอา GA มาใช้ค้นหาพารามิเตอร์ของเครือข่าย ทำให้ได้เครือข่ายที่มีพารามิเตอร์ที่เหมาะที่สุด เช่นมีจำนวนชั้นและนิวรอนในจำนวนที่เหมาะสมที่สุด รวมไปถึงค่าน้ำหนักประสาทที่เหมาะที่สุดด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ขั้นตอนการทำงาน

3.1 การออกแบบระบบและกระบวนการทำงาน

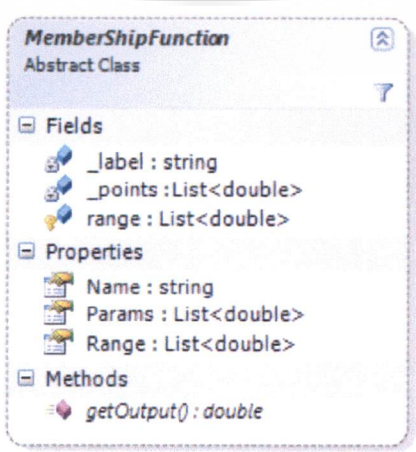
ระบบการพยากรณ์ข้อมูลที่พัฒนาขึ้นมานี้ เป็นการพยากรณ์ข้อมูลแบบระยะสั้นโดยใช้การประยุกต์กระบวนการพยากรณ์ข้อมูลแบบไฮบริดจ์อัลกอริทึมโดยการใช้ Fuzzy Logic, Neural Network และ Genetic Algorithm ในการพยากรณ์

โดยการออกแบบโครงสร้างของระบบการพยากรณ์ข้อมูลในครั้งนี้แบ่งออกเป็น 3 ส่วนใหญ่ๆ คือ

- การนำข้อมูลเข้าโดยการนำข้อมูลเข้านั้นสามารถนำเข้าได้ทั้งแบบไฟล์ Text หรือ Excel ก็ได้ ซึ่งข้อมูลที่นำเข้านั้นเป็นข้อมูลของตลาดหุ้นที่จะใช้ในการพยากรณ์
- ส่วนที่สองเป็นส่วนของการประมวลผลซึ่งในส่วนนี้จะใช้การทำงานของระบบไฮบริดจ์อัลกอริทึมในการพยากรณ์ข้อมูล
- ส่วนสุดท้ายจะเป็นส่วนของการแสดงผล ซึ่งในส่วนนี้จะทำการแสดงผลลัพธ์ของโปรแกรมหลังการประมวลผลแล้วออกมาในรูปแบบข้อมูลและกราฟ

3.2 เครื่องมือที่ใช้ในการพัฒนาระบบการพยากรณ์

ในการพัฒนาระบบการพยากรณ์หุ้นครั้งนี้ทางผู้พัฒนาได้พัฒนาโดยการใช้โปรแกรม Visual Studio 2010 ในการพัฒนาโดยเลือกใช้ C# เหตุผลที่เลือกใช้ C# นั้นเนื่องจากโปรแกรมนั้นมี Object ต่างๆ ที่ใช้ช่วยในการพัฒนาโปรแกรมมากมาย ทำให้สามารถพัฒนาได้รวดเร็ว สำหรับในส่วนของโปรแกรมนั้นประกอบด้วย Class ต่างๆดังนี้



รูปที่ 3.1 Class Membership Function

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Class MemberShipFunction: เป็น Abstract Class ที่ทำหน้าเหมือนกับฟังก์ชันสมาชิกในชั้นของฟัซซีอินพุต

Attribute

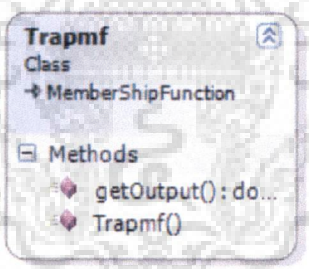
- _label เป็นข้อมูลชนิด string ทำหน้าที่เก็บชื่อของฟังก์ชันสมาชิก
- _points เป็นข้อมูลประเภท Array List ชนิด Double ทำหน้าเก็บตำแหน่งฐานของฟังก์ชันสมาชิกที่ใช้คำนวณค่าความเป็นสมาชิก
- range เป็นข้อมูลประเภท Array List ชนิด Double ทำหน้าเก็บช่วงของตำแหน่งฐานของฟังก์ชันนี้ที่มากที่สุดและน้อยที่สุด

Properties

- Name เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ _label
- Params เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ _points
- Range เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ range

Methods

- getOutput เป็น abstract ฟังก์ชันที่ถูก implement ใน Class Trapmf และ Class Trimf ทำหน้าที่คำนวณค่าความเป็นสมาชิกของฟังก์ชันสมาชิกแต่ละแบบ



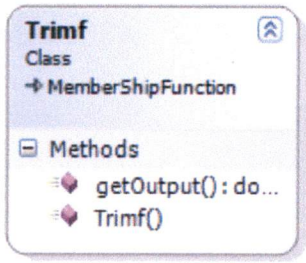
รูปที่ 3.2 Class Trapmf

Class Trapmf: เป็น Class สืบทอดจาก Class MemberShipFunction ที่ทำหน้าแทนรูปแบบฟังก์ชันสมาชิกแบบสี่เหลี่ยมคางหมูในชั้นของฟัซซีอินพุต

Methods

- getOutput เป็นฟังก์ชันทำหน้าที่คำนวณค่าความเป็นสมาชิกของฟังก์ชันสมาชิกแบบสี่เหลี่ยมคางหมู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

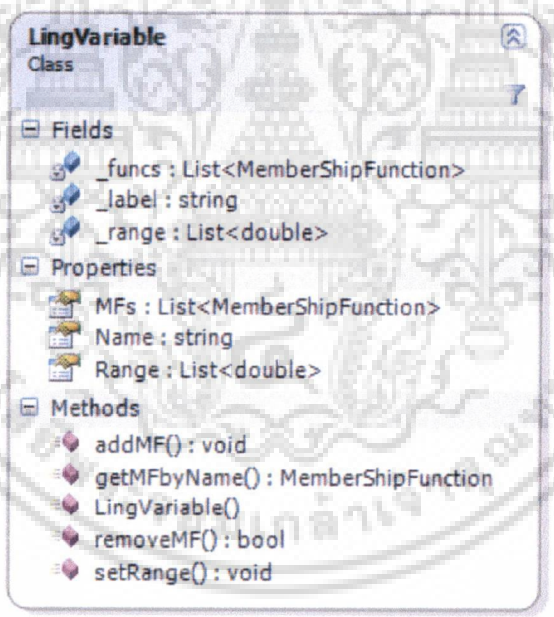


รูปที่ 3.3 Class Tripmf

Class Tripmf: เป็น Class สืบทอดจาก Class MemberShipFunction ที่ทำหน้าแทนรูปแบบฟังก์ชันสมาชิกแบบสามเหลี่ยมในชั้นของฟัซซีอินพุต

Methods

- getOutput เป็นฟังก์ชันทำหน้าที่คำนวณค่าความเป็นสมาชิกของฟังก์ชันสมาชิกแบบสามเหลี่ยม



รูปที่ 3.4 Class LingVariable

Class LingVariable: เป็น Class ที่ทำหน้าเหมือนกับตัวแปรภาษาในชั้นของอินพุต

Attribute

- _funcs เป็นข้อมูลประเภท Array List ชนิด Class MemberShipFunction เก็บฟังก์ชันสมาชิกแต่ละตัวของตัวแปรภาษานี้
- _label เป็นข้อมูลชนิด string ทำหน้าเก็บชื่อของตัวแปรภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

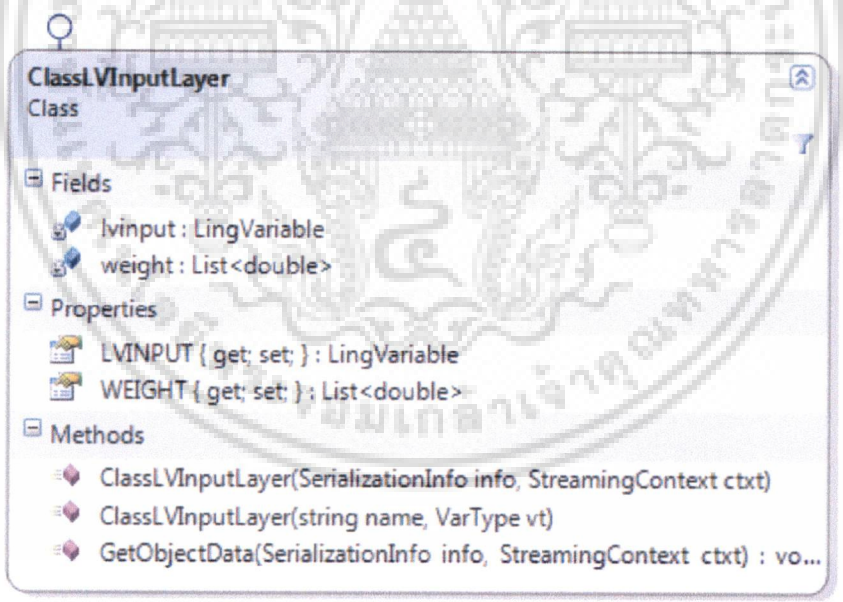
- _range เป็นข้อมูลประเภท Array List ชนิด Double ทำหน้าเก็บช่วงของตำแหน่งฐานของฟังก์ชันที่มากที่สุดและน้อยที่สุด

Properties

- Name เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ _label
- MFs เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ _funcs
- Range เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ _range

Methods

- addMF เป็นฟังก์ชันทำหน้าที่เพิ่มออบเจ็กต์ฟังก์ชันสมาชิกเข้าไปในแอทตริบิวต์ _funcs
- getMFbyName เป็นฟังก์ชันที่ค้นหาออบเจ็กต์ฟังก์ชันสมาชิกจากชื่อที่ต้องการค้นหา
- LingVariable เป็น Constructor ของ Class
- removeMF เป็นฟังก์ชันลบออบเจ็กต์ฟังก์ชันสมาชิกออกจากแอทตริบิวต์ _funcs
- setRange เป็นฟังก์ชันทำหน้าที่เซตค่าช่วงของตำแหน่งฐานของฟังก์ชันที่มากที่สุดและน้อยที่สุด



รูปที่ 3.5 Class LVInput Layer

Class ClassLVInputLayer: เป็น Class ที่ทำหน้าเหมือนกับตัวแปรภาษาในชั้นของอินพุตแต่จะเพิ่มน้ำหนักประสาทที่ใช้ในการเรียนรู้

Attribute

- lvinput เป็นข้อมูลชนิด Class LingVariable เก็บออกเจกต์ตัวแปรภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

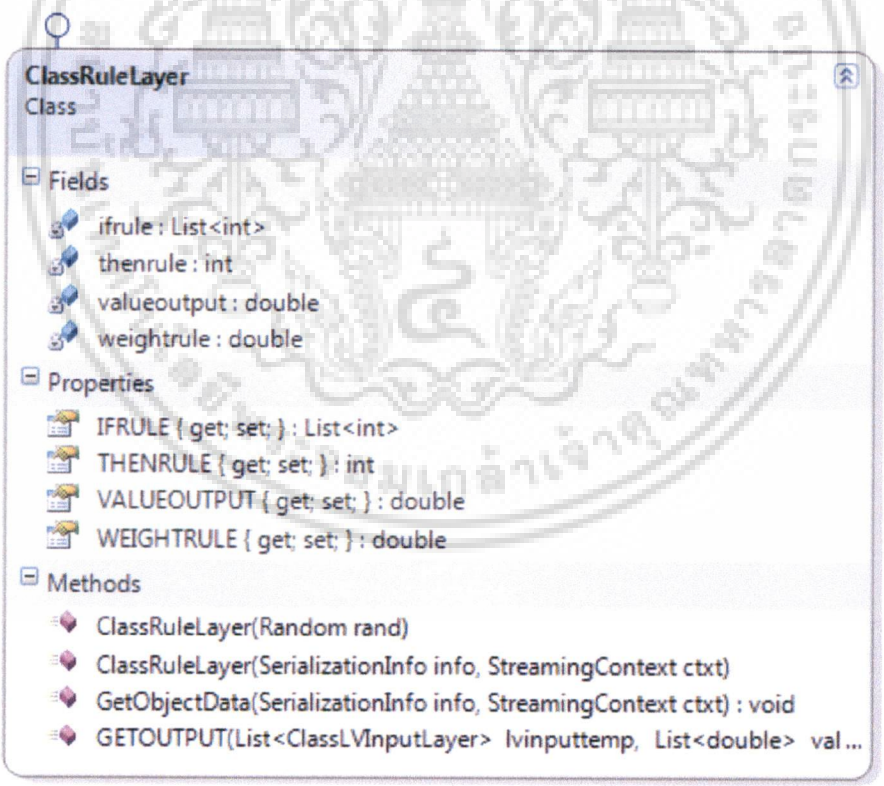
- weight เป็นข้อมูลประเภท Array List ชนิด Double ทำหน้าเก็บค่าน้ำหนักประสาทของแต่ละฟังก์ชันสมาชิกที่ตัวแปรภาษานั้นมีอยู่

Properties

- LVINPUT เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ lvinput
- WEIGHT เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ weight

Methods

- ClassLVInputLayer(SerializationInfo info, StreamingContext ctxt) เป็น Constructor ของ Class ที่จะทำงานเมื่อมีการบันทึกข้อมูลของ Class นี้ลงไฟล์ ทำหน้าที่นำข้อมูลจากแอตตริบิวต์ใน Class ลงไฟล์
- ClassLVInputLayer เป็น Constructor ของ Class
- GetObjectData เป็นฟังก์ชันทำหน้าที่คืนค่าข้อมูลจากไฟล์มาใส่ในแอตตริบิวต์ใน Class



รูปที่ 3.6 Class Rule Layer

Class ClassRuleLayer: เป็น Class ที่ทำหน้ากฎของฟัซซีในชั้นกฎของฟัซซี

Attribute

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

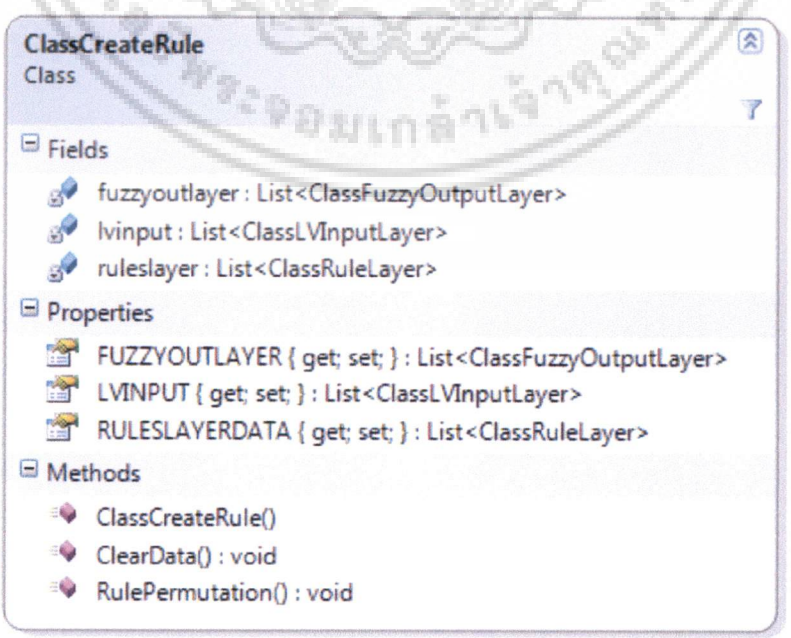
- ifrule เป็นข้อมูลประเภท Array List ชนิด int ทำหน้าที่เก็บดัชนีของฟังก์ชันสมาชิกที่เป็นอินพุตเข้าในโหนดของกฎนี้
- thenrule เป็นข้อมูลชนิด int ทำหน้าที่เก็บดัชนีของในชั้นฟัซซีของเอาต์พุตที่จะส่งออกไปเป็นอินพุตในชั้นต่อไป
- valueoutput เป็นข้อมูลชนิด Double ทำหน้าที่เก็บค่าเอาพุตของโหนดกฎนี้
- weightrule เป็นข้อมูลชนิด Double ทำหน้าที่เก็บค่าน้ำหนักประสาทในกฎของฟัซซี

Properties

- IFRULE เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ ifrule
- THENRULE เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ thenrule
- VALUEOUTPUT เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ valueoutput
- WEIGHTRULE เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ weightrule

Methods

- ClassRuleLayer(SerializationInfo info, StreamingContext ctxt) เป็น Constructor ของ Class ที่จะทำงานเมื่อมีการบันทึกข้อมูลของ Class นี้ลงไฟล์ ทำหน้าที่นำข้อมูลจากแอตตริบิวต์ใน Class ลงไฟล์
- ClassRuleLayer เป็น Constructor ของ Class
- GetObjectData เป็นฟังก์ชันทำหน้าที่คืนค่าข้อมูลจากไฟล์มาใส่ในแอตตริบิวต์ใน Class
- GETOUTPUT เป็นฟังก์ชันทำหน้าที่คำนวณเอาพุตของโหนดในชั้นกฎของฟัซซี



รูปที่ 3.7 Class Create Rule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Class ClassCreateRule: เป็น Class ที่ทำหน้าที่สร้างกฎให้กับชั้นกฎของฟัซซีแบบอัตโนมัติ

Attribute

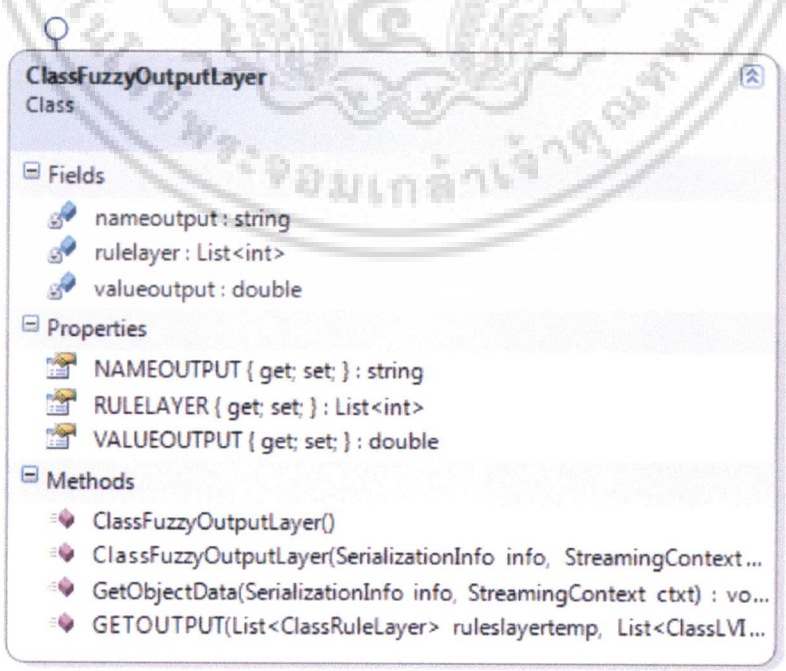
- fuzzyoutlayer เป็นข้อมูลประเภท Array List ชนิด Class ClassFuzzyOutputLayer ทำหน้าที่เก็บออบเจ็กต์ของ ClassFuzzyOutputLayer เพื่อนำมาใช้สร้างกฎในส่วน then
- lvinput เป็นข้อมูลประเภท Array List ชนิด Class ClassLVInputLayer ทำหน้าที่เก็บออกเจ็กต์ของ ClassLVInputLayer เพื่อนำมาใช้สร้างกฎในส่วน if
- ruleslayer เป็นข้อมูลประเภท Array List ชนิด Class ClassRuleLayer ทำหน้าที่เก็บออกเจ็กต์ของ ClassRuleLayer ที่ได้จากการสร้างแบบอัตโนมัติ

Properties

- FUZZYOUTLAYER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ fuzzyoutlayer
- LVINPUT เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ lvinput
- RULESLAYER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ ruleslayer

Methods

- ClassCreateRule เป็น Constructor ของ Class
- ClearData เป็นฟังก์ชันทำหน้าที่ reset ข้อมูลแอทตริบิวต์ของ Class
- RulePermutation เป็นฟังก์ชันทำหน้าที่สร้างชั้นกฎของฟัซซีแบบอัตโนมัติ



รูปที่ 3.8 Class Fuzzy Output Layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Class ClassFuzzyOutputLayer: เป็น Class ที่ทำหน้าที่การรวมกฎในชั้นการทำฟัซซีของเอาต์พุต

Attribute

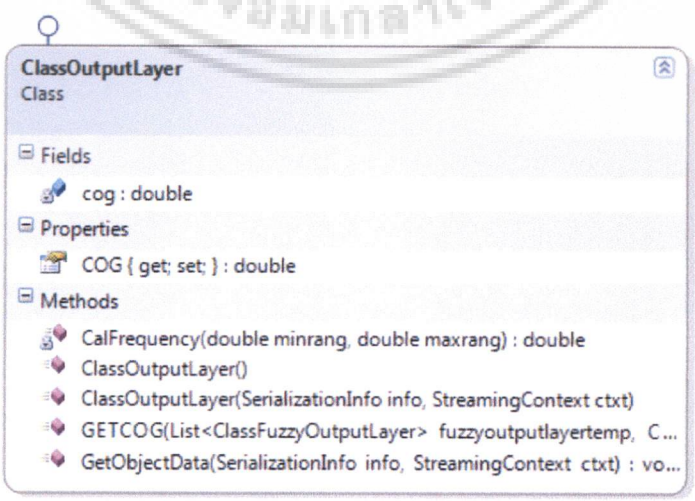
- nameoutput เป็นข้อมูลชนิด string ทำหน้าที่เก็บชื่อโหนดชั้นการทำฟัซซีของเอาต์พุต
- rulelayer เป็นข้อมูลประเภท Array List ชนิด int ทำหน้าที่เก็บดัชนีของกฎของฟัซซีเป็นอินพุตเข้าในโหนดการทำฟัซซีของเอาต์พุตนี้
- valueoutput เป็นข้อมูลชนิด Double ทำหน้าที่เก็บค่าเอาพุตที่ได้จากโหนดการทำฟัซซีของเอาต์พุตนี้

Properties

- NAMEOUTPUT เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ nameoutput
- RULELAYER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ rulelayer
- VALUEOUTPUT เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ valueoutput

Methods

- ClassFuzzyOutputLayer เป็น Constructor ของ Class
- ClassRuleLayer(SerializationInfo info, StreamingContext ctxt) เป็น Constructor ของ Class ที่จะทำงานเมื่อมีการบันทึกข้อมูลของ Class นี้ลงไฟล์ ทำหน้าที่นำข้อมูลจากแอตตริบิวต์ใน Class ลงไฟล์
- GetObjectData เป็นฟังก์ชันทำหน้าที่คืนค่าข้อมูลจากไฟล์มาใส่ในแอตตริบิวต์ใน Class
- GETOUTPUT เป็นฟังก์ชันทำหน้าที่คำนวณเอาพุตของโหนดในการทำฟัซซีของเอาต์พุต



รูปที่ 3.9 Class Output Layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Class ClassOutputLayer: เป็น Class ที่ทำหน้าที่การทำดีฟัซซีในชั้นการทำดีฟัซซี

Attribute

- cog เป็นข้อมูลชนิด double ทำหน้าที่เก็บค่าจุดศูนย์ถ่วงที่คำนวณได้

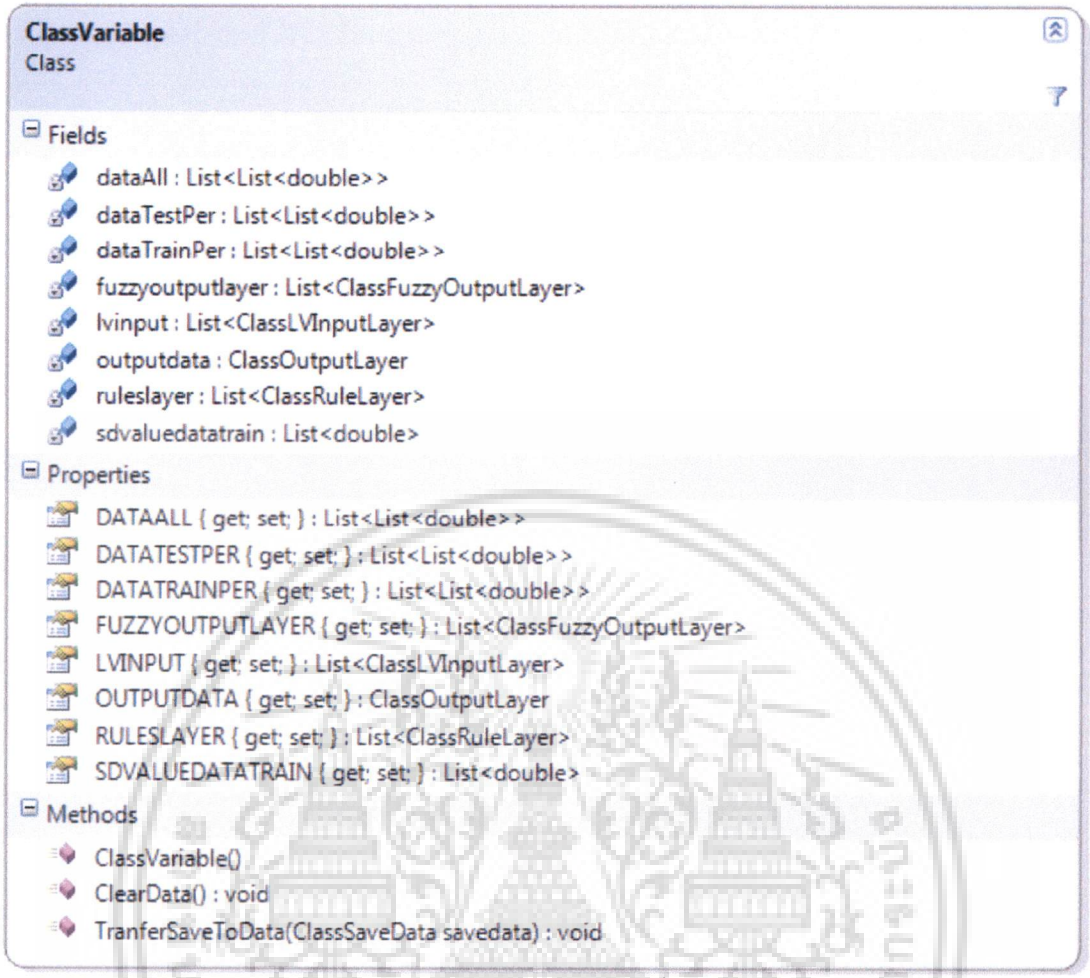
Properties

- COG เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ cog

Methods

- ClassOutputLayer เป็น Constructor ของ Class
- ClassOutputLayer(SerializationInfo info, StreamingContext ctxt) เป็น Constructor ของ Class ที่จะทำงานเมื่อมีการบันทึกข้อมูลของ Class นี้ลงไฟล์ ทำหน้าที่นำข้อมูลจากแอตตริบิวต์ใน Class ลงไฟล์
- GetObjectData เป็นฟังก์ชันทำหน้าที่คืนค่าข้อมูลจากไฟล์มาใส่ในแอตตริบิวต์ใน Class
- GETCOG เป็นฟังก์ชันทำหน้าที่คำนวณหาค่าจุดศูนย์ถ่วงซึ่งเอาพุตของโหนดการทำดีฟัซซี
- CalFrequency เป็นฟังก์ชันทำหน้าที่คำนวณค่าความละเอียดของการคำนวณค่าจุดศูนย์ถ่วง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 Class Variable

Class ClassVariable: เป็น Class ที่ทำหน้าที่เก็บข้อมูลต่างเพื่อใช้งานในระบบ

Attribute

- dataAll เป็นข้อมูลประเภท Array List 2 มิติ ชนิด Double ทำหน้าที่เก็บข้อมูลของชุดข้อมูลที่อ่านมาจาก file
- dataTestPer เป็นข้อมูลประเภท Array List 2 มิติ ชนิด Double ทำหน้าที่เก็บข้อมูลของชุดข้อมูลใช้ทดสอบที่แบ่งมาจาก dataAll
- dataTrainPer เป็นข้อมูลประเภท Array List 2 มิติ ชนิด Double ทำหน้าที่เก็บข้อมูลของชุดข้อมูลใช้เรียนรู้ที่แบ่งมาจาก dataAll
- fuzzyoutptulayer เป็นข้อมูลประเภท Array List ชนิด Class ClassFuzzyOutputLayer ทำหน้าที่เก็บข้อมูลชั้นการทำฟัซซีของเอาต์พุต
- lvinput เป็นข้อมูลประเภท Array List ชนิด Class ClassLVInputLayer ทำหน้าที่เก็บข้อมูลชั้นการทำฟัซซีของอินพุต
- outputdata เป็นข้อมูลชนิด Class ClassOutputLayer ทำหน้าที่เก็บข้อมูลชั้นทำดีฟัซซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ruleslayer เป็นข้อมูลประเภท Array List ชนิด Class ClassRuleLayer ทำหน้าที่เก็บข้อมูลชั้นกฎฟัซซี
- sdvaluedatatrain เป็นข้อมูลประเภท Array List ชนิด Double ทำหน้าที่เก็บข้อมูลส่วนเบี่ยงเบนมาตรฐานของแต่ละอินพุตโหนด

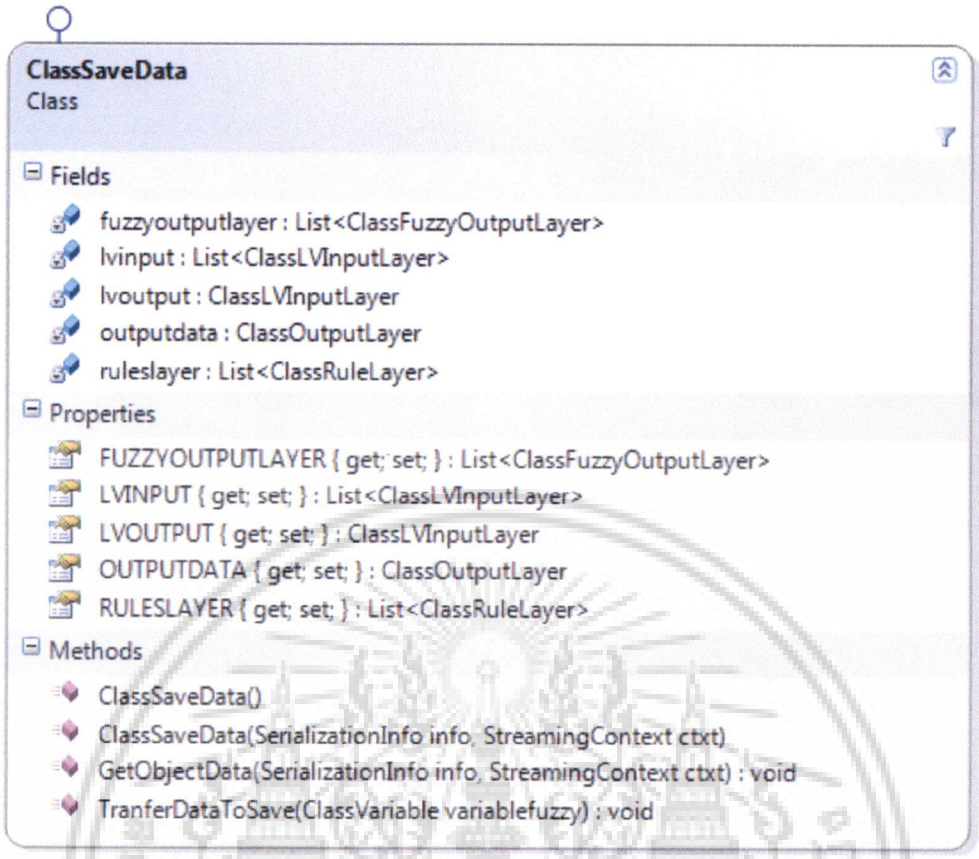
Properties

- DATAALL เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ dataAll
- DATATESTPER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ dataTestPer
- DATATRAINPER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ dataTrainPer
- FUZZYOUTPTULAYER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ fuzzyoutptulayer
- LVINPUT เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ lvinput
- OUTPUTDATA เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ outputdata
- RULESLAYER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ ruleslayer
- SDVALUEDATATRAIN เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ sdvaluedatatrain

Methods

- ClassVariable เป็น Constructor ของ Class
- ClearData เป็นฟังก์ชันที่ใช้ reset ค่าของแอทตริบิวต์ทั้งหมดใน Class
- TranferSaveToData เป็นฟังก์ชันทำหน้าที่นำข้อมูลจากออกเจ็กต์ ClassSaveData ที่อ่านจาก file ใส่เก็บลงในแอทตริบิวต์ของออบเจ็กต์ ClassVariable

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 Class Save Data

Class ClassSaveData: เป็น Class ที่ทำหน้าที่เก็บข้อมูลต่างเพื่อใช้งานในระบบ

Attribute

- fuzzyoutptulayer เป็นข้อมูลประเภท Array List ชนิด Class ClassFuzzyOutputLayer ทำหน้าที่เก็บข้อมูลชั้นการทำฟัซซีของเอาต์พุต
- lvinput เป็นข้อมูลประเภท Array List ชนิด Class ClassLVInputLayer ทำหน้าที่เก็บข้อมูลชั้นการทำฟัซซีของอินพุต
- outputdata เป็นข้อมูลชนิด Class ClassOutputLayer ทำหน้าที่เก็บข้อมูลชั้นทำดีฟัซซี
- ruleslayer เป็นข้อมูลประเภท Array List ชนิด Class ClassRuleLayer ทำหน้าที่เก็บข้อมูลชั้นกฎฟัซซี

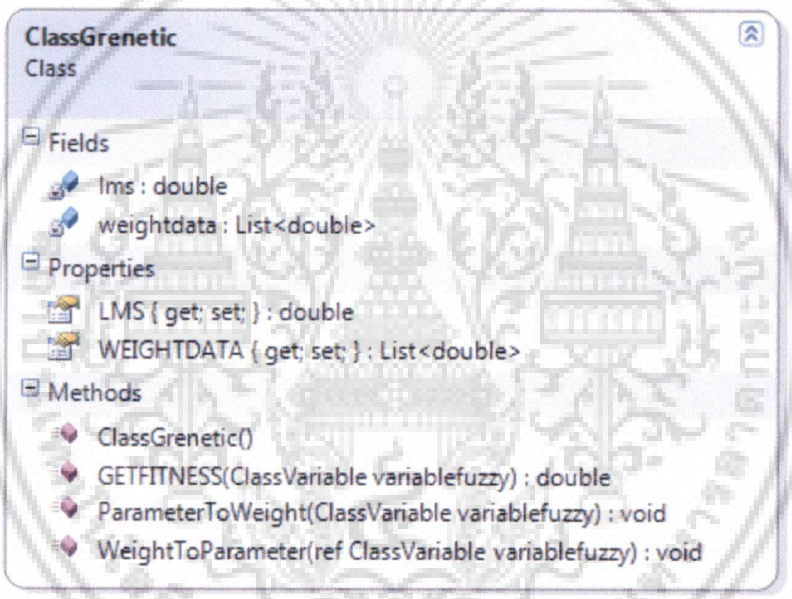
Properties

- FUZZYOUTPTULAYER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ fuzzyoutptulayer
- LVINPUT เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ lvinput
- OUTPUTDATA เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ outputdata
- RULESLAYER เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ ruleslayer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Methods

- ClassSaveData เป็น Constructor ของ Class
- ClassOutputLayer(SerializationInfo info, StreamingContext ctxt) เป็น Constructor ของ Class ที่จะทำงานเมื่อมีการบันทึกข้อมูลของ Class นี้ลงไฟล์ ทำหน้าที่นำข้อมูลจากแอตตริบิวต์ใน Class ลงไฟล์
- GetObjectData เป็นฟังก์ชันทำหน้าที่คืนค่าข้อมูลจากไฟล์มาใส่ในแอตตริบิวต์ใน Class
- TranferDataToSave เป็นฟังก์ชันทำหน้าที่นำข้อมูลจากออกเจ็กต์ ClassVariable ใส่เก็บลงในแอทตริบิวต์ของออบเจ็กต์ ClassSaveData



รูปที่ 3.12 Class Genetic

Class ClassGrenetic: เป็น Class ทำหน้าที่สร้างโครโมโซมที่ใช้ในการเรียนรู้เพื่อปรับน้ำหนักโดยวิธี Grenetic Algorithm

Attribute

- lms เป็นข้อมูลชนิด double ทำหน้าที่เก็บค่า mean squared error ของโครโมโซม
- weightdata เป็นข้อมูลประเภท Array List ชนิด Double เก็บค่าน้ำหนักประสาทของโครงสร้าง Fuzzy Neural Network

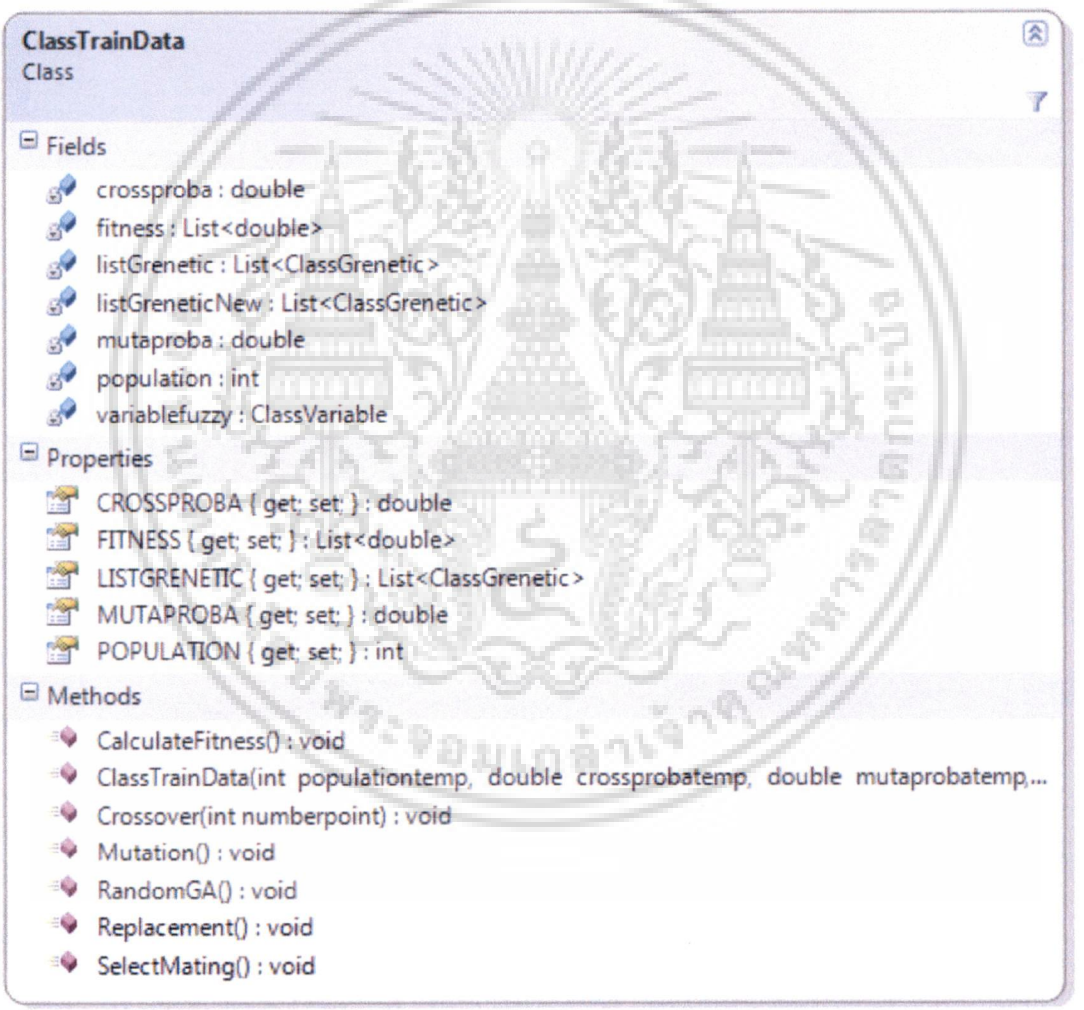
Properties

- LMS เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ lms
- WEIGHTDATA เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ weightdata

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Methods

- ClassGrenetic เป็น Constructor ของ Class
- GETFITNESS เป็นฟังก์ชันทำหน้าที่คำนวณค่าความเหมาะสมหรือคำนวณค่า mean squared error ของโครโมโซม
- ParameterToWeight เป็นฟังก์ชันทำหน้าที่นำค่าน้ำหนักประสาทจากโครงสร้าง Fuzzy Neural Network ใส่แอทตริบิวต์ weightdata
- WeightToParameter เป็นฟังก์ชันทำหน้าที่นำค่าน้ำหนักประสาทจากแอทตริบิวต์ weightdata ใส่โครงสร้าง Fuzzy Neural Network



รูปที่ 3.13 Class Train Data

Class ClassTrainData: เป็น Class ทำหน้าที่เรียนรู้ปรับน้ำหนักประสาทโดยใช้ Grenetic Algorithm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Attribute

- crossproba เป็นข้อมูลชนิด Double ทำหน้าที่เก็บค่าความน่าจะเป็นของการ Crossover
- fitness เป็นข้อมูลประเภท Array List ชนิด Double เก็บค่าความเหมาะสมของแต่ละโครโมโซม
- listGrenetic เป็นข้อมูลประเภท Array List ชนิด Class ClassGrenetic เก็บค่าออบเจ็กต์ของ ClassGrenetic ซึ่งเป็นโครโมโซมในแต่ละรุ่น
- listGreneticNew เป็นข้อมูลประเภท Array List ชนิด Class ClassGrenetic เก็บค่าออบเจ็กต์ของ ClassGrenetic ซึ่งเป็นโครโมโซมลูกที่ได้จากการ Crossover
- mutaproba เป็นข้อมูลชนิด Double ทำหน้าที่เก็บค่าความน่าจะเป็นของการกลายพันธ์
- population เป็นข้อมูลชนิด int ทำหน้าที่เก็บจำนวนโครโมโซมในแต่ละรุ่น
- variablefuzzy เป็นข้อมูลชนิด Class ClassVariable ทำหน้าที่เก็บออบเจ็กต์ของ ClassVariable ซึ่งมีโครงสร้างของ Fuzzy Neural Network เก็บอยู่

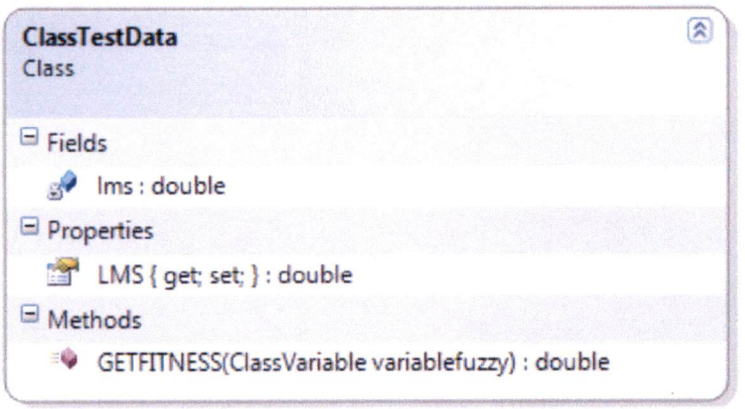
Properties

- CROSSPROBA เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ crossproba
- FITNESS เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ fitness
- LISTGRENETIC เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ listGrenetic
- MUTAPROBA เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ mutaproba
- POPULATION เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ population

Methods

- CalculateFitness เป็นฟังก์ชันคำนวณค่าความเหมาะสมของโครโมโซมทั้งหมด
- ClassTrainData เป็น Constructor ของ Class
- Crossover เป็นฟังก์ชันทำหน้าทำการ Crossover โครโมโซม
- Mutation เป็นฟังก์ชันทำหน้าที่กลายพันธ์โครโมโซมที่ได้จากการ Crossover
- RandomGA เป็นฟังก์ชันทำหน้าที่สุ่มประชากรโครโมโซมเริ่มต้น
- Replacement เป็นฟังก์ชันทำหน้าที่แทนที่โครโมโซมรุ่นใหม่หลังจากที่ทำปฏิบัติการทางพันธ์กรรมแล้ว
- SelectMating เป็นฟังก์ชันที่ทำหน้าที่เลือกคู่โครโมโซมมาทำการ Crossover

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 Class Test Data

Class ClassTestData: เป็น Class ทำหน้าที่นำโครงสร้างที่ได้หลังจากการเรียนรู้น้ำหนักประสาทมาทดสอบวัดค่าความเหมาะสมกับชุดข้อมูลทดสอบ

Attribute

- lms เป็นข้อมูลชนิด Double ทำหน้าที่เก็บค่าความเหมาะสมที่คำนวณกับชุดข้อมูลทดสอบ

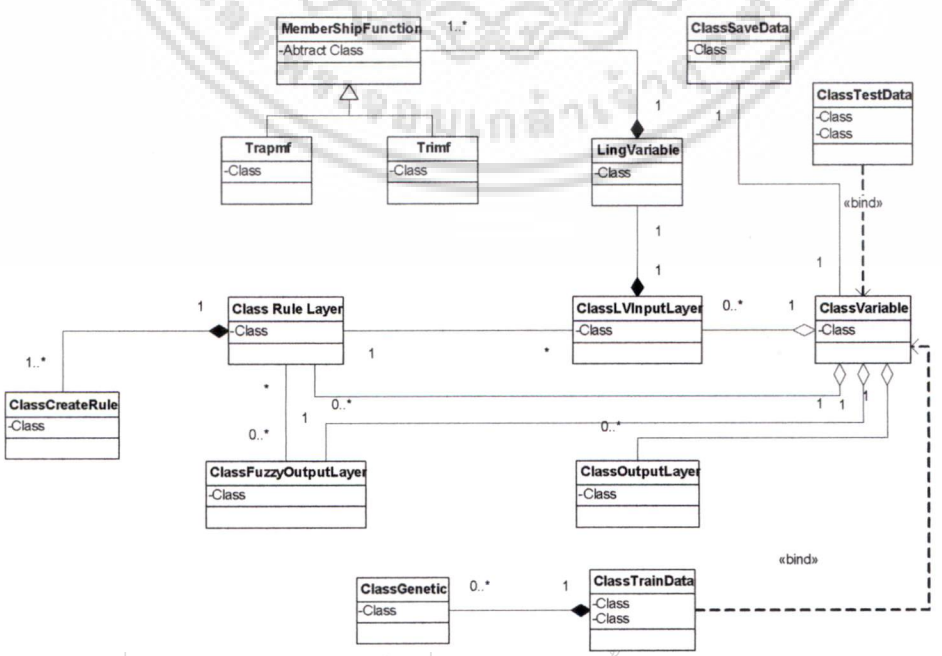
Properties

- LMS เป็นฟังก์ชันที่ใช้ในการเข้าถึงข้อมูลแอทตริบิวต์ lms

Methods

- GETFINESS เป็นฟังก์ชันคำนวณค่าความเหมาะสมกับชุดข้อมูลทดสอบ

Class โดยรวมของระบบสามารถแสดงได้ดังนี้



รูปที่ 3.15 Class ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก Class ของระบบนั้นสามารถเขียนเป็น Pseudo Code ได้ดังต่อไปนี้

```
Main()
        parent_chomosomes := InitPopulation()
        iter := 0
        While iter <  MaxIterance
                fitness := EvaluateFitness(parent_chomosomes, training_data)
                offset_pairing := RouletteWheelSelection(fitness)
                child_ chomosomes := Crossover(parent_chomosomes, offset_ pairing)
                child_ chomosomes := Mutation(child_ chomosomes)
                parent_chomosomes := Replacement(parent_chomosomes ,child_ chomosomes,
                training_data)
                iter := iter + 1

InitPopulation()
        i := 0
        first_chomosomes := {}
        while i < population
                number_rand := RandomCommand()
                new_chomosome := Create_Chomosome(number_rand)
                first_chomosomes := first_chomosomes ∪ new_chomosome
                i := i + 1
        return first_chomosomes

EvaluateFitness(parent_chomosomes, training_data)
        fitness := {}
        number_total_ training_data := Length(training_data)
        foreach  chomosome in parent_chomosomes
                MSE := 0
                foreach input in training_data
                        y := Getoutput(fuzzy_neural_network_structure, chromosome, input)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    MSE := (T_input – y)^2
                fitness := fitness ∪ (MSE/ number_total_ training_data)
        return fitness

RouletteWheelSelection(fitness)
        P_total := 0
        rang_value := 0
        offset_ pairing := {}
        rangP := {}
        foreach f in fitness
                P_total := P_total + 1/(f+Ɛ)
        foreach f in fitness
                P := (1/(f+Ɛ))/ P_total
                rangP := rangP ∪ [rang_value, rang_value + P]
        foreach each chromosome is paired
                number_rand := Random()
                foreach rang in rangP
                        if number_rand >= rang[0] and number_rand < rang[1]
                                value_offset1 := offset_rang
                                break
                number_rand := Random()
                foreach rang in rangP
                        if number_rand >= rang[0] and number_rand < rang[1]
                                value_offset2 := offset_rang
                                break
                if value_offspring1 ≠ value_offspring2 and pairing not in offset_ pairing
                        offset_ pairing := offset_ pairing ∪ [value_offset1, value_offset2]
        return offset_ pairing

Crossover(parent_chomosomes, offset_ pairing)
        child = {}
        foreach offset in offset_ pairing
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                chromosome1 := parent_chomosomes[offset[0]]
                chromosome2 := parent_chomosomes[offset[1]]
                randomly choose x-over point
                number_rand := Random()
                if number_rand < Crossover_Probability
                        [new_ chromosome1, new_ chromosome2] := chromosome1⊗
chromosome2
                        child := child ∪ new_ chromosome1
                        child := child ∪ new_ chromosome2
        return child

Mutation(child_ chomosomes)
        foreach chromosome in child_ chomosomes
                mutation_pt := RandomlyChooseMutationPoint(chromosome)
                foreach weight in chromosome
                        number_rand := Random()
                        if number_rand < Mutation_Probability
                                value_shift := GenerateRandom()
                                weight := weight + value_shift
                                update value in child_ chomosomes
        return child_ chomosomes

Replacement(parent_chomosomes ,child_ chomosomes, training_data)
        nextgeneration_ chomosomes = {}
        parent_fitness := EvaluateFitness(parent_chomosomes, training_data)
        child _fitness := EvaluateFitness(child_ chomosomes, training_data)
        fitness := parent_fitness ∪ child _fitness
        [fitness, offset_fitness_after_sort] := sort(fitness)
        chomosomes := parent_chomosomes ∪ child_ chomosomes
        foreach offset in offset_fitness_after_sort
                nextgeneration_ chomosomes := nextgeneration_ chomosomes ∪
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
chomosomes[offset]
        return nextgeneration_ chomosomes
```

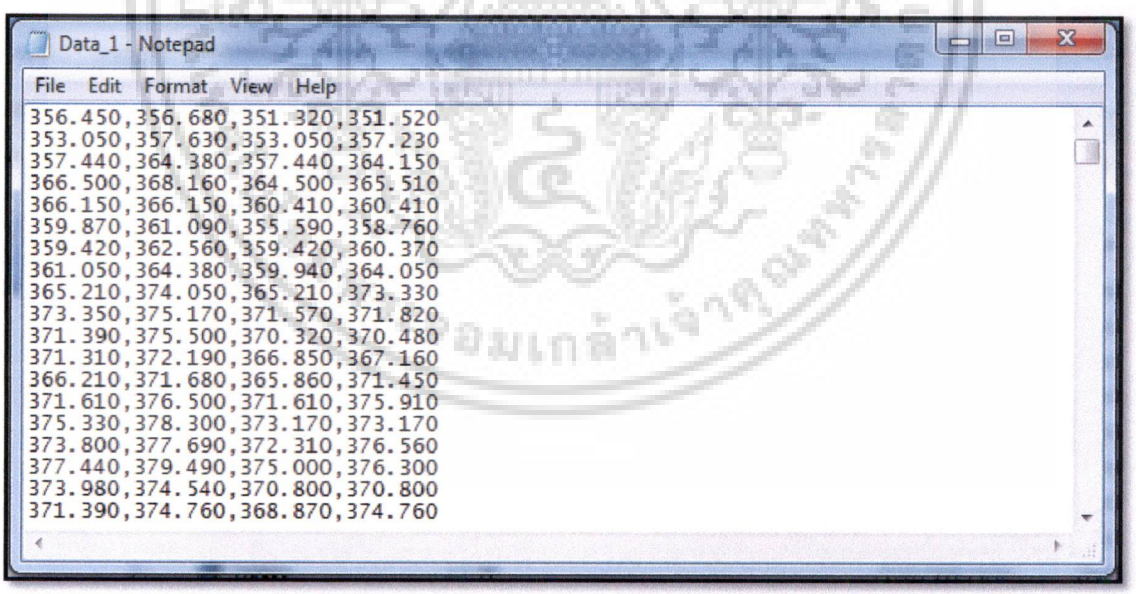
3.3 การทำงานของระบบการพยากรณ์หุ้น

ในกระบวนการทำงานของระบบการพยากรณ์หุ้นแบบไฮบริดจ์อัลกอริทึมนั้น ในระบบจะต้องมีการกำหนดค่าพารามิเตอร์ต่างๆให้ถูกต้องเหมาะสมจึงจะทำให้การพยากรณ์ราคาหุ้นที่ได้ออกมานั้นมีประสิทธิภาพ ถูกต้อง แม่นยำ และตรงกับความต้องการของผู้ใช้งานมากที่สุด ซึ่งทางผู้พัฒนาโปรแกรมจะอธิบายถึงการกำหนดค่าของพารามิเตอร์ต่างๆ ดังนี้

3.3.1 การนำข้อมูลเข้า

ขั้นตอนในการนำข้อมูลเข้านั้นจะมีการกำหนดรูปแบบของข้อมูลดังนี้

1. นำข้อมูลของราคาหุ้นที่จะใช้งานมาโดยต้องเป็นรูปแบบ Excel หรือ Text File เท่านั้น และจะต้องอยู่ในรูปแบบที่กำหนด โดยเรียงจากบนลงล่างเท่านั้น โดยจะมีกี่แถวกี่คอลัมน์ก็ได้ ซึ่งในแต่ละคอลัมน์นั้นจะทำการคั่นด้วนเครื่องหมายคอมม่า (,) เพื่อจะเป็นการบอกว่าข้อมูลแต่ละตัวสิ้นสุดที่ไหน ดังรูปที่ 3.16



รูปที่ 3.16 รูปแสดงไฟล์ Text ของตลาดหุ้น

2. เมื่อได้ข้อมูลที่จะใช้งานแล้วหลังจากนั้นต้องทำการเลือกคอลัมน์ที่จะใช้งานในการพยากรณ์ว่าต้องการใช้คอลัมน์ไหน จากนั้นทำการกำหนดค่าของอินพุตว่าต้องการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินพุตกี่ค่าในการพยากรณ์โดยขั้นตอนนี้สามารถทำได้ในตัวโปรแกรมที่ทางผู้พัฒนาได้จัดทำขึ้น

3.3.2 การเรียนรู้ของเครือข่าย

หลังจากที่มีการนำข้อมูลเข้าสู่ระบบเรียบร้อยแล้วระบบจะต้องทำการเรียนรู้ข้อมูลต่างๆ เพื่อที่จะได้ทำการพยากรณ์ได้ถูกต้อง โดยจะมีขั้นตอนต่างๆดังนี้

1. เมื่อทำการเลือกคอลัมน์และจำนวนอินพุตที่จะใช้งานในระบบเรียบร้อยแล้วหลังจากนั้นกดที่ปุ่ม สร้างโหนดฟัซซีแบบอัตโนมัติ
2. จากนั้นเมื่อระบบทำการสร้างโหนดเรียบร้อยแล้ว ต้องทำการสร้างกฎขึ้นเพื่อที่จะเรียนรู้กฎต่างๆเพื่อที่จะได้สามารถทำการปรับค่าน้ำหนักได้โดยการกดที่ปุ่ม สร้างกฎ ก็จะได้กฎทั้งหมดที่สามารถจะเป็นไปได้ออกมาหลังจากนั้นกดที่ปุ่ม บันทึกกฎ
3. จากนั้นจะเป็นส่วนของ Genetic Algorithm ที่จะใช้ในการเรียนรู้การพยากรณ์ข้อมูลต่างๆ โดยในส่วนนี้นั้นจะมีการกำหนดค่าพารามิเตอร์ต่างๆ เช่น ความน่าจะเป็นที่จะเกิดการครอสโอเวอร์ , ความน่าจะเป็นในการกลายพันธ์ , กำหนดจำนวนประชากร , กำหนดจำนวนรอบที่จะใช้ในการเรียนรู้ข้อมูลเพื่อที่จะใช้ในการปรับค่าน้ำหนักประสาท , การแบ่งข้อมูลออกเป็นข้อมูล Train Set และ Test Set โดยจะแบ่งออกเป็นเปอร์เซนต์ของข้อมูล
4. หลังจากที่ทำการกำหนดค่าของพารามิเตอร์ต่างๆแล้วจากนั้นกดที่ปุ่ม เรียนรู้ ซึ่งหลังจากที่กดแล้วระบบจะทำการเรียนรู้ข้อมูลตามจำนวนรอบที่ได้กำหนดไว้

3.3.3 การพยากรณ์ราคาหุ้น

การพยากรณ์หุ้นที่ดีนั้นควรที่จะออกแบบให้มีการใช้งานง่ายสามารถเข้าใจผลลัพธ์ที่เกิดขึ้นได้ ถูกต้องและแม่นยำ โดยในการพยากรณ์ราคาหุ้นในแต่ละครั้งนั้นจะต้องให้ระบบทำการเรียนรู้ข้อมูลเท่าที่ผู้ใช้งานต้องการก่อนเพื่อที่จะได้ทำการพยากรณ์ได้ถูกต้องและแม่นยำ เนื่องจากการพยากรณ์ข้อมูลนั้นจะทำได้ก็ต่อเมื่อระบบได้ทำการเรียนรู้แล้ว จากนั้นในการใช้งานเราต้องใส่ตัวแปรอินพุตตามรูปแบบที่ได้กำหนดไว้ ซึ่งทางระบบได้ทำการออกแบบให้มีการแสดงผลลัพธ์ได้ในแถบผลลัพธ์ (จากหน้าจอของโปรแกรม)

เมื่อระบบทำงานเสร็จเรียบร้อยแล้วจะมีการ Save ผลลัพธ์ออกไปเพื่อการใช้งานในรูปแบบ File Excel โดยการกดที่ปุ่ม ส่งข้อมูลเอาท์พุตหลังจากเรียนรู้ไปไฟล์ จากนั้นก็ทำการเลือกที่ๆจะจัดเก็บไฟล์และทำการกด Save

3.4 รายละเอียดหน้าจอของระบบ

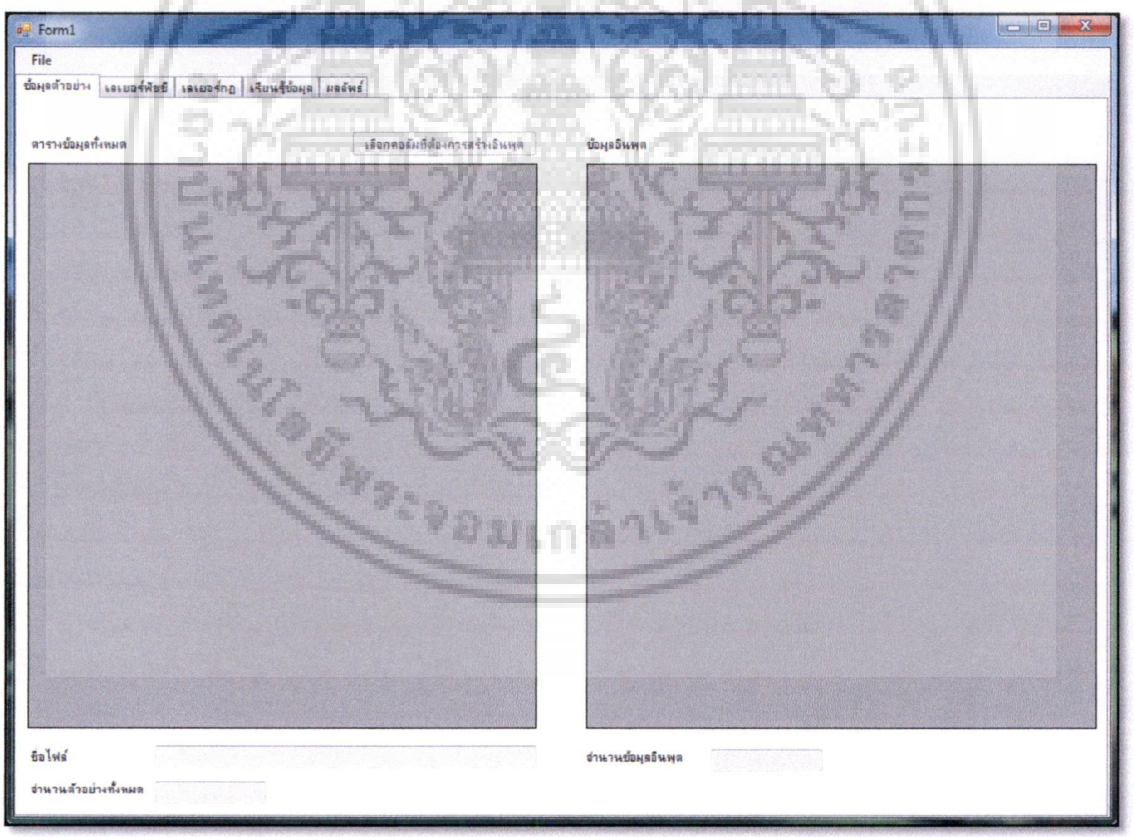
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าจอการทำงานของระบบที่พัฒนาขึ้นมีทั้งหมด 5 หน้าจอ ประกอบไปด้วยหน้าจอต่างๆ ดังนี้

1. น่าจอแสดงข้อมูลตัวอย่าง
2. เลเยอร์ฟัซซี
3. เลเยอร์กฎ
4. เรียนรู้ข้อมูล
5. ผลลัพธ์

3.4.1 หน้าจอแสดงข้อมูลตัวอย่าง

ซึ่งหน้าจอแสดงข้อมูลตัวอย่างนี้เป็นหน้าจอที่ใช้งานสำหรับการนำข้อมูลอินพุตเข้ามาในระบบ และรวมถึงการเลือกคอลัมน์ที่จะใช้ในการพยากรณ์และจำนวนของอินพุตที่เราต้องการที่จะใช้ในระบบ ซึ่งหน้าจอนี้จะแสดงในรูปที่ 3.17



รูปที่ 3.17 หน้าจอแสดงข้อมูลตัวอย่าง

3.4.2 หน้าจอแสดงเลเยอร์ฟัซซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

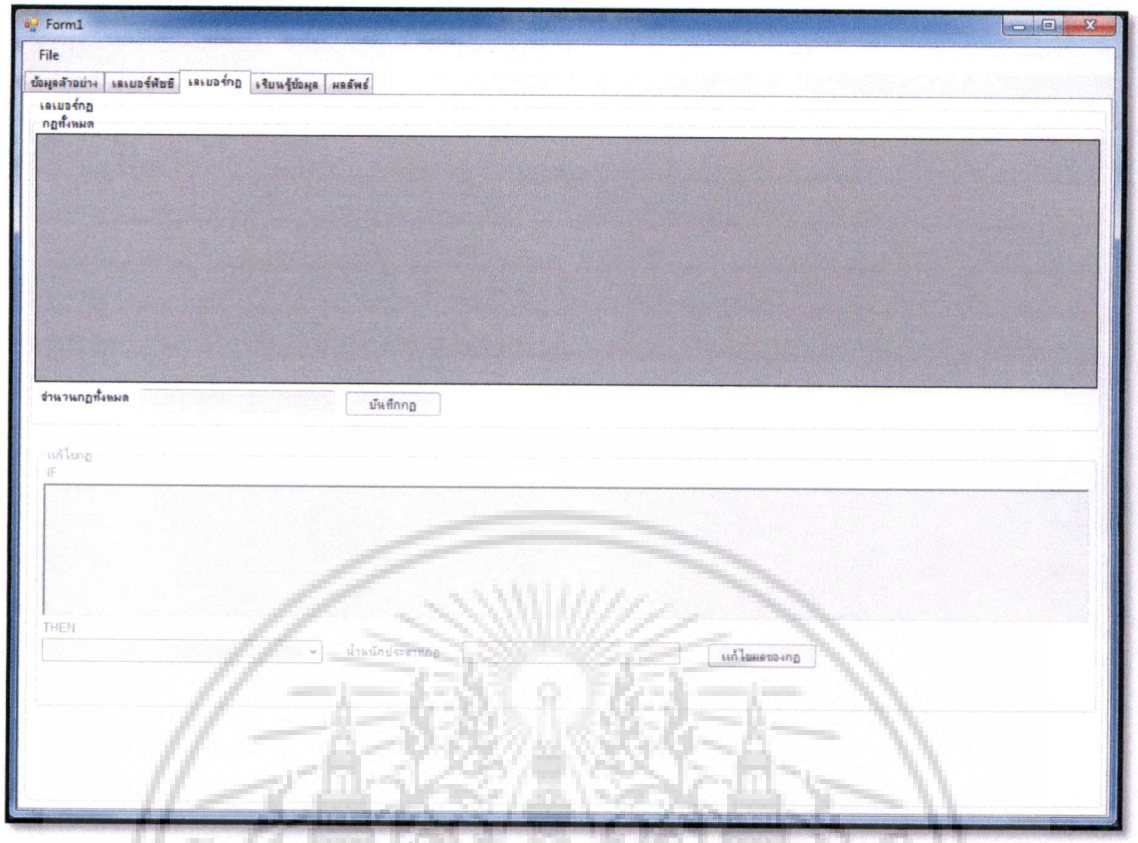
ในหน้าจอนี้จะบอกถึงจำนวนอินพุตฟัซซี , เอาท์พุตฟัซซี และจำนวนฟังก์ชันสมาชิก ซึ่งในหน้าจอนี้ สามารถแก้ไขข้อมูลได้ซึ่งการแก้ไขนั้นจะใช้เฉพาะผู้ที่เชี่ยวชาญเท่านั้น โดยปกติแล้วจะทำการสร้างและกำหนดค่าให้แบบอัตโนมัติ ซึ่งจะแสดงดังรูปที่ 3.18

รูปที่ 3.18 หน้าจอแสดงเลเยอร์ฟัซซี

3.4.3 หน้าจอแสดงเลเยอร์กฎ

ในหน้าจอนี้จะแสดงกฎทั้งหมดที่สร้างขึ้นว่ามีอะไรบ้างและยังสามารถแก้ไขได้ เมื่อเราได้กฎทั้งหมดของระบบแล้วก็กดที่ปุ่ม บันทึกกฎ จากนั้นกฎทั้งหมดก็จะถูกบันทึกลงระบบเพื่อทำการเรียนรู้ต่อไป หน้าจอแสดงเลเยอร์กฎดังรูปที่ 3.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



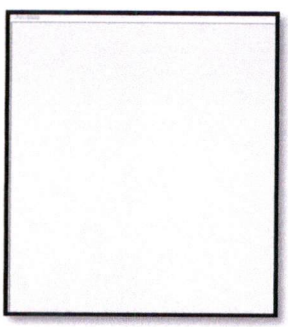
รูปที่ 3.19 หน้าจอแสดงเลเยอร์กฎ

3.4.4 หน้าจอแสดงการเรียนรู้ข้อมูล

ในหน้าจอนี้จะแสดงกระบวนการในการเรียนรู้ของระบบรวมทั้งการกำหนดค่าของพารามิเตอร์ต่างๆให้เหมาะสมด้วย อีกทั้งยังสามารถกฎบันทึกข้อมูลที่เรียนรู้มาเป็น File Excel ได้ ซึ่งหน้าจอนี้จะแบ่งออกเป็น 3 ส่วนหลักๆ คือ

- ส่วนของการเรียนรู้

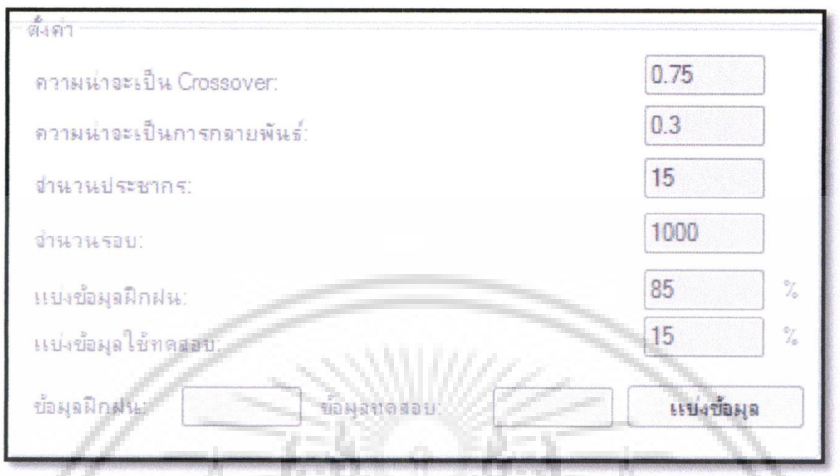
  ในส่วนของการเรียนรู้นั้นจะแสดงถึงค่า Mean square error ของระบบ โดยจะแสดงค่าที่คำนวนออกมาในแต่ละรอบ ซึ่งจำนวนรอบที่แสดงนั้นขึ้นอยู่กับว่าเรากำหนดให้คำนวนกี่รอบ



รูปที่ 3.20 แสดงส่วนของการเรียนรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนของการตั้งค่าพารามิเตอร์
  ในส่วนนี้ใช้สำหรับการกำหนดค่าต่างๆเพื่อที่จะทำให้ระบบสามารถเรียนรู้ข้อมูลได้ถูกต้องและตรงตามความต้องการมากที่สุด



รูปที่ 3.21 แสดงหน้าจอการตั้งค่าพารามิเตอร์

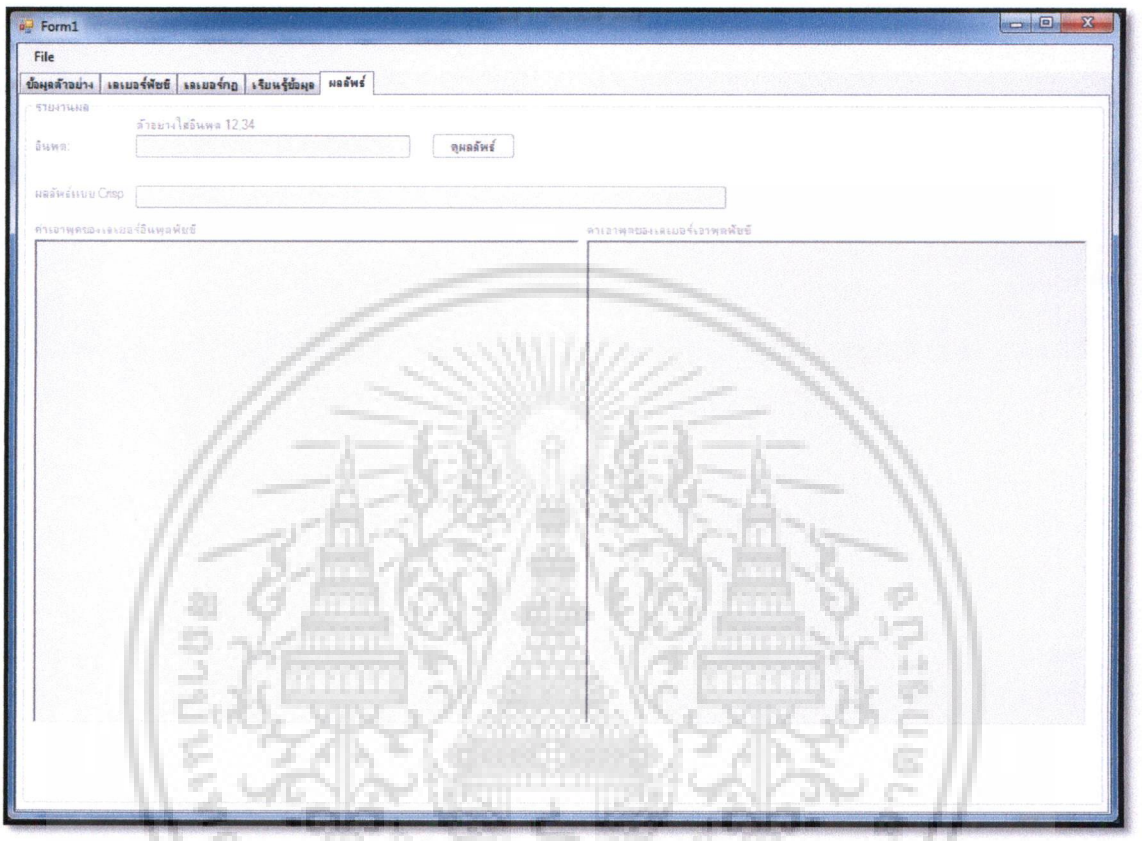
- ส่วนของการแสดงผลลัพธ์ของการเรียนรู้
  เมื่อเราทำการเรียนรู้เสร็จเรียบร้อยแล้วในส่วนนี้จะแสดงผลลัพธ์ทั้งหมดออกมา คือ ค่า Fitness ที่ดีที่สุด , ค่า Fitness ที่ดีที่สุดเฉลี่ย , ค่าน้ำหนัก Fitness ที่ดีที่สุด รวมทั้งปุ่มกดเพื่อบันทึกข้อมูลออกไปเป็น File Excel

รูปที่ 3.22 แสดงหน้าจอผลลัพธ์ของการเรียนรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.5 หน้าจอแสดงผลลัพธ์

ในส่วนของหน้าจอนี้ใช้สำหรับกรอกข้อมูลที่ต้องการจะใช้ในการทำนายแล้วกดให้โปรแกรมแสดงผลลัพธ์ออกมา รวมทั้งแสดงค่าเอาท์พุตต่างๆด้วย ดังที่จะแสดงในรูปที่ 3.23



รูปที่ 3.23 แสดงหน้าจอผลลัพธ์

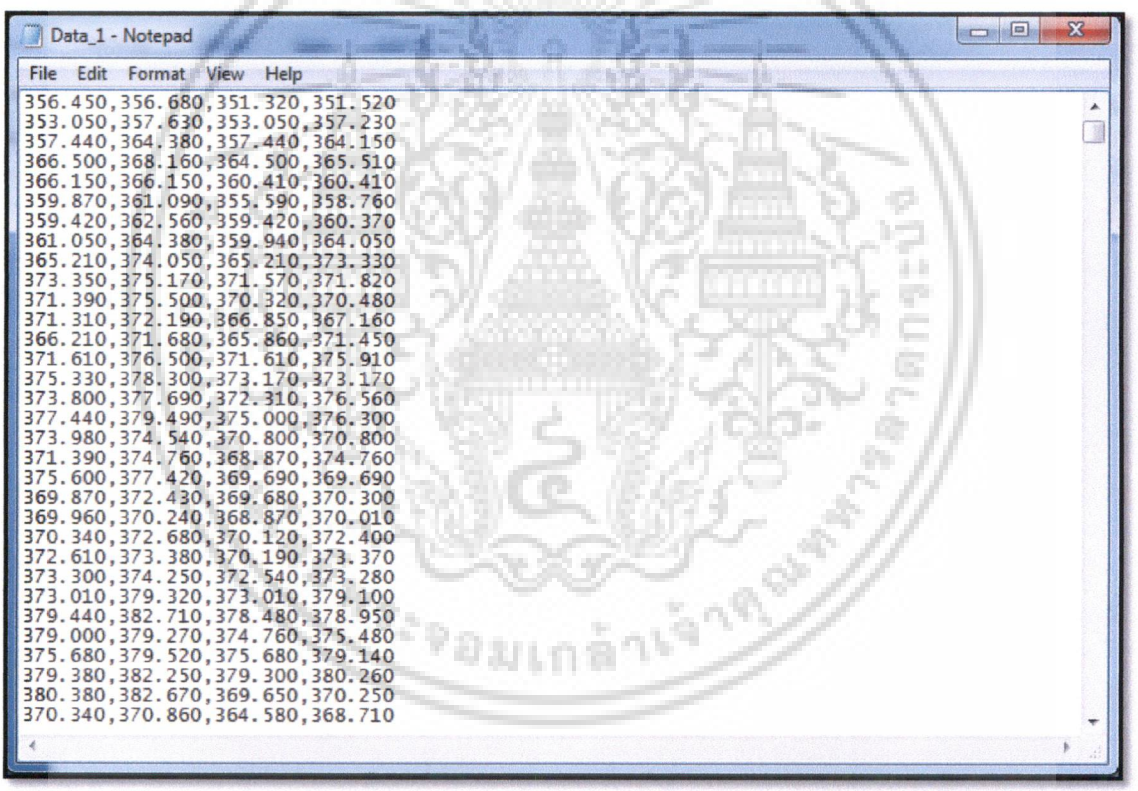
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 ข้อมูลที่ใช้ในการเรียนรู้และการพยากรณ์หุ้น

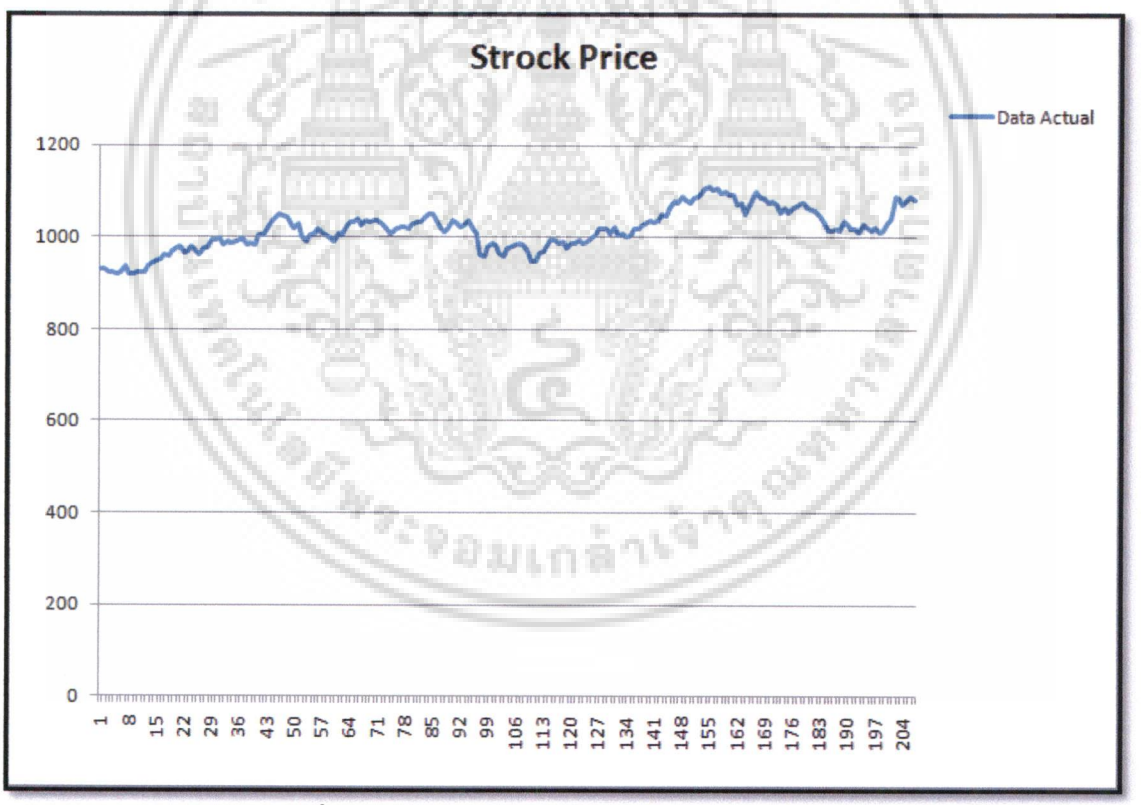
สำหรับข้อมูลที่ให้ในการพยากรณ์หุ้นในครั้งนี้เป็นข้อมูลราคาหุ้นในตลาดโดยรวม แบ่งออกเป็น 4 คอลัมน์ คือ ราคาเปิด , ราคาสูงสุด , ราคาต่ำสุด และราคาปิด ตามลำดับ โดยได้เลือกใช้ราคาปิดการซื้อขายของแต่ละวันมาใช้ในการพยากรณ์ ซึ่งข้อมูลนั้นเริ่มตั้งแต่ปี พ.ศ. 2545 จนถึงปี พ.ศ. 2554 จากนั้นได้ทำการแบ่งข้อมูลออกเป็นข้อมูลที่ใช้ในการฝึกสอนและข้อมูลสำหรับทำการทดสอบ ซึ่งวิธีการแบ่งข้อมูลนั้นแบ่งเป็นเปอร์เซนต์ ซึ่งทางผู้ใช้งานสามารถกำหนดเองได้



รูปที่ 4.1 แสดงหน้าจอข้อมูลที่จะใช้ในการพยากรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.2 กราฟแสดงราคาหุ้นที่ใช้ในการฝึกสอน



รูปที่ 4.3 กราฟแสดงราคาหุ้นที่ใช้ในการพยากรณ์

การแบ่งข้อมูลนั้นในการทดลองนี้กำหนดให้แบ่งข้อมูลเป็น ข้อมูลที่ใช้สำหรับการฝึกสอน 90% และ ข้อมูลทดสอบอีก 10% ดังที่แสดงในกราฟที่ 4.2 และ 4.3 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การกำหนดค่าเริ่มต้นสำหรับการฝึกสอน

สำหรับการพยากรณ์การทดลองนี้ได้ทำการกำหนดค่าของพารามิเตอร์ต่างๆที่ใช้สำหรับการเรียนรู้เพื่อการพยากรณ์ ดังนี้

1. ได้เลือกราคาปิดของตลาดหุ้น (คอลัมน์ที่ 3) มาใช้ในการเรียนรู้โครงข่ายประสาทเทียม
2. ได้กำหนดจำนวนโหนดที่จะใช้ในโครงข่าย ของอินพุตจำนวน 4 โหนด
3. ตั้งค่าของพารามิเตอร์เพื่อเริ่มต้นการเรียนรู้ ซึ่งในการทดลองครั้งนี้ได้กำหนดค่าต่างๆ ดังนี้

   3.1 ความน่าจะเป็นของการ Crossover = 0.75

   3.2 ความน่าจะเป็นการกลายพันธ์ = 0.3

   3.3 จำนวนประชากร = 15

   3.4 จำนวนรอบของการเรียนรู้ = 1000

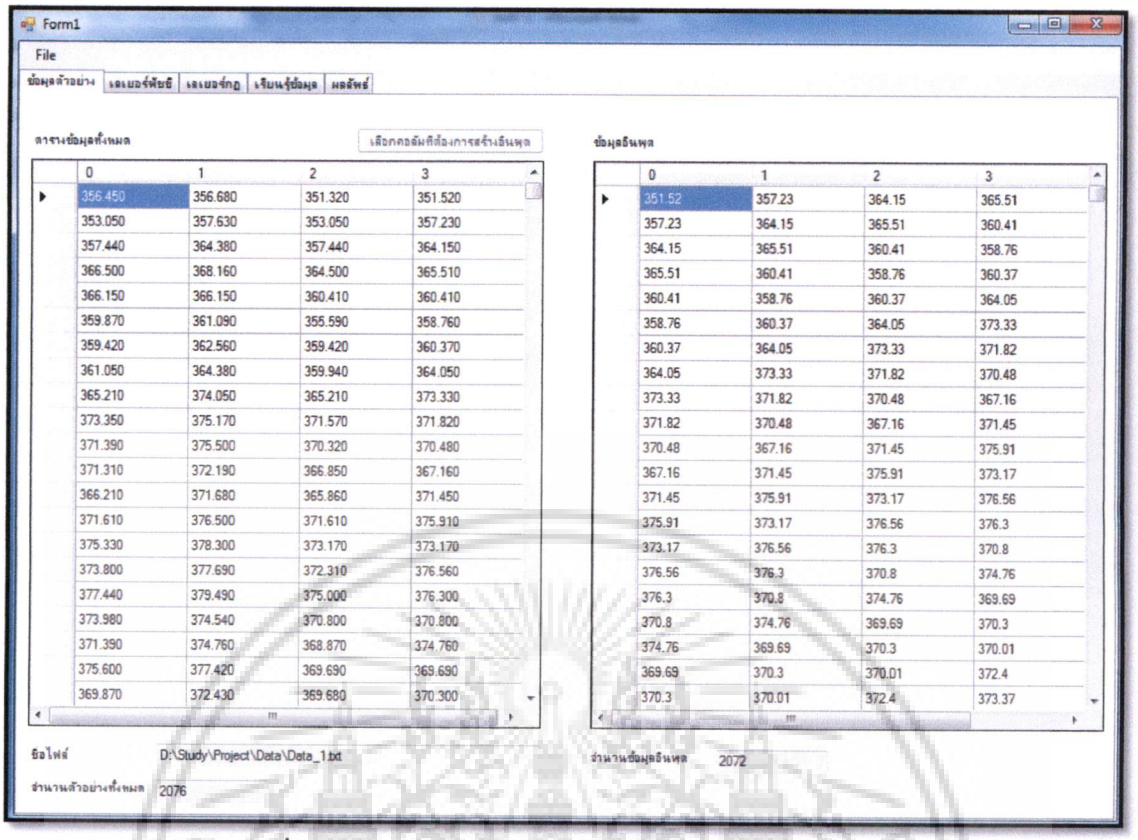
   3.5 การแบ่งข้อมูลฝึกสอน = 90% ข้อมูลทดสอบ = 10%

ส่วนค่าอื่นๆที่กำหนดนั้นได้ทำการสร้างโดยอัตโนมัติอยู่แล้ว เนื่องจากทางผู้พัฒนาโปรแกรมเห็นว่าเป็นค่าที่เหมาะสมแล้ว ซึ่งค่าเริ่มต้นต่างๆ (ข้อที่ 1-3) จะมีค่าเริ่มต้นให้อยู่แล้ว แต่ทางผู้ใช้งานสามารถปรับค่าได้ตามความเหมาะสม

4.3 การแสดงหน้าจอของโปรแกรมสำหรับการเรียนรู้และการพยากรณ์

ในส่วนนี้จะแสดงตั้งแต่การเริ่มการนำข้อมูลเข้าไปในระบบ รวมทั้งการกำหนดค่าต่างๆในการใช้งานตลอดจนการบันทึกผลลัพธ์จากการพยากรณ์ออกเป็น File Excel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

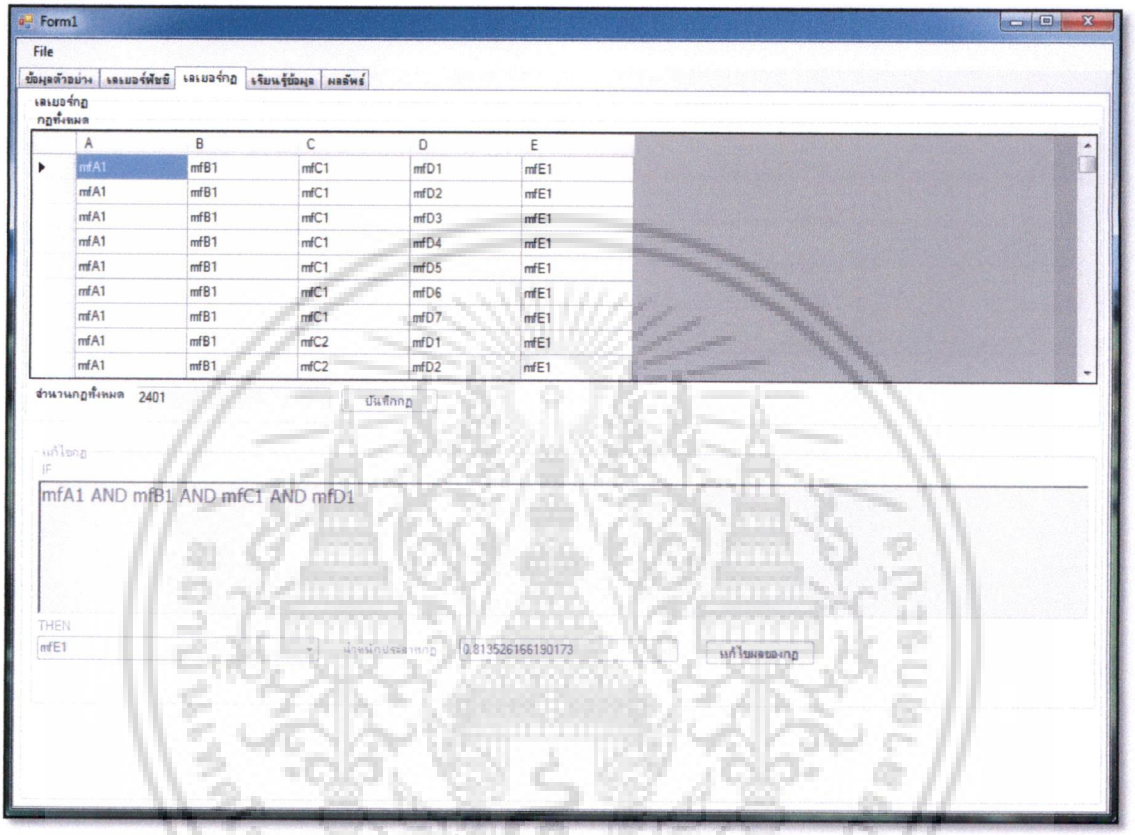


รูปที่ 4.4 แสดงข้อมูลนำเข้าและจำนวนอินพุตของระบบ

รูปที่ 4.5 แสดงเลเยอร์ฟัซซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งหน้าจอนี้จะบอกจำนวนอินพุตทั้งหมดรวมถึงจำนวนของฟังก์ชันสมาชิก ซึ่งสามารถแก้ไขค่าต่างๆได้ เช่น ชื่อฟังก์ชันสมาชิก ซึ่งสามารถที่จะเพิ่มหรือลบออกก็ได้ เป็นต้น ซึ่งการแก้ไขค่าพารามิเตอร์ต่างๆนั้นออกแบบมาสำหรับผู้เชี่ยวชาญใช้ เพื่อให้สามารถปรับค่าของพารามิเตอร์ต่างให้มีความเหมาะสมกับข้อมูลที่นำเข้ามา

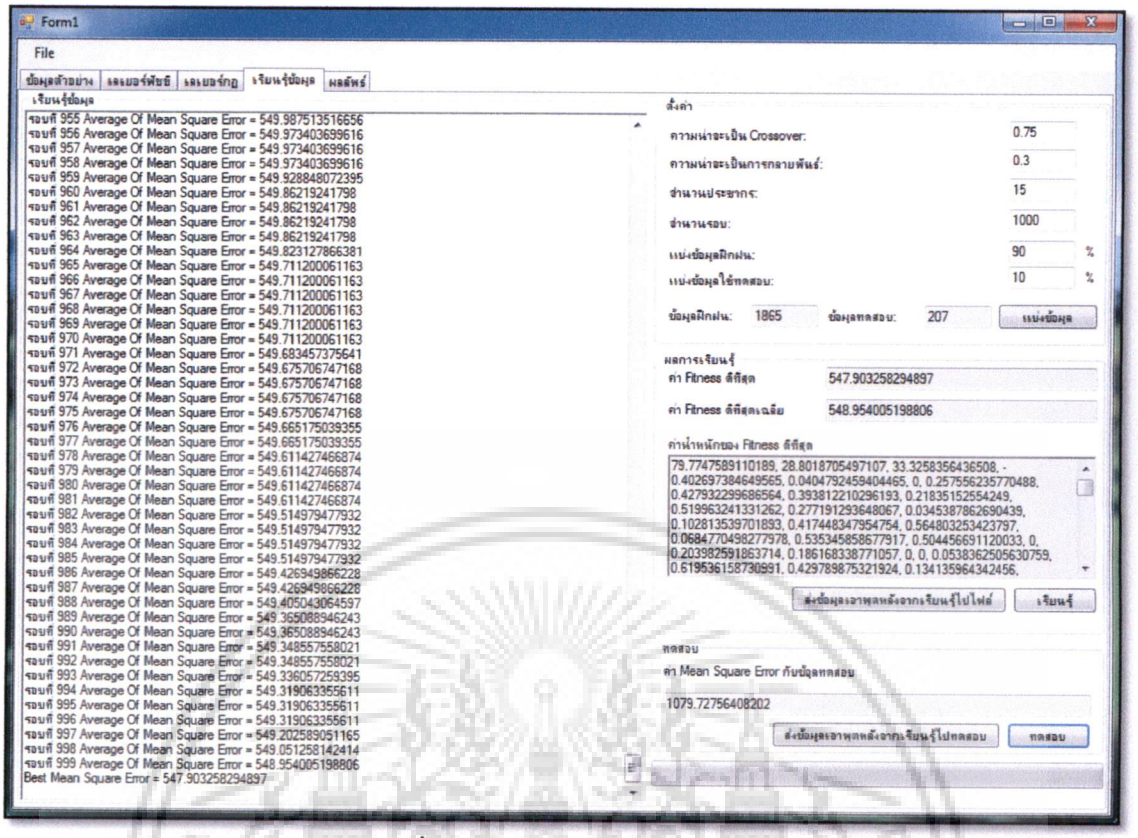


รูปที่ 4.6 แสดงเลเยอร์กฎ

ในหน้าจอนี้จะแสดงจำนวนของกฎที่สามารถเป็นไปได้ทั้งหมด จากรูปจะเห็นได้ว่ากฎทั้งหมดที่สร้างขึ้นมานั้นมีทั้งหมด 625 กฎ ที่สามารถเป็นไปได้ ซึ่งจะสามารถทำการแก้ไขกฎของแต่ละตัวได้และได้ทำการตั้งค่าเริ่มต้นของน้ำหนักประสาทของแต่ละกฎไว้ที่ 1 เพื่อเมื่อเวลาเกิดการเรียนรู้นั้นจะได้เรียนรู้จากทุกๆกฎที่มีจากนั้นกด “บันทึกกฎ” จะเข้าสู่หน้าการเรียนรู้ข้อมูล

จากนั้นจะเป็นขั้นตอนของการเรียนรู้ข้อมูลซึ่งจะมีการกำหนดค่าพารามิเตอร์ต่างๆให้ถูกต้องและเหมาะสมดังรูปที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงการเรียนรู้ข้อมูล

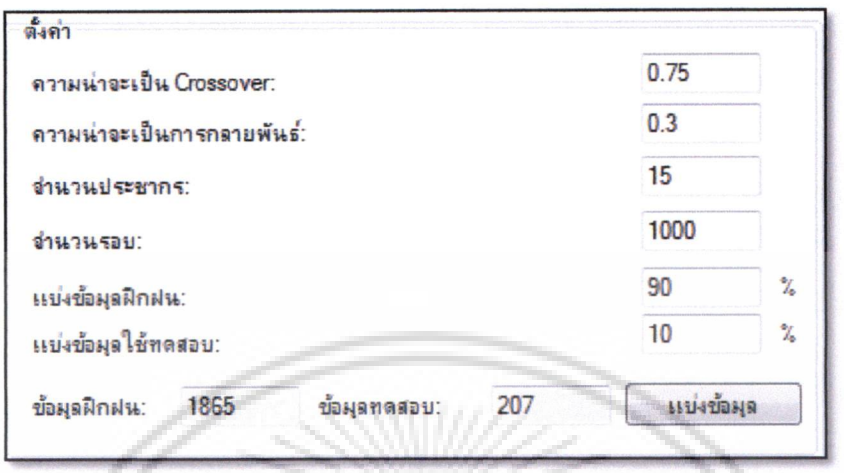
จากรูปที่ 4.7 จะแบ่งออกเป็น 3 ส่วนใหญ่ๆ ดังนี้

1. ส่วนการแสดงผลค่า mean square error
2. ส่วนของการตั้งค่าการเรียนรู้
3. ส่วนการแสดงผลลัพธ์จากการเรียนรู้

รูปที่ 4.8 แสดงส่วนของค่า Mean square error

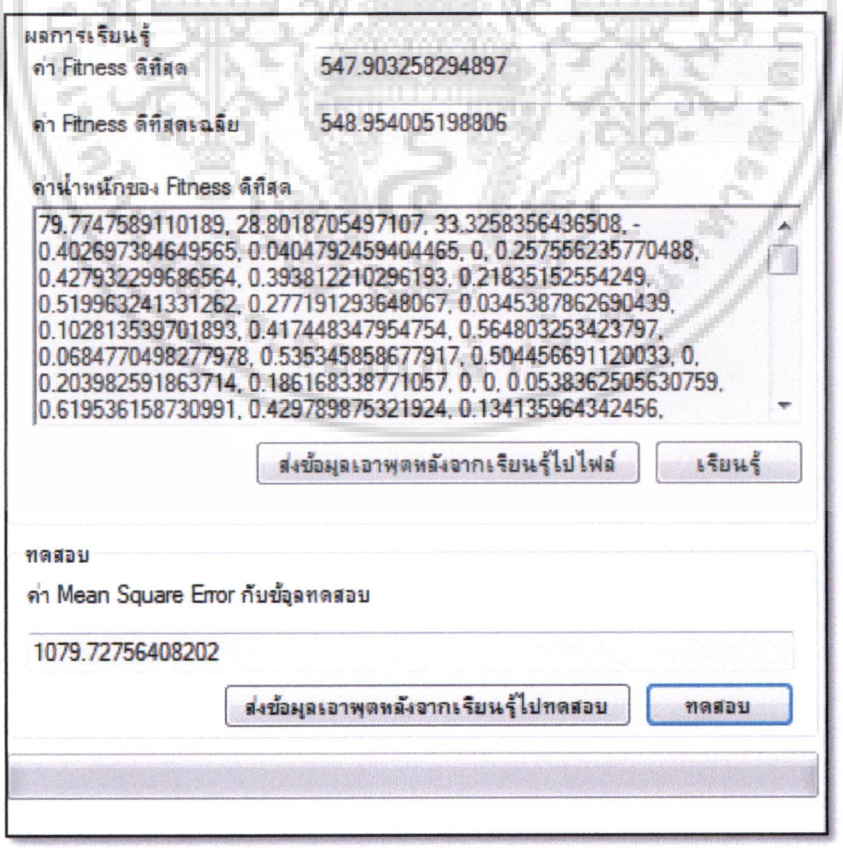
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.8 แสดงค่าเฉลี่ย mean square error ซึ่งค่าที่จะแสดงออกมานั้นจะออกมาเท่ากับจำนวนรอบของโปรแกรมที่ได้ทำการตั้งค่าไว้



รูปที่ 4.9 แสดงส่วนของการตั้งค่าการเรียนรู้

เริ่มต้นระบบนั้นจะตั้งค่ามาอยู่แล้วแต่ทางผู้ใช้งานสามารถแก้ไขค่าต่างๆ ได้ตามความต้องการเพื่อความเหมาะสมกับข้อมูลที่ใช้งาน



รูปที่ 4.10 แสดงส่วนผลลัพธ์จากการเรียนรู้

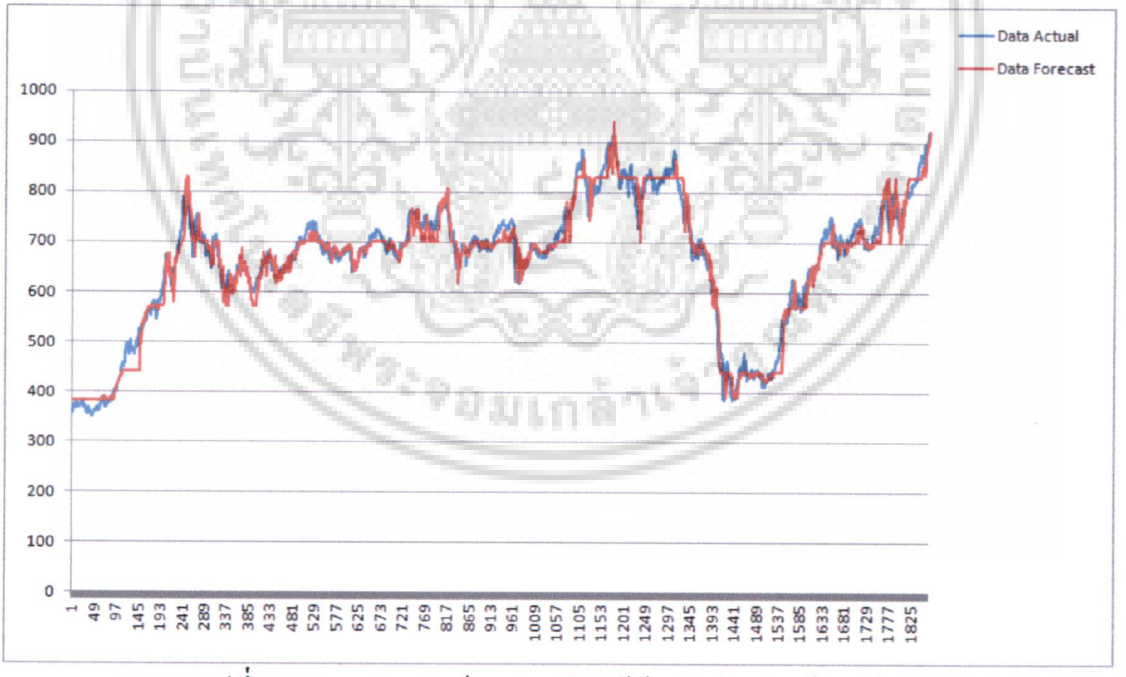
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ผลการทดสอบระบบ

หลังจากที่โปรแกรมได้ทำการเรียนรู้ตามเงื่อนไขที่ได้กำหนดไว้ตอนแรกแล้วนั้น โปรแกรมสามารถบันทึกข้อมูลออกไปเป็น Excel File ได้ โดยเลือกที่ "ส่งข้อมูลเอาพุตหลังจากเรียนรู้ไปไฟล์" ซึ่งจะสามารถดูเป็นการฟของ Excel File ได้ รวมทั้งยังสามารถดูของไฟล์ที่ใช้ทดสอบได้อีกด้วยโดยกดที่ "ส่งข้อมูลเอาพุตหลังจากเรียนรู้ไปทดสอบ" จากนั้นถ้าต้องการบันทึกรูปแบบของการเรียนรู้นี้เพื่อที่ในครั้งต่อๆไปจะได้ไม่ต้องเรียนรู้อีกสามารถทดสอบได้เลย ก็เลือกที่ File -> ส่งออกข้อมูลออกไปยังไฟล์ผลลัพธ์ File ที่ได้ออกไปจะกำหนดให้เป็น File.osl ซึ่งผลลัพธ์ที่ได้จากการทดลองซึ่งกำหนดให้มีดังนี้

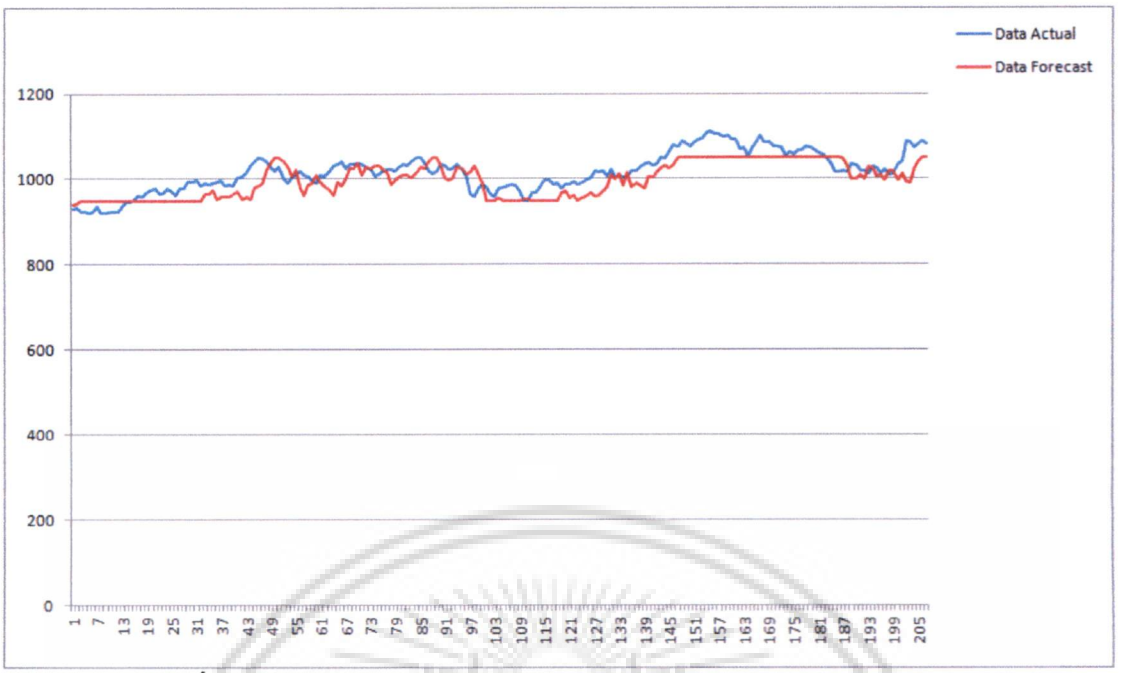
- 4 อินพุต
- กำหนดแบบสี่เหลี่ยมคางหมู
- 7 ฟังก์ชันสมาชิก
- เรียนรู้ 1000 รอบ
- ส่วนที่เหลือตั้งค่าเป็น Default

จะได้ผลลัพธ์ออกมาดังนี้



รูปที่ 4.11 แสดงผลลัพธ์ของการเรียนรู้ข้อมูลเทียบกับข้อมูลจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 แสดงผลลัพธ์ของการเรียนรู้ข้อมูลทดสอบเทียบกับข้อมูลจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

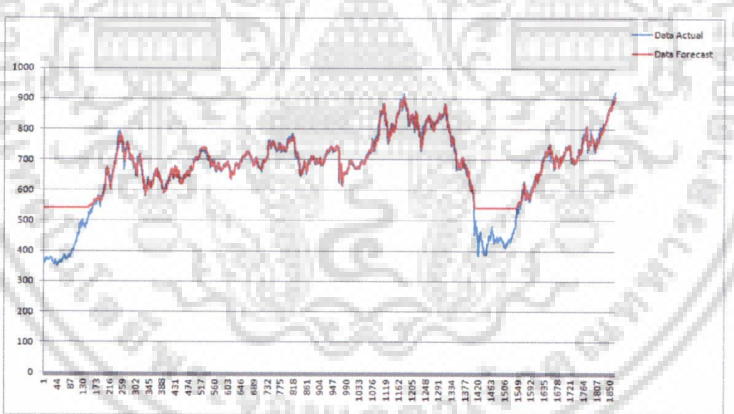
บทที่ 5
สรุปและข้อเสนอแนะ

5.1 เปรียบเทียบผลการทดลอง

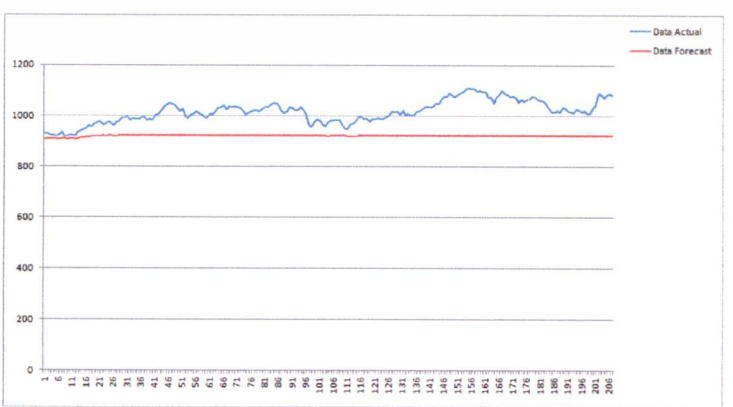
สำหรับผลลัพธ์ของผลการทดลองที่จะใช้ในการเปรียบเทียบนั้นทางผู้จัดทำได้แบ่งออกเป็นแบบต่างๆเพื่อแสดงให้เห็นถึงความชัดเจน โดยได้แบ่งออกเป็นการเปรียบเทียบจากจำนวนอินพุตของแต่ละแบบ (สามเหลี่ยม, สี่เหลี่ยมคางหมู) และการเปรียบเทียบโดยเทียบจากจำนวนฟังก์ชันสมาชิก ซึ่งกำหนดให้ฟังก์ชันสมาชิกที่ใช้คือ 3, 5 และ 7 ตามลำดับ ซึ่งกระบวนการของการเปรียบเทียบนั้นจะแสดงได้ดังนี้

5.1.1 เปรียบเทียบโดยใช้จำนวนอินพุตและจำนวนฟังก์ชันสมาชิกเท่ากัน

การเปรียบเทียบนี้จะใช้หลักการจำนวนอินพุตและจำนวนฟังก์ชันสมาชิกเท่ากัน เทียบกันในแต่ละแบบ

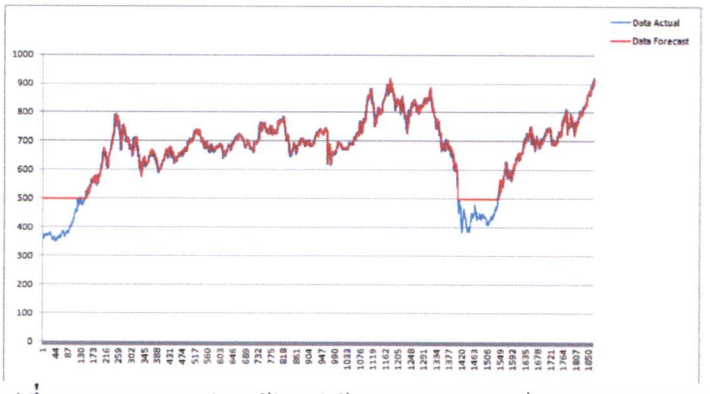


รูปที่ 5.1 แสดงการเรียนรู้โดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม

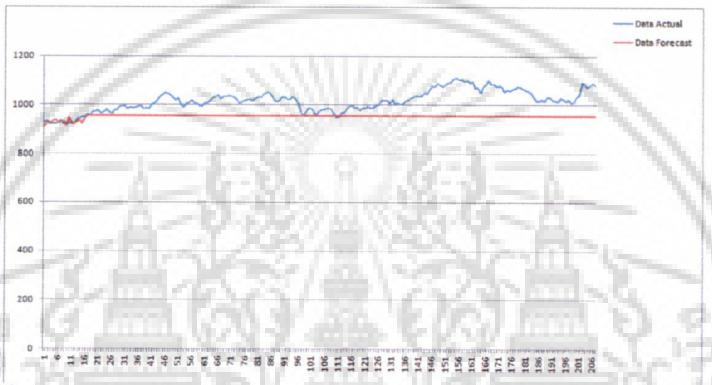


รูปที่ 5.2 แสดงผลการทดสอบโดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม

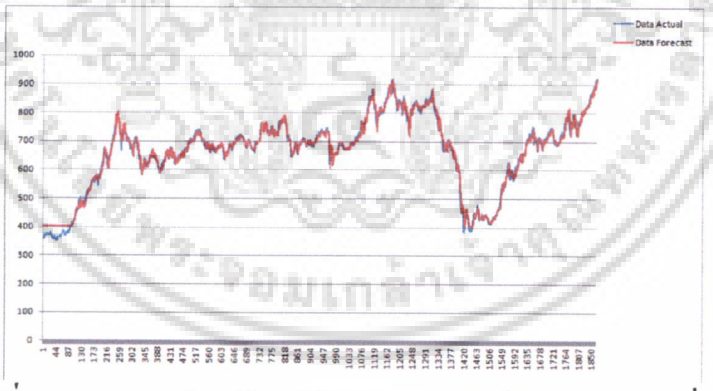
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



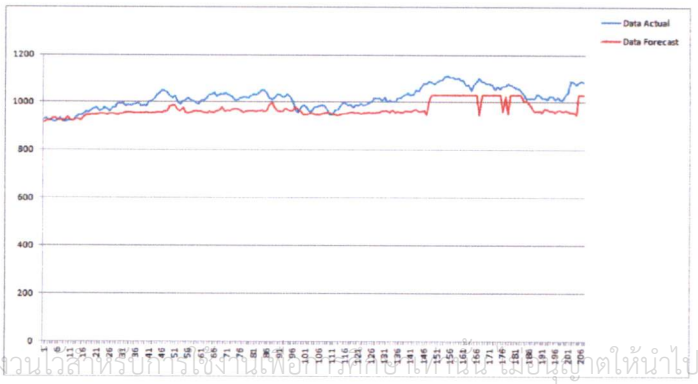
รูปที่ 5.3 แสดงการเรียนรู้โดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบคางหมู



รูปที่ 5.4 แสดงผลการทดสอบโดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบคางหมู



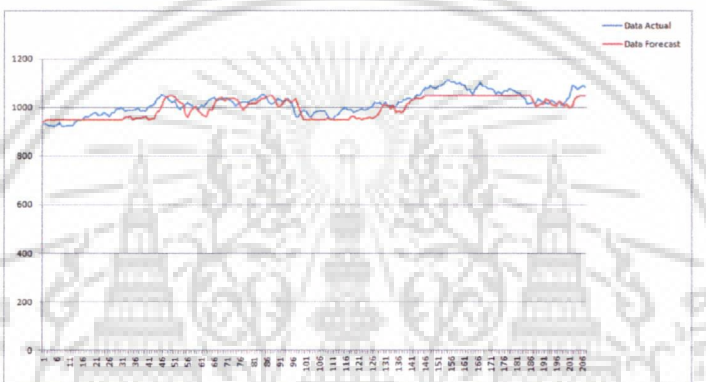
รูปที่ 5.5 แสดงการเรียนรู้โดยใช้ 3 อินพุต 7 ฟังก์ชัน แบบสามเหลี่ยม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.6 แสดงผลการทดสอบโดยใช้ 3 อินพุต 7 ฟังก์ชัน แบบสามเหลี่ยม

รูปที่ 5.7 แสดงการเรียนรู้โดยใช้ 3 อินพุต 7 ฟังก์ชัน แบบคางหมู



รูปที่ 5.8 แสดงผลการทดสอบโดยใช้ 3 อินพุต 7 ฟังก์ชัน แบบคางหมู

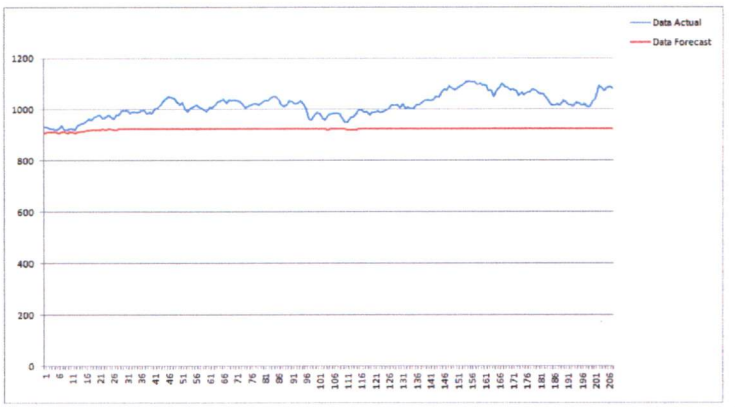
จากการเปรียบเทียบข้างต้นจะแสดงให้เห็นว่าถ้าใช้จำนวนอินพุตและจำนวนฟังก์ชันสมาชิกเท่ากันแล้ว แบบสี่เหลี่ยมคางหมูจะได้ผลออกมาดีกว่าแบบสามเหลี่ยม

5.1.2 เปรียบเทียบโดยใช้จำนวนอินพุตต่างกันฟังก์ชันสมาชิกเท่ากัน

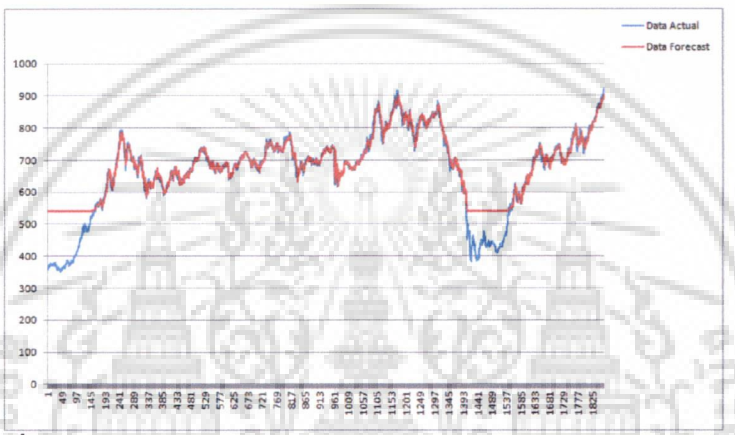
การเปรียบเทียบนี้เพื่อที่จะดูว่าจำนวนอินพุตมีผลต่อการเรียนรู้ของระบบหรือไม่ ทำการเปรียบเทียบทั้งสามเหลี่ยมและสี่เหลี่ยมคางหมู

รูปที่ 5.9 แสดงการเรียนรู้โดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม

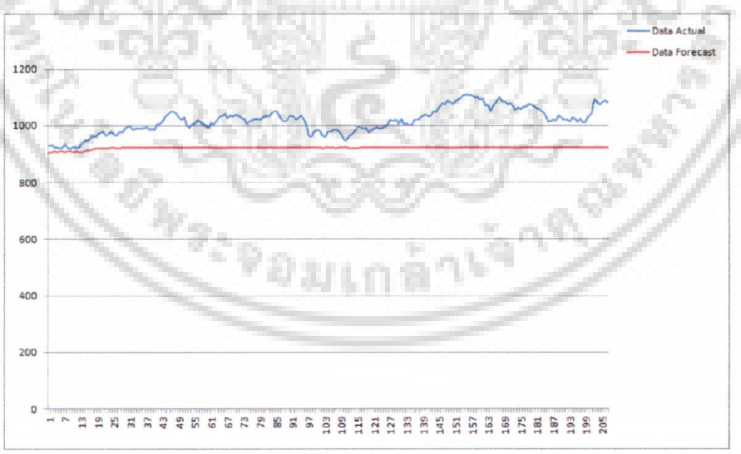
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.10 แสดงผลการทดสอบโดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม



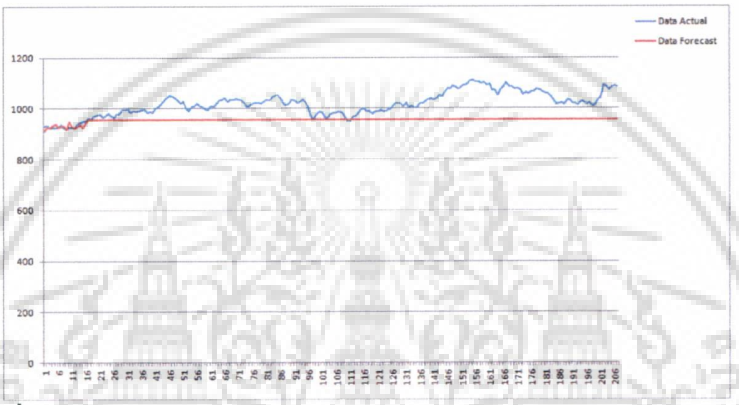
รูปที่ 5.11 แสดงการเรียนรู้โดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม



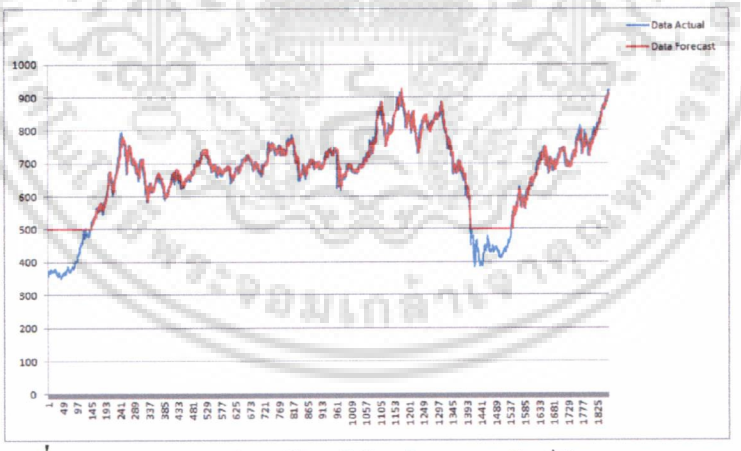
รูปที่ 5.12 แสดงผลการทดสอบโดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.13 แสดงการเรียนรู้โดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบคางหมู

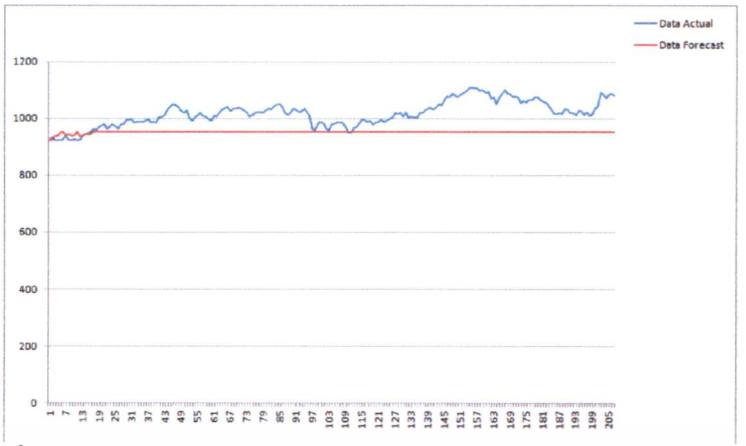


รูปที่ 5.14 แสดงผลการทดสอบโดยใช้ 3 อินพุต 3 ฟังก์ชัน แบบคางหมู



รูปที่ 5.15 แสดงการเรียนรู้โดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบคางหมู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

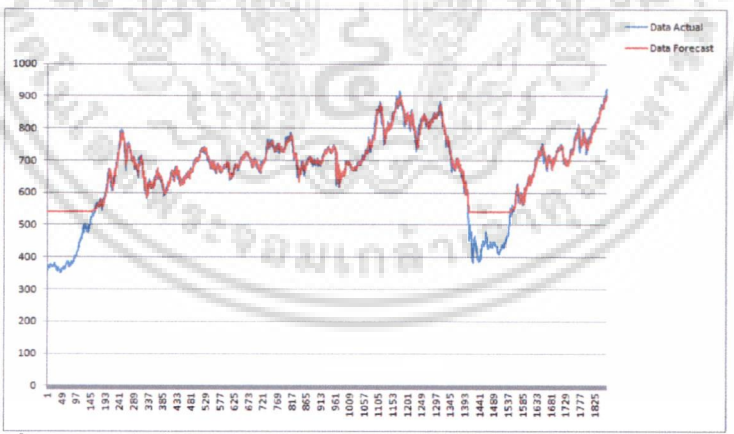


รูปที่ 5.16 แสดงผลการทดสอบโดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบคางหมู

จากการเปรียบเทียบจำนวนอินพุตต่างกันแต่จำนวนฟังก์ชันสมาชิกเท่ากันนั้น จะเห็นได้ว่าการกำหนดค่าของจำนวนอินพุตที่ต่างกันนั้นมีผลต่อระบบน้อยมาก ซึ่งถ้าสังเกตุจากรูปนั้นเกือบจะเหมือนกัน แต่ถ้าดูในโปรแกรมนั้นจะเห็นว่าถ้าเลือกอินพุตมากขึ้นค่า Fitness จะลดลง (ในโปรแกรมมีการบอกค่า Fitness ไว้)

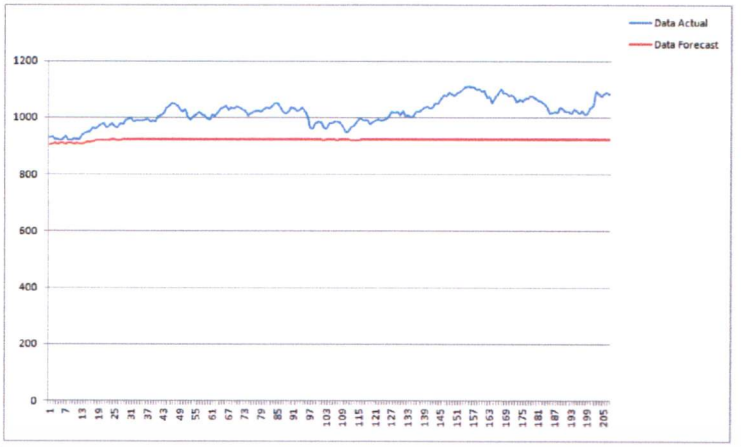
5.1.3 เปรียบเทียบโดยใช้จำนวนอินพุตเท่ากันฟังก์ชันสมาชิกต่างกัน

การเปรียบเทียบนี้เพื่อจะดูว่าจำนวนฟังก์ชันสมาชิกมีผลต่อการเรียนรู้หรือไม่

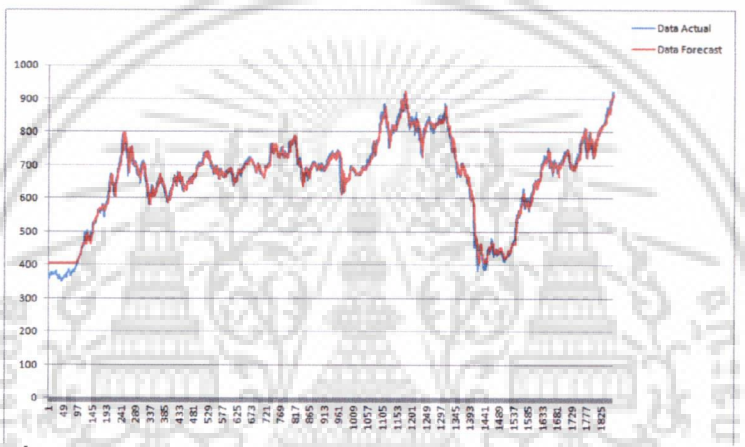


รูปที่ 5.17 แสดงการเรียนรู้โดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม

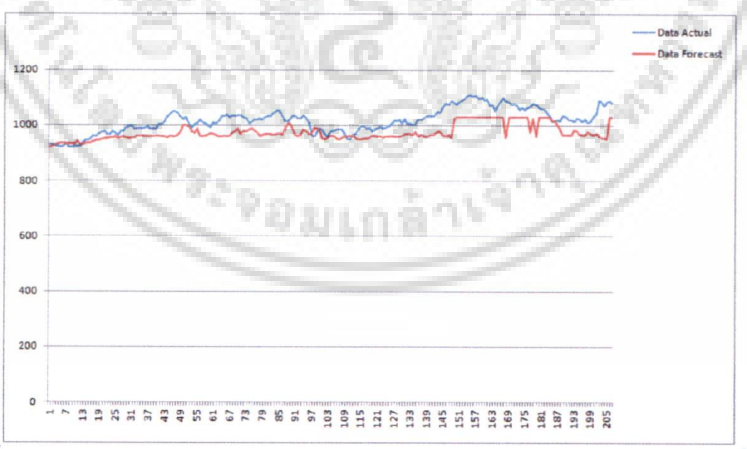
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.18 แสดงผลการทดสอบโดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบสามเหลี่ยม



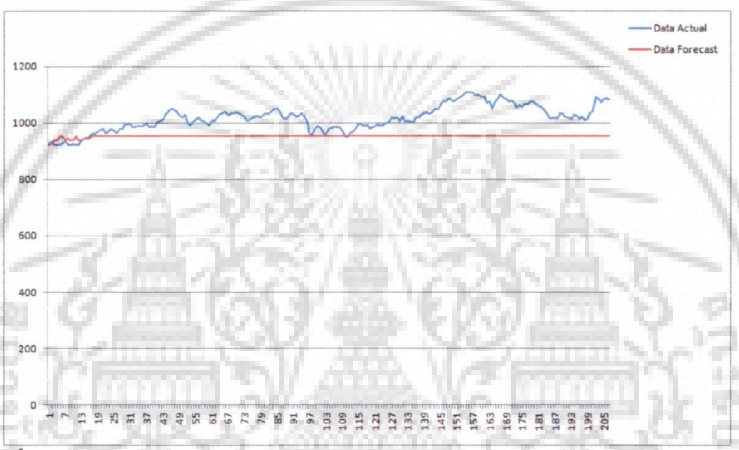
รูปที่ 5.19 แสดงการเรียนรู้โดยใช้ 4 อินพุต 7 ฟังก์ชัน แบบสามเหลี่ยม



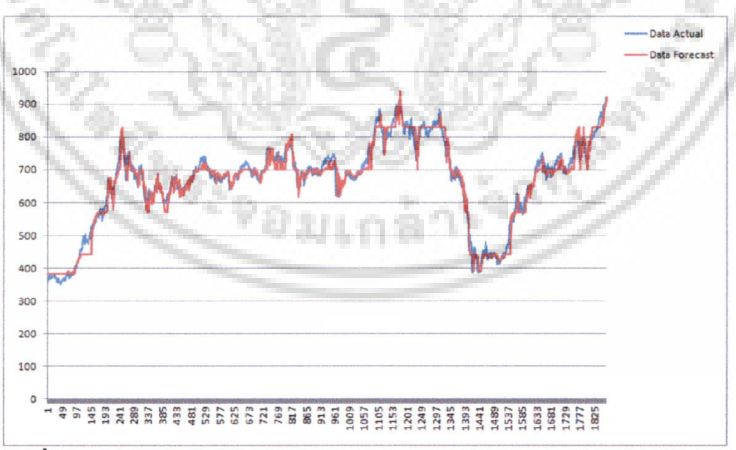
รูปที่ 5.20 แสดงผลการทดสอบโดยใช้ 4 อินพุต 7 ฟังก์ชัน แบบสามเหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.21 แสดงการเรียนรู้โดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบคางหมู



รูปที่ 5.22 แสดงผลการทดสอบโดยใช้ 4 อินพุต 3 ฟังก์ชัน แบบคางหมู



รูปที่ 5.23 แสดงการเรียนรู้โดยใช้ 4 อินพุต 7 ฟังก์ชัน แบบคางหมู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.24 แสดงผลการทดสอบโดยใช้ 4 อินพุต 7 ฟังก์ชัน แบบดางหมู

จากการเปรียบเทียบว่าถ้าใช้จำนวนอินพุตเท่ากันแต่จำนวนฟังก์ชันสมาชิกต่างกันนั้น จะสังเกตุเห็นได้ว่าถ้ามีจำนวนฟังก์ชันสมาชิกที่มากกว่าจะทำให้การพยากรณ์ได้ดีขึ้น

5.2 สรุป

จากการเปรียบเทียบผลการทดลองในแบบต่างๆทั้งหมด สามารถสรุปได้ว่าถ้าต้องการจะให้ได้ผลการพยากรณ์ที่ดีนั้น ผลที่ออกมาขึ้นอยู่กับค่าอินพุตและค่าฟังก์ชันสมาชิก ซึ่งการปรับค่าอินพุตนั้นจะส่งผลต่อระบบน้อยกว่าการปรับค่าฟังก์ชันสมาชิก ถ้าต้องการได้ผลการพยากรณ์ที่ดีต้องกำหนดค่าฟังก์ชันสมาชิกมากๆ ส่วนค่าอินพุตนั้นใช้ตามความเหมาะสม ก็จะทำให้ได้ผลลัพธ์ออกมาแม่นยำมากขึ้น

5.3 ข้อเสนอแนะ

ในการใช้งานการพยากรณ์หุ้นจริงๆนั้น ต้องเพิ่มฟังก์ชันต่างๆมากกว่านี้ เนื่องจากหุ้นนั้นมีหลายๆปัจจัยที่ทำให้เกิดการเปลี่ยนแปลงของราคาได้ ในส่วนของโปรแกรมต้องมีการปรับปรุงให้มีการเรียนรู้ที่เร็วขึ้น และทางผู้ใช้งานควรจะต้องมีความรู้พื้นฐานเพื่อที่จะใช้ในการปรับค่าต่างๆได้เหมาะสมมากยิ่งขึ้น ซึ่งจะทำให้โปรแกรมมีความแม่นยำในการพยากรณ์มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

อาทิตย์ ศรีแก้ว, ปัญญาเชิงคำนวณ Coputer Intelligence , 65

Huang Fu-yuan , 2008, **Forecasting Stock Price Using a Genetic Fuzzy Neural Network**

Gwo-Ching Liao, Member, IEEE, and Ta-Peng Tsao, June 2006 Application of a FuzzyNeural Network Combined With a Chaos Genetic Algorithm and Simulated Annealing to Short-Term Load Forecasting

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ-นามสกุล นายพิชญา ประทุมศิริ

วัน เดือน ปีเกิด 6 พฤศจิกายน พ.ศ. 2526

ที่อยู่ 80/309 ถ.ประชาอุทิศ อ.เมือง จ.พิษณุโลก 65000

ประวัติการศึกษา 2549 มหาวิทยาลัยนเรศวร คณะวิศวกรรมศาสตร์ สาขาคอมพิวเตอร์

ประสบการณ์การทำงาน

พ.ศ.2549 – 2550 บริษัท ไทยแอร์โรว์

พ.ศ.2553 – 2554 บริษัท แซจิเทท ประเทศไทย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้