ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA

TIME SERIES PREDICTION WITH RECURRENT NEURAL NETWORKS USING A HYBRID PSO-EA ALGORITHM

โดย

ใกล้รุ่ง สันติผลธรรม

KLAIRUNG SANTIPHOLTHAM

อาจารย์ที่ปรึกษา

รศ.ดร.อาริต ธรรมโน

H003322

b11752282
i1292997

วัน เดือน ปี 22 พ.ค. 2550
เลขทะเบียน 03322
เลขเรียกหนังสือ วพ. ก 9ง1ก 2549
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่ 1 ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIME SERIES PREDICTION WITH RECURRENT NEURAL NETWORKS USING A HYBRID PSO-EA ALGORITHM

KLAIRUNG SANTIPHOLTHAM

A SYSTEM DEVELOPMENT PROJECT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

1/ 2006

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2006

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ดัดแปลง ทำซ้ำ หรือจำหน่าย จ่ายแจก หรือใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์ก ร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA
นักศึกษา	นางสาวใกล้รุ่ง สันติผลธรรม
รหัสนักศึกษา	47066413
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขวงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2549
อาจารย์ที่ปรึกษา	รศ.ดร.อาริต ธรรมโน

บทคัดย่อ

การพยากรณ์ข้อมูลมีความสำคัญต่อการดำเนินงานทางธุรกิจ โดยการพัฒนาวิธีการพยากรณ์สำหรับธุรกิจนั้นมีจำนวนมาก สำหรับโครงการนี้จะพัฒนาโปรแกรมแบบจำลองด้วยรูปแบบโครงข่ายประสาทเทียมแบบรีเคอร์เร็นท์นิวรอลเน็ตเวิร์ก โดยอาศัยหลักการของการค้นหาค่าถ่วงน้ำหนักที่ถูกต้องสำหรับค่านำเข้าแต่ละค่า โดยใช้วิธีการทำซ้ำเพื่อแก้ไขค่าถ่วงน้ำหนักจนกว่าจะได้รับค่าความถูกต้องที่น่าพอใจ ซึ่งจะใช้เทคนิคของ Particle Swarm Optimization และ Evolutionary Algorithm มาใช้ร่วมด้วยกับโครงข่ายประสาทเทียม เพื่อช่วยในการค้นหาค่าที่เหมาะสมให้มีประสิทธิภาพดียิ่งขึ้น และเพื่อให้ได้ค่าพยากรณ์ที่มีความถูกต้องมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Time Series Prediction with Recurrent Neural Networks Using A Hybrid PSO-EA Algorithm
Student	Miss Klairung Santipholtham
Student ID.	47066413
Degree	Master of Science
Program	Information Science
Academic Year	2006
Advisor	Assoc. Prof. Dr. Arit thammano

ABSTRACT

Forecasting plays a key role in business policy making. A lot of methods for business forecasting have been developed. In this paper will develop an application program by using Neural Network Model with Recurrent algorithm. By Artificial Neural Network is use a principle of the search for optimum value of weight from data. Particle Swarm Optimization and Evolutionary Algorithm are new way to used for finding the optimize Neural Network weight . The methods for testing to find the correctness of the forecasting.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานนี้สำเร็จลุล่วงได้เป็นอย่างดีด้วยความช่วยเหลือจากบุคคลต่างๆ จึงขอขอบพระคุณบุคคลต่างๆ ดังนี้ ขอบคุณพระคุณบิดา มารดาและครอบครัวของข้าพเจ้าที่เป็นกำลังใจและสนับสนุนในทุกๆ เรื่อง รวมทั้งให้โอกาสในการศึกษาเล่าเรียนอย่างเต็มที่ ขอขอบพระคุณ รศ.ดร.อาริต ธรรมโน เป็นอย่างสูงที่ได้กรุณาให้คำแนะนำ คำปรึกษาจากและแก้ไขในสิ่งที่บกพร่องในการพัฒนาโครงการนี้ รวมทั้งอาจารย์ทุกท่านที่ให้ความรู้ต่างๆ เพื่อนำความรู้มาใช้ในการพัฒนาโครงการ ขอขอบคุณเพื่อนๆ ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา และสุดท้ายนี้ขอขอบคุณทุกท่านที่มอบความหวังดีที่ทำให้สามารถทำโครงการนี้สำเร็จลุล่วงด้วยดี

ใกล้รุ่ง สันติผลธรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VII

สารบัญรูป.....VIII

บทที่ 1 บทนำ.....1

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....4

2.1 ทฤษฎีดาต้าไมนิ่ง.....4

2.1.1 ดาต้าไมนิ่ง.....4

2.1.2 ทำไมต้องมีดาต้าไมนิ่ง.....5

2.1.3 กระบวนการทำงานของดาต้าไมนิ่ง.....5

2.1.4 เทคนิคที่สำคัญของดาต้าไมนิ่ง.....7

2.2 การพยากรณ์.....13

2.2.1 ประเภทของการพยากรณ์.....14

2.2.2 วิธีการพยากรณ์.....15

2.3 การวิเคราะห์อนุกรมเวลา.....17

2.3.1 ส่วนประกอบของอนุกรมเวลา.....18

2.3.2 กระบวนการสร้างรูปแบบอนุกรมเวลา.....21

2.4 การหาค่าเหมาะสมแบบกลุ่มอนุภาค.....21

2.4.1 การทำงานของการหาค่าเหมาะสมแบบกลุ่มอนุภาค.....21

2.4.2 อัลกอริทึมหาค่าความเหมาะสมของกลุ่มอนุภาค.....23

2.5 Evolutionary Algorithm.....24

2.5.1 อัลกอริทึมหาค่าความเหมาะสมของกลุ่มอนุภาค.....24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.5.2 การทำงานร่วมกันของ Particle Swarm Optimization และ Evolution Algorithm 27
2.6 โครงข่ายประสาทเทียม.....28
2.6.1 โครงสร้างของนิวรอลเน็ตเวิร์ก.....30
2.6.2 ประเภทของนิวรอลเน็ตเวิร์ก.....31

บทที่ 3 วิธีการดำเนินการศึกษา.....41
3.1 โครงสร้างการทำงานของระบบ.....41
3.2 ขั้นตอนการทำงานของระบบ.....44
3.2.1 การเลือกข้อมูลสำหรับที่จะนำมาใช้กับระบบ.....44
3.2.2 การกำหนดค่าตัวแปรให้กับระบบ.....45
3.2.3 การฝึกสอนโครงข่ายประสาทเทียม.....46
3.2.4 การทดสอบโครงข่ายประสาทเทียม.....57

บทที่ 4 การทดลองและผลการทดลอง.....61
4.1 การทำงานของโปรแกรม.....61
4.1.1 หน้าที่ 1 : RNN PSO&EA Prediction.....61
4.1.2 หน้าที่ 2 : หน้าจอหลักของโปรแกรม.....62
4.1.3 หน้าที่ 2.1 : หน้าจอหลักของเมนู File.....63
4.1.4 หน้าที่ 2.1.1 : หน้าจอหลักนำเข้าค่าของข้อมูล.....65
4.1.5 หน้าที่ 2.2 : หน้าจอหลักของเมนู Setup.....66
4.1.6 หน้าที่ 2.2.1 : หน้าจอ Normal Mode Training.....67
4.1.7 หน้าที่ 2.3 : หน้าจอหลักของเมนู Run.....68
4.1.8 หน้าที่ 2.3.1 : หน้าจอของเมนู Run Training.....69
4.1.9 หน้าที่ 2.3.2 : หน้าจอของเมนู Run Testing.....71
4.1.10หน้าที่ 2.3.3 : หน้าจอของเมนู Help.....72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

4.2 การทดสอบการใช้งาน....73
4.2.1 การทดลองข้อมูลชุดที่ 1....78
4.2.2 การทดลองข้อมูลชุดที่ 2....86
4.2.3 การทดลองข้อมูลชุดที่ 3....91
4.2.4 การทดลองข้อมูลชุดที่ 4....95

บทที่ 5 สรุปผลและข้อเสนอแนะ....101
5.1 สรุปผลการดำเนินงาน....101
5.2 สรุปผลการทดลอง....102
5.3 ข้อเสนอแนะ....103

บรรณานุกรม....104
ประวัติผู้เขียนโครงงาน....105

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

2.1 แสดงตัวอย่างของข้อมูลนำขาย....8

2.2 ราคากุ้งกุลาดำในอดีต....16

2.3 การเปรียบเทียบระหว่างเซลล์ประสาทของมนุษย์และนิวรอลเน็ตเวิร์ก.... 30

3.1 แสดงข้อมูลตัวอย่างที่เป็นข้อมูลนำเข้า....46

3.2 ค่าเริ่มต้นของค่าถ่วงน้ำหนักที่สมมุติขึ้นมาของพาติเคิล....48

3.3 ผลการคำนวณที่ได้จากการแทนค่าพาติเคิลเข้าไปในโครงข่ายประสาทเทียม....50

3.4 การคำนวณค่าความคลาดเคลื่อนของแต่ละพาติเคิล....50

3.5 การคำนวณค่าความเหมาะสมของแต่ละพาติเคิล....51

3.6 ค่าความเร็วที่ได้จากการคำนวณของแต่ละพาติเคิล....53

3.7 ปรับค่าความเร็วเพื่อให้อยู่ในช่วงความเร็วสูงสุดที่กำหนดไว้....54

3.8 ค่าตำแหน่งใหม่ของแต่ละพาติเคิล....55

3.9 นำข้อมูลตัวอย่างที่เป็นข้อมูลนำเข้ามาทดสอบ....58

3.10 ผลการคำนวณจากการแทนที่ค่าพาติเคิลที่ 1 เข้าไปในโครงข่ายประสาทเทียม....60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 ขั้นตอนการทำดาต้าไมนิ่ง....6

2.2 กระบวนการ Classification....9

2.3 ตัวอย่างของ Decision Tree เพื่อวิเคราะห์โอกาสที่ลูกค้าบ้านเช่าจะซื้อบ้าน....10

2.4 นิวรอลเน็ตเวิร์กเพื่อวิเคราะห์การให้เช่าและซื้อบ้านของลูกค้า....11

2.5 ตัวอย่างการรวมกลุ่ม....12

2.6 ราคากุ้งกุลาดำในปีต่างๆ....16

2.7 แสดงค่าแนวโน้มผลผลิตเคมีภัณฑ์ชนิดหนึ่ง....18

2.8 แสดงยอดขายรายเดือนของห้างสรรพสินค้าแห่งหนึ่ง....19

2.9 แสดงวัฏจักรธุรกิจ....20

2.10 อัลกอริทึมการหาค่าเหมาะสมแบบกลุ่มอนุภาค....24

2.11 การหาคำตอบของปัญหาโดยใช้ Evolutionary Algorithm....25

2.12 โครงสร้าง Evolutionary Algorithm ของประชากรภายในกลุ่มหนึ่ง....26

2.13 การนำอัลกอริทึมผสมแบบ PSO และ EA มาใช้ร่วมกัน....28

2.14 โครงสร้างเซลล์ประสาทของมนุษย์....29

2.15 โครงสร้างพื้นฐานของนิวรอลเน็ตเวิร์ก....30

2.16 โครงสร้างพื้นฐานและขั้นตอนการทำงานของโครงสร้างการส่งสัญญาณไปข้างหน้า....32

2.17 โครงสร้างพื้นฐานและขั้นตอนการทำงานของโครงสร้างการส่งสัญญาณป้อนกลับ....35

2.18 สถาปัตยกรรมของรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กแบบ Elman....36

2.19 Simple recurrent neural network....38

3.1 การคำนวณหาจำนวน Unknown Dimensions จากโครงข่ายประสาทเทียม....42

3.2 ขั้นตอนการทำงานของอัลกอริทึมแบบผสมของ PSO และ EA....43

3.3 การนำค่าตำแหน่งของแต่ละพาติเคิลไปแทนค่าถ่วงน้ำหนักในโครงข่ายประสาทเทียม....48

3.4 ตัวอย่างการแทนค่าความเหมาะสมของพาติเคิลตัวที่ 1 เพื่อทดสอบโครงข่ายประสาทเทียม....57

3.5 การแทนค่าข้อมูลนำเข้า Pattern 1 เพื่อทดสอบโครงข่ายประสาทเทียม....58

3.6 การแทนค่าข้อมูลนำเข้า Pattern 2 เพื่อทดสอบโครงข่ายประสาทเทียม....59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.1 แสดงหน้าที่ 1: RNN PSO&EA Prediction....61
4.2 แสดงหน้าที่ 2 : หน้าจอหลักของโปรแกรม....62
4.3 แสดงหน้าที่ 2.1 : หน้าจอหลักของเมนู File....64
4.4 แสดงหน้าที่ 2.1.1 : หน้าจอหลักนำเข้าค่าของข้อมูล....65
4.5 แสดงหน้าที่ 2.2 : หน้าจอหลักของเมนู Setup....66
4.6 แสดงหน้าที่ 2.2.1 : หน้าจอ Normal Mode Training. 67
4.7 แสดงหน้าที่ 2.3 : หน้าจอหลักของเมนู Run....69
4.8 แสดงหน้าที่ 2.3.1 : หน้าจอของเมนู Run Training....70
4.9 แสดงหน้าที่ 2.3.2 : หน้าจอของเมนู Run Testing....71
4.10 แสดงหน้าที่ 2.3.3 : หน้าจอของเมนู Help....72
4.11 แสดงการปรับตั้งค่าของโปรแกรมเพื่อทำการทดลองกับข้อมูลชุดที่ 1....78
4.12 แสดงการฝึกหัดโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 1....79
4.13 แสดงการทดสอบโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 1....80
4.14 กราฟแสดงผลทดลองโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 1....81
4.15 แสดงการปรับตั้งค่าของโปรแกรมเพื่อทำการทดลองกับข้อมูลชุดที่ 2....86
4.16 แสดงการฝึกหัดโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 2....87
4.17 แสดงการทดสอบโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 2....88
4.18 กราฟแสดงผลทดลองโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 2....89
4.19 แสดงการปรับตั้งค่าของโปรแกรมเพื่อทำการทดลองกับข้อมูลชุดที่ 3....86
4.20 แสดงการฝึกหัดโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 3 87
4.21 แสดงการทดสอบโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 3....88
4.22 กราฟแสดงผลทดลองโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 3....89
4.23แสดงการปรับตั้งค่าของโปรแกรมเพื่อทำการทดลองกับข้อมูลชุดที่ 4....86
4.24 แสดงการฝึกหัดโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 4....87
4.25 แสดงการทดสอบโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 4....88
4.26 กราฟแสดงผลทดลองโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 4....89

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

ในปัจจุบันนี้การพยากรณ์มีบทบาทสำคัญอย่างยิ่งต่อการวางแผนและการตัดสินใจที่เกี่ยวกับการดำเนินงานทางด้านธุรกิจขององค์กรต่างๆ เช่น ธุรกิจธนาคาร สถาบันการเงินต่างๆ ธุรกิจอุตสาหกรรม การเกษตร การศึกษา เป็นต้น ซึ่งการพยากรณ์ที่ใช้ในงานธุรกิจจำนวนมากใช้เทคนิคที่เป็นพื้นฐานทางด้านสถิติ แต่การพยากรณ์แบบนี้มีความซับซ้อน เนื่องจากการพยากรณ์ข้อมูลโดยใช้แบบจำลองทางสถิตินั้นทำได้ค่อนข้างยาก ซึ่งแบบจำลองทางสถิตินั้นจะเป็นแบบจำลองในลักษณะเชิงเส้น ซึ่งแบบจำลองแบบนี้มีข้อจำกัดทางด้านที่ไม่สามารถจัดการกับข้อมูลอินพุตที่มีลักษณะแปรปรวนมากๆ (Noisy signal) ได้ และยังไม่สามารถจัดการกับข้อมูลที่มีค่าเปลี่ยนแปลงอย่างมากในช่วงเวลาที่ไม่แน่นอนได้ ดังนั้นเพื่อแก้ปัญหาของข้อจำกัดของแบบจำลองในลักษณะ เชิงเส้นตามที่ได้กล่าวมาแล้วข้างต้น จึงได้มีการนำแบบจำลองในลักษณะที่ไม่เป็นเชิงเส้น มาประยุกต์ใช้งาน ซึ่งแบบจำลองนี้ก็คือ แบบ จำลองที่ใช้เทคนิคของโครงข่ายประสาทเทียมหรือนิวรอลเน็ตเวิร์ก โดยที่เป็นแบบจำลองสามารถปรับเปลี่ยนโครงสร้างหรือตัวแปรในแบบจำลองให้ใช้กับสภาพงานที่แตกต่างกันไปได้ง่ายกว่าแบบจำลองทางสถิติ นิวรอลเน็ตเวิร์กเป็นแบบจำลองที่มีความสามารถ เนื่องจากเป็นแบบจำลองที่ตัวระบบเครือข่ายสามารถเรียนรู้ได้เองจากข้อมูลตัวอย่าง แต่การนำนิวรอลเน็ตเวิร์กมาแก้ปัญหาในแต่ละแบบต้องมีการออกแบบโครงสร้างขึ้นมาเฉพาะสำหรับงานที่ใช้ จึงเห็นได้ว่าการใช้นิวรอลเน็ตเวิร์กนั้นให้ผลการทำนายที่ดีกว่าการใช้วิธีทางสถิติ ดังนั้นแนวความคิดของการพัฒนาโปรแกรมเพื่อการพยากรณ์จึงเกิดขึ้น เป็นโปรแกรมที่ชื่อว่าการทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์ก ร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA

การทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA เป็นเทคนิคอัลกอริทึมแบบใหม่ที่นำเอาสองอัลกอริทึม คือ Particle Swarm Optimization (PSO) และ Evolutionary Algorithm มาใช้งานร่วมกัน ซึ่งวัตถุประสงค์คือเป็นการนำเอาการวิธีการของการกลายพันธุ์ (Mutation) ที่เป็นหนึ่งในกระบวนการพื้นฐานของ Evolutionary Algorithm นำไปใช้กับ PSO ซึ่งเป็นการเพิ่มความหลากหลายของประชากรและความสามารถในการหลีกเลี่ยงปัญหาจุดที่คล้ายกับจุดที่ต่ำที่สุด (local minima) โดยที่อัลกอริทึมในการเรียนรู้แบบผสมของ PSO กับ EA นั้นปรากฏว่าประสบความสำเร็จในการฝึกเครือข่ายแบบรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กสำหรับการทำนายแบบอนุกรมเวลา ซึ่งการนำเอาหลักการรีเคอร์เร็นท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA นั้นมีข้อดีคือ ทำให้มีความสามารถมากในส่วนของการคัดเลือกประชากรและส่งผลต่อประชากรในรุ่นถัดไป

1.2 วัตถุประสงค์ของโครงงาน

การศึกษางานโครงงานพัฒนาระบบงานการทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA มีวัตถุประสงค์ดังนี้

1. เพื่อศึกษากระบวนการในการเรียนรู้และการพยากรณ์ จากวิธีโครงข่ายประสาทเทียมที่มีโครงสร้างการทำงานแบบมีการป้อนย้อนกลับ (Recurrent Neural Network)

2. เพื่อศึกษาเปรียบเทียบกระบวนการในการเรียนรู้และการพยากรณ์ จากวิธีโครงข่ายประสาทเทียมที่มีโครงสร้างการทำงานแบบมีการป้อนย้อนกลับ (Recurrent Neural Network) และการพยากรณ์อนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA

3. เพื่อศึกษาการค้นหาตัวแปรต้นที่สำคัญที่ส่งผลต่อกระบวนการในการเรียนรู้และการพยากรณ์ด้วยวิธีการพยากรณ์อนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA

4. เพื่อศึกษาข้อดีและข้อเสียของการเรียนรู้และการพยากรณ์จากวิธีของโครงข่ายประสาทเทียมที่มีโครงสร้างการทำงานแบบมีการป้อนย้อนกลับ (Recurrent Neural Network) และวิธีการพยากรณ์อนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA

5. เพื่อศึกษาอัลกอริทึมและสามารถพัฒนาโปรแกรมการใช้งานโครงข่ายประสาทเทียมได้อย่างถูกต้องและมีประสิทธิภาพ

1.3 ขอบเขตการศึกษา

การศึกษาโครงการพัฒนาระบบงานนี้มีขอบเขตของการศึกษา ดังนี้

1. เป็นการพัฒนาโปรแกรมที่ใช้พยากรณ์โดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA เพื่อใช้ในการพยากรณ์ สิ่งที่จะเกิดขึ้นในสถานการณ์ต่างๆ ได้ เพื่อช่วยในการตัดสินใจให้ดำเนินงานได้อย่างมีประสิทธิภาพ

2. เป็นการพยากรณ์แบบอนุกรมเวลา (Time series forecasting)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนและวิธีการดำเนินงาน

การศึกษาโครงการพัฒนาระบบงานนี้มีขั้นตอนและวิธีการดำเนินงาน ดังนี้

1. ศึกษาหลักการและเทคนิคของโครงข่ายประสาทเทียมที่มีโครงสร้างการทำงานแบบมีการป้อนย้อนกลับ (Recurrent Neural Network)

2. ศึกษาหลักการและเทคนิคของ Particle Swarm Optimization (PSO)

3. ศึกษาหลักการและเทคนิคของ Evolutionary Algorithm (EA)

4. ศึกษาหลักการและเทคนิคของการทำงานร่วมกันของโครงข่ายประสาทเทียมแบบมีการป้อนย้อนกลับและอัลกอริทึมที่นำ PSO และ EA มาใช้ร่วมกัน

5. รวบรวมข้อมูลสถิติของข้อมูลที่จะนำมาทดสอบ

6. ออกแบบและพัฒนาระบบ

7. ทดสอบการใช้งานระบบ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ในการศึกษาโครงการพัฒนาระบบงานนี้ ผลที่คาดว่าจะได้รับมีดังนี้

1. เพื่อเข้าใจหลักการและเทคนิคของโครงข่ายประสาทเทียมที่มีโครงสร้างการทำงานแบบมีการป้อนย้อนกลับ (Recurrent Neural Network)

2. เพื่อเข้าใจหลักการและเทคนิคของ Particle Swarm Optimization (PSO)

3. เพื่อเข้าใจลักการและเทคนิคของ Evolutionary Algorithm (EA)

4. เพื่อเข้าใจหลักการและเทคนิคของการเขียนโปรแกรมการทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA โดยใช้ภาษาโปรแกรม Java

5. เพื่อให้เป็นระบบที่ช่วยในการตัดสินใจได้อย่างมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

การพัฒนาระบบสารสนเทศสำหรับการทำนายอนุกรมเวลาโดยใช้หลักการรีเดอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอลิทึมแบบผสมของ Particle Swarm Optimization (PSO) และ Evolutionary Algorithm (EA) เพื่อช่วยในการตัดสินใจให้เป็นไปอย่างมีประสิทธิภาพยิ่งขึ้นนั้น ได้มีการนำเอาทฤษฎีต่างๆ ที่เกี่ยวข้องมาใช้ในการวิเคราะห์และออกแบบดังนี้

2.1 ทฤษฎีดาต้าไมนิ่ง (Data Mining)

2.1.1 ดาต้าไมนิ่ง

ดาต้าไมนิ่งหรือเหมืองข้อมูล หมายถึง กระบวนการที่กระทำกับข้อมูลจำนวนมาก เพื่อค้นหารูปแบบและความสัมพันธ์ที่ซ่อนอยู่ในชุดข้อมูลนั้น เป็นการที่ผู้ใช้ดึงและวิเคราะห์ตรวจสอบข้อมูลอย่างละเอียด ซึ่งผลลัพธ์ที่ได้จากการวิเคราะห์ตรวจสอบนั้นจะมีลักษณะของข้อมูลที่เป็นข้อมูลแบบ Unknown , ข้อมูลแบบ Valid และข้อมูลแบบ Actionable เพื่อนำข้อมูลดังกล่าวไปประยุกต์ใช้ในการประกอบการตัดสินใจในเชิงธุรกิจ ทำให้เข้าใจแนวโน้มและรูปแบบของตลาดงานด้านต่างๆ มากขึ้น ทั้งในด้านการส่งเสริมการขายสินค้าในห้างสรรพสินค้า, ด้านการวิเคราะห์เครดิตลูกค้าในธนาคาร และในด้านอื่นๆ อีกมาก

ข้อมูลแบบ Unknown คือ ข้อมูลที่ถูกใช้จะต้องเป็นข้อมูลผู้ใช้งานไม่รู้มาก่อนและไม่ชัดเจน ไม่สามารถตั้งสมมติฐานล่วงหน้าว่าควรจะเป็นแบบใด ตัวอย่างเช่น เจ้าของห้างสรรพสินค้าแห่งหนึ่งเพิ่งจะค้นพบว่าพฤติกรรมของผู้บริโภคใหม่ที่เป็นพ่อบ้านมักจะซื้อสินค้าเบียร์และผ้าอ้อมในวันศุกร์ตอนเย็น ดังนั้นเป็นสัญญาณให้เจ้าของกิจการควรจะเตรียมสินค้าไว้เพื่อจำหน่าย ซึ่งในขณะเดียวกันห้างสรรพสินค้าคู่แข่งอาจจะไม่รู้เรื่องนี้ก็ได้ แต่ลองสังเกตดูอีกหนึ่งตัวอย่างว่าเจ้าของร้านขายรถยนต์พบว่ารถขนาดใหญ่ราคาแพงมักจะถูกซื้อโดยคนที่สูงอายุ ซึ่งเจ้าของไม่รู้มาก่อน แต่ข้อมูลดังกล่าวไม่เป็นลักษณะ Unknown เพราะสมมติฐานดังกล่าวมีอยู่ เพราะคนที่มีอายุมักจะมีฐานะที่ดีขึ้นเมื่อเทียบคนในวัยที่อายุน้อยกว่า

ข้อมูลแบบ Valid นั้น เมื่อผู้ใช้ได้เริ่มใช้เทคนิคดาต้าไมนิ่งจะค้นพบสิ่งที่น่าสนใจตลอดเวลา แต่ว่าต้องพิจารณาด้วยว่าสิ่งนั้น Valid หรือไม่ เช่น ผู้ใช้มักจะพบว่ามีความสัมพันธ์ของการซื้อของ2 สิ่งเสมอ เมื่อจำนวนความหลากหลายสินค้ามากขึ้น แต่ไม่ได้หมายความว่าจะต้องให้ห้างสรรพสินค้าเก็บสินค้ามากขึ้น เพราะข้อมูลที่ได้อาจเกิดความคลาดเคลื่อน เพราะฉะนั้นจะต้องทำการ Validation และ Checking ความถูกต้องของข้อมูลและวิเคราะห์ความถูกต้องอีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลแบบ Actionable นั้น ข้อมูลจะต้องถูกแปลงออกมาและนำมาตัดสินใจให้เป็นความได้เปรียบเชิงธุรกิจ บางครั้งข้อมูลที่เราค้นพบเป็นสิ่งที่คู่แข่งได้ทำไปเสียแล้วหรือผิดกฎหมาย ซึ่งจะต้องมีวิจารณญาณในการใช้ด้วย บางทีข้อมูลดังกล่าว อาจจะไม่มีประโยชน์อะไร

2.1.2 ทำไมต้องมีดาต้าไมนิ่ง

ข้อมูลที่ถูกเก็บไว้ในฐานข้อมูลหากเก็บไว้เฉยๆ ก็จะไม่เกิดประโยชน์ ดังนั้นจึงต้องมีการสกัดสารสนเทศไปใช้ ซึ่งการสกัดสารสนเทศ หมายถึง การคัดเลือกข้อมูลออกมาใช้งานในส่วนที่ ต้องการ ซึ่งในอดีตเราใช้คนเป็นผู้สืบค้นข้อมูลต่างๆ ในฐานข้อมูลซึ่งผู้สืบค้นจะทำการสร้างเงื่อนไขขึ้นมาตามภูมิปัญญาของผู้สืบค้น ส่วนในปัจจุบันการวิเคราะห์ข้อมูลจากฐานข้อมูลเดียวอาจไม่ให้ความรู้เพียงพอและลึกซึ้งสำหรับการดำเนินงานภายใต้ภาวะที่มีการแข่งขันสูงและมีการเปลี่ยนแปลงที่รวดเร็วจึงจำเป็นที่จะต้องรวบรวมฐานข้อมูลหลายๆ ฐานข้อมูลเข้าด้วยกัน เรียกว่า “ คลังข้อมูล” (Data Warehouse)

ดังนั้นเราจึงจำเป็นต้องใช้ดาต้าไมนิ่งในการดึงข้อมูลจากฐานข้อมูลที่มีขนาดใหญ่ เพื่อที่จะนำข้อมูลนั้นมาใช้งานให้เกิดประโยชน์สูงที่สุด

2.1.3 กระบวนการทำงานของดาต้าไมนิ่ง

การทำดาต้าไมนิ่งประกอบด้วยขั้นตอนหลัก ดังแสดงในรูปที่ 2.1 ขั้นตอนมีดังต่อไปนี้

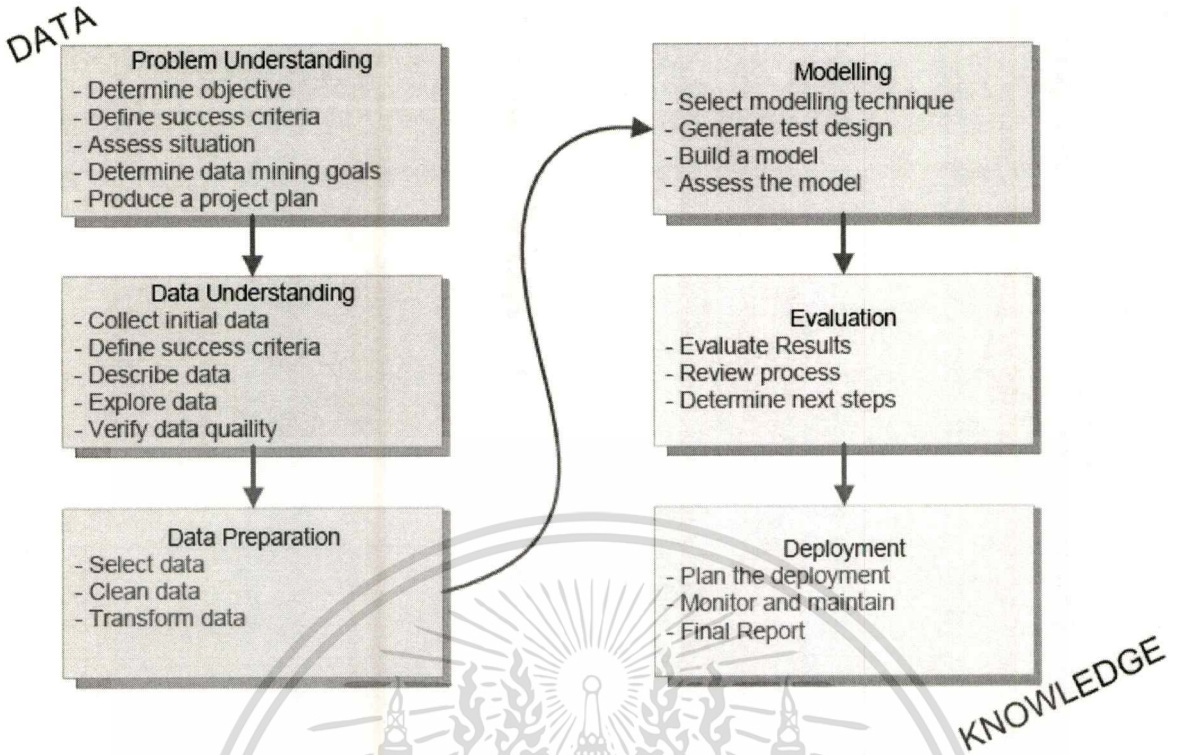
1. การทำความเข้าใจปัญหา ประกอบด้วยกระบวนการย่อยดังนี้

- ตั้งเป้าหมายว่าการทำดาต้าไมนิ่งครั้งนี้ต้องการที่จะแก้ปัญหาใด เช่น ทำดาต้าไมนิ่งเพื่อต้องการเพิ่มยอดขาดสินค้า เป็นต้น

- ตั้งเกณฑ์วัดความสำเร็จในการทำดาต้าไมนิ่ง ซึ่งอาจเป็นได้ทั้งความสำเร็จในด้านรูปธรรม เช่น สามารถเพิ่มยอดขายสินค้าได้ 5% และความสำเร็จในด้านนามธรรม เช่น สามารถค้นพบความรู้ใหม่จากข้อมูล

- ประเมินสถานการณ์ในด้านต่างๆ เช่น ความรู้พื้นฐานในเรื่องที่จะทำดาต้าไมนิ่งมีเพียงพอหรือไม่ และผลประโยชน์จากการทำดาต้าไมนิ่งจะคุ้มค่ากับต้นทุนที่เสียไปหรือไม่ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 ขั้นตอนการทำดาต้าไมนิ่ง

- ตั้งเป้าหมายในเชิงการทำดาต้าไมนิ่ง ซึ่งต่างไปจากเป้าหมายหลักในการแก้ปัญหา เช่น เป้าหมายหลัก คือ ต้องการเพิ่มยอดขาดสินค้า เป้าหมายในการเชิงการทำดาต้าไมนิ่ง คือ การหาลักษณะของลูกค้าที่แนวโน้มจะซื้อสินค้า

- วางแผนการทำดาต้าไมนิ่งว่าจะเก็บข้อมูลอย่างไร และใช้อัลกอริทึมใดในการทำดาต้าไมนิ่ง

2. การทำความเข้าใจข้อมูล ประกอบด้วยกระบวนการย่อยดังนี้
   - เก็บรวบรวมข้อมูล
   - กำหนดคุณสมบัติของข้อมูลที่เก็บมาได้
   - สำรวจดูข้อมูลอย่างคร่าวๆ ถึงค่าสถิติต่างๆ ของข้อมูล
   - ตรวจสอบข้อมูลขั้นต้น โดยตรวจสอบทั้งความสมบูรณ์และความถูกต้องของข้อมูล
3. การเตรียมข้อมูล ประกอบด้วยกระบวนการย่อยดังนี้
   - คัดเลือกข้อมูลที่จะนำมาใช้
   - ปรับเปลี่ยนรูปแบบข้อมูล
   - ทำความสะอาดข้อมูล เป็นกระบวนการเตรียมข้อมูลให้เหมาะสมที่สุดเพื่อนำไปใช้ในขั้นตอนต่อไป ซึ่งมีวิธีการต่างๆ หลายวิธี ได้แก่
     - การแก้ไขข้อมูลให้ถูกต้องสมบูรณ์ เช่น การแก้ไขค่าว่างของข้อมูลโดยใส่ค่าศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• ปรับเปลี่ยนข้อมูลให้มีค่าเหมาะสมในการตัดสินใจ เช่น ข้อมูลที่มีค่า "Coke" และ "Pepsi" อาจเปลี่ยนค่าให้เป็น "น้ำอัดลม"

• เลือกข้อมูลเฉพาะที่สนใจ เช่น ต้องการหาลักษณะลูกค้าที่ซื้อรถสปอร์ต ไม่ควรนำรายชื่อพนักงานขายเข้ามาเกี่ยวข้อง

• คอลัมน์ที่มีค่าสำหรับทุกแถวเป็นค่าเดียวกัน เช่น "สัญชาติไทย" หรือ คอลัมน์ที่มีค่าที่ไม่ซ้ำกันเลย เช่น "หมายเลขสมาชิก" ไม่ควรนำมาใช้ เนื่องจากไม่สามารถบอกรูปแบบของข้อมูลได้

4. การสร้างแบบจำลอง ประกอบด้วยกระบวนการย่อยดังนี้
   - เลือกอัลกอริทึมที่เหมาะสมในการทำดาต้าไมนิ่ง
   - กำหนดรูปแบบการทดสอบผลลัพธ์
   - สร้างแบบจำลองตามอัลกอริทึมที่เลือก
   - ทดสอบแบบจำลองที่ได้มานั้นว่ามีความถูกต้องและน่าเชื่อถือเพียงใด

5. การประเมิน อาจจะประเมินแบบจำลองที่สร้างขึ้นด้วยการลองนำไปใช้กับสถานการณ์จริงหรือนำไปใช้ในสถานการณ์ที่จำลอง เพื่อดูว่าแบบจำลองนี้ได้ผลหรือไม่เพียงใด และมีความผิดพลาดตรงไหน ถ้าผิดพลาดจะต้องดำเนินการแก้ไขในกระบวนการ ก่อนที่จะนำแบบจำลองนี้มาใช้งานจริง

6. การนำไปใช้ นำไปใช้และตรวจสอบผลว่าบรรลุเป้าหมายที่ตั้งไว้เพียงใด

2.1.4 เทคนิคที่สำคัญของดาต้าไมนิ่ง

ในทางปฏิบัติจริงดาต้าไมนิ่งจะประสบความสำเร็จกับงานบางกลุ่มเท่านั้น และต้องอยู่ภายใต้ภาวะที่จัดปัญหาให้เหมาะสมกับการใช้เทคนิคดาต้าไมนิ่ง ซึ่งเทคนิคดาต้าไมนิ่งที่สำคัญที่ได้นำมาประยุกต์ใช้ มีดังนี้

1. การค้นหากฎความสัมพันธ์ (Association Rule Discovery)

การค้นหากฎความสัมพันธ์ เป็นการค้นหาความสัมพันธ์ของข้อมูลจากฐานข้อมูลขนาดใหญ่ที่มีอยู่ เพื่อนำไปใช้ในการวิเคราะห์ หรือทำนายปรากฏการณ์ต่างๆ โดยเทคนิคนี้ใช้กันอย่างแพร่หลายในการขายสินค้าหรือการวิเคราะห์ข้อมูลที่เป็นทรานแซกชัน ตัวอย่างการนำเทคนิคการค้นหา กฎความสัมพันธ์มาประยุกต์ใช้ในข้อมูลการขายสินค้าดังตารางที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 แสดงตัวอย่างข้อมูลการขายสินค้า

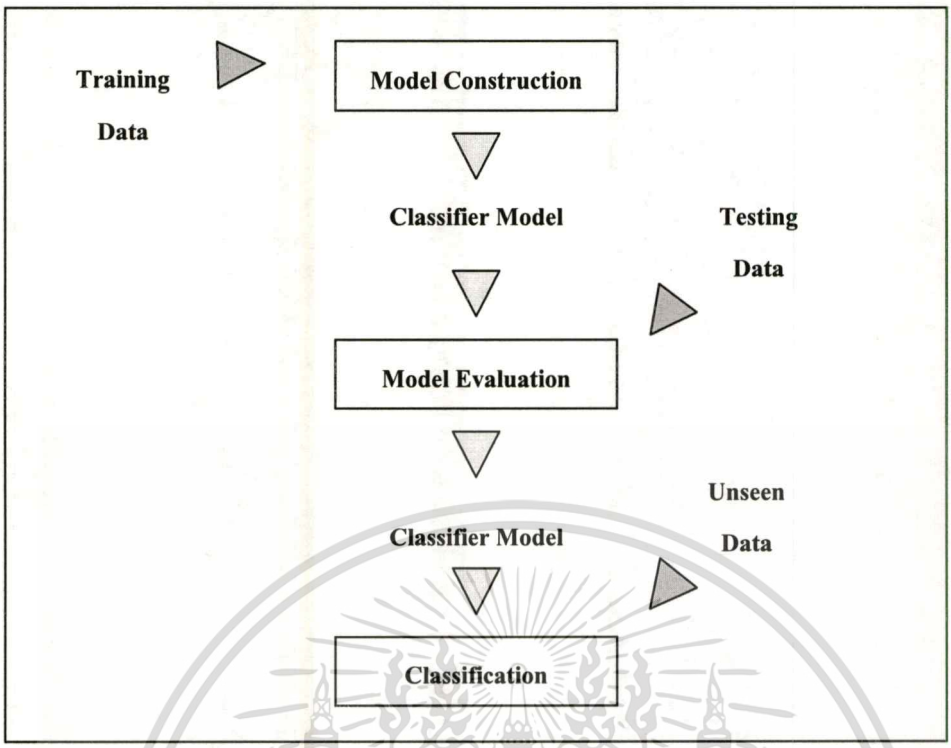
หมายเลขทรานแซกชัน	สินค้าที่ซื้อ
1	น้ำตาล, ขนมปัง
2	นมสด, น้ำตาล, ขนมปัง
3	ขนมปัง
4	นมสด, น้ำตาล

จากตารางที่ 2.1 สามารถบอกได้ว่าน้ำตาลและขนมปังจะถูกซื้อด้วยกันในทรานแซกชันที่ 1 หลังจากที่นำข้อมูลไปผ่านกระบวนการดาต้าไมนิ่งแล้ว จะได้ความสัมพันธ์อยู่ในรูป X -> Y หมายความว่า เมื่อซื้อ X แล้วจะซื้อ Y ด้วย ยกตัวอย่างเช่น นมสด -> น้ำตาล หมายความว่า เมื่อลูกค้าซื้อนมสดแล้วจะซื้อน้ำตาลด้วย

2. การจำแนกประเภทข้อมูล (Data Classification)

การจำแนกประเภทข้อมูล เป็นกระบวนการสร้างโมเดลจัดการข้อมูลให้อยู่ในกลุ่มที่กำหนดมาให้ โดยจะนำข้อมูลส่วนหนึ่งมาสอนให้ระบบเรียนรู้ (Training data) เพื่อจำแนกข้อมูลออกเป็นกลุ่มตามที่ได้กำหนดไว้ ผลลัพธ์ที่ได้จากการเรียนรู้คือ โมเดลจำแนกประเภทข้อมูล (Classifier model) และจะนำข้อมูลส่วนที่เหลือจากข้อมูลสอนระบบเป็นข้อมูลที่ใช้ทดสอบ (Testing data) ซึ่งกลุ่มที่แท้จริงของข้อมูลที่ใช้ทดสอบนี้จะถูกนำมาเปรียบเทียบกับกลุ่มที่หามาได้จากโมเดลเพื่อทดสอบความถูกต้อง และปรับปรุงโมเดลจนกว่าจะได้ค่าความถูกต้องในระดับที่น่าพอใจ หลังจากนั้น เมื่อมีข้อมูลใหม่เข้ามา เราจะนำข้อมูลมาผ่านโมเดล โดยโมเดลจะสามารถทำนายกลุ่มของข้อมูลนี้ได้ ตัวอย่างเช่น จัดกลุ่มนักเรียนว่า เรียนดีมาก ดี ปานกลาง ไม่ดี โดยพิจารณาจากประวัติและผลการเรียนหรือแบ่งประเภทของลูกค้าว่าเชื่อถือได้หรือไม่ โดยพิจารณาจากข้อมูลที่มีอยู่ กระบวนการ Classification นี้ แบ่งออกเป็น 3 ขั้นตอน ดังรูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 กระบวนการ Classification

- Model Construction (Learning)

เป็นขั้นการสร้างตัวแบบ (Model) โดยการเรียนรู้จากข้อมูลที่ได้กำหนดคลาสไว้เรียบร้อยแล้ว(Training data) ซึ่ง Model ที่ได้อาจแสดงในรูปของ

1) แบบต้นไม้ (Decision Tree)

2) แบบนิวรอลเน็ตเวิร์ก (Neural Network)

1) โครงสร้างแบบต้นไม้ของ Decision Tree

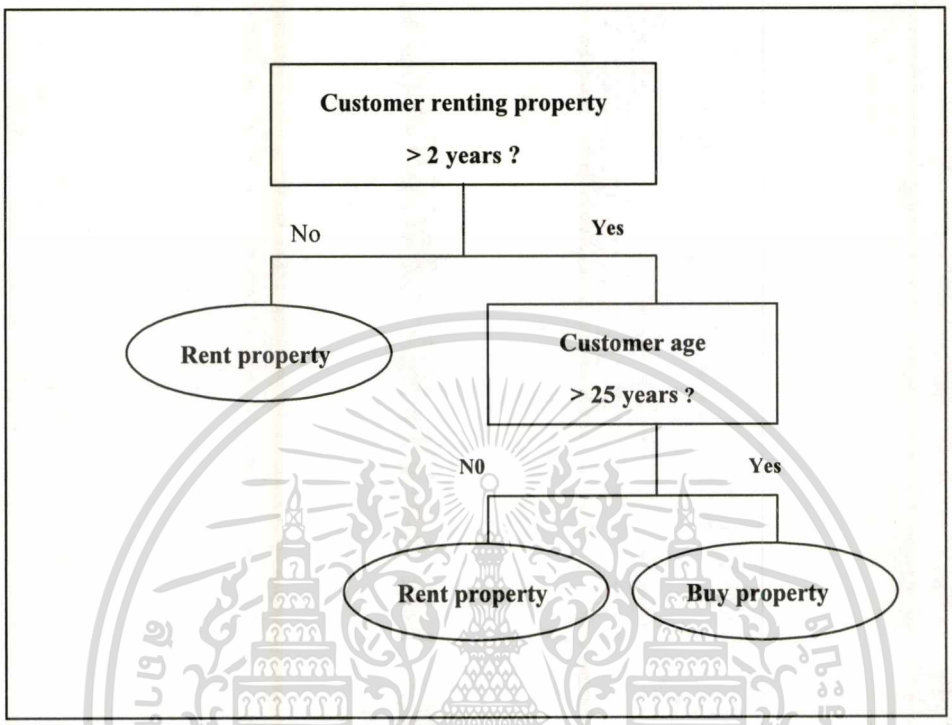
เป็นที่นิยมกันมากเนื่องจากเป็นลักษณะที่คนจำนวนมากคุ้นเคย ทำให้เข้าใจได้ง่าย มีลักษณะเหมือนแผนภูมิองค์กร โดยที่แต่ละโหนดแสดง Attribute แต่ละกิ่งแสดงผลในการทดสอบ และโหนดที่เป็นใบ (Leaf) แสดงคลาสที่กำหนดไว้

สมมติว่าบริษัทขนาดใหญ่แห่งหนึ่ง ทำธุรกิจอสังหาริมทรัพย์มีสำนักงานสาขาอยู่ประมาณ 50 แห่ง แต่ละสาขามีพนักงานประจำ เป็นผู้จัดการและพนักงานขาย พนักงานเหล่านี้แต่ละคนจะดูแลอาคารต่างๆ หลายแห่งรวมทั้งลูกค้าจำนวนมาก บริษัทจำเป็นต้องใช้ระบบฐานข้อมูลที่กำหนดความสัมพันธ์ระหว่างองค์ประกอบเหล่านี้ เมื่อรวบรวมข้อมูลแบ่งเป็นตารางพื้นฐานต่างๆ เช่น ข้อมูลสำนักงานสาขา (Branch) ข้อมูลพนักงาน (Staff) ข้อมูลทรัพย์สิน (Property) และข้อมูลลูกค้า (Client) พร้อมทั้งกำหนดความสัมพันธ์ (Relationship) ของข้อมูลเหล่านี้ เช่น ประวัติการเช่าบ้านของลูกค้า (Customer_rental) รายการให้เช่า (Rentals) รายการขายสินทรัพย์ (Sales) เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้น ต่อมาเมื่อมีประชุมกรรมการผู้บริหารของบริษัท ส่วนหนึ่งของรายงานจากฐานข้อมูลสรุปว่า

" 40 % ของลูกค้าที่เช่าบ้านนานกว่าสองปี และมีอายุเกิน 25 ปี จะซื้อบ้านเป็นของตนเอง โดยกรณีเช่นนี้เกิดขึ้น 35 % ของลูกค้าผู้เช่าบ้านของบริษัท"



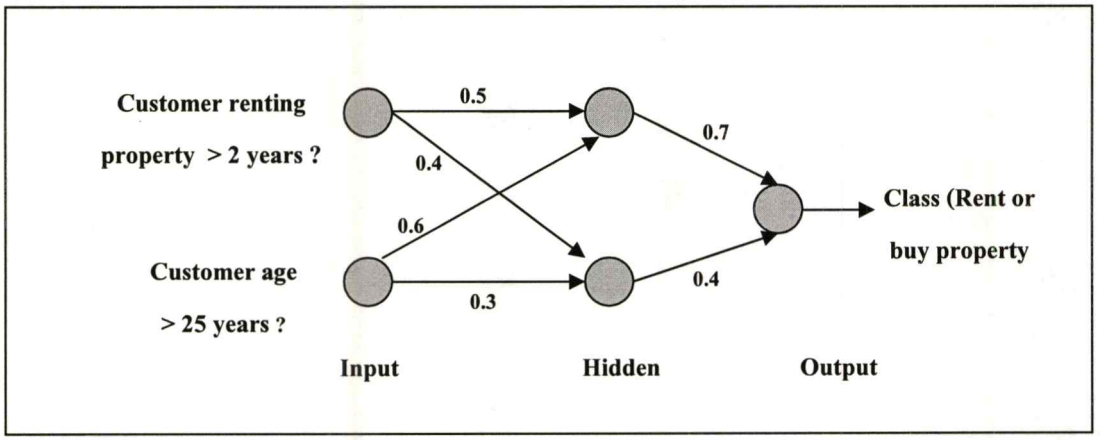
รูปที่ 2.3 ตัวอย่างของ Decision Tree เพื่อวิเคราะห์โอกาสที่ลูกค้าบ้านเช่าจะซื้อบ้าน

ดังรูปที่ 2.3 แสดงให้เห็นถึง Decision Tree สำหรับการวิเคราะห์ว่าลูกค้าบ้านเช่าจะมีความสนใจที่จะซื้อบ้านเป็นของตนเองหรือไม่ โดยใช้ปัจจัยในการวิเคราะห์คือ ระยะเวลาที่ลูกค้าได้เช่าบ้านมา และอายุของลูกค้า

2) นิวรอลเน็ตเวิร์ก (Neural Network)

เป็นเทคโนโลยีที่มีที่มาจากงานวิจัยด้านปัญญาประดิษฐ์ (Artificial Intelligence : AI) เพื่อใช้ในการคำนวณค่าฟังก์ชันจากกลุ่มข้อมูล วิธีการของนิวรอลเน็ตเวิร์กเป็นวิธีการที่ให้เครื่องเรียนรู้จากตัวอย่างต้นแบบ แล้วฝึก (Train) ให้ระบบได้รู้จักที่จะคิดแก้ปัญหาที่กว้างขึ้นได้ ในโครงสร้างของนิวรอลเน็ตจะประกอบด้วยโหนด (Node) สำหรับ Input – Output และการประมวลผล กระจายอยู่ในโครงสร้างเป็นชั้นๆ ได้แก่ Input layer, Output layer และ Hidden layers ซึ่งการประมวลผลของ นิวรอลเน็ตเวิร์กจะอาศัยการส่งการทำงานผ่านโหนดต่างๆ ใน layer เหล่านี้ สำหรับตัวอย่างรูปที่ 2.4 เป็นการวิเคราะห์แบบเดียวกับรูปที่ 2.3 ในโครงสร้างแบบนิวรอลเน็ตเวิร์ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 นิวรอลเน็ตเวิร์กเพื่อวิเคราะห์การเช่าและซื้อบ้านของลูกค้า

- Model Evaluation (Accuracy)

เป็นขั้นการประมาณความถูกต้องโดยอาศัยข้อมูลที่ใช้ทดสอบ (Testing data) ซึ่งคลาสที่แท้จริงของข้อมูลที่ใช้ทดสอบนี้จะถูกนำมาเปรียบเทียบกับคลาสที่หามาได้จาก Model เพื่อทดสอบความถูกต้อง

- Model Usage (Classification)

เป็นตัวแบบ (model) สำหรับใช้ข้อมูลที่ไม่เคยเห็นมาก่อน (Unseen data) โดยจะทำการกำหนดคลาสให้กับ Object ใหม่ที่ได้มา หรือ ทำนายค่าออกมาตามที่ต้องการ

3. การรวมกลุ่มข้อมูล (Database Clustering หรือ Segmentation)

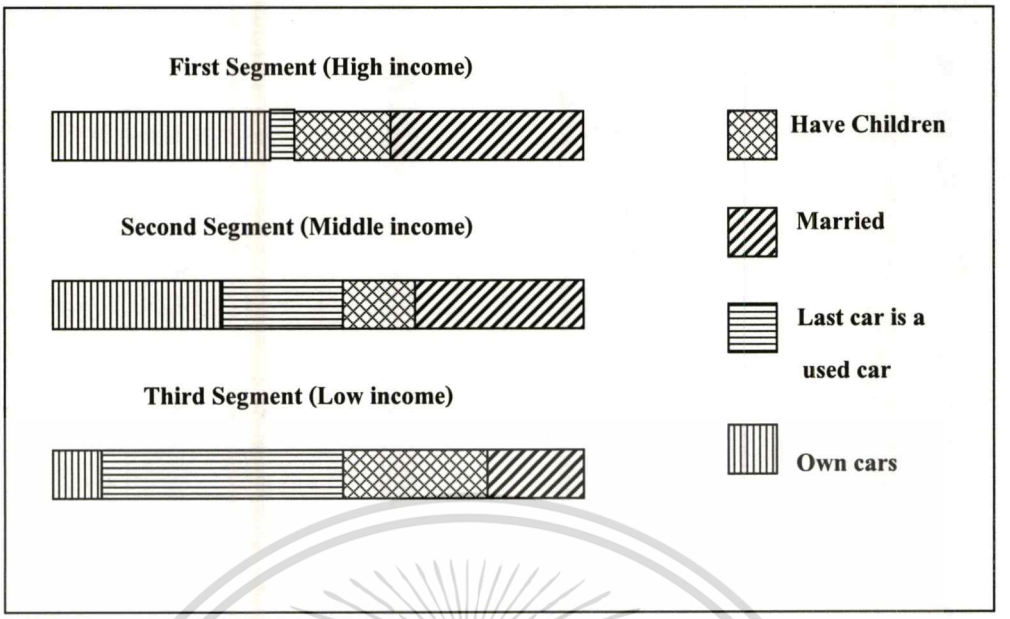
เป็นเทคนิคการลดขนาดของข้อมูลด้วยการรวมกลุ่มตัวแปรที่มีลักษณะเดียวกันไว้ด้วยกัน ตัวอย่างเช่น บริษัทจำหน่ายรถยนต์ได้แยกกลุ่มลูกค้าออกเป็น 3 กลุ่ม คือ

1) กลุ่มผู้มีรายได้สูง (>$80,000)
2) กลุ่มผู้มีรายได้ปานกลาง ($25,000 to $ 80,000)
3) กลุ่มผู้มีรายได้ต่ำ (less than $25,000)

และภายในแต่ละกลุ่มยังแยกออกเป็น

- Have Children
- Married
- Last car is a used car
- Own cars

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ตัวอย่างการรวมกลุ่ม

จากข้อมูลข้างต้นทำให้ทางบริษัทรู้ว่าเมื่อมีลูกค้าเข้ามาที่บริษัทควรจะเสนอขายรถประเภทใด เช่น ถ้าเป็นกลุ่มผู้มีรายได้สูงควรจะเสนอรถใหม่ เป็นรถครอบครัวขนาดใหญ่พอสมควร แต่ถ้าเป็นผู้มีรายได้ค่อนข้างต่ำควรเสนอรถมือสอง ขนาดค่อนข้างเล็ก

4. การพยากรณ์ข้อมูล (Data Prediction)

การพยากรณ์ข้อมูล เป็นกระบวนการสร้างโมเดลเพื่อทำนายหาค่าที่ต้องการจากข้อมูลที่มีอยู่ โดยมีกระบวนการสร้างโมเดลคล้ายกับการจำแนกประเภทข้อมูลดังที่ได้กล่าวมาข้างต้น ต่างกันตรงที่การพยากรณ์ข้อมูลไม่มีการจัดข้อมูลเข้ากลุ่มตามที่ได้กำหนด แต่การพยากรณ์ข้อมูลนี้เป็นการพยากรณ์หาค่าที่ต้องการออกมาเป็นตัวเลข ตัวอย่างเช่น หายอดขายของเดือนถัดไปจากข้อมูลการขายทั้งหมดที่ผ่านมา หรือทำนายเกรดเฉลี่ยของนักเรียนในปีการศึกษาหน้าจากข้อมูลการลงทะเบียนของนิสิตทั้งหมด เป็นต้น

5. Deviation Detection

เป็นกรรมวิธีในการหาค่าที่แตกต่างไปจากค่ามาตรฐาน หรือค่าที่คาดคิดไว้ว่าต่างไปมากน้อยเพียงใด โดยทั่วไปมักใช้วิธีการทางสถิติ หรือการแสดงให้เห็นภาพ (Visualization) สำหรับเทคนิคนี้ใช้ในการตรวจสอบ ลายเซ็นปลอม หรือบัตรเครดิตปลอม รวมทั้งการตรวจหาจุดบกพร่องของชิ้นงานในโรงงานอุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. การบรรยาย (Description)

ในบางครั้งวัตถุประสงค์ของดาต้าไมนิ่ง คือต้องการอธิบายความสับสนของฐานข้อมูลในทางที่จะเพิ่มความเข้าใจในส่วนของประชากร ผลิตภัณฑ์หรือกระบวนการให้มากขึ้น

ซึ่งไม่มีเทคนิคใดเลยที่จะสามารถแก้ปัญหาของดาต้าไมนิ่งได้ทุกปัญหา ดังนั้นความหลากหลายของเทคนิคจึงเป็นสิ่งที่จำเป็นในการไปสู่การแก้ปัญหาของดาต้าไมนิ่งได้ดีที่สุด

2.2 การพยากรณ์ (Prediction)

การพยากรณ์ หมายถึง การคาดคะเนหรือการทำนายค่าของตัวแปรโดยใช้วิธีการทางสถิติวิเคราะห์ข้อมูลในอดีตเพื่อประมาณค่าในอนาคต การพยากรณ์มีความจำเป็นอย่างยิ่งต่อการวางแผนและการตัดสินใจในการดำเนินงาน ซึ่งความสำเร็จของการพยากรณ์จะช่วยลดความเสี่ยงหรือความผิดพลาดในการดำเนินงานและช่วยให้เกิดความสะดวกรวดเร็วในการดำเนินงานมากขึ้น มีคำภาษา อังกฤษที่ความหมายคล้ายกัน คือ Forecasting กับคำว่า Estimating บางคนว่าสองคำนี้เหมือนกัน Forecasting คือ คาดคะเนว่าอนาคตจะเป็นอย่างไร แต่คำว่า Estimate มีความหมายทั่วๆ ไปว่าประมาณการ เช่น ประมาณว่า สร้างบ้านหลังนี้เสียเงินกี่บาทหรือช่วยประมาณว่า ปีนี้จะขายช้างได้กี่ตัว เป็นต้น

ซึ่งลักษณะของการพยากรณ์จะเกี่ยวข้องกับขั้นตอนดังต่อไปนี้

1. กำหนดวัตถุประสงค์ (Objectives) ของการพยากรณ์ เช่น พยากรณ์ปริมาณความต้องการสินค้าที่เพิ่มขึ้น การเปลี่ยนแปลงทางเทคโนโลยีที่มีผลต่อพฤติกรรมของผู้บริโภคขั้นสุดท้าย ทั้งนี้เพื่อทำการผลิตสินค้าให้ได้คุณภาพตามมาตรฐาน และปริมาณเพียงพอต่อความต้องการของตลาด

2. สร้างตัวแบบ (Model) สำหรับการพยากรณ์ โดยปกติในทางปฏิบัติจะใช้ Computer software

3. ทดลองตัวแบบที่สร้างขึ้น เพื่อทดสอบความถูกต้องของวิธีการพยากรณ์ สมการที่ใช้โดยใช้ข้อมูลในอดีต (เช่น ยอดขาย จำนวนหน่วยผลิตสำหรับแต่ละช่วงเวลา) ทั้งนี้เพื่อตรวจสอบค่าที่ได้จากการพยากรณ์กับค่าที่เกิดขึ้นจริงว่าแตกต่างกันอย่างไร ส่วนใหญ่ใช้ข้อมูลของปีที่ดำเนินการ แล้วทำการปรับแต่งตัวเลข/วิธีการให้สอดคล้องกับความเป็นจริง

4. นำตัวแบบไปใช้สำหรับการพยากรณ์ โดยนำข้อมูลในอดีต เช่น ข้อมูลการตลาด ข้อมูลการผลิต ตลอดจนข้อมูลของทั้งอุตสาหกรรมมาเปรียบเทียบกันกับการวิจัยตลาด

5. ประเมินผลและแก้ไขระบบ โดยทำการประเมินในหัวข้อต่อไปนี้

- วิธีการถูกต้องหรือไม่ ผลของการพยากรณ์สอดคล้องกับความเป็นจริงหรือไม่
- วิธีการและผลสามารถแก้ไข หรือสามารถให้มีข้อผิดพลาดน้อยที่สุดได้อย่างไร
- สอดคล้องกับวัตถุประสงค์หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ประเภทของการพยากรณ์

โดยแบ่งออกเป็น 4 ช่วงเวลา ดังต่อไปนี้

1. การพยากรณ์ 1 หน่วยเวลาล่วงหน้า (Immediate – Term Forecasting) เป็นการพยากรณ์ที่มีช่วงเวลาน้อยกว่า 1 เดือน โดยทั่วไปจะเกี่ยวข้องกับกิจกรรมด้านปฏิบัติงานที่อยู่ในความรับผิดชอบของผู้บริหารระดับกลางและระดับต่ำ เป้าหมายของการพยากรณ์จะมุ่งเพื่อการปรับปรุงวิธีการทำงานให้ ดีขึ้นมากกว่าการเปลี่ยนแปลงวิธีการ

2. การพยากรณ์ระยะสั้น (Short - Term Forecasting) เป็นการพยากรณ์ในช่วงเวลาที่ต่ำกว่า 3 เดือน ใช้พยากรณ์แต่ละสินค้าแยกเฉพาะ เพื่อใช้ในการบริหารสินค้าคงคลัง การจัดตารางการผลิต สายการประกอบหรือการใช้แรงงาน ในช่วงเวลาแต่ละสัปดาห์ แต่ละเดือน หรือแต่ละไตรมาส หรืออีก นัยหนึ่งคือการพยากรณ์ระยะสั้นใช้ในการวางแผนระยะสั้น

3. การพยากรณ์ระยะปานกลาง (Medium - Term Forecasting) เป็นการพยากรณ์ในช่วงเวลาที่มากกว่า 3 เดือนจนถึง 2 ปี ใช้พยากรณ์ทั้งกลุ่มของสินค้าหรือยอดขายรวมขององค์กร เพื่อใช้ในการวางแผนด้านบุคลากร การวางแผนการผลิต การจัดตารางการผลิตรวม การจัดซื้อและการกระจายสินค้า ระยะเวลาที่นิยมพยากรณ์คือ 1 ปี เพราะเป็นหนึ่งรอบระยะเวลาบัญชีพอดี การพยากรณ์ระยะปานกลางใช้ในการวางแผนระยะปานกลาง

4. การพยากรณ์ระยะยาว (Long - Term Forecasting) เป็นการพยากรณ์ในช่วงเวลา 2 ปีขึ้นไป ใช้พยากรณ์ยอดขายรวมขององค์กร เพื่อใช้ในการเลือกทำเลที่ตั้งของโรงงานและสิ่งอำนวยความสะดวก การวางแผนกำลังการผลิตและการจัดการกระบวนการผลิตในระยะยาว การพยากรณ์ระยะยาวใช้ในการวางแผนระยะยาว

2.2.2 วิธีการพยากรณ์

ซึ่งในที่นี้จะแบ่งวิธีการพยากรณ์เป็น 2 วิธี คือ วิธีไม่ใช้สถิติ (Nonstatistical forecasting) กับ วิธีใช้สถิติ (Statistical forecasting)

1. วิธีไม่ใช้สถิติ (Nonstatistical forecasting) ไม่ต้องใช้วิธีการหรือความรู้ทางสถิติ ได้แก่ การใช้ความรู้สึกจากประสบการณ์ คือ เป็นการคิดรวมๆ ว่า น่าจะเกิดเหตุการณ์นี้ขึ้น มีความรุนแรงเท่านี้ ถ้าจะถามว่าทำไมจึงเป็นเช่นนั้น ก็ตอบเหตุผลไม่ได้ นักธุรกิจ นักบริหาร นักการเมือง หลายคนใช้วิธีนี้ อย่างไรก็ตาม มิได้หมายความว่า จะเป็นวิธีที่ไร้เหตุผลเสียทีเดียว เพราะประสบการณ์ในอดีตที่นำมาใช้คาดคะเน ก็คือข้อมูลอย่างหนึ่ง การพยากรณ์ก็ต้องอาศัยข้อมูลจากประสบการณ์มาเป็นข้อพิจารณาว่าจะเกิดอะไรขึ้น ขนาดเท่าไร การพยากรณ์โดยอาศัยประสบการณ์นี้ อาจทำในรูป การประชุมปรึกษาหารือกันหลายคนก็ได้

2. วิธีใช้สถิติ (Statistical forecasting) วิธีนี้ใช้หลักวิชาสถิติในการพยากรณ์ อาจใช้มากหรือน้อยก็ตาม ได้แก่ วิธีดูแนวโน้มจากอดีตไปอนาคต และวิธีสร้างสมการเศรษฐมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

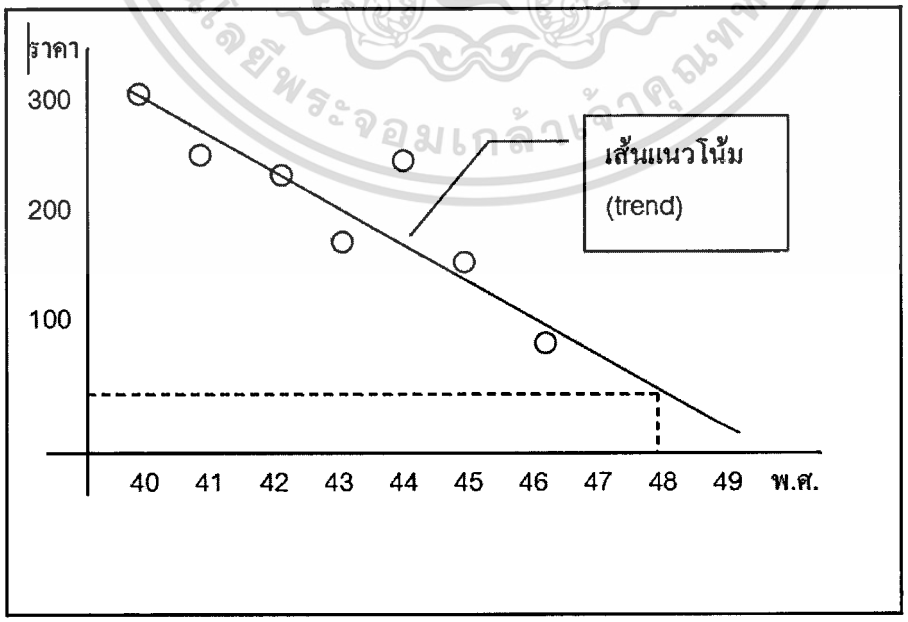
- วิธีดูแนวโน้มจากอดีตไปอนาคต ใช้ลากเส้นโดยประมาณ หรือใช้สมการถดถอย (regression)โดยใช้เวลาเป็นตัวแปรอิสระเพียงตัวเดียว ไม่คำนึงถึงความเป็นเหตุเป็นผล คือไม่สนใจว่า ค่าที่ต้องการพยากรณ์จะมากน้อยเท่าใดนั้น มีสาเหตุมาจากเรื่องใด วิธีดูแนวโน้มจากอดีตไปอนาคต ทำได้สองวิธี 1) วิธีลากเส้นโดยประมาณ 2) วิธีสร้างสมการ Regression มีเวลาเป็นตัวแปรอิสระตัวเดียว ดังนี้

1) วิธีลากเส้นโดยประมาณ วิธีการง่ายๆ ทำตารางบอกค่าตัวแปรกับเวลาแล้วจึงทำการเขียนกราฟลากเส้นผ่านโดยประมาณ แล้วก็อ่านจากกราฟ เพื่อดูว่าเมื่อถึงเวลาในอนาคตค่าตัวแปรจะเป็นเท่าไร ตัวอย่างดังตารางต่อไปนี้

ตารางที่ 2.2 ราคากุ้งกุลาดำในอดีต

ปี พ.ศ.	ราคา (บาท/กก.)
2540	300
2541	242
2542	220
2543	150
2544	230
2545	135
2546	90

จงทำนายราคากุ้งกุลาดำ ในปี พ.ศ. 2548



รูปที่ 2.6 ราคากุ้งกุลาดำในปีต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการเขียนกราฟ พยากรณ์ได้ว่า ราคากุ้งกุลาดำ พ.ศ. 2548 ราคา กก.ละ 50 บาท ซึ่งอาจจะมีปัญหาที่จะเกิดขึ้นบ้าง คือ การลากเส้นแนวโน้ม แต่ละคนจะลากไม่เหมือนกัน เพราะว่าการพยากรณ์ย่อมผิดพลาดได้ การผิดพลาดเพียงเท่านี้ ไม่มากนัก

2) วิธีสร้างสมการ Regression มีเวลาเป็นตัวแปรอิสระตัวเดียว จากตารางที่ 2.2 นำมาสร้างสมการ Regression ได้ คือ

$$P = 1,476 - 29.7858xT$$

อยากทราบราคา พ.ศ. 2548 ก็แทนค่า T ด้วย 48 จะได้ราคา (P) เท่ากับ 46.3571 บาท

อยากทราบราคา พ.ศ. 2549 ก็แทนค่า T ด้วย 49 จะได้ราคา (P) เท่ากับ 16.5714 บาท

รูปแบบของสมการพยากรณ์ มีหลายรูปแบบ การเลือกใช้ให้ดูจากความพอดีมากที่สุดระหว่างข้อมูลกับสมการ คือ รูปแบบใดได้ค่า R^2 (Coefficient of determination แสดงถึง Degree of fit) มากที่สุด ก็ใช้รูปแบบนั้นในการพยากรณ์ แปลว่าต้องสร้างสมการหลายสมการนั่นเอง แล้วนำสมการที่มีค่า R^2 สูงสุดมาใช้ รูปแบบสมการที่ควรสร้างขึ้นเพื่อคัดเลือกมาใช้ คือ

สมการเส้นตรง (straight line) $Y = a + b\,T$

สมการตัวแปรยกกำลัง (exponential line) $Y = a.bT$

สมการกำลังสอง (quadratic line) $Y = a + bT + cT2$

สมการ P = 1,476 - 29.7858xT ในตัวอย่าง คือ สมการเส้นตรง

บางคนต้องการสร้างสมการพยากรณ์ในรอบปีเก็บไว้ใช้ได้ทุกปี ก็ใช้ข้อมูลในปีก่อนๆ หลายปีมาสร้างสมการ แล้วอาจเพิ่มการเปลี่ยนแปลงขึ้นลงของค่าที่พยากรณ์เข้าไปด้วย เช่น ราคาสินค้าเกษตรขึ้นลงตามฤดูกาล ก็เพิ่มค่าการขึ้นลงตามฤดูกาลเข้าไปด้วย ถ้าพบว่าการขึ้นลงเป็นวัฏจักรธรรมชาติก็เพิ่มค่าการขึ้นลงตามวัฏจักรเข้าไปด้วย การทำเช่นนี้ เรียกว่า การวิเคราะห์สมการอนุกรมเวลา (Time series analysis)

2.3 การวิเคราะห์อนุกรมเวลา (Time Series Analysis)

อนุกรมเวลา (Time Series) คือ เซตของข้อมูลเชิงปริมาณที่จัดเก็บในช่วงเวลาหนึ่ง ตัวอย่างเช่น ดัชนีตลาดหลักทรัพย์ในแต่ละวันเมื่อปิดทำการซื้อขายในแต่ละวัน รายได้ประชาชาติ (GNP) รายไตรมาส รายรับในแต่ละปีของบริษัทแห่งหนึ่ง เป็นต้น

ข้อมูลอนุกรมเวลา (Time series data) หมายถึง ข้อมูลที่เกิดขึ้นตามลำดับเวลาที่มีระยะห่างเท่าๆ กันอย่างต่อเนื่องกันเป็นระยะเวลานาน อาจเป็น เดือน ปี สัปดาห์ วัน หรือชั่วโมงก็ได้ เช่น รายได้ต่อเดือน งบประมาณค่าใช้จ่ายต่อปี ราคาหุ้นของธนาคารแห่งหนึ่งในแต่ละวัน ยอดขายรายเดือนของบริษัทแห่งหนึ่ง เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวิเคราะห์อนุกรมเวลา (Time series analysis) เป็นระเบียบทางสถิติที่สามารถแปลงประสบการณ์ในอดีตไปพยากรณ์เหตุการณ์ในอนาคต ดังตัวอย่างเช่น

- ฝ่ายจัดซื้อของบริษัทผลิตเครื่องสำอางแห่งหนึ่งอาจใช้ประสบการณ์ในอดีต เพื่อไปพยากรณ์หรือตัดสินใจว่า ควรสั่งซื้อวัตถุดิบประเภทใด
- การไฟฟ้าฝ่ายผลิตอาจตัดสินว่าอุปสงค์สำหรับความต้องการพลังไฟฟ้าจะเติบโตในอัตราที่คล้ายคลึงกับทศวรรษก่อนหน้านี้ เพื่อจะคาดการณ์ความต้องการใน 10 หรือ 20 ปีข้างหน้า

เนื่องจากข้อมูลทางธุรกิจมีการเปลี่ยนแปลงตลอดเวลา ผู้นำทางธุรกิจหรือองค์กรต้องหาวิธีพัฒนาต่างๆ ที่สามารถนำไปใช้ประกอบการตัดสินใจวางแผน เกี่ยวกับผลที่เกิดจากความเปลี่ยนแปลงในการดำเนินการอย่างใดอย่างหนึ่ง ดังนั้นการวิเคราะห์อนุกรมเวลาจึงเข้ามามีบทบาทช่วยในการตัดสินใจ เทคนิคอย่างหนึ่งที่ใช้ช่วยในการควบคุมการดำเนินการในปัจจุบันและในการวางแผนความต้องการในอนาคต คือ การพยากรณ์ (Forecasting) ซึ่งการพยากรณ์นั้นทำได้หลายวิธี แต่ละวิธีต่างมีเป้าหมายเดียวกัน คือ ทำนายเหตุการณ์ในอนาคต

2.3.1 ส่วนประกอบของอนุกรมเวลา

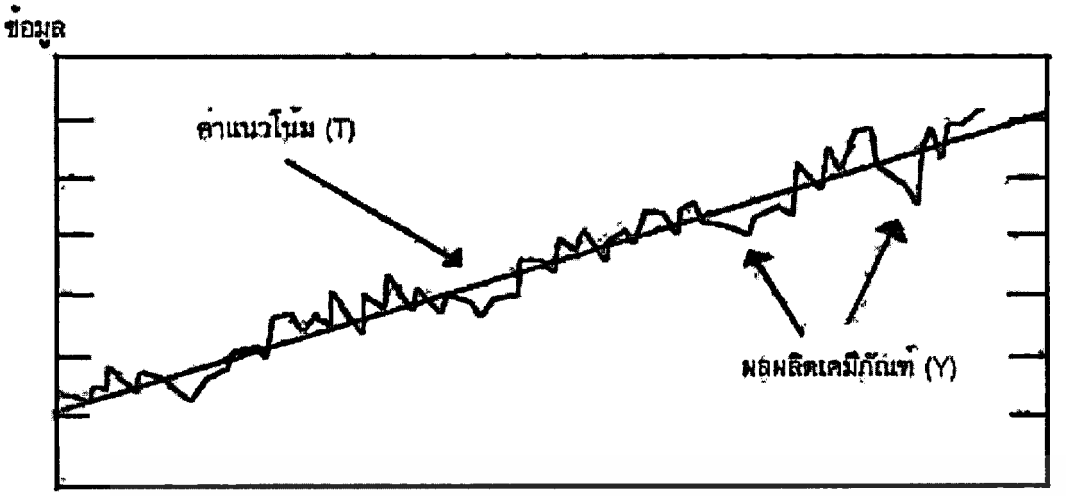
ในการวิเคราะห์อนุกรมเวลา ผู้วิเคราะห์จะแยกองค์ประกอบต่างๆ ที่ประกอบกันขึ้นเป็นอนุกรมเวลา โดยจะมีการเปลี่ยนแปลงไปตามอิทธิพลต่างๆ เช่น การเปลี่ยนแปลงการผลิต เทคโนโลยี สภาวะอากาศ เป็นต้น ในการหาคุณลักษณะของอนุกรมเวลา เราสามารถใช้แบบจำลองได้หลายแบบ แบบจำลองที่ใช้โดยนักเศรษฐศาสตร์แบบหนึ่ง คือ แบบจำลองแบบคลาสสิก (Classical model) เป็นการอธิบายถึงองค์ประกอบของการแปรผันของอนุกรมเวลา 4 ส่วน ดังนี้

1. ค่าแนวโน้ม (Secular Trend) แทนด้วย T_t

การเคลื่อนไหวขึ้นลงของข้อมูลเป็นระยะเวลายาวนาน พอที่จะสังเกตเห็นได้ว่ามีแนวโน้มการเปลี่ยนแปลงเป็นเส้นตรง (เมื่อข้อมูลมีอัตราการเปลี่ยนแปลงคงที่) หรือเป็นเส้นโค้ง (เมื่อข้อมูลมีอัตราการเปลี่ยนแปลงไม่คงที่) ค่าแนวโน้มของข้อมูลเป็นการเคลื่อนไหวในช่วงระยะเวลาที่ค่อนข้างนานพอสมควร ควรเป็นข้อมูลรายปี และควรมีข้อมูลอย่างน้อย 15 ปี ซึ่งจะแสดงทิศทางของอนุกรมเวลา

ตัวอย่างข้อมูลอนุกรมเวลาที่แสดงแนวโน้มการเคลื่อนไหวระยะยาว เช่น กราฟแสดงค่าแนวโน้มของผลผลิตเคมีภัณฑ์ชนิดหนึ่ง ดังรูปที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



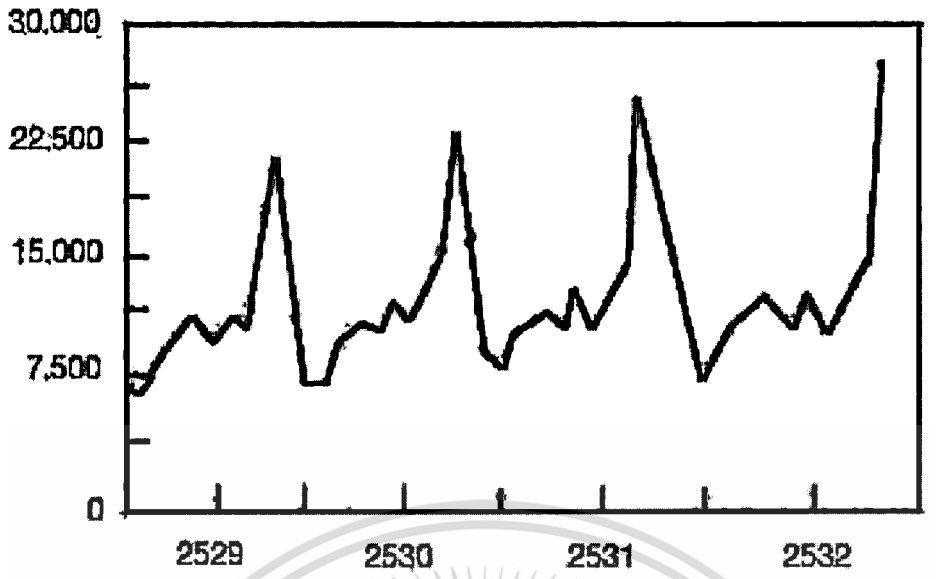
รูปที่ 2.7 แสดงค่าแนวโน้มผลผลิตเคมีภัณฑ์ชนิดหนึ่ง

จากกราฟ Y แทนข้อมูลอนุกรมเวลาของผลผลิตเคมีภัณฑ์ชนิดหนึ่ง ในช่วงเวลา 15 ปี ค่าแนวโน้มแทนด้วยกราฟเส้นตรง ซึ่งซ้อนอยู่บนเส้นกราฟของ Y เส้นกราฟแสดงแนวโน้มนอกจากจะมีลักษณะเป็นเส้นตรงดังรูปแล้ว อาจมีลักษณะเป็นเส้นโค้ง เช่น เส้นโค้งเอ็กโพเนนเชียลหรือพาราโบลา ก็ได้

2. การเปลี่ยนแปลงหรือความแปรผันตามฤดูกาล (Seasonal Variation) แทนด้วย S_t

เป็นการเคลื่อนไหวขึ้นลงของข้อมูลที่เกิดขึ้นเนื่องจากฤดูกาล หรือประเพณี และมักจะเกิดซ้ำกันเป็นประจำในช่วงเวลาเดียวกันของแต่ละปี มักจะเป็นรายไตรมาส (Quarterly) หรือรายเดือน ดังนั้น ข้อมูลที่บันทึกมาเป็นรายปีจะไม่สามารถหาความแปรผันตามฤดูกาลได้ จะต้องเป็นข้อมูลที่บันทึกมาเป็นช่วงเวลาต่ำกว่า 1 ปี เป็นไตรมาส เป็นเดือน เป็นสัปดาห์ หรือเป็นวัน จึงจะหาความแปรผันตามฤดูกาลได้ เช่น ยอดจำหน่ายเสื้อหนาวจะสูงในเดือนธันวาคม - มกราคม และจะลดต่ำลงในเดือนเมษายน - พฤษภาคม ตัวอย่างเช่น ยอดขายรายเดือนของห้างสรรพสินค้าแห่งหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงยอดขายรายเดือนของห้างสรรพสินค้าแห่งหนึ่ง

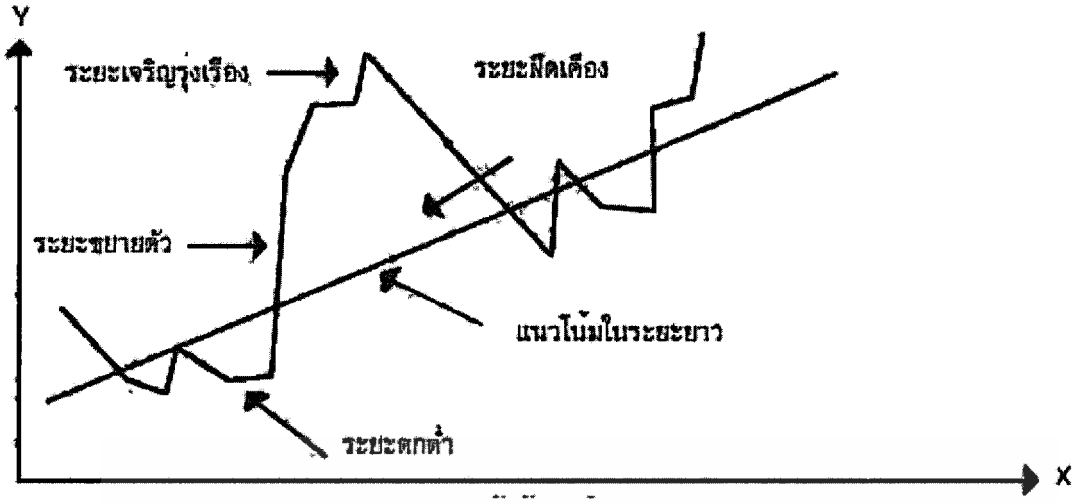
จากรูปที่ 2.8 จะเห็นว่ายอดขายของห้างสูงประมาณเดือนธันวาคมของทุกปี ซึ่งเป็นเทศกาลคริสต์มาส และปีใหม่ ประชาชนจึงมีการจับจ่ายใช่สอยมาก ส่วนในราวเดือนพฤษภาคมของทุกปี ยอดขายจะต่ำกว่าในเดือนอื่น ๆ ที่เป็นเช่นนี้เพราะเป็นช่วงเปิดภาคเรียน ประชาชนต้องเตรียมเงินไว้สำหรับค่าใช้จ่ายในการศึกษาของบุตรหลาน

3. การเปลี่ยนแปลงหรือความผันแปรตามวัฏจักร (Cyclical Variation) แทนด้วย C_t

เป็นการเคลื่อนไหวขึ้นลงของข้อมูล และมีการเกิดซ้ำกันคล้ายการแปรผันตามฤดูกาล แต่จะเกิดซ้ำกันในช่วงที่ยาวนานกว่าหนึ่งปี ความแปรผันตามวัฏจักรสามารถหาได้จากข้อมูลอนุกรมเวลาที่จัดเก็บมาเป็นปีหรือช่วงที่สั้นกว่า 1 ปี

รูปร่างของวัฏจักรประกอบไปด้วยช่วงซึ่งแสดงถึงความเจริญรุ่งเรือง (Prosperity) ช่วงเสื่อมถอยหรือชะงักงัน (Recession) ช่วงตกต่ำหรือหยุดอยู่กับที่(Depression) และช่วงการฟื้นตัว (Recovery) แต่ละรอบของการเกิดซ้ำจะไม่คงที่ จะแตกต่างกันไปในช่วงเวลาและช่วงความกว้าง เนื่องจากรูปแบบการเกิดซ้ำไม่เป็นปกติทำให้การทำนายการเกิดในอนาคตทำได้ยาก ความแปรผันตามวัฏจักรโดยทั่วๆ ไปจะเกิดจากสภาพเศรษฐกิจ การเปลี่ยนแปลงนโยบายของรัฐบาล การเปลี่ยนรสนิยมหรือพฤติกรรมการบริโภค จากรูปที่ 2.9 แสดงการเกิดระยะต่าง ๆ เหล่านี้ เกิดอย่างต่อเนื่องเป็นวัฏจักรและแต่ละรอบของวัฏจักรมีระยะเวลาไม่แน่นอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แสดงวัฏจักรธุรกิจ

4. การเปลี่ยนแปลงหรือความผันแปรเนื่องจากเหตุการณ์ผิดปกติ (Irregular Variation) แทนด้วย I_t

หมายถึง การเคลื่อนไหวขึ้นลงของข้อมูลที่ไม่มีรูปแบบที่แน่นอน และไม่สามารถจะคาดคะเน หรือทำนายได้ล่วงหน้าความแปรผันที่ไม่สม่ำเสมอนี้เกิดจากเหตุการณ์ไม่ปกติต่าง ๆ เช่น น้ำท่วม ภูเขาไฟระเบิด สงคราม แผ่นดินไหว การนัดหยุดงาน เป็นการเปลี่ยนแปลงที่เป็นเชิงสุ่ม (Random variation) เพราะไม่ได้อยู่ภายใต้เงื่อนไขที่เรากำหนด

จากองค์ประกอบของอนุกรมเวลาทั้ง 4 อย่าง คือ T S C และ I ในข้อมูลอนุกรมชุดหนึ่งๆ ไม่จำเป็นต้องครบองค์ประกอบข้างต้นก็ได้ ทั้งนี้ ขึ้นอยู่กับชนิดของมูลของเรา

2.3.2 กระบวนการสร้างรูปแบบอนุกรมเวลา

เราสามารถหาค่าพยากรณ์ในอนาคตจากการศึกษาข้อมูลของตัวแปรนั้นในอดีต เช่น การหารูปแบบของข้อมูล แล้วนำรูปแบบนั้นไปพยากรณ์ค่าในอนาคต ขั้นตอนการสร้างรูปแบบมีดังนี้

1. กำหนดหรือเลือกรูปแบบที่เหมาะสมกับข้อมูลอนุกรมเวลาที่ศึกษา โดยใช้ความรู้ทางสถิติและประสบการณ์ในอดีต หรือทำการเปรียบเทียบกับข้อมูลอื่นที่มีลักษณะคล้ายกัน

2. ประมาณค่าพารามิเตอร์ของรูปแบบที่ได้จากขั้นที่ 1

3. ทดสอบความถูกต้องเหมาะสมของรูปแบบที่ได้จากขั้นที่ 1 โดยใช้หลักการทางสถิติมาทดสอบ ถ้าสรุปผลการทดสอบได้ว่าเหมาะสมก็สามารถนำรูปแบบนั้นไปใช้การพยากรณ์ได้ แต่ถ้าสรุปได้ว่าไม่เหมาะสม ต้องสร้างรูปแบบใหม่ โดยต้องย้อนกลับไปทำขั้นตอนที่ 1 ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การหาค่าเหมาะสมแบบกลุ่มอนุภาค (Particle Swarm Optimization)

การหาค่าความเหมาะสมแบบกลุ่มอนุภาค (Particle Swarm Optimization : PSO) ซึ่งถูกคิดค้นโดย Russell Eberhart และ James Kennedy ในปี 1995 เป็นการหาที่เหมาะสมที่สุดแบบกลุ่มอนุภาคโดยใช้เทคนิคการหาค่าเหมาะสมที่สุดแบบสุ่ม (Stochastic) เป็นวิธีการหาค่าความเหมาะสมโดยอาศัยพื้นฐานของความน่าจะเป็น ทั้งคู่ได้รับแรงจูงใจมาจากพฤติกรรมทางสังคมในการอพยพของฝูงนก และการเรียนรู้ของฝูงปลา

ซึ่ง PSO เป็นอัลกอรึทึมที่มีวิธีการที่คล้ายกันกับเทคนิคของจีเนติกอัลกอริทึม เป็นเทคนิคของ Evolutionary Computation ซึ่ง PSO เป็นเครื่องมือที่หาคำตอบที่เหมาะสมที่สุดของภายในกลุ่มประชากรนั้นๆ โดยเริ่มต้นจะทำการสุ่มสร้างประชากรจำนวนหนึ่งขึ้นมา แล้วทำการหาค่าที่เหมาะสมที่สุดโดยการพัฒนาประชากรที่มีอยู่แล้วไปเรื่อยๆ สำหรับไปใช้ในรุ่นต่อไป ซึ่งทั้งสองอัลกอรึทึมนี้จะเหมือนกันตรงขั้นตอนการสร้างประชากรเท่านั้น เนื่องจาก PSO จะให้ประชากรที่สร้างขึ้นมา ที่เรียกว่า "Particle" นั้นบินผ่านช่องว่างของปัญหาและทำการปรับทิศทางของตัวมันไปในทิศทางของตำแหน่งของพาติเคิลที่มีค่าความเหมาะสมมากกว่าตัวมัน เป็นจำนวนหลายๆ รอบ จนพาติเคิลได้ตำแหน่งที่ดีที่สุด ซึ่ง PSO นี้ไม่ได้ทำการสร้างประชากรใหม่ด้วยการ Crossover และ Mutation เพื่อสร้างประชากรใหม่ๆ เหมือนกับจีเนติกอัลกอริทึม

2.4.1 การทำงานของการหาค่าเหมาะสมแบบกลุ่มอนุภาค

ขั้นตอนการหาค่าความเหมาะสมแบบกลุ่มอนุภาค สมมติสถานการณ์สร้างแบบจำลองการค้นหาอาหารของฝูงนก ถ้าในบริเวณที่ต้องการหามีอาหารเพียงที่เดียว นกทั้งฝูงจะไม่รู้ว่าอาหารอยู่ที่ไหนแต่เมื่อผ่านการค้นหาในแต่ละรอบ ถึงจะรู้ว่าอาหารอยู่ใกล้นกตัวไหน แล้วตามนกตัวที่อยู่ใกล้อาหารมากที่สุด

หลักการของการหาค่าความเหมาะสมแบบกลุ่มอนุภาค คือ เรียนรู้สถานการณ์และใช้ข้อมูล
ร่วมกัน ทำให้สามารถแก้ปัญหาเกี่ยวกับการหาคำตอบที่เหมาะสม ในการหาค่าความเหมาะสมแบบ
กลุ่มอนุภาคนั้น แต่ละคำตอบที่เป็นไปได้จะมีค่าความเหมาะสม (Fitness Value) หรือฟังก์ชั่นเป้าหมาย (Objective Function) กำกับไว้ ทำให้ได้มาของตำแหน่งและกำหนดให้เป็นค่าความเหมาะสม (Fitness Value) โดยตำแหน่งที่มีค่าความเหมาะสมสูงที่สุดจากทั้งหมดจะถูกเรียกว่า global best (gbest) และ แต่ละพาร์ทิเคิลจะมีการเก็บค่าความเหมาะสมสูงที่สุดของตัวเองไว้ โดยตำแหน่งของค่าเหล่านี้จะเป็น personel best (pbest) ของตัวเอง อัลกอริทึมพื้นฐานจะมีส่วนเกี่ยวข้องกับการจัดการกับจำนวนประชากรของพาร์ทิเคิลตลอดการค้นหาและวิธีที่ดีสุดเหมาะสมที่สุดจะถูกค้นพบ แต่ละครั้งในการทำซ้ำทุกๆ พาร์ทิเคิลจะมีการปรับขนาดของเวกเตอร์ความเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Velocity Vector) โดยยึดหลักโมเมนตัมของพาร์ทิเคิลและใช้การชักจูงจากของทั้งวิธีที่ดีที่สุดของพาร์ทิเคิลเองและวิธีที่ดีที่สุดของบริเวณใกล้เคียง จากนั้นตำแหน่งใหม่จะถูกคำนวณเพื่อตรวจสอบซึ่งทำให้ได้มาของผลลัพธ์ที่ดีที่สุด

หลังจากที่เราสร้างพาติเคิลขึ้นมาจำนวนหนึ่งโดยการสุ่มตำแหน่งให้กับพาติเคิลและคำนวณค่าความเหมาะสมทั้งหมดแล้ว ขั้นตอนต่อไปเป็นการปรับตำแหน่งใหม่ของพาติเคิลแต่ละตัว ซึ่งจะมีวิธีการเลือกตำแหน่งใหม่จากค่า pbest และ gbest ที่พาติเคิลได้จำเอาไว้จากสมการดังต่อไปนี้

$$V(t+1) = V(t) + C_1 * rand()_1 * (pbest(t) - present(t)) + C_2 * rand()_2 * (gbest(t) - present(t)) \quad (2.1)$$

และ

$$present(t+1) = present(t) + V(t+1) \quad (2.2)$$

เมื่อ

V(t)	คือ	ค่าความเร็วปัจจุบันของพาติเคิล
present(t)	คือ	ค่าตำแหน่งปัจจุบันของพาติเคิล
C_1	คือ	ค่าคงที่มีความสัมพันธ์กับค่า pbest นิยมใช้ค่าประมาณ 2.00 ถ้าค่านี้มากแสดงว่าให้ความสำคัญกับค่า pbest มาก
C_2	คือ	ค่าคงที่มีความสัมพันธ์กับค่า gbest นิยมใช้ค่าประมาณ 2.00 ถ้าค่านี้มากแสดงว่าให้ความสำคัญกับค่า gbest มาก
$rand()_1$	คือ	ค่าที่ถูก random ขึ้นมามีค่า 0 - 1
$rand()_2$	คือ	ค่าที่ถูก random ขึ้นมามีค่า 0 - 1
pbest	คือ	ค่าตำแหน่งที่มีค่าความเหมาะสมสูงสุดที่พาติเคิลได้วิ่งผ่าน
gbest	คือ	ค่าตำแหน่งที่มีค่าความเหมาะสมสูงสุดที่ทุกพาติเคิลเคยเจอ

2.4.2 อัลกอริทึมหาค่าความเหมาะสมของกลุ่มอนุภาค

การหาตำแหน่งใหม่ของแต่ละพาติเคิลจะประกอบไปด้วย 2 ขั้นตอนหลัก คือ 1) ขั้นตอนการหาความเร็วใหม่ และ 2) ขั้นตอนการปรับตำแหน่ง (นำความเร็วใหม่บวกกับตำแหน่งพาติเคิลเดิม)

การเริ่มโดยการนำเอาสมการ (2.1) ทำการคำนวณหาความเร็วที่จะต้องเปลี่ยนให้กับพาติเคิลก่อนแล้วจึงนำค่าที่ได้มารวมกับตำแหน่งเดิมของพาติเคิลตัวนั้นๆ โดยใช้สมการที่ (2.2) จากนั้นเราก็จะได้ค่าตำแหน่งใหม่ของพาติเคิล และหลังจากนั้นทุกพาติเคิลก็จะได้ตำแหน่งใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และก็เริ่มทำการคำนวณหาค่าความเหมาะสมอีกวนไปเรื่อยๆ จนกว่าจะได้ค่าความเหมาะสมที่เราพอใจ

นอกจากนั้นในการปรับความเร็วและตำแหน่งของแต่ละพาติเคิลนั้นเรายังต้องพิจารณาค่าอีก 1 ค่าด้วย นั่นคือค่าความเร็วสูงสุด โดยค่านี้เป็นค่าที่มีความสำคัญอย่างมาก ถ้าเรากำหนดค่านี้ให้มากเกินไปแล้วก็จะมีโอกาสมากที่จะทำให้พาติเคิลวิ่งผ่านจุดที่มีค่าความเหมาะสมมากที่สุดไปได้

ซึ่งเราสามารถสรุปขั้นตอนการทำงานต่างๆ ของการแบ่งกลุ่มอนุภาค (PSO) ได้ดังนี้

1. สร้างประชากรของพาติเคิลขึ้นมาพร้อมทั้งสุ่มค่าตำแหน่งให้กับแต่ละพาติเคิล
2. คำนวณหาค่าความเหมาะสมแต่ละพาติเคิลและจัดเก็บค่า pbest ไว้
3. ถ้าค่าที่ดีที่สุดของ pbest ดีกว่าค่า gbest ปัจจุบัน gbest จะถูกแทนด้วยค่าที่ดีที่สุดนั้นและจัดเก็บค่า gbest ไว้
4. คำนวณค่าความเร็วใหม่ ตามค่า pbest และ gbest ของแต่ละพาติเคิล
5. ปรับค่าตำแหน่งของทุกๆ พาติเคิลใหม่
6. ตรวจสอบเงื่อนไขการจบการทำงาน ถ้าจำนวนการทำซ้ำสูงที่สุดที่กำหนดไว้ล่วงหน้ายังไม่หมด ให้กลับไปทำที่ขั้นตอนที่ 2 จนครบ

ซึ่งเราสามารถเขียนอัลกอริทึมของ PSO ได้อีกแบบหนึ่งดังรูปที่ 2.10 ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
For each particle /* กำหนดค่าเริ่มต้น*/
        Initialize particle
End
Do
        For each particle /*ประเมินค่าความเหมาะสมของแต่ละ particle*/
                Calculate fitness value
                If the fitness value is better than the best fitness value(pBest)
                In history then set current value as the new pBest
        End
        Choose the particle with the best fitness value of all the particles as the gBest
        For each particle /*ปรับข้อมูลและความเร็วในการมุ่งสู่เป้าหมาย*/
                Calculate particle velocity according equation(13)
                Update particle position according equation(14)
While maximum iterations or minimum error criteria is not attained
```

รูปที่ 2.10 อัลกอริทึมการหาค่าเหมาะสมแบบกลุ่มอนุภาค

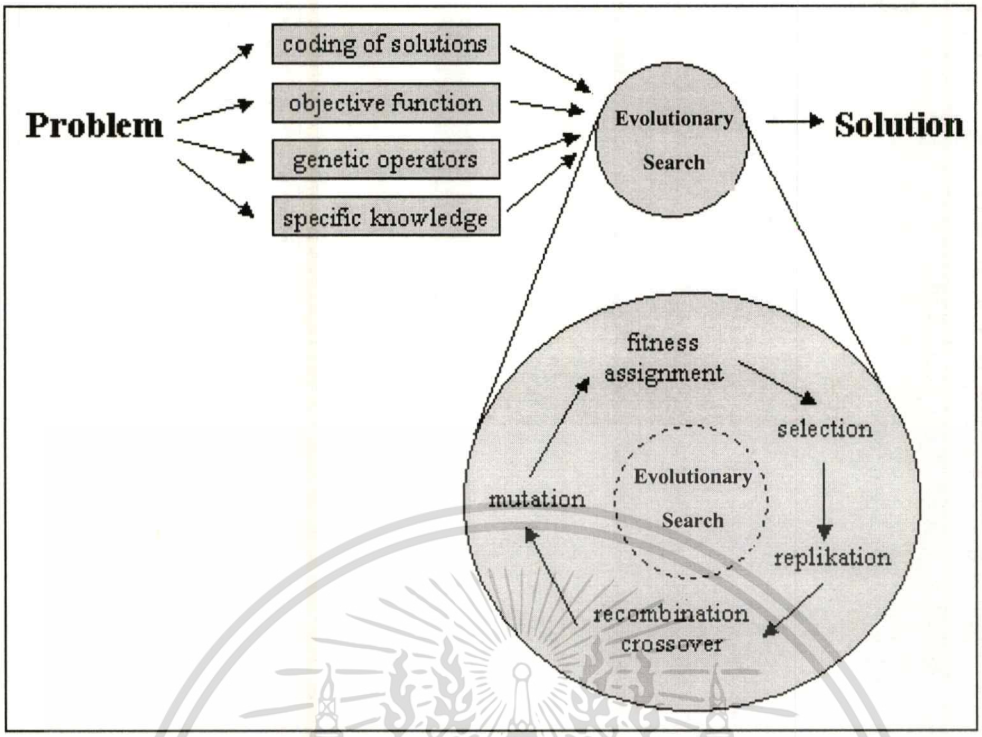
2.5 Evolutionary Algorithm

2.5.1 การทำงานของ Evolutionary Algorithm

ในช่วงทศวรรษที่ผ่านมาได้มีสิ่งต่างๆ ถูกพัฒนาขึ้นมากมายโดยการนำเอาอัลกอริทึมมาหาคำตอบที่เหมาะสมที่สุด เป็นลักษณะเกี่ยวกับ Evolutionary Algorithm (EA) ที่เป็นที่รู้จักกันว่าเป็น Evolutionary Computation ซึ่ง Evolutionary Algorithm นั้น เป็นอัลกอริทึมที่ใช้ค้นหาคำตอบที่เหมาะสมหรือดีที่สุดของประชากรภายในกลุ่ม ซึ่งมีรากฐานแนวคิดมากจากทฤษฎีวิวัฒนาการของชาร์ล ดาร์วิน (Charles Darwin) โดยอิงจากหลักการแนวความคิดของการอยู่รอดของผู้ที่แข็งแรงที่สุด ซึ่งวิธีที่ใช้ในการหาคำตอบที่เหมาะสมที่สุดของประชากรนั้นจะสามารถค้นพบคำตอบที่ดีที่สุดที่มีประสิทธิภาพและมีประสิทธิผล ส่วนมากในการใช้วิธีการค้นหานั้นจะใช้ทฤษฎีวิวัฒนาการ ซึ่ง Evolutionary Computation นั้นประกอบด้วย 3 วิธีหลัก ดังนี้

1. Genetic Algorithm
2. Evolutionary Programming
3. Evolutionary Strategies

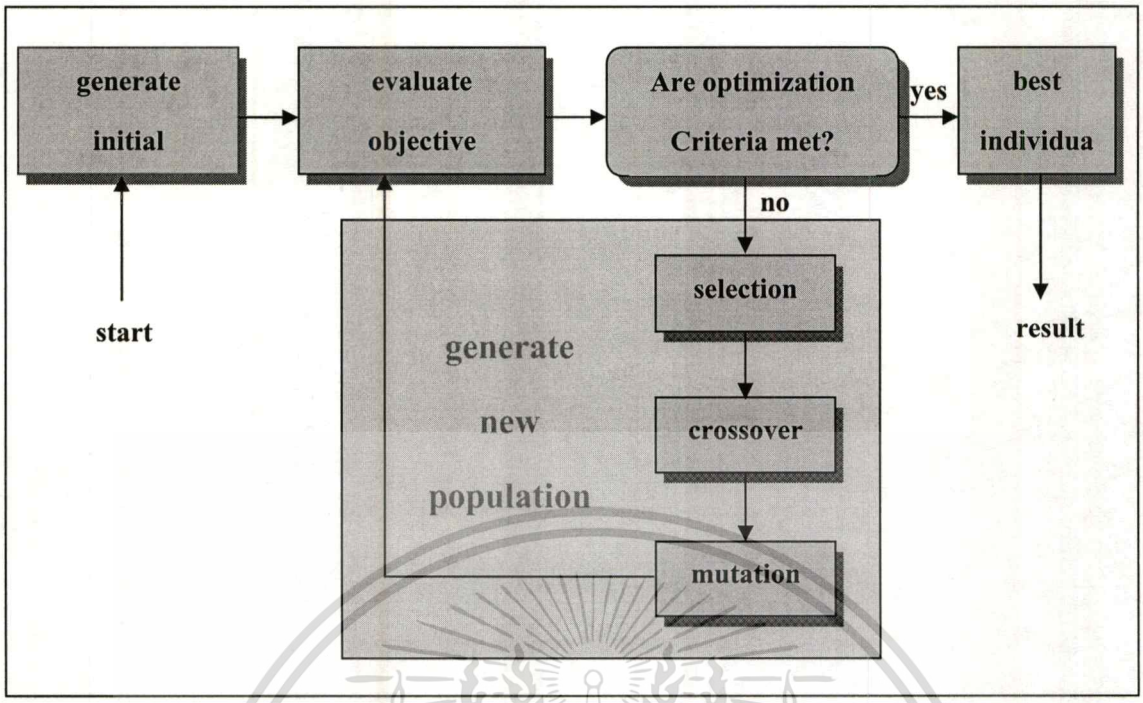
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 การหาคำตอบของปัญหาโดยใช้ Evolutionary Algorithm

จากรูปที่ 2.11 เป็นกระบวนการหาคำตอบที่เหมาะสมที่สุดของปัญหาโดยการใช้ Evolutionary Algorithm สำหรับประชากรในแต่ละรุ่นจะเป็นประชากรที่ถูกสร้างขึ้นมาโดยกระบวนการคัดเลือกโดยดูจากค่าความเหมาะสม ซึ่งแบบจำลอง (model) ของกระบวนการทำงานใน Evolutionary Algorithm นั้น นอกจากจะประกอบไปด้วยกระบวนการคัดเลือกแล้ว ก็มีการดำเนินการแลกเปลี่ยน (crossover) และการกลายพันธุ์ (mutation) เป็นต้น ดังรูปที่ 2.12 ที่เป็นแสดงโครงสร้างของ Genetic Algorithm แบบง่าย ซึ่งเป็นการทำงานของ Evolutionary Algorithm ในการหาคำตอบของประชากรในกลุ่มหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 โครงสร้าง Evolutionary Algorithm ของประชากรภายในกลุ่มหนึ่ง

จากรูปที่ 2.12 เป็นวิธีการดำเนินการหาประชากรที่มีค่าความเหมาะสมที่สุด โดยเริ่มต้นในการคำนวณจะทำการสร้างประชากรต้นกำเนิดโดยวิธีการสุ่มค่าและใช้ฟังก์ชันเป้าหมาย (Object function) ในการกำหนดทิศทางการหาคำตอบเพื่อใช้ในการหาค่าความแข็งแรง จากนั้นถ้าในการหาคำตอบที่เหมาะสมที่สุดนั้นยังไม่ได้ประชากรที่มีค่าตามที่ต้องการ ให้ดำเนินวิธีการ โดยทำการคัดเลือกประชากรเพื่อจะนำมาเป็นประชากรที่มีค่าความเหมาะสมจากรุ่นปัจจุบันถูกส่งไปยังรุ่นถัดไป หลังจากผ่านกระบวนการคัดเลือกแล้ว ประชากรรุ่นใหม่ที่เกิดขึ้นจะมาจากสมาชิกที่มีความเหมาะสมจากรุ่นก่อน ซึ่งความเหมาะสมนี้จะมาจากค่าความแข็งแรงของประชากรแต่ละตัว เมื่อเทียบกับผลรวมของค่าความแข็งแรงของประชากรทั้งหมดในรุ่นก่อน ทำให้ประชากรที่มีค่าความแข็งแรงสูงมีสัดส่วนมากที่จะถูกพิจารณาหรือรับเลือกให้เข้าสู่ในรุ่นถัดไป ในขณะเดียวกันประชากรที่มีค่าความแข็งแรงต่ำก็จะมีสัดส่วนน้อยที่จะถูกพิจารณาให้เข้าสู่ในรุ่นถัดไป

จากนั้นสมาชิกของประชากรที่ผ่านการคัดเลือกมาแล้วสองตัวกำหนดให้เป็นประชากรรุ่นพ่อกับประชากรรุ่นแม่ (Parent individual) จากนั้นทำการแลกเปลี่ยนยีนระหว่างประชากรรุ่นพ่อกับประชากรรุ่นแม่ จะทำให้เกิดเป็นประชากรรุ่นลูก (Offspring Individuals) ประชากรรุ่นนี้จะถูกนำไปเป็นสมาชิกของประชากรรุ่นถัดไป การสุ่มเลือกประชากรรุ่นพ่อกับประชากรรุ่นแม่มาทำการสลับสายพันธุ์ (Crossover) จะถูกกำหนดโดยความน่าจะเป็นในการสลับสายพันธุ์ (Crossover probability)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนมิวเตชั่น(Mutation) หรือการกลายพันธุ์ คือการเปลี่ยนแปลงลักษณะของยีนของประชากรไปจากเดิมที่ควรเป็นไปตามการถ่ายทอด ซึ่งเป็นสาเหตุของการเกิดลักษณะที่แปลกๆ อย่างมากมายของสิ่งมีชีวิต การกลายพันธุ์นั้นทุกลักษณะในแต่ละยีนย่อมมีโอกาสที่จะเกิดการเปลี่ยนแปลงได้พอกัน และถ้าเหมาะสมกับสภาพแวดล้อมก็จะอยู่รอดต่อไป ส่วนถ้าไม่เหมาะสมกับสภาพขณะนั้น ก็จะไม่ถูกคัดเลือกและหายไป

ในการคำนวณค่าความเหมาะสมของประชากรรุ่นถัดไปนั้น จะได้ประชากรรุ่นลูกไปแทนที่ตำแหน่งของประชากรรุ่นพ่อและประชากรรุ่นแม่ ซึ่งเป็นวัฏจักรในการคำนวณซ้ำจนกระทั่งถึงรุ่นสุดท้ายที่กำหนดไว้

ดังนั้นจะเห็นว่า Evolutionary Algorithm จะแตกต่างจากวิธีการค้นหาและหาคำตอบที่เหมาะสมที่สุดแบบวิธีเดิม ซึ่งสิ่งที่แตกต่างกันมีดังนี้ คือ

- การทำงานของ Evolutionary Algorithm นี้ จะเป็นไปในลักษณะของการหาคำตอบแบบคู่ขนาน (Parallel Search) โดยคำตอบที่ได้จากการหาคำตอบในหนึ่งรุ่น (Generation) จะผ่านการแปลง (Transformation) เพื่อที่จะนำไปสู่การค้นหาคำตอบที่ดีขึ้นในรุ่นถัดไป

- Evolutionary Algorithm จะใช้ข้อมูลที่เป็นผลที่เกิดจากค่าเป้าหมาย (Objective Value) ในการกำหนดทิศทางในการหาคำตอบในพื้นที่การค้นหา ซึ่งเทคนิคอื่นจะใช้ข้อมูลที่ได้มาจากอนุพันธ์ (Derivative) ของฟังก์ชันเป้าหมาย (Objective Function) ในการกำหนดทิศทางในการหาคำตอบ

- Evolutionary Algorithm จะใช้กฎการส่งผ่านเชิงความน่าจะเป็น (Probabilistic transmission rule) ในกระบวนการหาคำตอบ ซึ่งเทคนิคอื่นจะใช้กฎการส่งผ่านเชิงกำหนด (Deterministic transmission rule) ในกระบวนการหาคำตอบ

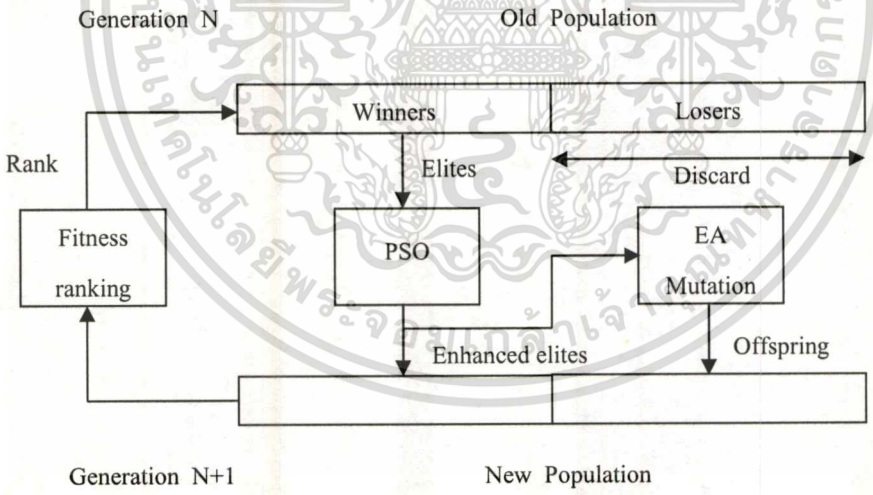
2.5.2 การทำงานร่วมกันของ Particle Swarm Optimization และ Evolutionary Algorithm

PSO นั้น เป็นวิธีที่เลียนแบบพฤติกรรมการอยู่ร่วมกันเป็นกลุ่ม เป็นการปรับค่าความเหมาะสมไปในทิศทางที่เหมือนกัน ซึ่งการปรับปรุงประชากรในกลุ่มนั้นจะใช้วิธีการหาค่าที่เหมาะสมที่สุดที่มีอยู่ในบริเวณสภาพแวดล้อมนั้นๆ ซึ่งต่างจาก Evolutionary Algorithm เนื่องจาก Evolutionary Algorithm ใช้ทฤษฎีวิวัฒนาการจากรุ่นหนึ่งไปยังอีกรุ่นหนึ่ง ดังนั้นการเปลี่ยนแปลงของประชากรในกลุ่มจึงเป็นการเปลี่ยนแปลงตามในแต่ละยุค ซึ่ง Evolutionary Algorithm นั้นเมื่อสิ้นสุดการทำงานในแต่ละยุคแล้วจะไม่มีการเก็บข้อมูลเอาไว้และจะทำการสุ่มประชากรในรุ่นถัดไป ในขณะที่ PSO จะเก็บข้อมูลของตำแหน่งเฉพาะที่ดีที่สุด (Local best) ที่พาติเคิลเคยเจอมาแล้วและตำแหน่งที่ดีที่สุดของทั้งหมดภายในกลุ่ม (Global best) ซึ่งการกลายพันธุ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Mutation) ใน Evolutionary Algorithm จะช่วยสนับสนุน PSO ในส่วนของการทำให้ประชากรสามารถบินเข้าไปสู่พื้นที่สำหรับการค้นหาใหม่ได้

วิธีการที่นำเอา PSO และ Evolutionary Algorithm มาใช้ร่วมกันนั้น ถือว่าเป็นอัลกอริทึมแบบผสมวิธีใหม่ที่เป็นการนำเอาแนวคิดจากสองอัลกอริทึมมาใช้ด้วยกัน โดยนำเอาข้อมูลของประชากรทั้งหมดมาคำนวณหาค่าความเหมาะสม จากนั้นก็ทำการแบ่งข้อมูลที่ได้จากการคำนวณของการหาความเหมาะสมแล้วออกเป็นครึ่งหนึ่งของข้อมูลทั้งหมด ซึ่งข้อมูลจะประกอบด้วยข้อมูลที่มีค่าความเหมาะสมสูง (Winners) จะอยู่ส่วนครึ่งแรกและข้อมูลที่มีความเหมาะสมต่ำ (Losers) จะอยู่ส่วนของครึ่งหลัง โดยที่ประชากรที่มีค่าความเหมาะสมสูงนั้นจะดำเนินการด้วยเทคนิคของ PSO ซึ่งประชากรที่มีค่าความเหมาะสมสูงจะมีการใช้ข้อมูลร่วมกันกับประชากรคนอื่นๆ ซึ่งเป็นประโยชน์อย่างมากเนื่องจากเป็นการเรียนรู้จาก สิ่งที่เคยเกิดขึ้นมาก่อนแล้ว ซึ่งนำมาเปรียบเทียบกับ Evolutionary Algorithm นั้นจะไม่มีการเก็บข้อมูลจากรุ่นก่อน และส่วนประชากรส่วนที่เหลืออีกครึ่งหนึ่งจะเป็นประชากรที่มีค่าความเหมาะสมต่ำนั้น จะถูกตัดทิ้งและถูกแทนที่โดยประชากรรุ่นลูกที่สร้างมาจาก Evolutionary Algorithm ซึ่งเป็นการจะได้รับอิทธิพลที่มีของประชากรพ่อและประชากรแม่จากเทคนิค PSO ซึ่งกระบวนการนำ PSO และ Evolutionary Algorithm มาใช้ผสมร่วมกันนี้สามารถอธิบายได้ ดังรูป ที่ 2.13



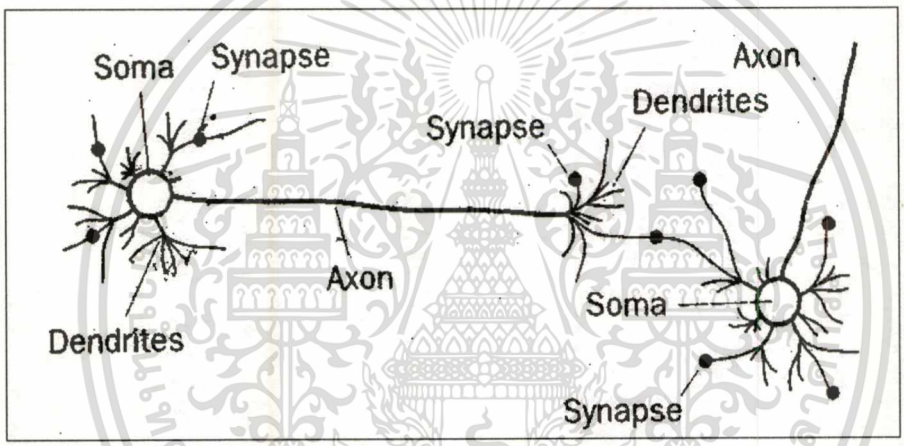
รูปที่ 2.13 การนำอัลกอริทึมผสมแบบ PSO และ EA มาใช้ร่วมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 โครงข่ายประสาทเทียม (Artificial Neural Network)

ในธรรมชาติเซลล์ประสาทของสิ่งมีชีวิตจะประกอบ ด้วยเซลล์จำนวนมาก ซึ่งสิ่งมีชีวิตในที่นี้จะขอกล่าว ถึงเซลล์ประสาทสมองของมนุษย์ โดยปกติสมองของมนุษย์ประกอบ ด้วยเซลล์ประสาทหรือที่เรียกว่า นิวรอล (Neural) จำนวนมากเชื่อมต่อกันเป็นโครงข่าย ดังรูปที่ 2.14 ในแต่ละเซลล์ประกอบ ด้วย 4 ส่วน คือ

- **ตัวเซลล์ (Soma)** ทำหน้าที่เป็นจุดรวมสัญญาณ
- **เดนไดรต์ (Dendrite)** เป็นเส้นใยประสาทจำนวนมาก ซึ่งทำหน้าที่รับสัญญาณอินพุต
- **แอกซอน (Axon)** ทำหน้าที่ส่งสัญญาณเอาต์พุต
- **ซินแนปติก (Synaptic)** ทำหน้าที่เป็นจุดเชื่อมต่อระหว่างเซลล์



รูปที่ 2.14 โครงสร้างเซลล์ประสาทของมนุษย์

โดยการทำงานของเซลล์ประสาทเริ่มจากเดนไดรต์ (Dendrites) ที่ทำหน้าที่รับสัญญาณอินพุตจะรับข้อมูลมาจากช่องว่างซินแนปติก (Synaptic) โดยวิธีการทางเคมี ซึ่งเราจะเรียกว่าเป็นการถ่วงน้ำหนัก (Weight) จากนั้นตัวเซลล์ (Soma) จะทำหน้าที่รวมอินพุตต่างๆ ที่ผ่านการถ่วงน้ำหนักแล้วมาประมวลผลสัญญาณโดยใช้วิธีกระตุ้น ซึ่งเรียกว่าเป็นฟังก์ชันกระตุ้น (Activation function) ของเซลล์ ทำให้ได้เอาต์พุตออกมา และนำเอาต์พุตที่ได้จากตัวเซลล์นี้ส่งเป็นอินพุตให้กับเซลล์ประสาทหรือนิวรอลอีกตัวหนึ่งต่อเนื่องกันไป ซึ่งภายในเวลา 1 วินาที ของร่างกายมนุษย์จะเกิดปฏิกิริยาเช่นนี้นับเป็นล้านๆ ครั้ง

สำหรับในโครงสร้างอาร์ติฟิเชียลนิวรอลเน็ตเวิร์ก (Artificial Neural Network : ANN) หรือเรียกสั้นๆ ว่านิวรอลเน็ตเวิร์กนั้น ก็มีการทำงานในลักษณะเดียวกันกับเซลล์ประสาทของสิ่งมีชีวิต ซึ่งนิวรอลเน็ตเวิร์กเป็นสิ่งประดิษฐ์ที่สร้างขึ้นเพื่อจำลองการทำงานเครือข่ายประสาทของมนุษย์ มีการส่งข้อมูลการเรียนรู้ของระบบเชื่อมต่อกันเป็นโครงข่ายจากอินพุตจนถึงเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

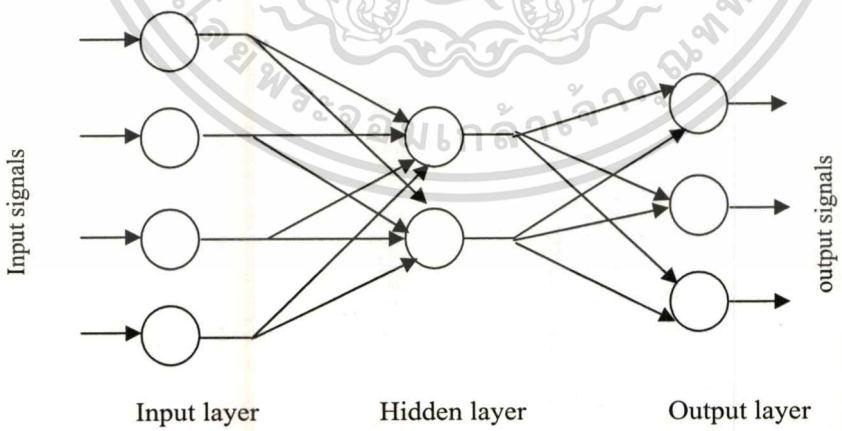
โดยในแบบจำลองของนิวรอลเน็ตเวิร์กนั้นจะมีการเชื่อมต่อกันของข้อมูลในลักษณะที่เป็นลำดับชั้น (Layer) ซึ่งในปัจจุบันนิวรอลเน็ตเวิร์กมีใช้กันหลายรูปแบบ ไม่ว่าจะเป็นด้านการจดจำรูปแบบ (pattern recognition) เช่น จดจำเสียงพูด (Voice recognition) การจดจำตัวอักษร (Character recognition) การประมวลภาพ (Image processing) เป็นต้น ทางด้านระบบควบคุม (Control system) เช่น การควบคุมการเคลื่อนไหวของแขนหุ่นยนต์ เป็นต้น

ตารางที่ 2.3 การเปรียบเทียบระหว่างเซลล์ประสาทของมนุษย์และนิวรอลเน็ตเวิร์ก

เซลล์ประสาทมนุษย์	นิวรอลเน็ตเวิร์ก
Soma	Neuron
Dendrite	Input
Axon	Output
Synapse	Weight

2.6.1 โครงสร้างของนิวรอลเน็ตเวิร์ก

นิวรอลเน็ตเวิร์กประกอบด้วยส่วนประมวลผลต่างๆ ที่เรียกว่า นิวรอน (Neuron) ซึ่งทุกๆ นิวรอนสามารถมีอินพุตหรือข้อมูลที่ป้อนเข้าเพื่อที่จะประมวลผลได้หลายอินพุต แต่มีเอาต์พุต หรือ ผลที่เกิดจากการประมวลผลได้เพียงเอาต์พุตเดียว และทุกๆ เอาต์พุตจะแยกไปเป็นอินพุตของนิวรอนอื่นๆ ภายในโครงข่ายต่อไป



รูปที่ 2.15 โครงสร้างพื้นฐานของนิวรอลเน็ตเวิร์ก

จากรูปที่ 2.15 แสดงให้เห็นว่าโครงสร้างของนิวรอลเน็ตเวิร์กเป็นลักษณะของโครงข่ายที่มีการเชื่อมต่อกันเป็นลำดับชั้น (Layer) ต่างๆ ซึ่งประกอบด้วยชั้นอินพุต (Input layer) ชั้นฮิดเดน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Hidden layer) และชั้นของกลุ่มเอาต์พุต (Output layer) โดยผ่านนิวรอนต่างๆ ตามลำดับชั้นของการเชื่อมต่อ ซึ่งในการเชื่อมต่อกันระหว่างแต่ละนิวรอนจากอินพุตจนถึงเอาต์พุตนั้น จะมีค่าการถ่วงน้ำหนัก (Weight) เฉพาะที่แตกต่างกันไป โดยในระหว่างการสอนให้เน็ตเวิร์กเกิดการเรียนรู้นั้น ค่าถ่วงน้ำหนักเหล่านี้ก็จะถูกปรับเปลี่ยนไปตามค่าที่ได้จากการคำนวณตามอัลกอริทึมที่ใช้จนกระทั่งได้ค่าที่เหมาะสมที่สุด ซึ่งองค์ประกอบที่สำคัญของการพัฒนาแบบจำลองนิวรอลเน็ตเวิร์กมีดังนี้

1. **โครงสร้างของเน็ตเวิร์ก (Network architecture)** ประกอบด้วยจำนวนชั้นภายในเน็ตเวิร์ก จำนวนของโหนดในชั้นที่เป็นอินพุต (Input layer) ชั้นฮิดเดนแต่ละชั้น (Hidden layer) และรวมถึงจำนวนโหนดชั้นเอาต์พุต (Output layer) ที่เชื่อมต่อกันอยู่ในแต่ละชั้นด้วย

2. **ฟังก์ชันกระตุ้น (Activation function)** ที่ใช้คำนวณค่าที่ได้จากแต่ละโหนดภายในเน็ตเวิร์กจากอินพุตจนถึงเอาต์พุต

3. **ฟังก์ชันที่ใช้ทดสอบหาค่าความผิดพลาดของผลลัพธ์ที่ได้ (Cost function)** เป็นการเปรียบเทียบค่าผลลัพธ์ที่คำนวณได้จากแบบจำลองกับค่าข้อมูลเป้าหมายที่เกิดขึ้นจริง

4. **อัลกอริทึมที่ใช้สอนให้เน็ตเวิร์กเกิดการเรียนรู้ (Training algorithm)** โดยมีการกำหนดวิธีการปรับค่าตัวแปรต่างๆ ภายในเน็ตเวิร์ก เพื่อให้ค่าความผิดพลาดที่คำนวณได้จากฟังก์ชันทดสอบความผิดพลาดมีค่าน้อยที่สุด

2.6.2 ประเภทของนิวรอลเน็ตเวิร์ก

นิวรอลเน็ตเวิร์กมี 2 ประเภท คือ โครงสร้างแบบส่งสัญญาณไปข้างหน้า (Feed-forward Neural Network) และโครงสร้างแบบมีการป้อนกลับ (Recurrent Neural Network) ซึ่งลักษณะในการทำงานระหว่างทั้ง 2 โครงสร้างดังกล่าว มีดังนี้

1. โครงสร้างแบบส่งสัญญาณไปข้างหน้า (Feed-forward Neural Network)

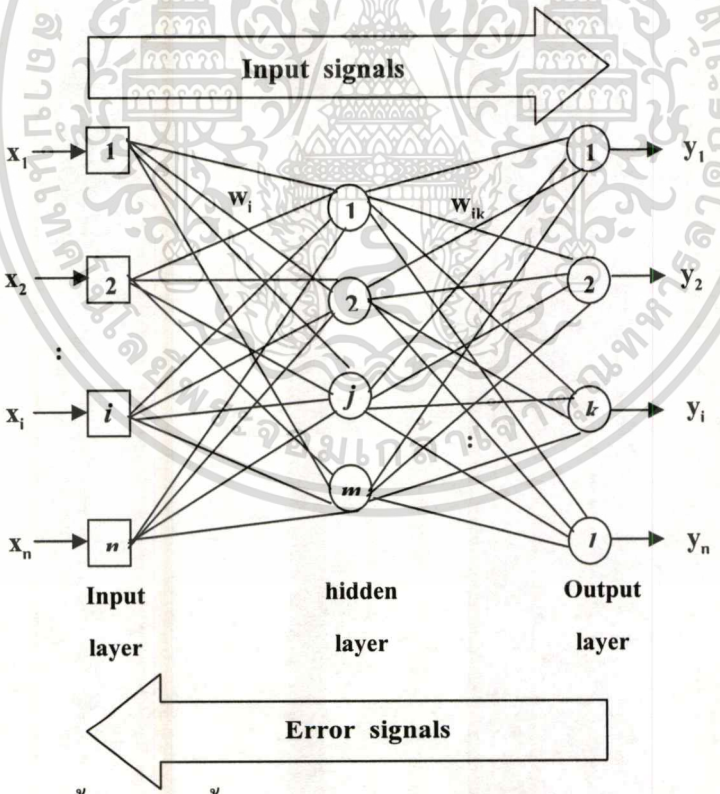
โครงสร้างแบบส่งสัญญาณไปข้างหน้า หรือ Feed-forward Neural Network นั้นมีลักษณะเหมือนกับโครงข่ายประสาทเทียมชนิดอื่นๆ ที่มีสิ่งที่ต้องพิจารณา คือ การเชื่อมต่อกันของนิวรอน ฟังก์ชันกระตุ้นที่ใช้ และการปรับปรุงค่าถ่วงน้ำหนัก ปกติโครงสร้างแบบส่งสัญญาณไปข้างหน้าจะประกอบกันด้วย 3 layer ขึ้นไป ดังรูปที่ 2.16 โดยในการเชื่อมถึงกันจะเชื่อมกันแบบทั่วถึงกัน (Fully connected) คือ ทุกนิวรอนในแต่ละลำดับชั้น (Layer) จะเชื่อมต่อกันกับทุกนิวรอนของนิวรอนในลำดับชั้นถัดไป โดยจะไม่มีการเชื่อมต่อกันภายในลำดับชั้นหรือกระโดดข้ามลำดับชั้นกัน ดังนั้น Input layer จะเชื่อมต่อกับ Hidden layer ที่อยู่ตรงกลาง และ Hidden ก็จะเชื่อมต่อกับ Output layer เสมอ

จากรูปที่ 2.16 กำหนดให้นิวรอน i, j และ k เป็นนิวรอนใน Input layer, Hidden layer และ Output layer ตามลำดับ สัญญาณอินพุต $x_1, x_2, \ldots, x_n$ นั้น จะถูกส่งผ่านเน็ตเวิร์กจากทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้านซ้ายไปยังทางด้านขวาและสัญญาณ error $e_1, e_2, \ldots, e_n$ ที่ได้มาจากการคำนวณของการนำค่าเอาต์พุตเป้าหมายลบด้วยเอาต์พุตที่คำนวณได้จากโครงข่ายและค่า error นี้จะถูกส่งผ่านเน็ตเวิร์กจากทางด้านขวาไปยังด้านซ้ายเพื่อเป็นการปรับค่าความผิดพลาด และ w_{ij} เป็นค่าถ่วงน้ำหนักที่เชื่อมระหว่างนิวรอน i ใน input layerและนิวรอน j ใน hidden layer และw_{jk} เป็นค่าถ่วงน้ำหนักที่เชื่อมระหว่างนิวรอน j ใน hidden layer และนิวรอน k ใน output layer

ซึ่งโครงสร้างแบบนี้จะการทำงานเป็น 2 ขั้นตอน คือ ขั้นแรก เป็นการใส่ข้อมูลอินพุตที่ได้เตรียมไว้ (Input pattern) ให้กับ Input layer ของนิวรอลเน็ตเวิร์ก จากนั้นค่าอินพุตจะถูกนำมาคำนวณโดยจะคำนวณจาก Input layer จนถึง Output layer ตามลำดับ สุดท้ายจะได้ผลลัพธ์ออกมาที่ Output layer ส่วนขั้นตอนที่ 2 เป็นการเปรียบเทียบค่าเอาต์พุตเป้าหมายกับค่าเอาต์พุตที่คาดว่าจะได้จากที่ใส่ค่าอินพุต (Input pattern) เข้าไปในแบบจำลองนิวรอลเน็ตเวิร์ก หากค่าที่ได้ไม่ตรงกันจะหมายถึงมี Error เกิดขึ้น ซึ่งค่า Error จะถูกส่งย้อนกลับมาในเครือข่าย โดยจะถูกส่งกลับจาก Output layer มายัง Input layer และค่า Error ที่ถูกส่งกลับมายังลำดับชั้นต่างๆ มีผลทำให้ค่าถ่วงน้ำหนัก (Weight) ของชั้นนั้นๆ ถูกปรับไปตามค่า Error ที่ถูกส่งกลับมา เพื่อที่ให้ค่า Error ที่จะเกิดต่อไปนั้นมีค่าลดลง



รูปที่ 2.16 โครงสร้างพื้นฐานและขั้นตอนการทำงานของโครงสร้างการส่งสัญญาณไปข้างหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันกระตุ้น (Activation function) ที่นิวรอนใช้ คือ Sigmoid function โดยในการคำนวณนั้นเริ่มคำนวณชั้น Hidden layer โดยที่แต่ละนิวรอนทำการคำนวณค่าอินพุตที่รับเข้ามาคูณกับค่าถ่วงน้ำหนัก ดังสมการ

$$net_j = \sum_{i=1}^{n} x_i w_{ij} + \theta_j \tag{2.3}$$

n คือ จำนวนนิวรอนในชั้นอินพุต

net_j คือ ผลรวมของค่าอินพุต

x_i คือ ค่าอินพุตที่ป้อนเข้าสู่นิวรอนที่ i

w_{ij} คือ ค่าถ่วงน้ำหนักระหว่างของนิวรอนที่ i และ นิวรอนที่ j

θ_j คือ ค่า bias ของโหนด j

จากนั้นค่าอินพุตที่ได้จะถูกส่งไปคำนวณที่ Activation function ซึ่งอยู่ในรูปดังสมการ

$$f(netx_j) = \frac{1}{1 + e^{-netx_j}} \tag{2.4}$$

ซึ่งค่าเอาต์พุตที่ได้จากการคำนวณนี้ จะมีค่าอยู่ในช่วงระหว่าง 0 และ 1

เมื่อได้ค่าเอาต์พุตของ Hidden layer แล้ว จากนั้นจึงทำการคำนวณในชั้น Output layer ต่อไป โดยที่เอาต์พุตของ Hidden layer จะเป็นอินพุตของ Output layer ซึ่งในการทำการคำนวณค่าอินพุตที่รับเข้ามาและการคำนวณ Activation function จะกระทำกันในลักษณะเช่นเดียวกับการคำนวณใน Hidden layer

เมื่อได้ค่าของเอาต์พุตครบทุกๆ นิวรอนแล้ว ก็จะคำนวณค่าความผิดพลาดของเอาต์พุตที่ได้ในทิศทางย้อนกลับ เพื่อปรับค่าถ่วงน้ำหนักสำหรับการเชื่อมต่อระหว่างนิวรอนใดๆ ในเน็ตเวิร์ก รวมทั้งปรับค่า Bias ให้กับแต่ละนิวรอนด้วย

สำหรับนิวรอน k สามารถคำนวณค่าความผิดพลาด (Error) ได้ดังสมการ

$$e_k = \frac{1}{2} \sum \left(y_{d,k}(p) - y_k(p) \right)^2 \tag{2.5}$$

เมื่อ $y_{d,k}(p)$ คือ ค่าเอาต์พุตเป้าหมายที่เกิดขึ้นจริงจากอดีต

$y_k(p)$ คือ ค่าเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์ก

$e_k(p)$ คือ ค่า error ที่ได้จากการนำค่าเอาต์พุตเป้าหมายลบค่าเอาต์พุตที่ได้จากแบบจำลอง

p คือ pattern ที่ p ใน input data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากพบว่าค่าที่คำนวณได้มีความผิดพลาดจากค่าเอาต์พุตจริงที่เราต้องการ ดังนั้นจะต้องทำการปรับค่าถ่วงน้ำหนักเพื่อให้นิวรอลเน็ตเวิร์กหาค่าเอาต์พุตที่เราต้องการ การปรับค่าถ่วงน้ำหนักจะทำการปรับค่าจาก output layer ไปยัง hidden layer เป็นดังสมการ

$$W_{jk}(p+1) = w_{jk}(p) + \Delta w_{jk}(p) \quad (2.6)$$

$\Delta w_{jk}(p)$ คือ ค่าถ่วงน้ำหนักที่ทำการเปลี่ยนแปลง

ซึ่งหาค่า $\Delta w_{jk}(p)$ นี้ได้จากการดังสมการ

$$\Delta w_{jk}(p) = \alpha \times y_j(p) \times \delta_k(p) \quad (2.7)$$

α คือ อัตราการเรียนรู้ (Learning rate)

$\delta_k(p)$ คือ ค่า error gradient ของนิวรอน k ใน Output layer ของ pattern ที่ p

โดยที่ ค่า error gradient เป็นค่า derivative ของฟังก์ชันซิกมอยด์ โดยคูณกับ error ที่นิวรอนเอาต์พุต ซึ่งสามารถแสดงค่า error gradient ได้ดังสมการ

$$\delta_k(p) = y_k(p) \times [1 - y_k(p)] \times e_k(p) \quad (2.8)$$

เมื่อ $$y_k(p) = \frac{1}{1 + e^{-net_k(p)}} \quad (2.9)$$

เมื่อทำการปรับค่าถ่วงน้ำหนักที่ Output layer แล้ว จากนั้นจะทำการปรับค่าถ่วงน้ำหนักที่ Hidden layer ซึ่งใช้สมการเหมือนกันกับ Output layer ดังสมการ

$$\Delta w_{ij}(p) = \alpha \times x_i(p) \times \delta_j(p) \quad (2.10)$$

$\delta_j(p)$ คือ ค่า error gradient ของนิวรอน j ใน Hidden layer

$$\delta_j(p) = y_j(p) \times [1 - y_j(p)] \times \sum_{k=1}^{l} \delta_k(p) w_{jk}(p) \quad (2.11)$$

l คือ จำนวนของนิวรอนใน Output layer

เมื่อ $$Y_j(p) = \frac{1}{1 + e^{-net_j(p)}}$$

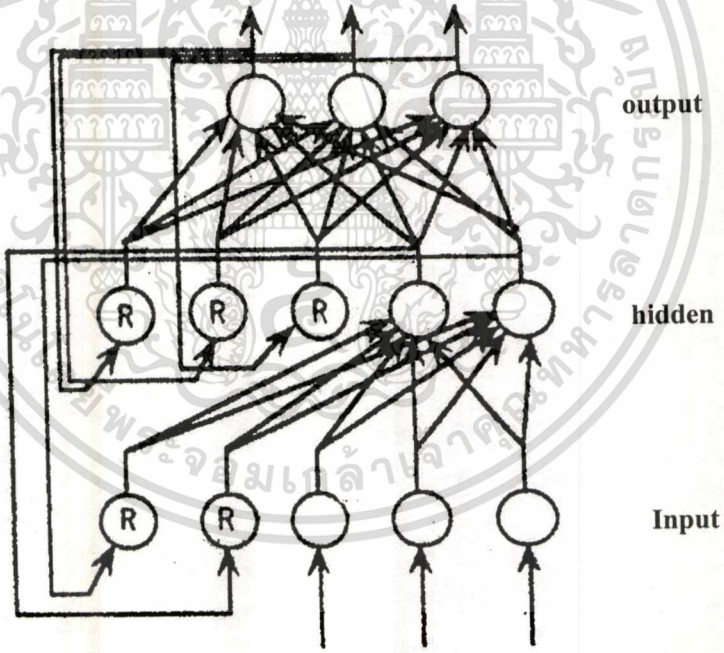
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. โครงสร้างแบบมีการป้อนกลับ (Recurrent Neural Network)

โครงสร้างแบบมีการป้อนกลับนี้เป็นโครงสร้างที่มีลักษณะแตกต่างกับโครงสร้างแบบ Feedforward หรือโครงสร้างแบบส่งสัญญาณไปข้างหน้าตรงที่รูปแบบโครงสร้างของเน็ตเวิร์กแบบนี้นั้นจะมีส่วนที่เพิ่มขึ้นมา คือ มีการป้อนกลับ (Feedback connection) จากบางชั้นของเน็ตเวิร์กมายังชั้นในลำดับก่อนหน้า เพื่อช่วยทำให้เน็ตเวิร์กสามารถเกิดการเรียนรู้และสร้างแบบจำลองที่เหมาะสมกับข้อมูลในลักษณะที่เป็นอนุกรมเวลาได้มากขึ้น

โครงสร้างแบบมีการป้อนกลับหรือรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กนั้น มีการนำค่าเอาต์พุตที่ได้จากนิวรอนใดๆ นั้นมาย้อนกลับเป็นค่าอินพุตให้กับเครือข่ายได้อีก ซึ่งส่งผลให้เครือข่ายประเภทนี้เสมือนมีความทรงจำระยะสั้นเกี่ยวกับเหตุการณ์ที่เกิดขึ้นในอดีตที่จะส่งผลกับเหตุการณ์ที่จะเกิดขึ้นในอนาคตหรือผลของการพยากรณ์เหตุการณ์ต่อไปของเครือข่ายนั่นเอง ซึ่งการพยากรณ์ข้อมูลแบบป้อนกลับจะมีประสิทธิภาพในการพยากรณ์ข้อมูลมากกว่าเครือข่ายแบบส่งสัญญาณไปข้างหน้าด้วยเช่นกัน

วิธีการสร้างเครือข่ายแบบมีการป้อนกลับโดยทั่วๆไปนั้น คือ การนำโครงสร้างแบบส่งสัญญาณไปข้างหน้ามาทำการปรับปรุง ซึ่งจะเป็นดังรูปที่ 2.17



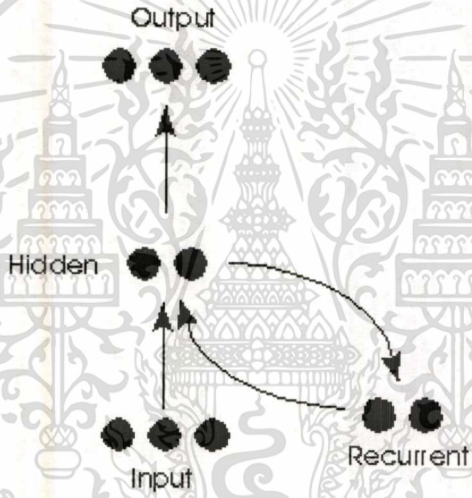
รูปที่ 2.17 โครงสร้างพื้นฐานและขั้นตอนการทำงานของโครงสร้างการส่งสัญญาณป้อนกลับ

จากรูปที่ 2.17 แสดงให้เห็นว่าเป็นรูปของเครือข่ายแบบส่งสัญญาณไปข้างหน้าขนาดเล็ก ซึ่งประกอบด้วยนิวรอนที่อยู่ใน Input layer 3 นิวรอน นิวรอนที่อยู่ใน Hidden layer จำนวน 2 นิวรอนและนิวรอนที่อยู่ใน Output layer จำนวน 3 นิวรอน โดยที่ Input layer นั้นจะมีรีเคอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เร็นท์นิวรอล (Recurrent neural) เพิ่มขึ้นมา 2 นิวรอน ซึ่งใช้สัญลักษณ์แทนด้วย R โดยที่แต่ละนิวรอน R จะเป็นเอาต์พุตของนิวรอนใน Hidden layer และลักษณะเดียวกันกับใน Hidden layer นั้นจะมีรีเคอร์เร็นท์นิวรอล (Recurrent neural) จาก Output layer เพิ่มขึ้นมา 3 นิวรอน จะเห็นว่าเครือข่ายแบบป้อนกลับนั้นจะมีการเชื่อมโยงกันของนิวรอนมากกว่าในเครือข่ายแบบส่งสัญญาณไปข้างหน้า

สำหรับเอกสารฉบับนี้จะใช้เครือข่ายแบบมีการป้อนกลับที่มีโครงสร้างแบบ Elman ซึ่งอัลกอริทึมที่ใช้ในการสอนให้เน็ตเวิร์กเกิดการเรียนรู้นั้นใช้มาตรฐานอัลกอริทึมของ Backpropagation โดยที่มีการปรับค่าความผิดพลาดและค่าถ่วงน้ำหนักตามแต่ละช่วงเวลาที่ทำการป้อนค่าข้อมูลอินพุต เข้ามา ซึ่งโครงสร้างนี้เป็นที่ยอมรับโดยทั่วไปว่าเป็นโครงสร้างแบบหนึ่งที่นำไปประยุกต์ ใช้งานได้ง่ายและมีประสิทธิภาพ ดังรูปที่ 2.18



รูปที่ 2.18 สถาปัตยกรรมของรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กแบบ Elman

จากรูปที่ 2.18 โครงสร้างแบบ Elman จะเป็นโครงสร้างที่มีการป้อนกลับบางส่วน (Partial recurrent network) ประเภทหนึ่งที่ประกอบด้วย Hidden layer 1 ชั้น โดยที่เอาต์พุตของ Hidden layer นี้จะถูกป้อนกลับมาเป็นอินพุตให้กับชั้นนั้นๆ โดยผ่าน Context layer โดยนิวรอนที่เป็นรีเคอร์เร็นท์นั้นจะถูกเก็บอยู่ใน Context layer ซึ่งเป็น layer ที่ถูกเพิ่มเข้ามาและอยู่ขนานระดับเดียวกับ Input layer และแต่ละนิวรอนใน Hidden layer จะต่อกับนิวรอน 1 นิวรอนใน Context layer สำหรับ Context layer นี้เป็นส่วนประกอบของเน็ตเวิร์กที่บันทึกและจดจำสถานะของ Hidden layer จากการคำนวณในครั้งที่ผ่านมา ดังนั้นจำนวนนิวรอนใน Context layer จะเท่ากับจำนวนนิวรอนใน Hidden layer

โดยทั่วๆ ไปแล้วการสอนให้เน็ตเวิร์กเกิดการเรียนรู้นั้น เครือข่ายจะต้องทำการสอนหลายๆ รอบ และในกรณีที่เป็นเครือข่ายแบบป้อนกลับนี้ เมื่อจะใส่ข้อมูลอินพุตเข้าสู่เน็ตเวิร์กใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

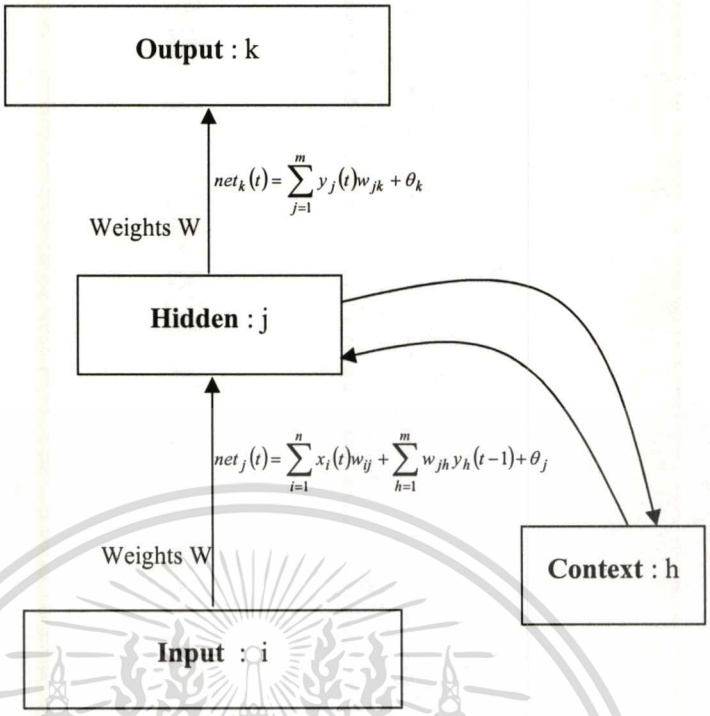
แต่ละช่วงเวลานั้น แต่ละนิวรอนในเน็ตเวิร์กจะทำการคำนวณค่าตามฟังก์ชันกระตุ้นในลักษณะเช่นเดียวกับเครือข่ายแบบการส่งสัญญาณไปข้างหน้ามาใช้ในการสอนให้เน็ตเวิร์กเกิดการเรียนรู้ และจะทำการคำนวณเพิ่มเติมในส่วนของเน็ตเวิร์กตรงที่มีการนำนิวรอนที่เป็นรีเคอร์เร็นท์นิวรอลเพิ่มเข้ามาคำนวณด้วย ก็จะยิ่งทำให้การทำงานของเครือข่ายมีความซับซ้อนมากขึ้นเช่นกัน ในการสอนให้เน็ตเวิร์กเกิดการเรียนรู้นั้นสามารถทำการสอนได้ คือ เมื่อเราใส่ค่าอินพุต (Input pattern) ต่างๆ ให้กับเครือข่ายเป็นจำนวน n ครั้ง การทำงานของเครือข่ายก็จะเป็นลักษณะดังนี้

ณ เวลา t = 1 ข้อมูลอินพุต (Input pattern) ชุดแรกที่ถูกใส่เข้ามาที่ Hidden layer และ Output layer จะต้องคำนวณ ค่าฟังก์ชันกระตุ้นของ Input layer โดยจะต้องคำนวณค่าจากรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กด้วย ต้องกำหนดค่าเริ่มต้นของรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กเอาไว้ก่อน ในการคำนวณนั้น Hidden layer จะเริ่มคำนวณเมื่อได้รับค่าอินพุตเข้ามาจากนั้นค่าอินพุตที่ได้จะถูกทำการคำนวณด้วยฟังก์ชันกระตุ้นจะได้ค่าเอาต์พุตออกมาและนำค่าเอาต์พุตที่ได้ไปเก็บไว้ที่ Context layer ต่อจากนั้น Output layer ก็ทำการคำนวณค่าอินพุตที่รับเข้ามาจาก Hidden layer และคำนวณค่ากระตุ้นด้วยวิธีเดียวกันกับ Hidden layer เมื่อได้ค่าเอาต์พุตที่ได้จากแบบจำลองเครือข่ายของข้อมูลอินพุต (Input pattern) ที่ใส่เข้าไปในรอบนี้ออกมาได้ค่าเอาต์พุตแล้ว จึงทำการเปรียบเทียบค่าเอาต์พุตเป้าหมายกับเอาต์พุตที่ได้จากเครือข่าย ถ้าพบว่ามีค่าความผิดพลาดจึงทำการปรับค่าถ่วงน้ำหนัก เพื่อให้นิวรอลเน็ตเวิร์กหาค่าเอาต์พุตที่เราต้องการได้

ณ เวลา t = 2 ข้อมูลอินพุต (Input pattern) ลำดับต่อมาจะถูกใส่ให้กับเครือข่าย โดยที่ Hidden layer และ Output layer จะคำนวณค่าฟังก์ชันกระตุ้นโดยที่จะใช้ค่ากระตุ้นที่ได้จากการทำงานที่เวลา t = 1 มาเป็นค่าของรีเคอร์เร็นท์นิวรอลเน็ตเวิร์ก ที่เพิ่มขึ้นมาและนำมาคำนวณค่าร่วมกับข้อมูลอินพุตในปัจจุบัน จากนั้นก็ทำการเปรียบเทียบและปรับปรุงค่าถ่วงน้ำหนักในลักษณะเช่นเดียวกัน

ซึ่งการทำงานของเครือข่ายแบบป้อนกลับนี้จะดำเนินไปลักษณะเช่นเดียวกันกับเมื่อเวลา t = 2 จนกระทั่งครบจำนวน n ครั้ง ซึ่งค่าของรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กจะเปลี่ยนไปในแต่ละรอบ คือ เป็นค่ากระตุ้นที่ได้จากเวลามาคำนวณร่วมกับข้อมูลอินพุตที่ใส่เข้าในเครือข่าย อย่างไรก็ตามค่า ถ่วงน้ำหนักของการป้อนกลับ (Recurrent neural) จะถูกกำหนดเป็นค่าที่แน่นอนไว้ ซึ่งมีค่าเท่ากับ 1 โดยมีค่าถ่วงน้ำหนักของการเชื่อมต่อแบบการส่งสัญญาณไปข้างหน้าเท่านั้น ที่ได้รับการปรับเปลี่ยนในช่วงที่มีการเรียนรู้ ซึ่งจะใช้อัลกอริทึมแบบ Backpropagation ตามวิธีการที่ได้อธิบายไว้ข้างต้นในการปรับปรุงเครือข่ายของเน็ตเวิร์กเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 simple recurrent neural network

ณ เวลา t ใดๆ จะมีการคำนวณค่าอินพุตของนิวรอน โดยในการคำนวณนั้นเริ่มคำนวณ Hidden layer โดยที่แต่ละนิวรอนทำการคำนวณค่าอินพุตที่รับเข้ามา ดังสมการ

$$net_j(t) = \sum_{i=1}^{n} x_i(t) w_{ij} + \sum_{h=1}^{m} w_{jh} y_h(t-1) + \theta_j \qquad (2.12)$$

เมื่อ

n คือ จำนวนนิวรอนของ Input layer

m คือ จำนวนนิวรอนของ Hidden layer

h คือ รีเคอร์เร็นท์นิวรอนของ Hidden layer

$y_h(t-1)$ คือ ค่า activity จากรอบที่แล้วซึ่งเป็นค่า Sigmoid ของ Hidden layer

จากนั้นค่าอินพุตที่ได้จะถูกส่งไปคำนวณที่ Activation function ซึ่งอยู่ในรูปดังสมการ

$$y_j(t) = f(net_j(t))$$

ซึ่ง
$$f(net_j(t)) = \frac{1}{1 + e^{-net_j(t)}} \qquad (2.13)$$

เมื่อ $y_j(t)$ คือ ค่าเอาต์พุตของนิวรอน j ที่เวลา t

จากนั้นชั้น Output layer ทำการคำนวณค่าอินพุตที่รับเข้ามา ดังสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$net_k(t) = \sum_{j=1}^{m} y_j(t) w_{jk} + \theta_k \tag{2.14}$$

ส่วนการคำนวณค่า Activation function จะคำนวณเหมือนกันกับชั้น Hidden layer ดังสมการ

$$y_k(t) = g(net_k(t)) \tag{2.15}$$

หลังจากที่เราได้ค่าเอาต์พุตครบทุกๆ นิวรอนแล้วจากนั้นทำการคำนวณค่าความผิดพลาดที่ได้ในลักษณะเช่นเดียวกับอัลกอริทึมแบบ Backpropagation ของโครงสร้างแบบส่งสัญญาณไปข้างหน้า เพื่อปรับค่าถ่วงน้ำหนักสำหรับการเชื่อมต่อระหว่างนิวรอนใดๆ ในเน็ตเวิร์ก

สำหรับนิวรอน k สามารถคำนวณค่าความผิดพลาด (error) ได้ดังสมการ

$$e_k(t) = \frac{1}{2} \sum \left(y_{d,k}(t) - y_k(t) \right)^2 \tag{2.16}$$

หากพบว่าค่าที่คำนวณได้มีความผิดพลาดจากค่าเอาต์พุตจริงที่เราต้องการ ดังนั้นจะต้องทำการปรับค่าถ่วงน้ำหนักเพื่อให้นิวรอลเน็ตเวิร์กหาค่าเอาต์พุตที่เราต้องการ การปรับค่าถ่วงน้ำหนักจะทำการปรับค่าจาก Output layer ไปยัง Hidden layer เช่นเดียวกับอัลกอรึทึมแบบ Backpropagation ดังนี้

$$W_{jk}(t+1) = w_{jk}(t) + \Delta w_{jk}(t) \tag{2.17}$$

ซึ่งค่า $\Delta w_{jk}(t)$ นี้หาได้ดังสมการ

$$\Delta w_{jk}(t) = \alpha \times y_j(t) \times \delta_k(t) \tag{2.18}$$

ในส่วนของการคำนวณค่า error gradient ของค่าถ่วงน้ำหนักแต่ละค่านั้น จะทำการคำนวณจาก Output layer ที่เวลา $t = t_1$ ซึ่งสามารถคำนวณได้ดังสมการ

$$\delta_k(t) = y_k(t_1) \times [1 - y_k(t_1)] \times e_k(t_1) \tag{2.19}$$

เมื่อทำการปรับค่าถ่วงน้ำหนักที่ output layer แล้ว จากนั้นทำการปรับค่าถ่วงน้ำหนักที่ hidden layer ซึ่งใช้สมการเหมือนกันกับอัลกอริทึมแบบ backpropagation สำหรับการคำนวณค่า error gradient คำนวณดังสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\delta_j(\tau) = y_j(\tau) \times [1 - y_j(\tau)] \times \left(e_k(\tau) + \sum_{k=1}^{l} w_{jk}\delta_j(\tau+1)\right) \quad (2.20)$$

2.7 ความผิดพลาดของการพยากรณ์

ในการพยากรณ์นั้น ความถูกต้องของการพยากรณ์เป็นสิ่งสำคัญสำหรับผู้พยากรณ์ โดยความถูกต้องขึ้นอยู่กับผลต่างของค่าความผิดพลาดที่เกิดขึ้นในการพยากรณ์ ซึ่งเกิดจากค่าเป้าหมายลบด้วยค่าที่ได้จากการพยากรณ์ ซึ่งถ้าผลต่างนี้ไม่มีความต่างกันมากนัก ก็ถือเป็นการพยากรณ์ที่สามารถควบคุมได้ คือ ตัวแบบ(Model) นั้นมีความเหมาะสมกับชุดข้อมูลนั้นๆ แต่ถ้าผลต่างที่ได้จากการพยากรณ์มีความแตกต่างกันมาก แสดงว่าการพยากรณ์นั้นไม่สามารถควบคุมได้ แสดงถึงว่าตัวแบบที่ใช้ในการพยากรณ์นั้นไม่เหมาะสม จึงควรทำการปรับเปลี่ยนตัวแบบพยากรณ์นั้นให้เหมาะสมก่อนที่จะนำค่าพยากรณ์นั้นไปใช้งาน ซึ่งค่าสถิติที่ใช้วัดความถูกต้องของการพยากรณ์นั้น มีดังนี้

1. Sum Square Error (SSE) เป็นค่าวัดความถูกต้องของการพยากรณ์ที่วัดจากค่าความคลาดเคลื่อนในแต่ละครั้งยกกำลังสอง

$$SSE = \left(\sum_{t=1}^{n} e_t^2\right) \quad (2.21)$$

2. Mean Square Error (MSE) เป็นค่าวัดความถูกต้องของการพยากรณ์ที่วัดจากค่าความคลาดเคลื่อน ค่า MSE จะไวต่อความคลาดเคลื่อนที่มีขนาดใหญ่ เพราะได้จากการนำค่าความคลาดเคลื่อนแต่ละค่ามายกกำลังสอง

$$MSE = \left(\sum_{t=1}^{n} e_t^2\right) / n \quad (2.22)$$

3. Root Mean Square Error (RMSE) เป็นค่าเบี่ยงเบนมาตรฐานของค่าความคลาดเคลื่อนกำลังสองเฉลี่ย

$$RMSE = \sqrt{\left(\frac{1}{n}\sum_{t=1}^{n} e_t^2\right)} \quad (2.23)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีการดำเนินการศึกษา

การนำเอาเทคนิควิธีการของ Particle Swarm Optimization มาทำงานผสมร่วมกับเทคนิค Evolutionary Algorithm นั้นถือเป็นวิธีที่มีประสิทธิภาพมากอย่างหนึ่ง ซึ่งทั้งสองเทคนิคนี้เป็นเทคนิคที่มีลักษณะคล้ายกันคือ การหาคำตอบหรือค่าที่เหมาะสมที่สุด ดังนั้นจึงได้นำเทคนิคนี้มาใช้ทำงานร่วมกับการทำนายข้อมูลอนุกรมเวลาโดยใช้หลักการของรีเคอร์เร็นท์นิวรอลเน็ตเวิร์ก เพื่อให้การทำนายมีความถูกต้องมากยิ่งขึ้น

3.1 โครงสร้างการทำงานของระบบ

โครงสร้างของแบบจำลองการทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์ก ร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA นั้น เป็นแบบจำลองที่ใช้สำหรับการพยากรณ์ข้อมูลที่จะเกิดขึ้นในอนาคต ซึ่งในโครงสร้างนี้จะประกอบไปด้วยการทำงานส่วน 2 ส่วน คือ ส่วนแรกเป็นส่วนที่นำมาใช้งานแทนฟังก์ชั่นในการพยากรณ์ ซึ่งส่วนนี้เป็นส่วนของโครงข่ายประสาทเทียม และส่วนที่สองเป็นส่วนที่นำเอาอัลกอริทึมของ PSO และ EA มาใช้ เพื่อทำการหาค่าที่เหมาะสมที่สุด ซึ่งนำทั้งสองอัลกอริทึมนี้มาเพื่อใช้สำหรับปรับและค้นหาค่าถ่วงน้ำหนักที่เหมาะสมที่สุดเพื่อไปใช้ในการพยากรณ์กับโครงข่ายประสาทเทียมที่ได้จำลองขึ้นมา

ในส่วนของโครงข่ายประสาทเทียมแบบรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กที่ใช้ในโครงงานนี้ประกอบด้วยโครงสร้างการทำงานที่แบ่งเป็นจำนวนชั้นการทำงานทั้งหมด 4 ชั้น ดังนี้

1. ชั้นอินพุต (Input layer) จำนวนโหนดในชั้นนี้ ที่ผู้ใช้งานสามารถกำหนดได้ตามต้องการ

2. ชั้นคอนเท็กซ์ (Context layer) จำนวนโหนดของชั้นนี้จะต้องเท่ากับจำนวนโหนดของ Hidden layer เนื่องจากเป็นการนำค่าเอาท์พุตที่ได้จาก hidden layer มาเป็นค่าอินพุตของโครงข่ายประสาทเทียม

3. ชั้นฮิดเดน (Hidden layer) จำนวนโหนดในชั้นนี้ ผู้ใช้งานสามารถกำหนดได้ตามต้องการ

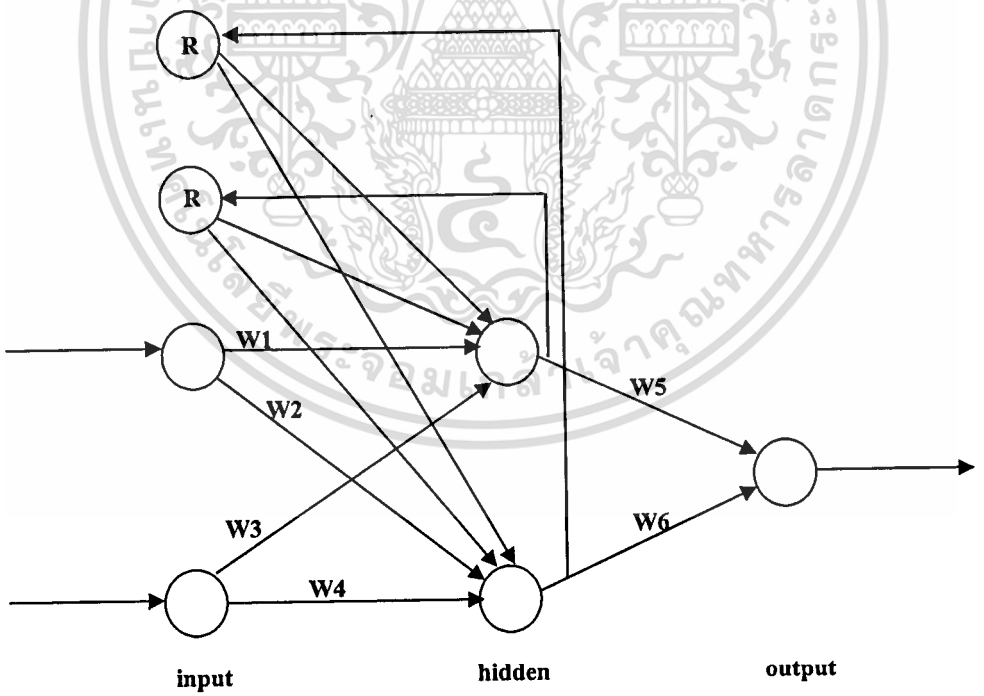
4. ชั้นเอาท์พุต (Output layer) มีจำนวนเอาท์พุตเพียง 1 โหนด เพราะมีค่าเอาท์พุตเพียงค่าเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการทำงานส่วนของ PSO นั้น ทำงานโดยเริ่มจากที่ให้พาติเคิลหนึ่งพาติเคิลทำงานโดยใช้โครงข่ายประสาทเทียมเป็นส่วนของการค้นหาค่าความเหมาะสมที่สุด ซึ่งโครงข่ายประสาทเทียมหนึ่งชุดจะทำงานต่อการค้นหาของพาติเคิลของหนึ่งตัว สำหรับการสร้างพาติเคิลแต่ละตัวนั้นเราจะต้องสร้างพาติเคิลแต่ละตัวให้มีจำนวน Unknown Dimensions เท่ากับจำนวนค่าถ่วงน้ำหนักทั้งหมดที่มีอยู่ในโครงข่ายประสาทเทียม แต่จะไม่คิดส่วนค่าถ่วงน้ำหนักของนิวรอนที่เป็นรีเคอร์เร็นท์ ตัวอย่างเช่น ถ้าโครงข่ายประสาทเทียมของ Input layer มี 2 นิวรอน Hidden layer มีจำนวน 2 นิวรอน และ output layer มีจำนวน 1 นิวรอน จะมีจำนวนของ Unknown Dimensions ดังนี้

Unknown Dimensions = (จำนวนนิวรอนของ input x จำนวนนิวรอลของ hidden) +
(จำนวนนิวรอลของ hidden x จำนวนนิวรอลของ output)
= (2 x 2) + (2 x 1)
= 6 ตัวแปร

แสดงดังรูปที่ 3.1 จะเห็นว่าในโครงข่ายมีค่าถ่วงน้ำหนัก 6 ค่า ซึ่งเท่ากับค่า Unknown Dimensions ที่ได้คำนวณมา

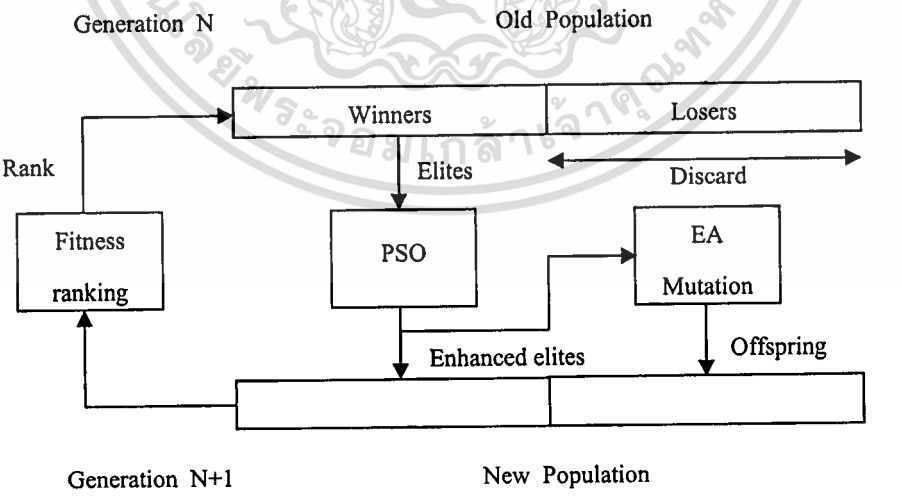


รูปที่ 3.1 การคำนวณหาจำนวน Unknown Dimensions จากโครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1 นั้นค่าถ่วงน้ำหนักของรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กนั้น จะถูกกำหนดให้เป็นค่าที่แน่นอนไว้ ซึ่งมีค่าเท่ากับ 1 โดยมีค่าถ่วงน้ำหนักของการเชื่อมต่อแบบส่งสัญญาณไปข้างหน้าเท่านั้น ที่ได้รับการปรับเปลี่ยนในช่วงที่มีการเรียนรู้

สำหรับการทำงานของ Evolutionary Algorithm (EA) นั้น จะเริ่มต้นทำงานที่วิธีการดำเนินงานในส่วนของการกลายพันธุ์ (Mutation) ซึ่งในการทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA เริ่มจากทำการรับค่าของจำนวน Particle (ในที่นี้ Particle จะลักษณะเช่นเดียวกันกับ Population) และต่อจากนั้นทำการคำนวณค่าความเหมาะสม (Fitness value) ของแต่ละ Particle เมื่อได้ค่าความเหมาะสมแล้วก็ให้แบ่งครึ่งจำนวน Particle เช่น Particle ที่รับมาจำนวน 10 Particle หลังจากทำการคำนวณหาค่าความเหมาะสมแล้ว ให้แบ่งจำนวน Particle ออกเป็น 2 ส่วน ในที่นี้จะได้ส่วนละ 5 Particle เป็นต้น ซึ่งจะกำหนดค่าความเหมาะสมที่มีค่าสูงไว้ในส่วนแรกและค่าความเหมาะสมที่ต่ำจะอยู่ในส่วนที่สอง หลังจากนั้นให้ลบส่วนที่มีค่าความเหมาะสมต่ำออกไปและนำส่วนแรกที่มีค่าความเหมาะสมสูงมาคำนวณโดยวิธี PSO เมื่อคำนวณด้วยวิธี PSO แล้วจะได้ Particle ที่มีค่าความเหมาะสมที่สุด ซึ่งจากตัวอย่าง คือ Particle ทั้ง 5 ตัวที่ได้ทำการปรับตำแหน่งแล้ว โดยนำ Particle ทั้งหมดนี้ส่งไปยัง EA เพื่อทำการดำเนินงานในส่วนของการกลายพันธุ์ (Mutation) ทำให้ได้ Particle ใหม่สำหรับในส่วนของ EA ออกมา ดังนั้น Particle หรือ Population รุ่นใหม่จะมีจำนวนเท่ากับก่อนทำการแบ่งออกเป็นครึ่งหนึ่ง จากตัวอย่างก็คือจะมีจำนวน Particle เป็น 10 ตัว ซึ่งขั้นตอนของการนำอัลกอริทึม PSO มาผสมกับ Evolutionary algorithm แสดงได้ดังรูปที่ 3.2



รูปที่ 3.2 ขั้นตอนการทำงานของอัลกอริทึมแบบผสมของ PSO และ EA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทำงานของระบบนั้นมีจุดประสงค์หลัก คือ ทำการค้นหาค่าถ่วงน้ำหนักทั้งชุดนี้ โดยใช้อัลกอริทึมของ PSO ผสมกับอัลกอริทึมของ EA ซึ่งเราจะต้องทำการใส่ค่าข้อมูลอินพุตและข้อมูลเอาท์พุตให้กับโครงข่ายประสาทเทียม และทำการสุ่มข้อมูลภายในรีเดอร์เร็นทนิวรอลด้วย เพื่อให้โครงข่ายประสาทเทียมนี้สามารถทำการคำนวณค่าความคลาดเคลื่อนที่เกิดขึ้นได้เพื่อคำนวณหาค่าความเหมาะสมอื่นต่อไป ค่าความคลาดเคลื่อนที่ใช้ในการคำนวณจะใช้การคำนวณค่าความคลาดเคลื่อนแบบ SSE เมื่อได้ค่าถ่วงน้ำหนัก ชุดนั้นออกมาจะต้องพิจารณาว่าค่าถ่วงน้ำหนักชุดนั้นเหมาะสมแล้วหรือยัง โดยพิจารณาจากค่าความเหมาะสม ถ้ายังเป็นค่าที่ยังไม่ดีพอ PSO ก็จะต้องทำการย้ายตำแหน่งของพาติเคิลของมันตามค่า pbest และ gbest ของชุดพาติเคิลชุดนั้นเพื่อทำการค้นหาค่าถ่วงน้ำหนักในโครงข่ายประสาทเทียมที่เราฝึกสอนต่อไป

3.2 ขั้นตอนการทำงานของระบบ

การพัฒนาระบบการทำนายอนุกรมเวลาโดยใช้หลักการรีเดอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA มีลักษณะการทำงาน คือ ให้อัลกอริทึม PSO และ EA ทำการค้นหาค่าถ่วงน้ำหนักที่มีค่าที่เหมาะสมที่สุด เพื่อให้การทำนายเกิดค่าความคลาดเคลื่อนให้น้อยที่สุด ซึ่งขั้นตอนการทำงานหลักๆ ที่สำคัญ ประกอบด้วย

1. การเลือกข้อมูลสำหรับที่จะนำมาใช้กับระบบ
2. การกำหนดค่าตัวแปรให้กับระบบ
3. การฝึกหัดโครงข่ายประสาทเทียมของระบบ
4. การทดสอบโครงข่ายประสาทเทียมของระบบ

3.2.1 การเลือกข้อมูลสำหรับที่จะนำมาใช้กับระบบ

ในการเลือกข้อมูลนั้นเป็นส่วนที่มีผลต่อการพยากรณ์ของระบบ การเลือกข้อมูลมีสิ่งที่ต้องพิจารณาอยู่หลายประเด็นด้วยกัน ดังนี้

1. ชนิดข้อมูลที่นำมาใช้นั้น พิจารณาว่าชนิดข้อมูลนั้นมีนัยสำคัญหรือไม่ ข้อมูลที่ไม่มีนัยสำคัญ เช่น เบอร์โทรศัพท์ บ้านเลขที่ เป็นต้น

2. ปริมาณของข้อมูล ในการฝึกโครงข่ายเทียมให้มีการเรียนรู้นั้นถ้าหากจำนวนของข้อมูลมีน้อยเกินไป จะทำให้เกิดความผิดพลาดขึ้นได้สูงเนื่องจากโครงข่ายประสาทเทียมเกิดการเรียนรู้น้อย

3. เมื่อได้ข้อมูลมาแล้ว ต้องมีการปรับค่าข้อมูลโดยการนอร์มอลไลเซชัน (Normalization) ซึ่งเป็นการนำข้อมูลที่ได้จากขั้นตอนที่ 1 มาปรับเปลี่ยนให้ข้อมูลมีค่าอยู่ในช่วงระหว่าง 0 ถึง 1 เพื่อให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลสามารถนำไปใช้งานกับอัลกอริทึมได้ และให้สอดคล้องกับฟังก์ชันที่นำมาใช้ ซึ่งในที่นี้คือ ซิกมอยด์ฟังก์ชัน (Sigmoid Function) โดยสูตรที่ใช้ในการนอร์มอลไลเซชัน คือ

$$newdata = (data - newmin) / (newmax - newmin)$$

โดยที่

$$newmin = min - 10\%$$

$$newmax = max + 10\%$$

เมื่อ

data คือ ข้อมูลที่ต้องการนอร์มอลไลเซชัน

max คือ ข้อมูลที่มีค่าสูงสุด

min คือ ข้อมูลที่มีค่าต่ำสุด

newmin คือ ข้อมูลที่มีค่าต่ำสุดที่ถูกปรับแล้ว

newmax คือ ข้อมูลที่มีค่าสูงสุดที่ถูกปรับแล้ว

newdata คือ ข้อมูลที่ทำการนอร์มอลไลเซชันแล้ว

ซึ่งหลังจากนอร์มอลไลเซชันแล้ว จะทำให้ได้ข้อมูลภายในชุดนั้นมีค่าอยู่ระหว่าง 0 ถึง 1 จึงจะสามารถข้อมูลนั้นมาใช้งานสำหรับการการฝึกสอนโครงข่ายประสาทเทียมของระบบได้ และหลังจากที่เราได้ผลในการทำนายแล้ว ถ้าเราต้องการแปลงค่าที่เหมาะสมได้จาก 0 ถึง 1 ให้เป็นค่าจริง เราก็ต้องทำการแปลงกลับข้อมูลได้จากสมการ

$$data = newdata \times (newmax - newmin) + newmin$$

3.2.2 การกำหนดค่าตัวแปรให้กับระบบ

สำหรับการกำหนดค่าตัวแปรเบื้องต้นให้กับระบบนั้น จะประกอบด้วยขั้นตอนดังนี้

1. กำหนดค่าจำนวนพาติเคิลที่จะใช้
2. กำหนดค่าจำนวนรอบที่จะทำการคำนวณหาค่าความเหมาะสม
3. กำหนดความเร็วสูงสุดให้กับพาติเคิล
4. กำหนดลักษณะของโครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อรู้ลำดับขั้นตอนในการกำหนดค่าตัวแปรให้กับระบบแล้ว ในที่นี่จะทำการสมมุติตัวอย่างขึ้นเมื่อทำการฝึกหัดโครงข่ายประสาทเทียมต่อไป เริ่มต้นโดยการสมมุติให้รับจำนวนพาติเคิลเข้ามา 10 พาติเคิล จากนั้นทำการคำนวณหาค่าความเหมาะสมของทุกพาติเคิลและแบ่งครึ่งพาติเคิลออกเป็นสองส่วน โดยพาติเคิลที่มีค่าความเหมาะสมสูงจะอยู่ในส่วนครึ่งแรกและพาติเคิลที่มีค่าความเหมาะสมต่ำอยู่ในส่วนที่สอง และทำการตัดส่วนที่สองทิ้งไป จากนั้นก็จะทำการคำนวณด้วยวิธี PSO ซึ่งในที่นี้พาติเคิลจะมีจำนวนครึ่งหนึ่ง โดยสมมุติให้จำนวนพาติเคิลที่อยู่ในส่วนครึ่งแรกคือ ส่วนที่มีค่าความเหมาะสมสูงมีพาติเคิลจำนวน 5 พาติเคิล จำนวนรอบเท่ากับ 10 รอบ ความเร็วสูงสุดที่ให้กับพาติเคิลคือ 0.5 และลักษณะโครงข่ายประสาทเทียมกำหนดให้มีนิวรอนใน input layer เท่ากับ 2 นิวรอนและให้ hidden layer เท่ากับ 2 นิวรอน ซึ่งลักษณะของโครงข่ายที่กำหนดขึ้นมีลักษณะดังรูปที่ 3.1 ที่แสดงไว้ด้านบน

จากนั้นให้ทำการนำข้อมูลเข้ามาเพื่อใช้ในการฝึกสอนโครงข่ายประสาทเทียม ซึ่งจะทำการกำหนดข้อมูลสมมุติขึ้น โดยให้ข้อมูลที่กำหนดขึ้นมามีค่าดังตารางที่ 3.1

ตารางที่ 3.1 แสดงข้อมูลตัวอย่างที่เป็นข้อมูลนำเข้า

Pattern	Input1	Input2	Output
1	0.3	0.4	0.2
2	0.5	0.7	0.6

3.2.3 การฝึกสอนโครงข่ายประสาทเทียม

- วิธีดำเนินการในส่วนของ PSO

ขั้นตอนในการฝึกสอนโครงข่ายประสาทเทียมนั้น จะประกอบด้วยขั้นตอนดังนี้

1. สร้างพาติเคิลที่ได้กำหนดไว้ตั้งแต่ขั้นตอนการกำหนดค่าให้กับตัวแปร ซึ่งถูกกำหนดไว้ให้มี 5 พาติเคิล (คือ ค่าส่วนครึ่งแรกที่เป็นค่าที่มีความเหมาะสมสูงจากตัวอย่างการหาค่าความเหมาะสมด้านบน) ซึ่งแต่ละพาติเคิลจะมีข้อมูลบรรจุอยู่ภายใน คือ

<u>พาติเคิลตัวที่ 1</u> :

ตำแหน่งปัจจุบัน คือ 0.1, 0.3, 0.2, -0.1, 0.5, -0.2

ตำแหน่งที่ดีที่สุด คือ 0.1, 0.3, 0.2, -0.1, 0.5, -0.2

ความเร็ว คือ 0.0, 0.0, 0.0, 0.0, 0.0

ค่าความเหมาะสมที่ดีที่สุด คือ 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พาติเคิลตัวที่ 2 :

ตำแหน่งปัจจุบัน คือ -0.3, 0.2, 0.4, 0.1, 0.6, -0.4
ตำแหน่งที่ดีที่สุด คือ -0.3, 0.2, 0.4, 0.1, 0.6, -0.4
ความเร็ว คือ 0.0, 0.0, 0.0, 0.0, 0.0
ค่าความเหมาะสมที่ดีที่สุด คือ 0

พาติเคิลตัวที่ 3 :

ตำแหน่งปัจจุบัน คือ 0.2, -0.5, 0.3, -0.1, 0.4, -0.3
ตำแหน่งที่ดีที่สุด คือ 0.2, -0.5, 0.3, -0.1, 0.4, -0.3
ความเร็ว คือ 0.0, 0.0, 0.0, 0.0, 0.0
ค่าความเหมาะสมที่ดีที่สุด คือ 0

พาติเคิลตัวที่ 4 :

ตำแหน่งปัจจุบัน คือ 0.2, 0.4, 0.5, -0.4, -0.8, -0.3
ตำแหน่งที่ดีที่สุด คือ 0.2, 0.4, 0.5, -0.4, -0.8, -0.3
ความเร็ว คือ 0.0, 0.0, 0.0, 0.0, 0.0
ค่าความเหมาะสมที่ดีที่สุด คือ 0

พาติเคิลตัวที่ 5 :

ตำแหน่งปัจจุบัน คือ -0.8, -0.7, 0.3, -0.5, 0.5, 0.9
ตำแหน่งที่ดีที่สุด คือ -0.8, -0.7, 0.3, -0.5, 0.5, 0.9
ความเร็ว คือ 0.0, 0.0, 0.0, 0.0, 0.0
ค่าความเหมาะสมที่ดีที่สุด คือ 0

ค่าตำแหน่งนั้นเกิดจากการสุ่มค่าขึ้นมา ซึ่งการสุ่มค่านั้นมีความสัมพันธ์กับค่าถ่วงน้ำหนักในโครงข่ายประสาทเทียม เนื่องจากค่าที่สุ่มมานั้นข้อมูลจะต้องอยู่ในระหว่างค่า [-1, 1] จากการที่สมมุติให้โครงข่ายประสาทเทียมที่มีนิวรอนใน Input layer เท่ากับ 2 นิวรอนและมีนิวรอนใน Hidden layer เท่ากับ 2 นิวรอนนั้น นำมาทำการคำนวณหา Unknown Dimensions จะได้ 6 ตัวแปรซึ่งค่าที่คำนวณหาได้จาก Unknown Dimensions นั้นเป็นจำนวนของค่าถ่วงน้ำหนักที่มีในโครงข่ายประสาทเทียมที่จะทำการฝึกสอน คือ W1, W2, W3, W4, W5, W6 ดังรูปที่ 3.1 ที่ได้แสดงไว้ในส่วนด้านบน และกำหนดให้ความเร็วเริ่มต้นมีค่าเท่ากับศูนย์ ดังนั้นสามารถสรุปเขียนข้อมูลของพาติเคิลได้ดังนี้

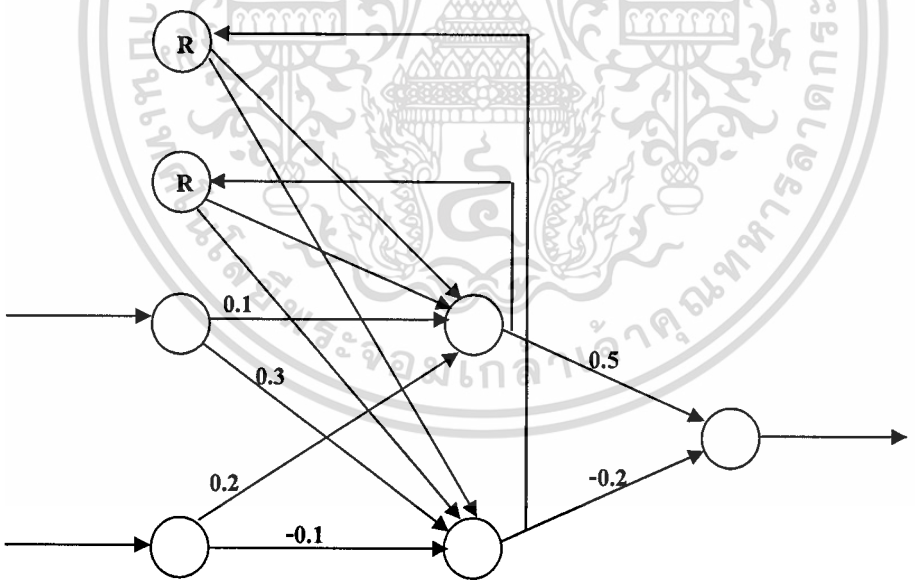
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 ค่าเริ่มต้นของค่าถ่วงน้ำหนักที่สมมุติขึ้นมาของพาติเคิล

พาติเคิล	W1	W2	W3	W4	W5	W6
1	0.1	0.3	0.2	-0.1	0.5	-0.2
2	-0.3	0.2	0.4	0.1	0.6	-0.4
3	0.2	-0.5	0.3	-0.1	0.4	-0.3
4	0.2	0.4	0.5	-0.4	-0.8	-0.3
5	-0.8	-0.7	0.3	-0.5	0.5	0.9

2. คำนวณค่าความเหมาะสม ซึ่งแบ่งออกเป็น 2 ขั้นตอน คือ
   - ขั้นตอนแรก ทำการคำนวณค่าผ่านทางโครงข่ายประสาทเทียม
   - ขั้นตอนที่สอง คำนวณค่าความเหมาะสม (Fitness) ของแต่ละพาติเคิล

   โดยเริ่มต้นนำค่าแต่ละตำแหน่งของพาติเคิลไปแทนค่าที่เป็นตำแหน่งของค่าถ่วงน้ำหนักตามลำดับ ดังแสดงในรูปที่ 3.3



รูปที่ 3.3 การนำค่าตำแหน่งของแต่ละพาติเคิลไปแทนค่าถ่วงน้ำหนักในโครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นจึงทำการคำนวณค่าข้อมูลอินพุตที่ได้รับเข้ามา โดยใช้สมการ ดังนี้

$$net_j(t) = \sum_{i=1}^{n} x_i(t) w_{ij} + \sum_{h=1}^{m} w_{jh} y_h(t-1) + \theta_j$$

ซึ่งจากสมการข้างบนจะใช้สมการนี้กับ Hidden layer ที่เป็น Hidden layer แรกเท่านั้น (First hidden layer) เพราะเราสามารถกำหนดจำนวนชั้นของ Hidden layer ได้ ซึ่งโครงสร้างแบบ Elman เป็นโครงสร้างที่มีการป้อนกลับแบบบางส่วน (Partial recurrent network)

โหนดที่อยู่ใน Hidden layer แรกนั้น เมื่อได้รับข้อมูลอินพุตจากแต่ละโหนดแล้วนั้น จะทำการหาผลรวมของข้อมูลที่ได้รับมา จนได้ผลรวมของข้อมูลอินพุต เมื่อได้ผลรวมของข้อมูลอินพุตแล้ว หลังจากนั้นก็นำค่าที่ได้มาแทนค่าในสมการ ดังนี้

$$y_j(t) = f(net_j(t))$$

ซึ่ง $$f(net_j(t)) = \frac{1}{1 + e^{-net_j(t)}}$$

จากนั้นก็ทำการคำนวณในชั้นของ Hidden ที่สองต่อไป (ในกรณีที่ Hidden layer มีหลายชั้น) และทำการคำนวณในชั้นของ Output layer โดยสมการที่ใช้ในการคำนวณ ดังนี้

$$net_k(t) = \sum_{j=1}^{m} y_j(t) w_{jk} + \theta_k$$

การคำนวณค่า Activation function จะคำนวณเหมือนกันกับชั้น Hidden layer แสดงดังสมการ

$$y_k(t) = g(net_k(t))$$

การคำนวณค่าเพื่อหาค่าความเหมาะสมของตำแหน่งปัจจุบันของแต่ละพาติเคิลนั้น จะทำการคำนวณไล่ลำดับชั้นตามโครงข่ายประสาทเทียมไปเรื่อยๆ จนกระทั่งได้รับค่าเอาท์พุตออกมา ดังแสดงในตารางที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 ผลการคำนวณที่ได้จากการแทนค่าพาติเดิลเข้าไปในโครงข่ายประสาทเทียม

พาติเดิล	ผลการคำนวณ	
	Input Pattern 1	**Input Pattern 2**
1	0.50605	0.56360
2	0.53037	0.55985
3	0.52551	0.52586
4	0.27974	0.27758
5	0.75195	0.73936

จากนั้นให้นำค่าที่ได้มาหาค่าความคลาดเคลื่อน (Error) ที่เกิดขึ้น เมื่อเทียบกับค่าจากข้อมูลเป้าหมายจริง โดยสำหรับระบบนี้จะใช้วิธีการหาค่าความคลาดเคลื่อนแบบ Sum Square Error (SSE) ซึ่งสมการในการหาค่าความคลาดเคลื่อนมีดังนี้

$$SSE = \left(\sum_{i=1}^{n} e_i^2 \right)$$

ซึ่งในการคำนวณนั้นจะได้ผลแสดงดังตารางที่ 3.4 คือ

ตารางที่ 3.4 การคำนวณค่าความคลาดเคลื่อนของแต่ละพาติเดิล

พาติเดิล	Pattern 1		Pattern 2		SSE
	e	e^2	e	e^2	
1	-0.30605	0.09366	0.03640	0.00132	0.09498
2	-0.33037	0.10914	0.04015	0.00161	0.11075
3	-0.32551	0.10595	0.07414	0.00549	0.11144
4	-0.07974	0.00635	0.32242	0.10395	0.11030
5	-0.55195	0.30464	-0.13936	0.01942	0.32406

เมื่อได้ทำการคำนวณค่าความคลาดเคลื่อนแล้ว ให้ทำการคำนวณหาค่าความเหมาะสม (Fitness) จากสมการ

$$Fitness = \frac{1}{SSE}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 การคำนวณค่าความเหมาะสมของแต่ละพาติเคิล

พาติเคิล	Fitness
1	10.52853
2	9.02934
3	8.97343
4	9.06618
5	3.08584

3. ขั้นตอนการปรับตำแหน่งของพาติเคิล ซึ่งขั้นตอนนี้เป็นขั้นตอนหลักที่สำคัญของอัลกอริทึม PSO ซึ่งในการปรับตำแหน่งจะประกอบด้วยขั้นตอนต่อไปนี้

1) คำนวณหาค่าตำแหน่งของพาติเคิลที่มีค่าความเหมาะสมมากที่สุด (Fitness value) เท่าที่พาติเคิลตัวนั้นได้วิ่งผ่านมาของแต่ละพาติเคิลเอง ซึ่งก็คือ ค่า pbest โดยการทำรอบแรกนั้น pbest ก็คือตำแหน่งปัจจุบันแรกที่ถูกสุ่มค่ามา เช่น ในรอบแรกนั้นพาติเคิลตัวที่ 2 จะมีตำแหน่งปัจจุบัน คือ -0.3, 0.2, 0.4, 0.1, 0.6, -0.4 ดังนั้นจึงให้ตำแหน่งที่ดีที่สุดของพาติเคิลตัวที่ 2 เป็น -0.3, 0.2, 0.4, 0.1, 0.6, -0.4

2) คำนวณหาค่าตำแหน่งของพาติเคิลที่มีค่าความเหมาะสมมากที่สุด (Fitness value) ของพาติเคิลภายในกลุ่ม เท่าที่ทุกพาติเคิลเคยวิ่งผ่านมาแล้ว ซึ่งก็คือ ค่า gbest ซึ่งในตอนนี้ค่า gbest จะเท่ากับ ค่าตำแหน่งของพาติเคิลที่ 1 ซึ่งตำแหน่งคือ 0.1, 0.3, 0.2, -0.1, 0.5, -0.2 (เพราะมีค่า Fitness มากที่สุด)

3) คำนวณหาความเร็วใหม่ของทุกพาติเคิล ซึ่งในตอนนี้ได้กำหนดให้ความเร็วในปัจจุบันนี้มีค่าเป็น 0 และโดยตามปกติแล้วเราควรทำการกำหนดค่าความเร็วสูงสุดด้วย ซึ่งถ้าเราไม่กำหนดค่าความเร็วสูงสุดเอาไว้จะทำให้พาติเคิลอาจวิ่งเร็วเกินไป มีผลทำให้การทำงานของอัลกอริทึมอาจไม่มีประสิทธิภาพเท่าที่ควร ในการคำนวณหาค่าความเร็วใหม่ของแต่ละพาติเคิลนั้น สามารถทำการคำนวณได้ดังสมการต่อไปนี้

$$V(t+1) = V(t) + C_1 * rand()_1 * (pbest(t) - present(t)) + C_2 * rand()_2 * (gbest(t) - present(t))$$

เมื่อ $V(t)$ คือ ค่าความเร็วปัจจุบันของพาติเคิล

$present(t)$ คือ ค่าตำแหน่งปัจจุบันของพาติเคิล

C_1 คือ ค่าคงที่มีความสัมพันธ์กับค่า pbest มีค่าตั้งแต่ 0 ขึ้นไป ซึ่งนิยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้ค่าประมาณ 2.00 ถ้าค่านี้มากแสดงว่าให้ความสำคัญกับค่าของ pbest มาก

C_2 คือ ค่าคงที่มีความสัมพันธ์กับค่า gbest มีค่าตั้งแต่ 0 ขึ้นไป ซึ่งนิยมใช้ค่าประมาณ 2.00 ถ้าค่านี้มากแสดงว่าให้ความสำคัญกับค่าของ gbest มาก

$rand()_1$ คือ ค่าที่ถูก random ขึ้นมามีค่า 0 - 1

$rand()_2$ คือ ค่าที่ถูก random ขึ้นมามีค่า 0 - 1

pbest คือ ค่าตำแหน่งที่มีค่าความเหมาะสมสูงสุดที่พาติเคิลได้วิ่งผ่าน

gbest คือ ค่าตำแหน่งที่มีค่าความเหมาะสมสูงสุดที่ทุกพาติเคิลเคยเจอ

สำหรับในที่นี้ กำหนดให้ตัวแปรต่างๆ มีค่าดังนี้

ความเร็วสูงสุด = 0.3

C_1 และ C_2 = 2

$rand()_1$ และ $rand()_2$ = 0.5

จากนั้นแทนค่าตัวแปร และค่าของพาติเคิล ดังต่อไปนี้

<u>พาติเคิลตัวที่ 1</u> :

$V(t+1) = V(t) + C_1 * rand()_1 * (pbest(t) - present(t)) + C_2 * rand()_2 * (gbest(t) - present(t))$

$= 0 + 2 * 0.5 * [(0.1, 0.3, 0.2, -0.1, 0.5, -0.2) - (0.1, 0.3, 0.2, -0.1, 0.5, -0.2)] +$

$2 * 0.5 * [(0.1, 0.3, 0.2, -0.1, 0.5, -0.2) - (0.1, 0.3, 0.2, -0.1, 0.5, -0.2)]$

$= (0, 0, 0, 0, 0, 0)$

<u>พาติเคิลตัวที่ 2</u> :

$V(t+1) = V(t) + C_1 * rand()_1 * (pbest(t) - present(t)) + C_2 * rand()_2 * (gbest(t) - present(t))$

$= 0 + 2 * 0.5 * [(-0.3, 0.2, 0.4, 0.1, 0.6, -0.4) - (-0.3, 0.2, 0.4, 0.1, 0.6, -0.4)] +$

$2 * 0.5 * [(0.1, 0.3, 0.2, -0.1, 0.5, -0.2) - (-0.3, 0.2, 0.4, 0.1, 0.6, -0.4)]$

$= (0.4, 0.1, -0.2, -0.2, -0.1, 0.2)$

<u>พาติเคิลตัวที่ 3</u> :

$V(t+1) = V(t) + C_1 * rand()_1 * (pbest(t) - present(t)) + C_2 * rand()_2 * (gbest(t) - present(t))$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

= 0 + 2 * 0.5 * [(0.2, -0.5, 0.3, -0.1, 0.4, -0.3) – (0.2, -0.5, 0.3, -0.1, 0.4, -0.3)]+

2 * 0.5 * [(0.1, 0.3, 0.2, -0.1, 0.5, -0.2) – (0.2, -0.5, 0.3, -0.1, 0.4, -0.3)]

= (-0.1, 0.8, -0.1, 0, 0.1, 0.1)

พาติเคิลตัวที่ 4 :

$V(t+1) = V(t) + C_1 * rand()_1 * (pbest(t) - present(t)) + C_2 * rand()_2 * (gbest(t) - present(t))$

= 0 + 2 * 0.5 * [(0.2, 0.4, 0.5, -0.4, -0.8, -0.3) – (0.2, 0.4, 0.5, -0.4, -0.8, -0.3)]+

2 * 0.5 * [(0.1, 0.3, 0.2, -0.1, 0.5, -0.2) – (0.2, 0.4, 0.5, -0.4, -0.8, -0.3)]

= (-0.1, -0.1, -0.3, 0.3, 1.3, 0.1)

พาติเคิลตัวที่ 5 :

$V(t+1) = V(t) + C_1 * rand()_1 * (pbest(t) - present(t)) + C_2 * rand()_2 * (gbest(t) - present(t))$

= 0 + 2 * 0.5 * [(-0.8, -0.7, 0.3, -0.5, 0.5, 0.9) – (-0.8, -0.7, 0.3, -0.5, 0.5, 0.9)]+

2 * 0.5 * [(0.1, 0.3, 0.2, -0.1, 0.5, -0.2) – (-0.8, -0.7, 0.3, -0.5, 0.5, 0.9)]

= (0.9, 1.0, -0.1, 0.4, 0, -1.1)

เมื่อเราได้ทำการคำนวณครบทุกพาติเคิลแล้วนั้น จะได้ค่าดังตารางที่ 3.6

ตารางที่ 3.6 ค่าความเร็วที่ได้จากการคำนวณของแต่ละพาติเคิล

พาติเคิล	V1	V2	V3	V4	V5	V6
1	0	0	0	0	0	0
2	0.4	0.1	-0.2	-0.2	-0.1	0.2
3	-0.1	0.8	-0.1	0	0.1	0.1
4	-0.1	-0.1	-0.3	0.3	1.3	0.1
5	0.9	1.0	-0.1	0.4	0	-1.1

เนื่องจากว่าเราได้ทำการกำหนดค่าความเร็วสูงสุดไว้ในเบื้องต้นแล้ว ดังนั้นจึงต้องพิจารณาปรับค่าความเร็วที่เกินกว่าความความเร็วสูงสุดที่เราได้กำหนดไว้ เพื่อไม่ให้ความเร็วมีค่าเกินขอบเขตที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราได้กำหนดขึ้น ในที่นี่กำหนดให้ความเร็วสูงสุดเท่ากับ 0.3 ซึ่งจะทำให้ได้ความเร็วใหม่ที่จะนำไปใช้ได้ แสดงดังตารางที่ 3.7

ตารางที่ 3.7 ปรับค่าความเร็วเพื่อให้อยู่ในช่วงความเร็วสูงสุดที่กำหนดไว้

พาติเคิล	V1	V2	V3	V4	V5	V6
1	0	0	0	0	0	0
2	0.3	0.1	-0.2	-0.2	-0.1	0.2
3	-0.1	0.3	-0.1	0	0.1	0.1
4	-0.1	-0.1	-0.3	0.3	0.3	0.1
5	0.3	0.3	-0.1	0.3	0	0.3

4) คำนวณหาค่าตำแหน่งใหม่ของแต่ละพาติเคิล เมื่อได้ความเร็วใหม่แล้วก็ทำการคำนวณหาค่าตำแหน่งใหม่ ดังสมการ

$$present(t+1) = present(t) + V(t+1)$$

พาติเคิลตัวที่ 1 :

$$present(t+1) = present(t) + V(t+1)$$
$$= (0.1, 0.3, 0.2, -0.1, 0.5, -0.2) + (0, 0, 0, 0, 0, 0)$$
$$= (0.1, 0.3, 0.2, -0.1, 0.5, -0.2)$$

พาติเคิลตัวที่ 2 :

$$present(t+1) = present(t) + V(t+1)$$
$$= (-0.3, 0.2, 0.4, 0.1, 0.6, -0.4) + (0.3, 0.1, -0.2, -0.2, -0.1, 0.2)$$
$$= (0 , 0.3, 0.2, -0.1, 0.5, -0.2)$$

พาติเคิลตัวที่ 3 :

$$present(t+1) = present(t) + V(t+1)$$
$$= (0.2, -0.5, 0.3, -0.1, 0.4, -0.3) + (-0.1, 0.3, -0.1, 0, 0.1, 0.1)$$
$$= (0.1, -0.2, 0.2, -0.1, 0.5, -0.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พาติเคิลตัวที่ 4 :

present(t+1) = present(t) + V(t+1)

= (0.2, 0.4, 0.5, -0.4, -0.8, -0.3) + (-0.1, -0.1, -0.3, 0.3, 0.3, 0.1)

= (0.1, 0.3, 0.2, -0.1, -0.5, -0.2)

พาติเคิลตัวที่ 5 :

present(t+1) = present(t) + V(t+1)

= (-0.8, -0.7, 0.3, -0.5, 0.5, 0.9) + (0.3, 0.3, -0.1, 0.3, 0, 0.3)

= (-0.5, -0.4, 0.2, -0.2, 0.5, 1.1)

เมื่อคำนวณค่าตำแหน่งใหม่ของแต่ละพาติเคิลได้แล้ว สามารถเขียนสรุปตำแหน่งใหม่ของแต่ละพาติเคิลได้ดังตารางที่ 3.8

ตารางที่ 3.8 ค่าตำแหน่งใหม่ของแต่ละพาติเคิล

พาติเคิล	W1	W2	W3	W4	W5	W6
1	0.1	0.3	0.2	-0.1	0.5	-0.2
2	0	0.3	0.2	-0.1	0.5	-0.2
3	0.1	-0.2	0.2	-0.1	0.5	-0.2
4	0.1	0.3	0.2	-0.1	-0.5	-0.2
5	-0.5	-0.4	0.2	-0.2	0.5	1.1

เมื่อสิ้นสุดกระบวนการในขั้นนี้แล้วจากนั้นก็จะนำเอาพาติเคิลที่ได้รับการปรับตำแหน่งใหม่แล้ว ส่งให้กับส่วนของ Evolutionary algorithm เพื่อที่จะทำการ Mutation ต่อไป

- วิธีดำเนินการในส่วนของ EA

สำหรับในส่วนของการดำเนินการของ EA นั้น เริ่มจากการที่ PSO ส่งค่าของพาติเคิลทั้งหมดเข้ามาที่ EA เพื่อที่จะทำการ Mutation ต่อไป ซึ่งในส่วนของการทำ Mutation นั้นเป็นการสุ่มเพื่อปรับเปลี่ยนค่าถ่วงน้ำหนักบางค่าของพาติเคิล โดยในที่นี้จะทำการปรับเปลี่ยนค่าถ่วงน้ำหนักเพียง 1 ตำแหน่งเท่านั้น ทำให้ได้ค่าถ่วงน้ำหนักที่แตกต่างไปจากเดิม โดยที่ Mutation จะทำหน้าที่ป้องกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อผิดพลาดของวิธีการแก้ไขปัญหาทั้งหมดในการเข้าไปในเฉพาะ ปัญหา (Local minimum) ที่ดีที่สุดของการแก้ปัญหา

ขั้นตอนของกระบวนการ Evolutionary algorithm มีดังต่อไปนี้

1. นำพาติเคิลที่ได้รับจากการดำเนินการของ PSO เข้ามา เพื่อสุ่มตำแหน่งค่าถ่วงน้ำหนักของพาติเคิลแต่ละตัว เมื่อได้ตำแหน่งที่ต้องการปรับเปลี่ยนค่าถ่วงน้ำหนักแล้วจึงทำการสุ่มค่า σ_i ขึ้นมา ซึ่ง σ_i คือ ค่า self-adaptive parameter vector โดยค่าที่สุ่มจะมีค่าอยู่ในช่วง -1 ถึง 1 ซึ่งสามารถคำนวณได้ดังสมการ

$$\sigma'_i(j) = \sigma_i(j) \exp(\tau N_j(0,1))$$

โดยที่ $j = 1,\ldots, N_w$

$$\tau = \frac{1}{\sqrt{2\sqrt{N_w}}}$$

เมื่อ σ_i คือ self-adaptive parameter vector

N_w คือ จำนวนของค่าถ่วงน้ำหนักที่อยู่ในรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กทั้งหมด

$N_j(0,1)$ คือ ค่า standard Gaussian random เป็นค่าสุ่มที่มีค่าอยู่ในช่วงตั้งแต่ 0 ถึง 1

ซึ่งค่าของ N_w นั้นมีค่าเท่ากับขนาดของรีเคอร์เร็นนิวรอลเน็ตเวิร์ก (size = จำนวนของนิวรอนทั้งหมดใน Input layer x จำนวนของนิวรอนทั้งหมดใน Hidden layer x จำนวนของนิวรอนทั้งหมดใน Output layer) เช่น จากตัวอย่างนั้นเราให้ Input layer มี 2 นิวรอน Hidden layer มี 2 นิวรอน และ Output layer มี 1 นิวรอน จะได้ 2 x 2 x 1 = 5

2. จากนั้นทำการปรับค่าถ่วงน้ำหนัก ณ ตำแหน่งที่ทำการ Mutation ดังสมการ

$$W'_i(j) = W_i(j) + \sigma'_i(N_j(0,1))$$

3. เมื่อได้ค่าถ่วงน้ำหนักค่าใหม่แล้วก็นำค่าที่ได้ไปแทนที่ในตำแหน่งที่ทำการ Mutation ซึ่งทำการ Mutation ให้ครบทุกพาติเคิล

4. ค่าพาติเคิลใหม่จะถูกนำไปรวมกับพาติเคิลที่คำนวณได้จากกระบวนการของ PSO จะได้ค่าเท่ากับค่าที่รับข้อมูลมาก่อนที่จะทำการแบ่งครึ่งของพาติเคิลออกเป็นสองส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

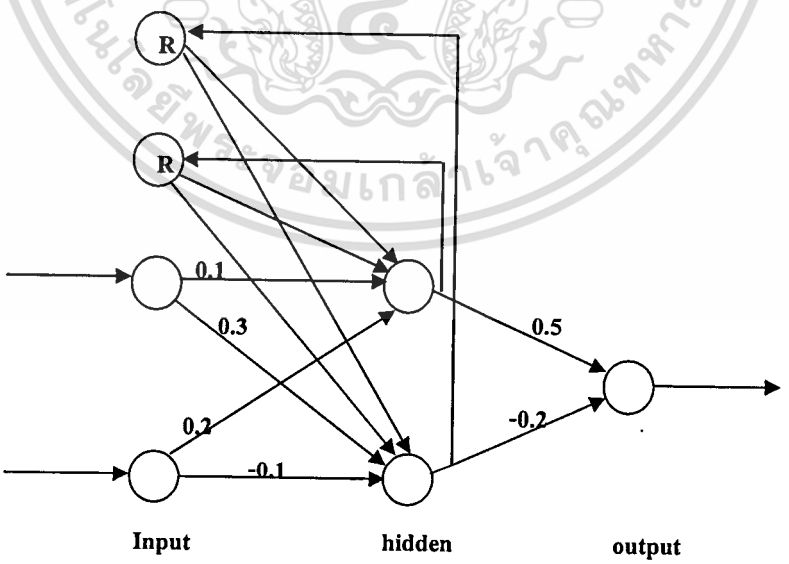
เมื่อสิ้นสุดกระบวนการในขั้นนี้แล้ว ก็ให้ทำการวนกลับไปทำซ้ำในขั้นตอนของการคำนวณหาค่าความเหมาะสมเพื่อแบ่งข้อมูลออกเป็นสองส่วน คือ ส่วนที่มีค่าความเหมาะสมสูงและส่วนที่มีค่าความเหมาะสมต่ำ จากนั้นทำในส่วนของกระบวนการ PSO และ EA ต่อไป จนกว่าจะครบจำนวนรอบที่ได้กำหนดไว้

ซึ่งจะสังเกตได้เมื่อหาค่าความเหมาะสมแต่ละรอบของพาติเคิลนั้น พาติเคิลจะทำการปรับความเร็วและตำแหน่งของตัวพาติเคิลเอง ภายใต้เงื่อนไขของฟังก์ชันที่เราต้องการจะสอนโครงข่ายประสาทเทียมเลียนแบบให้มีค่าความเหมาะสมสูงขึ้นเรื่อยๆ โดยที่มีการปรับค่าพาติเคิลซึ่งเป็นการย้ายตำแหน่งของแต่ละพาติเคิลให้เข้าใกล้ตำแหน่งที่มีค่าความเหมาะสมสูงสุด

3.2.4 การทดสอบโครงข่ายประสาทเทียม

ในการทำงานของขั้นตอนการฝึกสอนโครงข่ายประสาทเทียมโดยใช้อัลกอริทึมแบบผสมของ PSO และ EAนั้น ทำให้ได้ผลลัพธ์ของการหาค่าถ่วงน้ำหนักที่มีค่าที่เหมาะสมหรือที่ดีที่สุดออกมา ซึ่งเราจะนำเอาค่าถ่วงน้ำหนักค่าที่ดีที่สุดนี้มาทำการทดสอบโครงข่ายประสาทเทียม ดังนี้

1. จากตัวอย่างที่เราได้กำหนดไว้ตั้งแต่แรกและได้ทำการหาค่าความเหมาะสม (Fitness) แล้วนั้น พาติเคิลที่มีความเหมาะสมนั้น คือ พาติเคิลตัวที่ 1 ดังนั้นเราจะนำค่าถ่วงน้ำหนักที่ดีที่สุดของพาติเคิลตัวนี้มาแทนค่าลงในค่าถ่วงน้ำหนักของโครงข่ายประสาทเทียมที่เราได้กำหนดโครงสร้างของโครงข่ายประสาทเทียมไว้ตั้งแต่เบื้องต้นแล้ว ซึ่งจะแสดงดังรูปที่ 3.4



รูปที่ 3.4 ตัวอย่างการแทนค่าความเหมาะสมของพาติเคิลตัวที่ 1 เพื่อทดสอบโครงข่ายประสาทเทียม

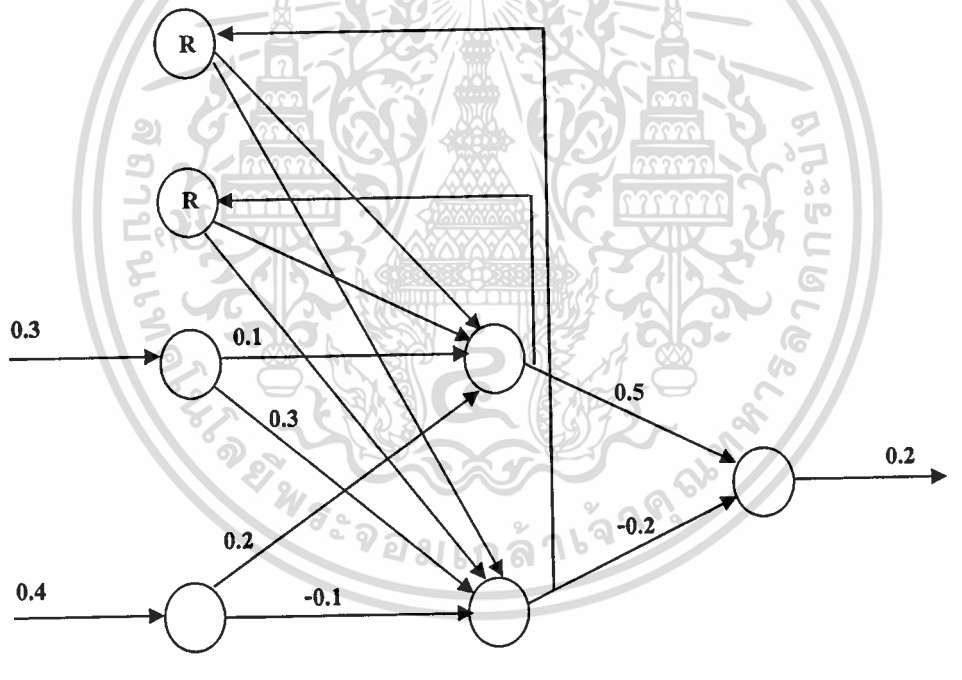
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. นำข้อมูลอินพุตเข้ามาใส่ในโครงข่ายประสาทเทียม เพื่อทำการป้อนค่าข้อมูลเข้ามา สมมุติให้ข้อมูลที่นำเข้ามานั้นแสดงดังตารางที่ 3.9

ตารางที่ 3.9 นำข้อมูลตัวอย่างที่เป็นข้อมูลนำเข้ามาทดสอบ

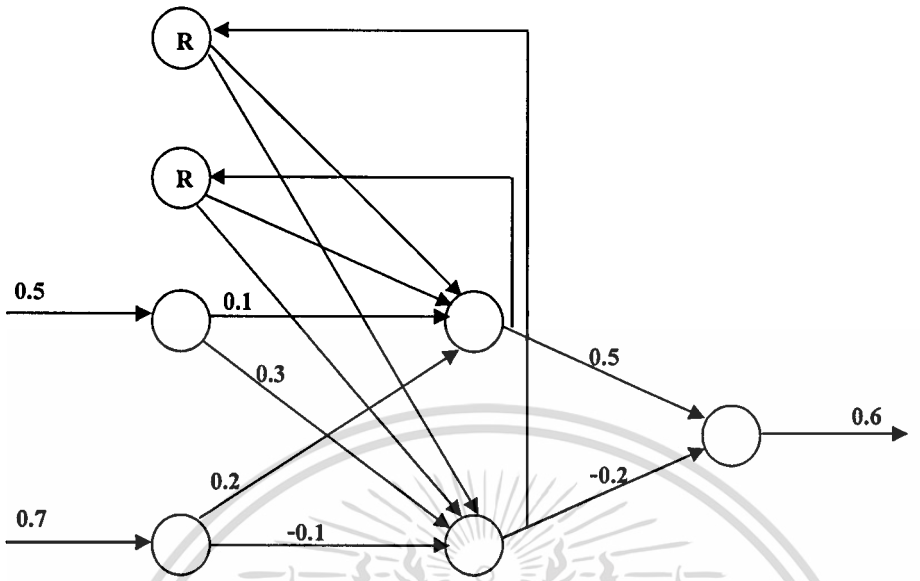
Pattern	Input1	Input2	Output
1	0.3	0.4	0.2
2	0.5	0.7	0.6

เมื่อทราบข้อมูลอินพุตและข้อมูลเอาท์พุตแล้ว ในที่นี้จะแสดงตัวอย่างของการนำข้อมูลเข้าไปแทนค่าในโครงข่ายประสาทเทียม ซึ่งแสดงดังรูปที่ 3.5 และรูปที่ 3.6



รูปที่ 3.5 การแทนค่าข้อมูลนำเข้า Pattern 1 เพื่อทดสอบโครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 การแทนค่าข้อมูลนำเข้า Pattern 2 เพื่อทดสอบโครงข่ายประสาทเทียม

จากนั้นจึงทำการคำนวณค่าข้อมูลอินพุตที่ได้รับเข้ามา โดยใช้สมการ ดังนี้

$$net_j(t) = \sum_{i=1}^{n} x_i(t) w_{ij} + \sum_{h=1}^{m} w_{jh} y_h(t-1) + \theta_j$$

ซึ่งจากสมการข้างบนจะใช้สมการนี้กับ Hidden layer ที่เป็น Hidden layer แรกเท่านั้น (First hidden layer) เพราะเราสามารถกำหนดจำนวนชั้นของ Hidden layer ได้ ซึ่งโครงสร้างแบบ Elman เป็นโครงสร้างที่มีการป้อนกลับแบบบางส่วน (Partial recurrent network)

โหนดที่อยู่ใน Hidden layer แรกนั้น เมื่อได้รับข้อมูลอินพุตจากแต่ละโหนดแล้วนั้น จะทำการหาผลรวมของข้อมูลที่ได้รับมา จนได้ผลรวมของข้อมูลอินพุต เมื่อได้ผลรวมของข้อมูลอินพุตแล้ว หลังจากนั้นก็นำค่าที่ได้มาแทนค่าในสมการ ดังนี้

$$y_j(t) = f(net_j(t))$$

ซึ่ง
$$f(net_j(t)) = \frac{1}{1 + e^{-net_j(t)}}$$

จากนั้นก็ทำการคำนวณในชั้นของ Hidden ที่สองต่อไป (ในกรณีที่ Hidden layer มีหลายชั้น) และทำการคำนวณในชั้นของ Output layer โดยสมการที่ใช้ในการคำนวณ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$net_k(t) = \sum_{j=1}^{m} y_j(t) w_{jk} + \theta_k$$

สำหรับการคำนวณค่า Activation function จะคำนวณเหมือนกันกับชั้น Hidden layer แสดงดังสมการ

$$y_k(t) = g(net_k(t))$$

ซึ่งเมื่อถึงชั้นสุดท้าย คือ Output layer จะทำให้เราได้ค่าเอาท์พุตออกมา เมื่อนำมาเปรียบเทียบกับเอาท์พุตเป้าหมายแล้ว ผลลัพธ์จะได้ดังตารางที่ 3.10

ตารางที่ 3.10 ผลการคำนวณจากการแทนค่าพาติเคิลที่ 1 เข้าไปในโครงข่ายประสาทเทียม

Input	พาติเคิล 1	ผลลัพธ์เป้าหมาย
Pattern 1	0.50605	0.2
Pattern 2	0.56360	0.6

ซึ่งจะเห็นว่าค่าที่คำนวณได้นั้นห่างจากค่าผลลัพธ์เป้าหมายที่เราต้องการอยู่มาก เนื่องจากว่าในขั้นตอนของการกำหนดค่าเบื้องต้นให้กับตัวแปรนั้น ได้ทำการกำหนดอย่างง่ายๆ และในการแสดงการฝึกสอนโครงข่ายประสาทเทียมนั้นเป็นการฝึกสอนเพียงรอบเดียวเท่านั้น เพื่อที่จะแสดงการคำนวณเป็นตัวอย่างให้เห็นวิธีการดำเนินงานของระบบ ซึ่งในการทำการคำนวณจริงๆ ต้องมีการปรับเปลี่ยนค่าตัวแปรให้มีความซับซ้อนมากยิ่งขึ้นกว่านี้ เพื่อให้ผลลัพธ์ที่มีประสิทธิภาพมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

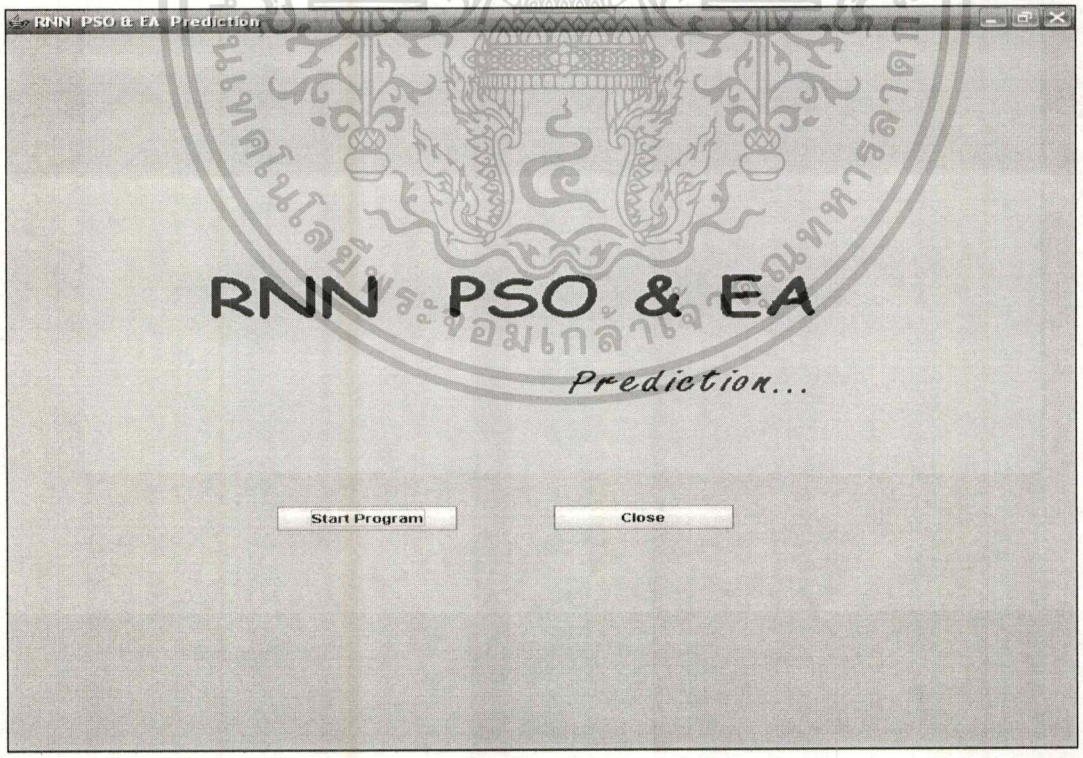
จากการพัฒนาโปรแกรมซึ่งได้ทำการทดลองใช้งานโปรแกรม สามารถสรุปผลการทำงานของโปรแกรมการทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA โดยแบ่งการทำงานสำหรับใช้งานได้เป็น 2 ส่วน

1. การทำงานของโปรแกรม
2. การทดลองใช้งานและปรับค่าโปรแกรม

4.1 การทำงานของโปรแกรม

การทำงานของโปรแกรมนี้ จะมีขั้นตอนในการใช้งานโปรแกรมที่ง่าย ทำให้ผู้ใช้สามารถเข้าใจในการใช้งานและใช้งานได้อย่างถูกต้อง ซึ่งขั้นตอนในการทำงานของโปรแกรมมีดังต่อไปนี้

4.1.1 หน้าที่ 1 : RNN PSO&EA Prediction



รูปที่ 4.1 แสดงหน้าที่ 1 : RNN PSO&EA Prediction

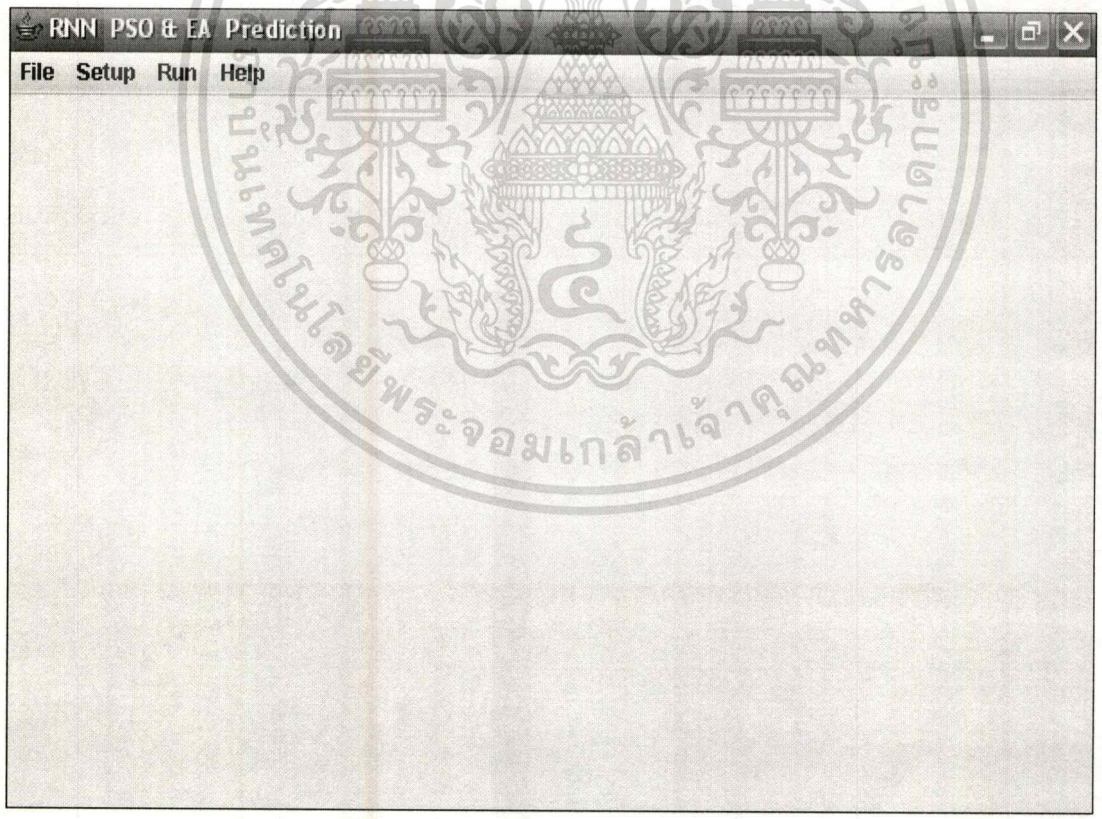
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเริ่มต้นเข้าสู่การทำงานของโปรแกรมนั้น จะปรากฏหน้าจอแรกดังรูปที่ 4.1 ซึ่งเป็นหน้าจอที่จะบอกให้ผู้ใช้งานทราบว่าขณะนี้โปรแกรมพร้อมที่จะเข้าสู่การทำงานภายในเรียบร้อยแล้ว ซึ่งหากผู้ใช้ยังไม่พร้อมที่จะใช้งานก็สามารถทำการปิดโปรแกรมออกจากการทำงานได้โดยใช้ปุ่ม Close โดยที่หน้าจอนี้จะมีปุ่มการทำงาน ดังนี้คือ

- ปุ่ม Start Program ใช้สำหรับให้ผู้ใช้เลือกเมนูเมื่อผู้ใช้พร้อมที่จะเริ่มใช้งานโปรแกรมในการทำนายข้อมูลที่ต้องการ
- ปุ่ม Close ใช้สำหรับออกจากโปรแกรมหากผู้ใช้ยังไม่พร้อมใช้งานโปรแกรม

4.1.2 หน้าที่ 2 : หน้าจอหลักของโปรแกรม

เมื่อผู้ใช้ทำการกดปุ่ม Start Program แล้วนั้น จากนั้นก็จะเข้าสู่หน้าจอการทำงานหลักของโปรแกรม ดังรูปที่ 4.2



รูปที่ 4.2 แสดงหน้าที่ 2 : หน้าจอหลักของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

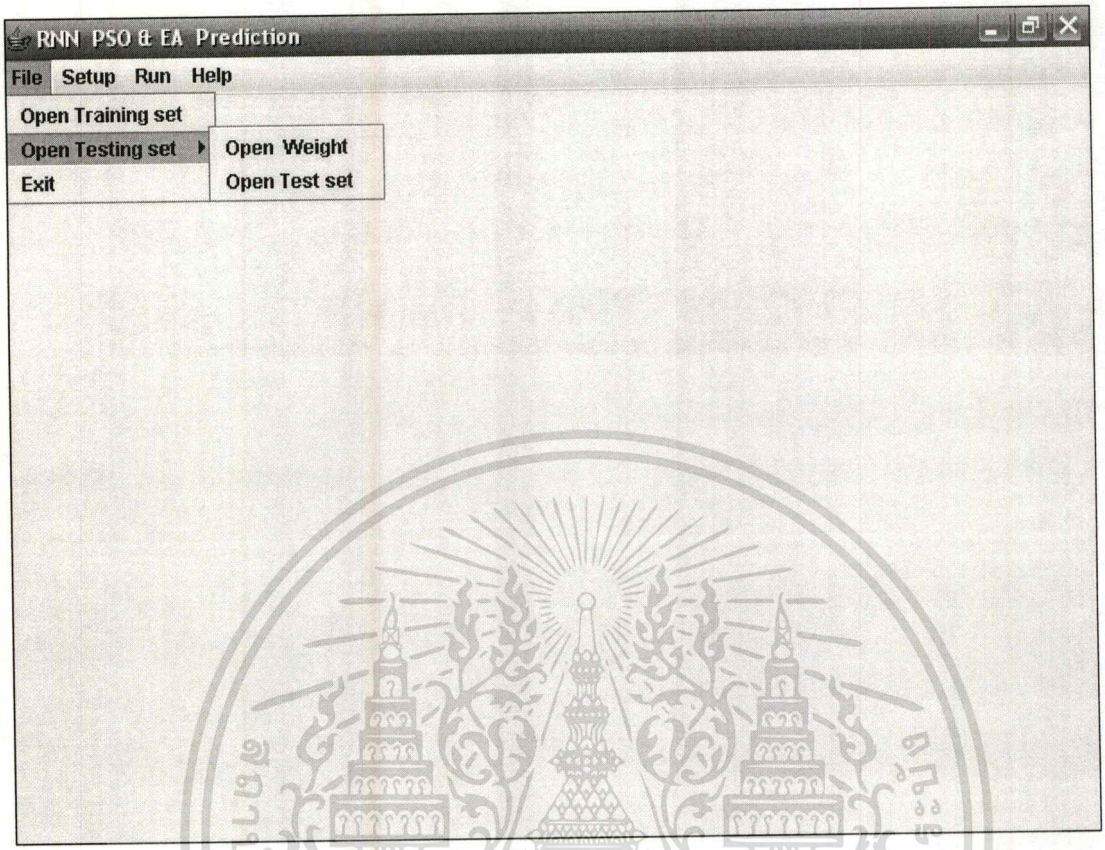
จากรูปที่ 4.2 ผู้ใช้ต้องพิจารณาว่าจะเลือกลักษณะการใช้งานโปรแกรมเพื่อทำงานอะไร ซึ่งในการทำงานของโปรแกรมนี้จะแบ่งเป็นการทำงานออกเป็น 2 ส่วนหลัก ได้แก่ ส่วนสำหรับนำข้อมูลมาฝึกสอนโครงข่าย (Training) และส่วนสำหรับนำข้อมูลมาทดสอบกับโครงข่าย (Testing) สำหรับการเลือกทำรายการนั้นจะต้องเลือกเมนูเรียงลำดับการทำงานกันไป File > Setup > Run และ Help ซึ่งหน้าจอหลักประกอบด้วยเมนูต่อไปนี้

- ปุ่ม File ใช้สำหรับเลือกเมนู เมื่อผู้ใช้งานต้องการจัดการเกี่ยวกับการโหลดไฟล์เพื่อนำมาใช้งานกับโปรแกรม ซึ่งในปุ่ม File ประกอบด้วย 2 ส่วนหลัก ได้แก่ Open Training Set และ Open Testing Set
- ปุ่ม Setup ใช้สำหรับเลือกเมนูเมื่อผู้ใช้งานต้องการปรับแต่งค่าเกี่ยวกับอัลกอริทึมแบบผสมของ PSO กับ EA และโครงข่ายประสาทเทียม เพื่อให้เหมาะสมกับข้อมูลที่จะนำเข้า
- ปุ่ม Run ใช้สำหรับเลือกเมนูเมื่อผู้ใช้งานต้องการให้โปรแกรมคำนวณและทำนายค่าจากข้อมูลที่ผู้ใช้ได้ป้อนเข้ามา ซึ่งประกอบด้วย 2 ส่วน ได้แก่ Run Training และ Run Testing
- ปุ่ม Help ใช้สำหรับเลือกเมนูเมื่อผู้ใช้งานต้องการทราบข้อมูลเบื้องต้นเกี่ยวกับที่มา ความสามารถ และภาพรวมในการทำงานของโปรแกรม เพื่อใช้สร้างและเพิ่มความเข้าใจให้กับผู้ใช้งานมากยิ่งขึ้น

4.1.3 หน้าที่ 2.1 : หน้าจอหลักของเมนู File

สำหรับในส่วนของหน้าจอหลักที่แสดงส่วนของเมนู File นั้น เป็นเมนูที่ประกอบด้วยเมนูย่อยต่างๆ เพื่อนำไปใช้ในการเลือกไฟล์ที่ผู้ใช้ต้องการนำค่าเข้ามากำหนดให้กับโปรแกรม โดยที่เมนู File นี้ ประกอบด้วยเมนู Open Training Set และเมนู Open Testing Set ซึ่งเมนู Open Testing Set นี้จะประกอบด้วยเมนูย่อย (Menu Item) คือ เมนู Open Weight และเมนู Open Test Set แสดงดังรูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงหน้าที่ 2.1 : หน้าจอหลักของเมนู File

สำหรับการทำงานของเมนูต่างๆ มีดังต่อไปนี้

- เมนู Open Training Set ใช้สำหรับเมื่อผู้ใช้งานต้องการให้โปรแกรมโหลดนำเข้าค่าของข้อมูลเข้ามาในโปรแกรม ซึ่งจะเป็นข้อมูลที่ผู้ใช้งานต้องการจะนำมาฝึกหัด (Training) กับโครงข่ายประสาทเทียม ซึ่งประกอบด้วยข้อมูลอินพุตและเอาต์พุตตามลักษณะที่ผู้ใช้ต้องการ

- เมนู Open Testing Set > Open Weight ใช้สำหรับเมื่อผู้ใช้งานต้องการให้โปรแกรมนำเข้าข้อมูลที่เป็นค่าถ่วงน้ำหนักเก่า (Weight) ที่ได้เคยทำการฝึกหัดโครงข่ายประสาทเทียมและได้บันทึกค่าถ่วงน้ำหนักไว้นำค่าเข้ามา ซึ่งการบันทึกค่าถ่วงน้ำหนักไว้ก็เพื่อเป็นการประหยัดเวลาในการฝึกหัดโครงข่ายประสาทเทียม ทำให้สามารถผ่านขั้นตอนของการฝึกหัดโครงข่ายประสาทเทียมไปได้

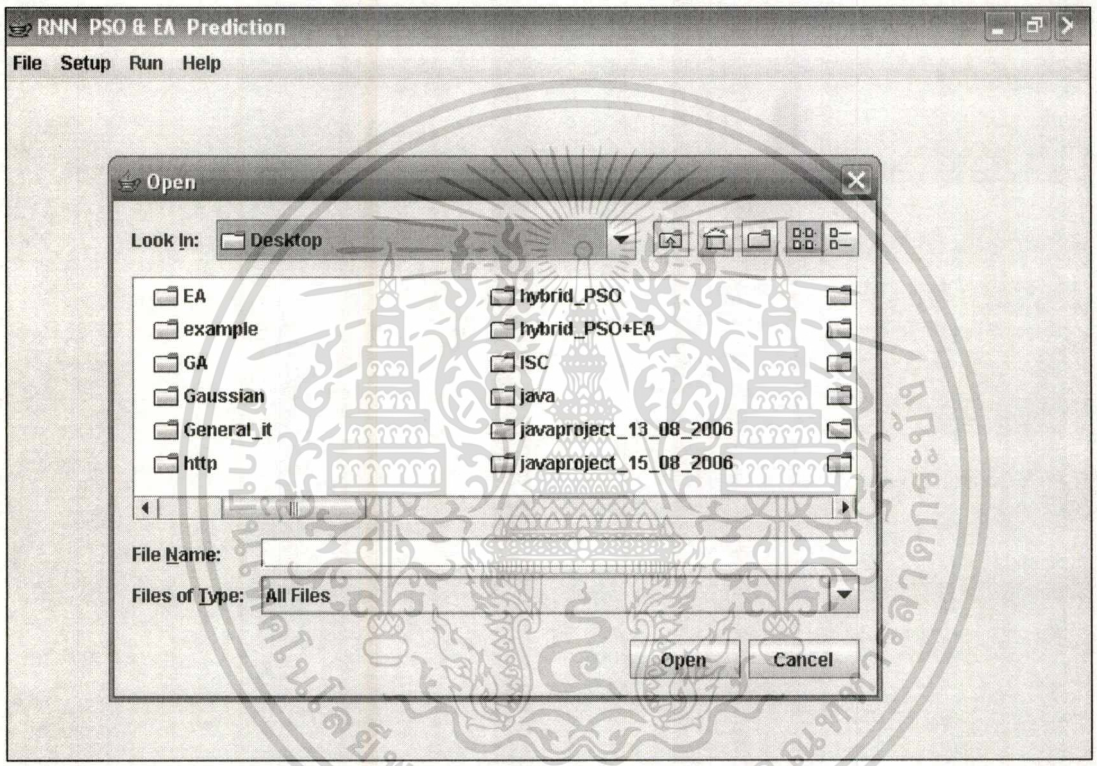
- เมนู Open Testing Set > Open Test Set ใช้สำหรับเมื่อผู้ใช้งานต้องการให้โปรแกรมนำเข้าค่าของข้อมูลเข้ามาในโปรแกรม ซึ่งเป็นข้อมูลที่ผู้ใช้งานต้องการจะนำมาทดสอบ (Testing) กับโครงข่ายประสาทเทียม ซึ่งประกอบด้วยข้อมูลอินพุตและเอาต์พุตตามลักษณะที่ผู้ใช้ต้องการ

- เมนู Exit ใช้สำหรับเมื่อผู้ใช้งานต้องการออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 หน้าที่ 2.1.1 : หน้าจอนำเข้าค่าของข้อมูล

หน้าจอนำเข้าค่าของข้อมูลนี้เป็นหน้าจอที่ผู้ใช้งานต้องการที่จะโหลดข้อมูลเข้ามาทำงานภายในโปรแกรม ซึ่งหน้าจอนำเข้าค่าของข้อมูลจะปรากฏขึ้นในส่วนของเมนู File ที่ผู้ใช้งานทำการเลือก โดยเมนูนี้ประกอบด้วยเมนูย่อย (Menu Item) คือ เมนู Open Training set เมนู Open Testing set เมนู Open Weight และเมนู Open Test set ซึ่งแสดงดังรูปที่ 4.4



รูปที่ 4.4 แสดงหน้าที่ 2.1.1 : หน้าจอนำเข้าค่าของข้อมูล

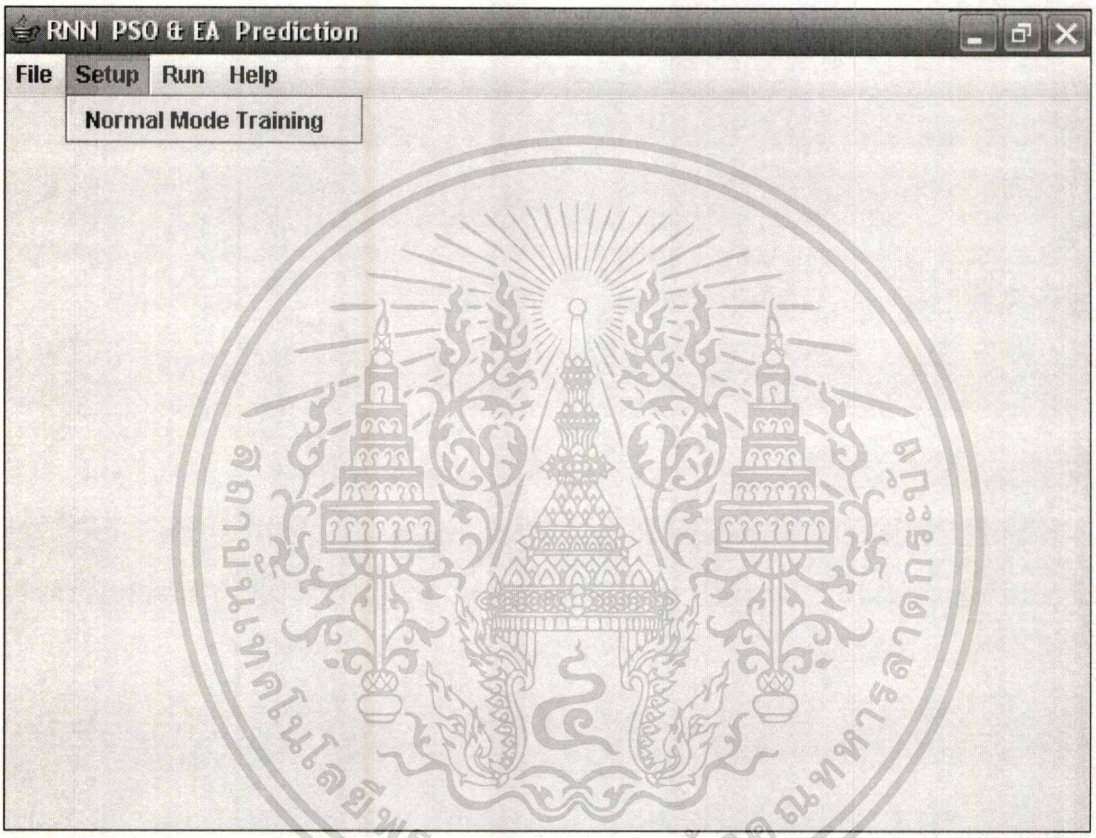
โดยที่หน้าจอนำเข้าค่าของข้อมูลจะปรากฏหน้าต่าง (Window) ที่มีส่วนประกอบดังนี้

- ปุ่ม Open ใช้สำหรับให้ผู้ใช้งานที่ต้องการจะทำการเลือกไฟล์ที่เป็นไฟล์ข้อมูลนำเข้ามาใช้สำหรับการฝึกหัดโครงข่าย หรือเพื่อเป็นการนำเข้าค่าถ่วงน้ำหนักที่ได้บันทึกไว้แล้วของโครงข่ายประสาทเทียม
- ปุ่ม Cancel ใช้สำหรับให้ผู้ใช้งานเลือกหากไม่ต้องการเลือกไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.5 หน้าที่ 2.2 : หน้าจอหลักของเมนู Setup

ในส่วนหน้าจอหลักของเมนู Setup นั้น ประกอบด้วยเมนูย่อย คือ เมนู Normal Mode Training เป็นเมนูที่เอาไว้ใช้ในการปรับตั้งค่าการใช้งานโปรแกรมให้เหมาะสมกับข้อมูลนำเข้าตามที่ผู้ใช้งานต้องการ แสดงดังรูปที่ 4.5



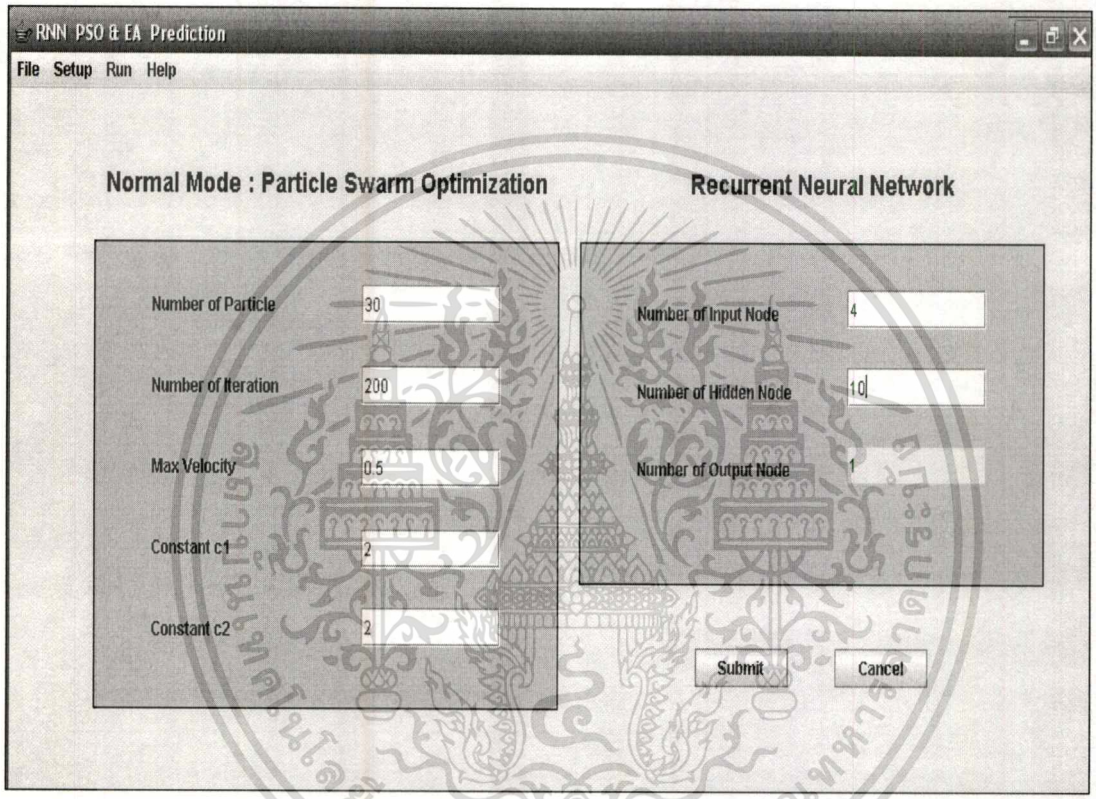
รูปที่ 4.5 แสดงหน้าที่ 2.2 : หน้าจอหลักของเมนู Setup

จากรูปที่ 4.5 นั้น เมนู Normal Mode Training นั้น มีไว้สำหรับให้ผู้ใช้งานกำหนดค่าข้อมูลขึ้นมา เพื่อนำไปทำการฝึกหัดกับโครงข่ายประสาทเทียมให้โครงข่ายเกิดการเรียนรู้ขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.6 หน้าที่ 2.2.1 : หน้าจอ Normal Mode Training

สำหรับหน้าจอ Normal Mode Training นี้ จะเป็นการเริ่มต้นการใช้งานสำหรับการปรับตั้งค่าที่ผู้ใช้ได้ทำการเลือกเพื่อที่จะใส่ค่าของข้อมูลให้กับโครงข่ายประสาทเทียม โดยที่หน้าจอของ Normal Mode Training แสดงดังรูปที่ 4.6



รูปที่ 4.6 แสดงหน้าที่ 2.2.1 : หน้าจอ Normal Mode Training

จากรูปที่ 4.6 จะเห็นได้ว่าหน้าจอ Normal Mode Training นั้น ประกอบด้วยค่าที่ผู้ใช้ต้องใส่ข้อมูลเข้าไปในโปรแกรม ซึ่งข้อมูลเหล่านั้นประกอบด้วยดังต่อไปนี้

- แถบ Number of Particle ใช้สำหรับให้ผู้ใช้งานใส่ค่าจำนวนของพาติเคิลที่ต้องการเพื่อให้ระบบสร้างพาติเคิลขึ้นมา ซึ่งโดยปกติจะสร้างพาติเคิลประมาณ 20 – 40 พาติเคิล
- แถบ Number of Iteration ใช้สำหรับให้ผู้ใช้งานใส่ค่าจำนวนของการวนรอบทำซ้ำของโปรแกรมเพื่อให้พาติเคิลทำการปรับตำแหน่งของตัวพาติเคิลเอง ซึ่งจำนวนที่ทำการวนรอบนี้จะเป็นไปตามจำนวนที่ผู้ใช้ได้ใส่เอาไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แถบ Max Velocity ใช้สำหรับให้ผู้ใช้งานใส่ค่าจำนวนของความเร็วที่มากที่สุดที่จะให้พาติเคิลมีได้ โดยที่ค่าที่มากที่สุดนี้จะมีค่าอยู่ในช่วงระหว่าง 0.0 ถึง 1.0

- แถบ Constant C1 ใช้สำหรับให้ผู้ใช้งานใส่ค่าที่เป็นค่าคงที่ ปกติค่านี้จะมีค่าเท่ากับ 2.00 โดยจะนำค่านี้ไปใช้ในโปรแกรม ซึ่งเป็นค่าที่ใช้สำหรับคูณเพื่อเพิ่มความสำคัญให้กับตำแหน่งที่ดีที่สุดของแต่ละพาติเคิล (ค่า pbest)

- แถบ Constant C2 ใช้สำหรับให้ผู้ใช้งานใส่ค่าที่เป็นค่าคงที่ ปกติค่านี้จะมีค่าเท่ากับ 2.00 โดยจะนำค่านี้ไปใช้ในโปรแกรม ซึ่งเป็นค่าที่ใช้สำหรับคูณเพื่อเพิ่มความสำคัญให้กับตำแหน่งที่ดีที่สุดของทุกๆ พาติเคิล (ค่า gbest)

- แถบ Number of Input Node ใช้สำหรับให้ผู้ใช้งานใส่ค่าจำนวนโหนดของอินพุตที่ผู้ใช้ต้องการให้มี เพื่อสร้างโครงข่ายประสาทเทียมสำหรับการทำนาย ซึ่งจำนวนของโหนดอินพุตนั้นเป็นค่าที่ค่อนข้างสำคัญมาก ดังนั้นควรจะใส่ค่านี้ให้มีความเหมาะสมกับจำนวนอินพุตที่รับเข้ามา

- แถบ Number of Hidden Node ใช้สำหรับให้ผู้ใช้งานใส่ค่าจำนวนโหนดของฮิดเดนที่ผู้ใช้ต้องการให้มี ซึ่งโหนดฮิดเดนนั้นถ้าจำนวนโหนดยิ่งมากจำให้โครงข่ายสามารถเลียนแบบฟังก์ชันยากๆ ได้ แต่ก็จะทำให้เสียเวลาในการคำนวณนานขึ้น ดังนั้นควรจะใส่ค่านี้ให้มีความเหมาะสม

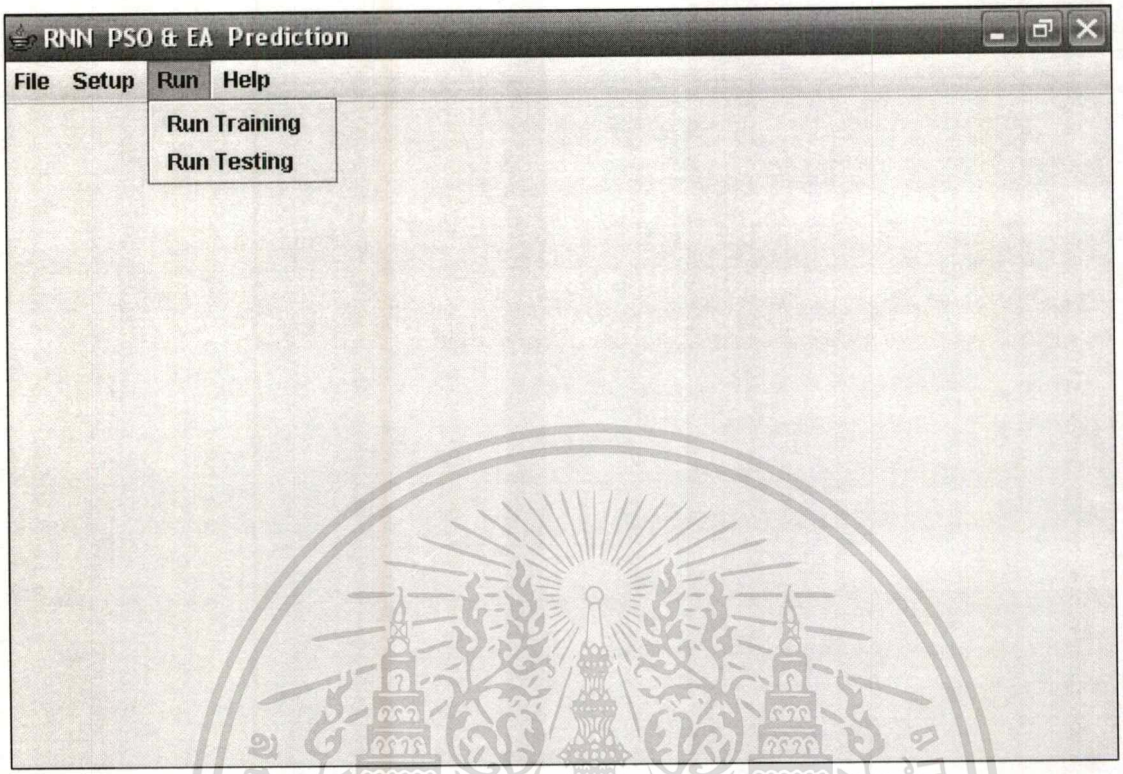
- ปุ่ม Submit ใช้สำหรับให้ผู้ใช้งานกดปุ่ม Submit เมื่อผู้ใช้ได้ใส่ข้อมูลครบทุกข้อมูลแล้วและเป็นข้อมูลที่ผู้ใช้ได้ตรวจสอบแล้วว่าใส่ค่าถูกต้อง

- ปุ่ม Cancel ใช้สำหรับให้ผู้ใช้งานกดปุ่ม Cancel เมื่อผู้ใช้ต้องการลบค่าที่กรอกข้อมูลลงในแถบต่างๆ

4.1.7 หน้าที่ 2.3 : หน้าจอหลักของเมนู Run

ในส่วนของหน้าจอหลักของเมนู Run นั้น จะถูกใช้งานหลังจากที่ผู้ใช้งานได้ทำการเปิดไฟล์ข้อมูลขึ้นมาเพื่อเป็นการนำเข้าข้อมูลมาใช้ในโครงข่ายและใส่ค่าที่ใช้สำหรับการปรับตั้งค่าที่ผู้ใช้ได้ทำการเลือกเพื่อที่จะใส่ค่าของข้อมูลให้กับโครงข่ายประสาทเทียมแล้ว หรือผู้ใช้ได้ทำการเปิดไฟล์ของค่าถ่วงน้ำหนักและเปิดไฟล์ของข้อมูลที่จะนำมาทดสอบกับโครงข่ายประสาทเทียมเรียบร้อยแล้วเช่นกัน จากนั้นให้โปรแกรมดำเนินงาน (Run) เพื่อทำการฝึกหัดโครงข่ายประสาทเทียม (Training) และทดสอบโครงข่าย (Testing) ต่อไป โดยที่ส่วนของเมนู Run นั้นจะประกอบด้วยการทำงานของการฝึกหัดโครงข่ายและการทำงานส่วนของการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงหน้าที่ 2.3 : หน้าจอหลักของเมนู Run

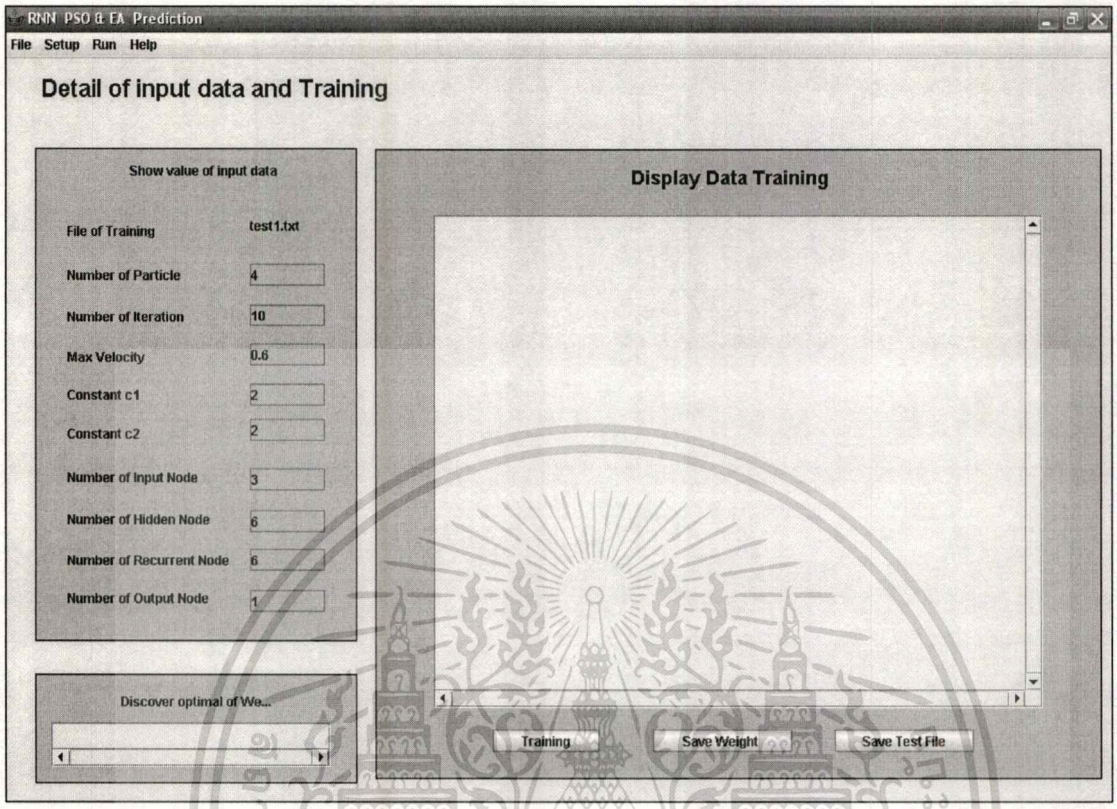
จากรูปที่ 4.7 ประกอบด้วยเมนูย่อยดังนี้

- เมนู Run > Run Training ใช้สำหรับให้ผู้ใช้งานเลือกเมื่อผู้ใช้ต้องการดำเนินการในส่วนของขั้นตอนการฝึกหัดโครงข่ายประสาทเทียม
- เมนู Run > Run Testing ใช้สำหรับให้ผู้ใช้งานเลือกเมื่อผู้ใช้ต้องการดำเนินการในส่วนของขั้นตอนการทดสอบโครงข่ายประสาทเทียม

4.1.8 หน้าที่ 2.3.1 : หน้าจอของเมนู Run Training

เมื่อเริ่มเข้าสู่การดำเนินการของเมนู Run Training นั้น ผู้ใช้งานจะต้องทำหน้าที่ในส่วนของการพิจารณาเพื่อตรวจสอบความถูกต้องของค่าที่ได้ใส่เพื่อทำการปรับตั้งค่าให้กับโปรแกรมที่ผู้ใช้ได้ดำเนินการในขั้นตอนของเมนู Setup > Normal Mode Training เพื่อความสะดวกในการฝึกหัดโครงข่ายประสาทเทียมทำให้ไม่เสียเวลาในการฝึกหัดโครงข่าย ซึ่งรายละเอียดของค่าที่ผู้ใช้งานได้ใส่ลงไปนั้นแสดงดังรูปที่ 4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แสดงหน้าที่ 2.3.1 : หน้าจอ ของเมนู Run Training

จากรูปที่ 4.8 นั้น ในหน้าจอของเมนู Run Training นั้น จะประกอบด้วยปุ่มต่างๆ ที่เกี่ยวข้องกับการทำงานของการฝึกหัดโครงข่าย ดังนี้

- ปุ่ม Training ใช้สำหรับให้ผู้ใช้งานเลือกเมื่อผู้ใช้ได้ทำการตรวจสอบข้อมูลแล้วว่าข้อมูลที่ใส่ในการปรับตั้งค่านั้นมีความถูกต้องตรงกับความต้องการแล้ว จากนั้นจึงเลือกกดปุ่ม Training เพื่อทำการฝึกหัดโครงข่ายต่อไป

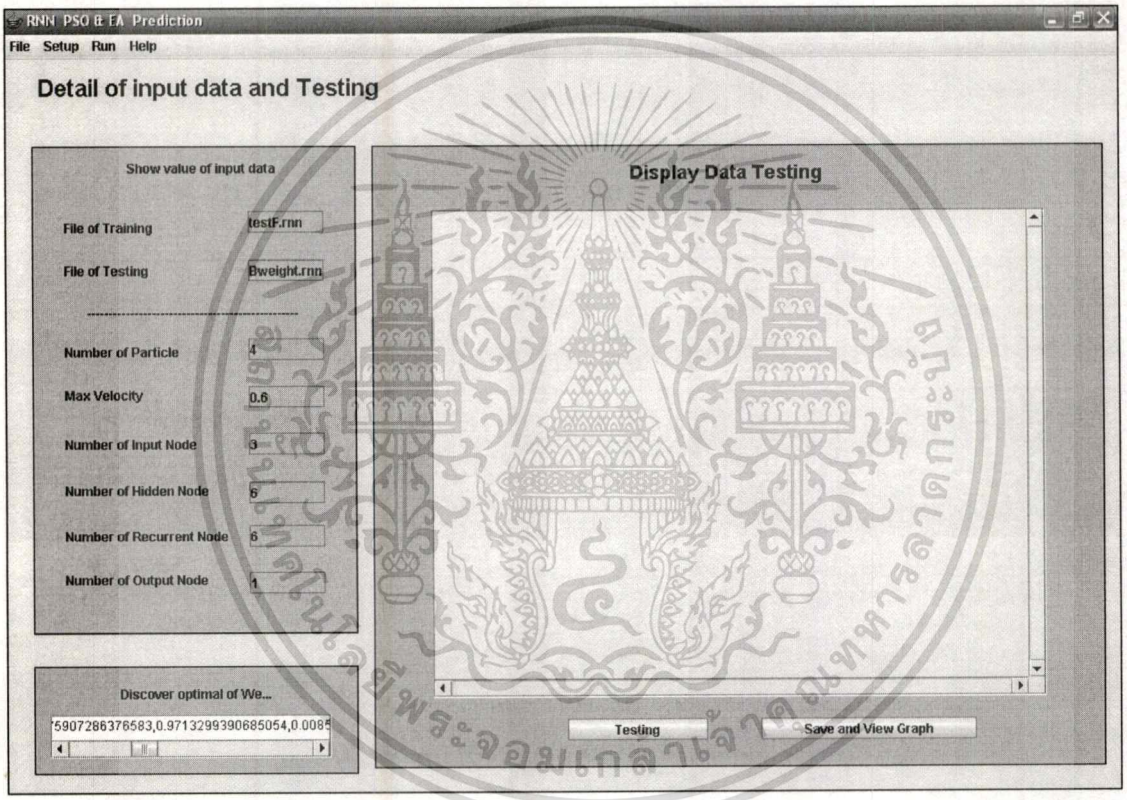
- ปุ่ม Save Weight ใช้สำหรับให้ผู้ใช้งานเลือกเมื่อต้องการที่จะบันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเอาไว้ใช้งานในการทดสอบโครงข่ายประสาทเทียม จะทำให้ไม่เสียเวลาฝึกสอนโครงข่ายซ้ำอีกครั้ง

- ปุ่ม Save Test File ใช้สำหรับให้ผู้ใช้งานเลือกการบันทึกข้อมูลที่ถูกแบ่งเอาไว้เป็นส่วนทดสอบ โดยที่ข้อมูลที่นำเข้าเพื่อทำการ Training และ Testing นั้น เราจะแบ่งข้อมูลจากหนึ่งไฟล์แยกออกเป็นสองไฟล์ คือ ไฟล์สำหรับการฝึกหัดและไฟล์สำหรับทดสอบโครงข่ายประสาทเทียม โดยที่ปุ่ม Save Test File จะให้ผู้ใช้ทำการเลือกกดเพื่อที่จะได้ทราบว่าไฟล์ที่จะทำการทดสอบเป็นไฟล์ใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.9 หน้าที่ 2.3.2 : หน้าจอของเมนู Run Testing

สำหรับการทำงานของเมนู Run Testing นั้น เป็นการทำงานเพื่อทำการทดสอบข้อมูล ซึ่งได้จากการนำค่าถ่วงน้ำหนักที่ได้จากการคำนวณของอัลกอริทึมแบบผสมของ PSO และ EA ที่บันทึกไว้มาทำการทดสอบกับข้อมูลในส่วนของการทดสอบที่ได้แบ่งไว้ตอนต้นแล้วว่าข้อมูลหนึ่งไฟล์จะแยกออกเป็นข้อมูลสำหรับฝึกหัดและข้อมูลสำหรับทดสอบกับโครงข่ายประสาทเทียม ซึ่งหน้าจอของเมนู Run Testing แสดงดังรูปที่ 4.9



รูปที่ 4.9 แสดงหน้าที่ 2.3.2 : หน้าจอ ของเมนู Run Testing

จากรูปที่ 4.9 ประกอบด้วยส่วนของการทำงาน คือ

- ปุ่ม Testing ใช้สำหรับให้ผู้ใช้งานทำการกดปุ่ม เพื่อเป็นการยืนยันว่าจะทำการทดสอบข้อมูลกับโครงข่ายประสาทเทียม

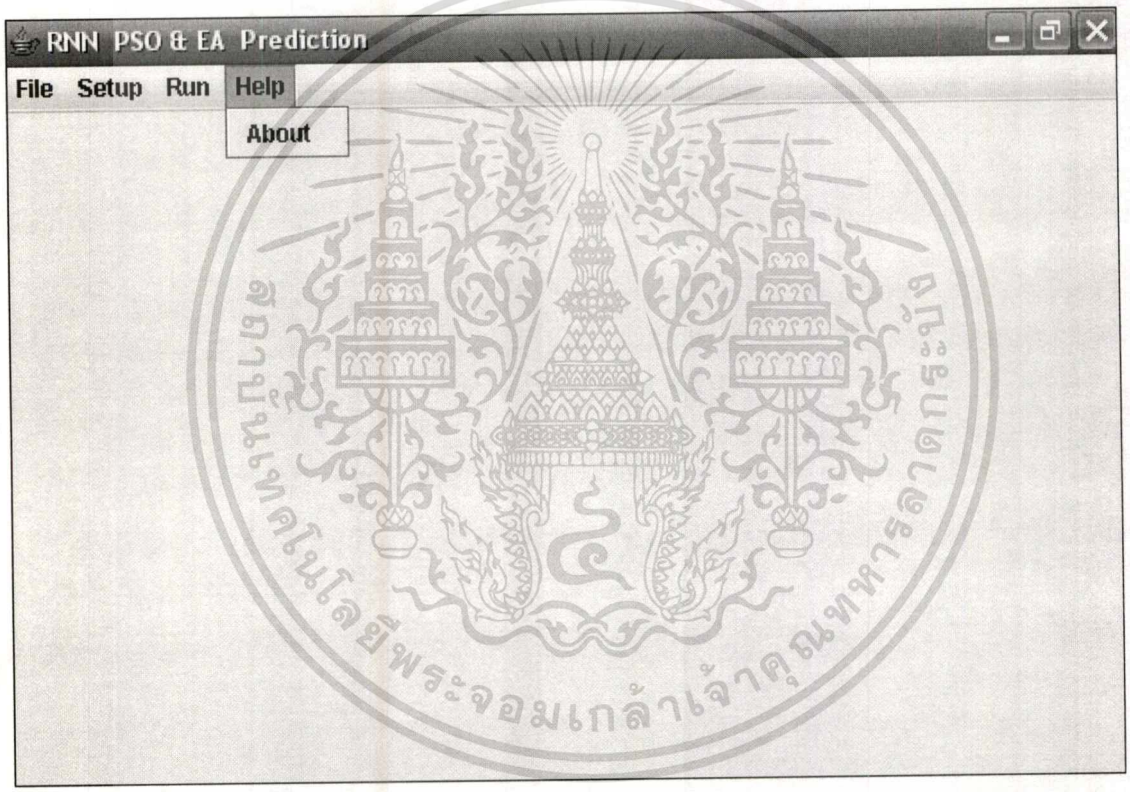
- ปุ่ม Save and View Graph ใช้สำหรับให้ผู้ใช้งานกดปุ่มเลือกการบันทึกข้อมูลที่เราได้ทำการทดสอบและแสดงกราฟซึ่งเป็นผลลัพธ์จากการทำนายค่าของข้อมูลออกมา ซึ่งเป็นกราฟที่แสดงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบเอาต์พุตเป้าหมายที่เกิดขึ้นจริงในอดีตกับเอาต์พุตที่ได้จากแบบจำลองของรีเคอร์เร็นท์นิวรอลเน็ตเวิร์ก

4.1.10 หน้าที่ 2.3.3 : หน้าจอของเมนู Help

เมนู Help ใช้สำหรับเมื่อผู้ใช้งานต้องการทราบข้อมูลเบื้องต้นเกี่ยวกับที่มา ความสามารถและภาพรวมในการทำงานของโปรแกรม เพื่อใช้สร้างและเพิ่มความเข้าใจให้กับผู้ใช้งานมากยิ่งขึ้น แสดงดังรูปที่ 4.10



รูปที่ 4.10 แสดงหน้าที่ 2.3.3 : หน้าจอของเมนู Help

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดสอบการใช้งาน

จากการที่ได้พิจารณาลักษณะการทำงานของโปรแกรมการทำนายอนุกรมเวลาโดยใช้หลักการรีเดอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA แล้ว ต่อจากนั้นจะเป็นการทดสอบการทำงานของโปรแกรม โดยขั้นตอนในการทดสอบโปรแกรมนั้นจะต้องพิจารณาตามขั้นตอนต่อไปนี้

1. นำตัวอย่างของข้อมูลมาทดสอบการใช้งานกับโปรแกรม ซึ่งประกอบด้วยข้อมูลอนุกรมเวลา 2 ชุด ข้อมูลที่นำมาทดสอบแสดงดังต่อไปนี้

1) แสดงรายละเอียดของข้อมูลชุดแรกที่นำมาทดสอบกับโปรแกรมการทำนาย ซึ่งมีข้อมูลดังนี้

Australia monthly production of electricity.

Actual m.KWH

Jan 1956 - August 1995

1254 1290 1379 1346 1535 1555 1655 1651 1500 1538 1486 1394 1409 1387 1543

1502 1693 1616 1841 1787 1631 1649 1586 1500 1497 1463 1648 1595 1777 1824

1994 1835 1787 1699 1633 1645 1597 1577 1709 1756 1936 2052 2105 2016 1914

1925 1824 1765 1721 1752 1914 1857 2159 2195 2287 2276 2096 2055 2004 1924

1851 1839 2019 1937 2270 2251 2382 2364 2129 2110 2072 1980 1995 1932 2171

2162 2489 2424 2641 2630 2324 2412 2284 2186 2184 2144 2379 2383 2717 2774

3051 2891 2613 2600 2493 2410 2390 2463 2616 2734 2970 3125 3342 3207 2964

2919 2764 2732 2622 2698 2950 2895 3200 3408 3679 3473 3154 3107 3052 2918

2786 2739 3125 3033 3486 3661 3927 3851 3456 3390 3280 3166 3080 3069 3340

3310 3798 3883 4191 4213 3766 3628 3520 3322 3250 3287 3552 3440 4153 4265

4655 4492 4051 3967 3807 3639 3647 3560 3929 3858 4485 4697 4977 4675 4596

4491 4127 4144 4014 3994 4320 4400 5002 5091 5471 5193 4997 4737 4546 4498

4350 4206 4743 4582 5191 5457 5891 5618 5158 5030 4800 4654 4453 4440 4945

4788 5425 5706 6061 5846 5242 5408 5114 5042 5008 4657 5359 5193 5891 5980

6390 6366 5756 5640 5429 5398 5413 5141 5695 5554 6369 6592 7107 6917 6353

6205 5830 5646 5379 5489 5824 5907 6482 6795 7028 6776 6274 6362 5940 5958

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5769 5887 6367 6165 6868 7201 7601 7581 7090 6841 6408 6435 6176 6138 6717

6470 7312 7763 8171 7788 7311 6679 6704 6724 6552 6427 7105 6869 7683 8082

8555 8386 7553 7398 7112 6886 7077 6820 7426 7143 8261 8240 8977 8991 8026

7911 7510 7381 7366 7414 7824 7524 8279 8707 9486 8973 8231 8206 7927 7999

7834 7521 8284 7999 8940 9381 10078 9796 8471 8572 8150 8168 8166 7903 8606

8071 9178 9873 10476 9296 8818 8697 8381 8293 7942 8001 8744 8397 9115 9773

10358 9849 9083 9143 8800 8741 8492 8795 9354 8796 10072 10174 11326 10744 9806

9740 9373 9244 9407 8827 9880 9364 10580 10899 11687 11280 10208 10212 9725 9721

9846 9407 10265 9970 10801 11246 12167 11578 10645 10613 10104 10348 10263 9973 10803

10409 11458 11845 12559 12070 11221 11338 10761 11012 10923 10790 11427 10788 11772 12104

12634 12772 11764 11956 11646 11750 11485 11198 12265 11704 12419 13259 13945 13839 12387

12546 12038 11977 12336 11793 12877 11923 13306 13988 14002 14336 12867 12721 12449 12686

12810 12015 12888 12431 13499 13014 14296 14125 12817 12862 12449 12489 12621 12380 13023

12302 13339 13825 14428 14151 13355 13094 12656 12435 13287 12434 13209 12817 13746 14259

14590 14354 13254 13464 13302 13456 13171 12517 13489 12509 13785 13921 14603 14749 13540

13457 13243 13590 13487 12776 13812 13032 14268 14473 15359 14457

2) แสดงรายละเอียดของข้อมูลชุดที่สองที่นำมาทดสอบกับโปรแกรมการทำนาย ซึ่งมีข้อมูลดังนี้

monthly production of electricity.

Actual

1956 - 1995

227.86 198.24 194.97 184.88 196.79 205.36 226.72 226.05 202.50 194.79 192.43 219.25

217.47 192.34 196.83 186.07 197.31 215.02 242.67 225.17 206.69 197.75 196.43 213.55

222.75 194.03 201.85 189.50 206.07 225.59 247.91 247.64 213.01 203.01 200.26 220.50

237.90 216.94 214.01 196.00 208.37 232.75 257.46 267.69 220.18 210.61 209.59 232.75

232.75 219.82 226.74 208.04 220.12 235.69 257.05 258.69 227.15 219.91 219.30 259.04

237.29 212.88 226.03 211.07 222.91 249.18 266.38 268.53 238.02 224.69 213.75 237.43

248.46 210.82 221.40 209.00 234.37 248.43 271.98 268.11 233.88 223.43 221.38 233.76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

243.97 217.76 224.66 210.84 220.35 236.84 266.15 255.20 234.76 221.29 221.26 244.13

245.78 224.62 234.80 211.37 222.39 249.63 282.29 279.13 236.60 223.62 225.86 246.41

261.70 225.01 231.54 214.82 227.70 263.86 278.15 274.64 237.66 227.97 224.75 242.91

253.08 228.13 233.68 217.38 236.38 256.08 292.83 304.71 245.57 234.41 234.12 258.17

268.66 245.31 247.47 226.25 251.67 268.79 288.94 290.16 250.69 240.80

3) แสดงรายละเอียดของข้อมูลชุดที่สามที่นำมาทดสอบกับโปรแกรมการทำนาย ซึ่งมีข้อมูลดังนี้

Dow Jones index

28 Aug - 18 Dec 1972

Brockwell and Davis

110.94 110.69 110.43 110.56 110.75 110.84 110.46 110.56 110.46 110.05 109.6

109.31 109.31 109.25 109.02 108.54 108.77 109.02 109.44 109.38 109.53 109.89

110.56 110.56 110.72 111.23 111.48 111.58 111.9 112.19 112.06 111.96 111.68

111.36 111.42 112 112.22 112.7 113.15 114.36 114.65 115.06 115.86 116.4

116.44 116.88 118.07 118.51 119.28 119.79 119.7 119.28 119.66 120.14 120.97

121.13 121.55 121.96 122.26 123.79 124.11 124.14 123.37 123.02 122.86 123.02

123.11 123.05 123.05 122.83 123.18 122.67 122.73 122.86 122.67 122.09 122

121.23

4) แสดงรายละเอียดของข้อมูลชุดที่สี่ที่นำมาทดสอบกับโปรแกรมการทำนาย ซึ่งมีข้อมูลดังนี้

D.J. index

1995 - 1999

3651 3645 3626 3634 3620.5 3607 3589 3590 3622 3634 3616 3634

3631 3613 3576 3537 3547 3540 3543 3566 3566 3555 3581 3578

3587 3599 3584 3585 3593 3593 3603 3622 3630 3642 3635 3645

3636 3649 3674 3672 3665 3688 3681 3693 3698 3662 3625 3643

3648 3640 3664 3662 3684 3678 3711 3704 3685 3694 3670 3674

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3688	3686	3684	3678	3684	3697	3702	3704	3710	3719	3734	3730
3741	3764	3743	3717	3726	3752	3755	3745	3762	3758	3776	3794
3776	3757	3784	3799	3804	3821	3866	3850	3849	3842	3867	3870
3870	3884	3892	3914	3913	3895	3926	3945	3978	3964	3976	3968
3871	3906	3906	3932	3895	3895	3904	3928	3937	3923	3888	3900
3912	3892	3840	3839	3832	3809	3832	3824	3832	3856	3852	3853
3831	3863	3863	3850	3848	3865	3896	3865	3863	3869	3821	3775
3762	3699	3627	3636	3675	3680	3693	3674	3689	3682	3662	3663
3662	3620	3620	3599	3653	3649	3700	3684	3668	3682	3701	3714
3698	3696	3670	3629	3656	3629	3653	3660	3672	3721	3733	3759
3766	3742	3745	3755	3754	3757	3757.5	3758	3761	3759	3772	3768
3756	3749	3753	3773	3815	3790	3811	3777	3742	3708	3725	3699
3637	3686	3670	3667	3625	3647	3649.5	3652	3674	3688	3709	3703
3703	3704	3739	3754	3755	3748	3727	3732	3735	3742	3736	3720
3731	3764	3798	3796	3793	3766	3747	3754	3756	3767	3751	3769
3760	3785	3776	3755	3755	3751	3776	3847	3830	3881	3899	3917
3913	3901	3886	3892.5	3899	3886	3908	3875	3860	3880	3895	3954
3933	3937	3869	3852	3837	3832	3849	3863	3878	3855	3843	3847
3801	3787	3776	3797	3821	3877	3875	3890	3910	3924	3918	3936
3911	3891	3855									

2. เมื่อได้ชุดข้อมูลเพื่อนำมาทดลองกับโปรแกรมแล้ว จากนั้นจะต้องทำการปรับเปลี่ยนข้อมูล (Normalization) ให้มีค่าอยู่ในช่วงระหว่าง 0.0 ถึง 1.0 เพื่อให้ข้อมูลมีความเหมาะสมในการใช้งานกับอัลกอริทึมได้ โดยสูตรที่ใช้ในการปรับเปลี่ยนข้อมูล คือ

newdata = (data – newmin) / (newmax – newmin)

เมื่อ

data คือ ข้อมูลที่ต้องการนอร์มอลไลเซชัน

max คือ ข้อมูลที่มีค่าสูงสุด

min คือ ข้อมูลที่มีค่าต่ำสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

newmin คือ ข้อมูลที่มีค่าต่ำสุดที่ถูกปรับแล้ว

newmax คือ ข้อมูลที่มีค่าสูงสุดที่ถูกปรับแล้ว

newdata คือ ข้อมูลที่ทำการนอร์มอลไลเซชันแล้ว

ซึ่งหลังจากนอร์มอลไลเซชันแล้ว จะทำให้ได้ข้อมูลภายในชุดนั้นมีค่าอยู่ระหว่าง 0.0 ถึง 1.0 จึงจะสามารถข้อมูลนั้นมาใช้งานสำหรับการการฝึกสอนโครงข่ายประสาทเทียมของระบบได้ และหลังจากที่เราได้ผลในการทำนายแล้ว ถ้าเราต้องการแปลงค่าที่เหมาะสมได้จาก 0.0 ถึง 1.0 ให้เป็นค่าจริง เราก็ต้องทำการแปลงกลับข้อมูลได้จากสมการ

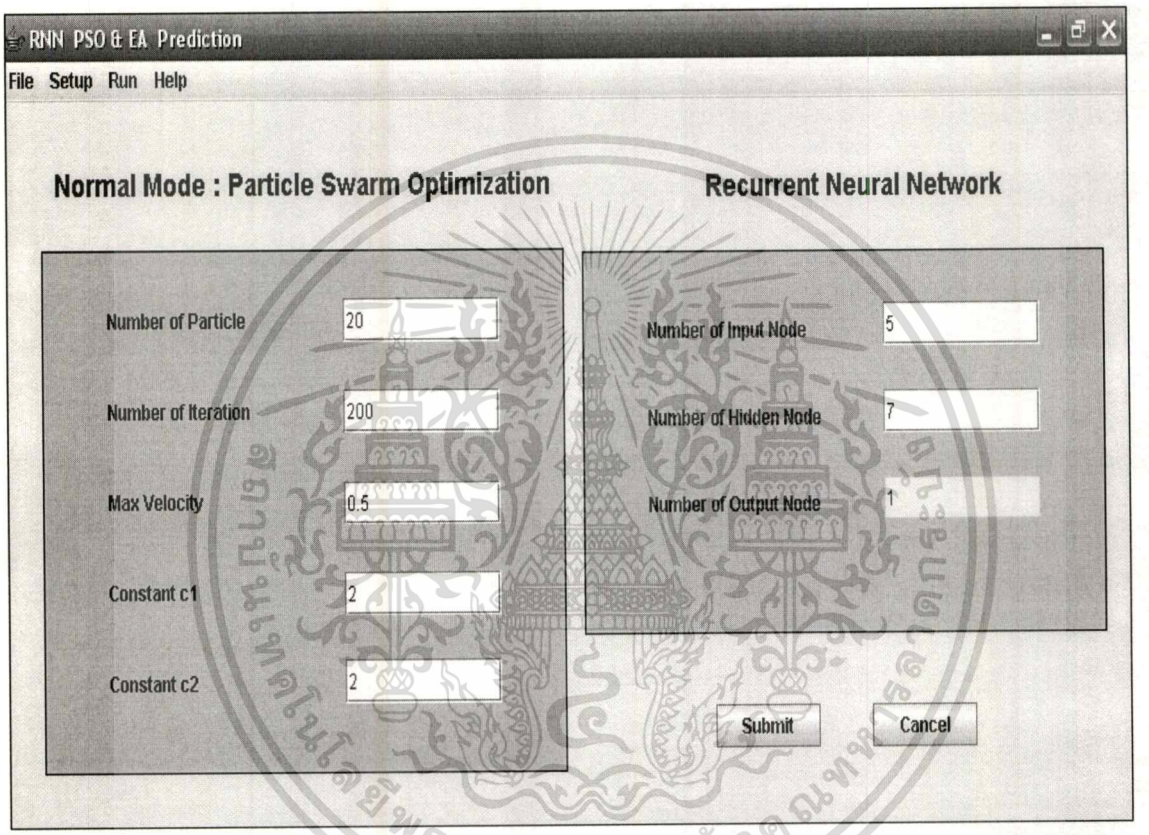
$$data = newdata \times (newmax - newmin) + newmin$$

3. ในการทดลองกับข้อมูลแต่ละชุดนั้น จะทำการแบ่งชุดของข้อมูลในการทดลองนั้นออกเป็น 2 ส่วนด้วยกัน ได้แก่ ข้อมูลสำหรับใช้เป็นชุดฝึกหัดโครงข่ายประสาทเทียม 75% และชุดข้อมูลสำหรับใช้เป็นชุดทดสอบโครงข่ายประสาทเทียม 25%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 การทดลองข้อมูลชุดที่ 1

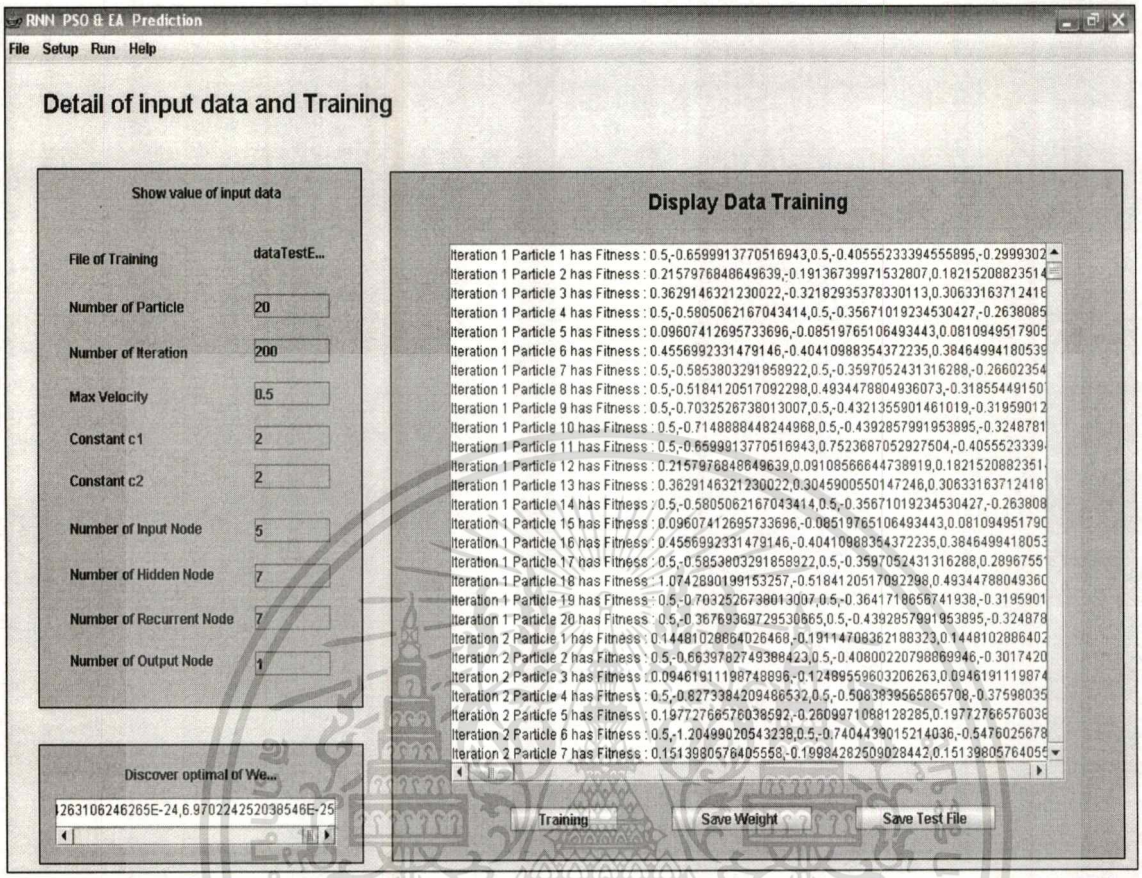
1. เริ่มแรกของการทำงานนั้นก่อนอื่นจะต้องทำการเปิดไฟล์ข้อมูลเข้ามา ในที่นี้ทำการเปิดไฟล์ข้อมูลชุดที่ 1 เข้ามา จากนั้นก็ทำการปรับตั้งค่าโปรแกรมในเมนู Setup > Normal Mode Training ซึ่งจะทำการใส่ค่าดังแสดงในรูปที่ 4.11



รูปที่ 4.11 แสดงการปรับตั้งค่าของโปรแกรมเพื่อทำการทดลองกับข้อมูลชุดที่ 1

2. หลังจากที่ใส่ค่าของข้อมูลเรียบร้อยแล้ว ต่อจากนั้นก็ทำการ Run โปรแกรม ซึ่งทำการฝึกหัดโครงข่ายประสาทเทียม โดยเลือกไปที่ Run > Run Training จะปรากฏหน้าจอของเมนู Run Training ขึ้นมา ซึ่งหน้าจอนี้ก็จะแสดงค่าของข้อมูลที่เราใส่เข้ามาให้ตรวจสอบดูว่าข้อมูลที่เราใส่นั้นถูกต้องหรือไม่ เพื่อที่จะทำให้ไม่ต้องเสียเวลาในการฝึกหัดโครงข่ายอีกครั้ง หากตรวจสอบแล้วว่าข้อมูลที่ใส่เข้าไปถูกต้องก็ทำการฝึกหัดโครงข่าย แสดงดังรูปที่ 4.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

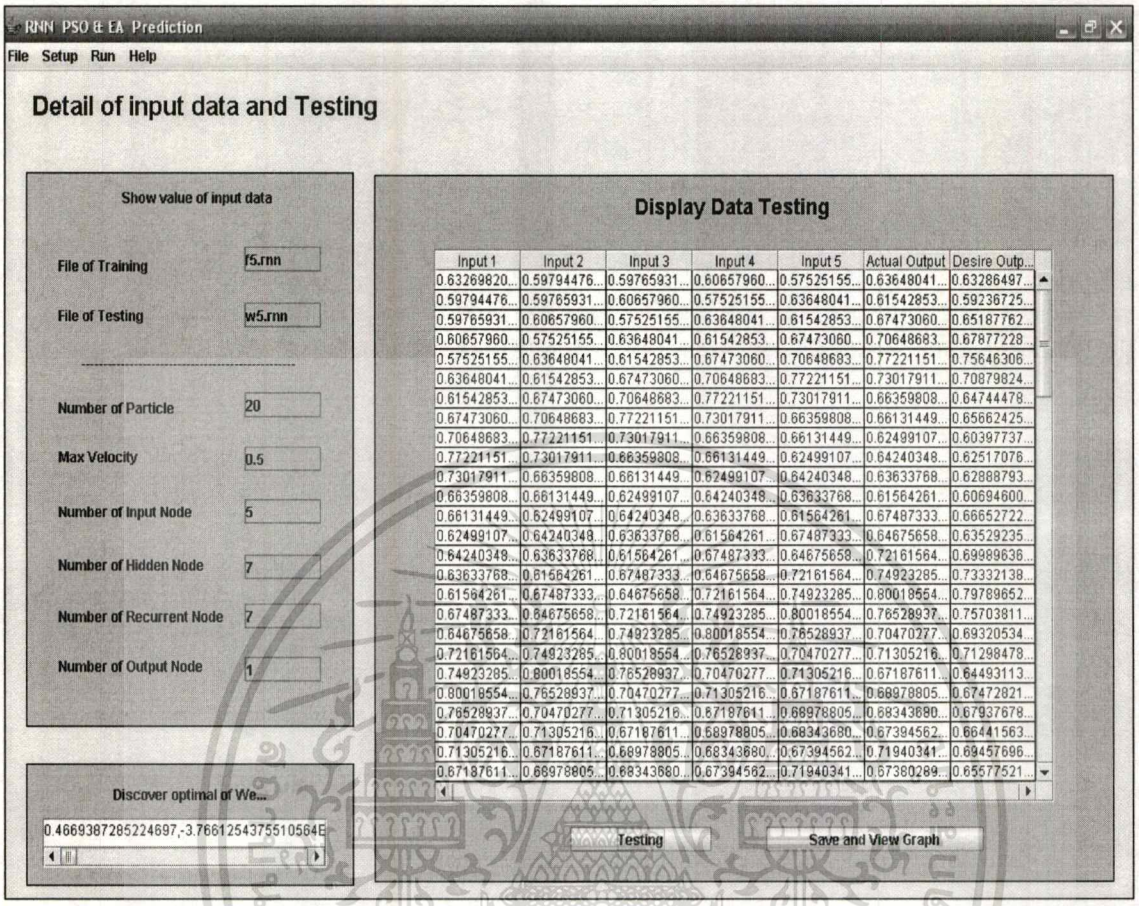


รูปที่ 4.12 แสดงการฝึกหัดโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 1

จากรูปที่ 4.12 เป็นการแสดงผลของการฝึกหัดโครงข่ายประสาทเทียม ซึ่งสามารถจะทำการบันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเพื่อเอาไว้ใช้ในส่วนของการทดสอบโครงข่ายได้ ทำให้ประหยัดเวลาในการทำงาน เนื่องจากไม่ต้องเสียเวลาในการฝึกหัดโครงข่าย สามารถเปิดไฟล์ที่บันทึกค่าถ่วงน้ำหนักมาใช้ทดสอบได้ทันที

3. ทำการทดสอบโครงข่ายประสาทเทียม โดยจะต้องเปิดไฟล์ข้อมูลที่เป็นไฟล์ที่ได้บันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเอาไว้ขึ้นมาทำการทดสอบ จากนั้นทำการทดสอบโครงข่ายประสาทเทียม โดยไปที่ Run > Run Testing แสดงดังรูปที่ 4.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

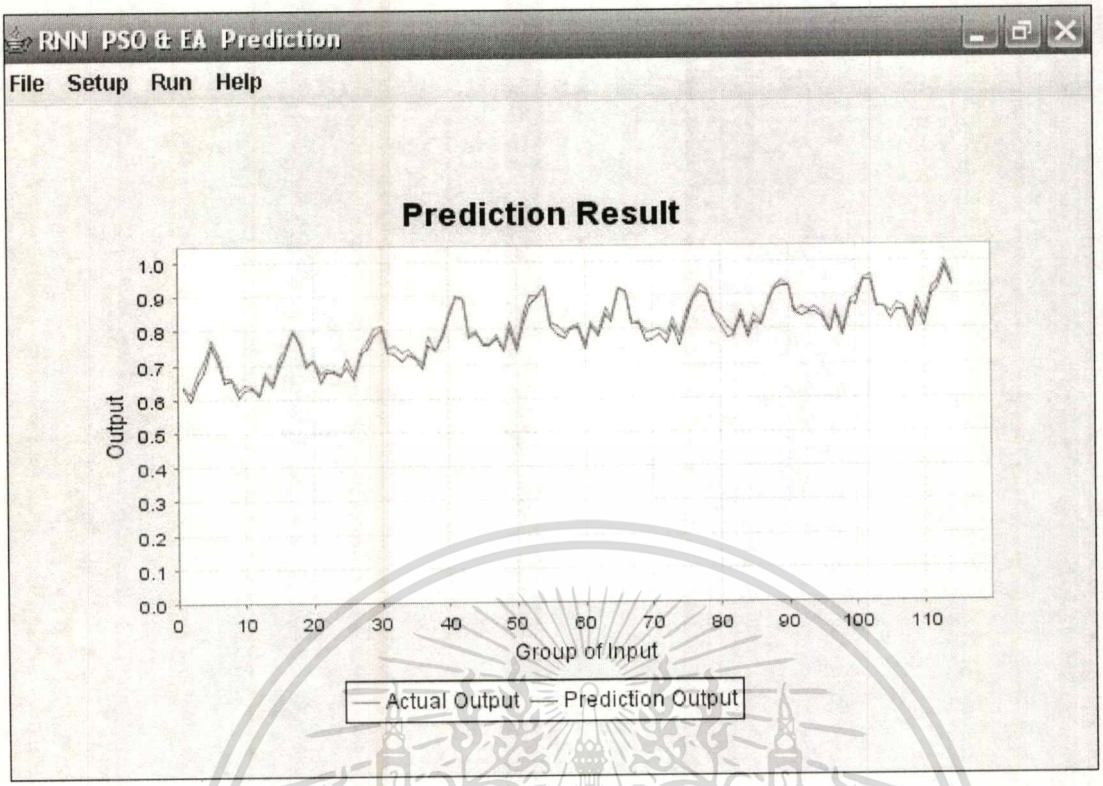


รูปที่ 4.13 แสดงการทดสอบโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 1

จากรูปที่ 4.13 แสดงหน้าจอของผลการทดลองในส่วนของการทดสอบโครงข่ายประสาทเทียม ซึ่งแสดงได้เป็นตารางการเปรียบเทียบของเอาต์พุตที่เป็นเอาต์พุตเป้าหมายที่เกิดขึ้นจริงในอดีต (Actual output) กับเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์ก (Prediction output)

4. สามารถนำข้อมูลเอาต์พุตมาแสดงเป็นกราฟได้ เพื่อให้การดูผลการทำนายของโปรแกรมที่เป็นการเปรียบเทียบเอาต์พุตที่เป็นเอาต์พุตเป้าหมายกับเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์กง่ายขึ้น แสดงดังรูปที่ 4.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 กราฟแสดงผลทดลองโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 1

5. นำค่าจากผลการทดสอบของข้อมูลชุดที่ 1 มาหาค่าความผิดพลาด (Sum Square Error) โดยค่าที่คำนวณได้มีดังนี้

ตารางที่ 4.1 การหาค่าความผิดพลาดของข้อมูลชุดที่ 1

Pattern	Actual Output	Desire Output	Square Error
1	0.636480411	0.632864976	0.00001307137
2	0.615428531	0.592367259	0.00053182224
3	0.674730607	0.651877628	0.00052225867
4	0.706486834	0.678772282	0.00076809639
5	0.772211518	0.756463065	0.00024801376
6	0.730179119	0.70879825	0.00045714159
7	0.663598087	0.647444782	0.00026092927
8	0.661314494	0.656624252	0.00002199837
9	0.62499108	0.603977374	0.00044157584
10	0.642403482	0.625170761	0.00029696670
11	0.636337686	0.628887937	0.00005549876

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pattern	Actual Output	Desire Output	Square Error
12	0.615642618	0.606946002	0.00007563113
13	0.674873332	0.666527223	0.00006965754
14	0.646756583	0.635292359	0.00013142845
15	0.721615643	0.699896363	0.00047172712
16	0.749232855	0.733321388	0.00025317479
17	0.800185542	0.797896523	0.00000523961
18	0.765289374	0.75703812	0.00006808320
19	0.704702776	0.693205344	0.00013219093
20	0.713052166	0.71298478	0.00000000454
21	0.671876115	0.644931136	0.00072603192
22	0.689788054	0.674728214	0.00022679877
23	0.683436809	0.679376788	0.00001648377
24	0.673945622	0.664415639	0.00009082058
25	0.719403411	0.694576962	0.00061635257
26	0.673802897	0.655775212	0.00032499742
27	0.744023407	0.732555394	0.00013151533
28	0.767715693	0.743317622	0.00059526585
29	0.805537715	0.785356619	0.00040727665
30	0.815385713	0.803109964	0.00015069401
31	0.743452508	0.736125966	0.00005367822
32	0.757154071	0.730774923	0.00069585944
33	0.735031756	0.707085423	0.00078099755
34	0.742453436	0.731653802	0.00011663209
35	0.723542425	0.719816284	0.00001388412
36	0.703061443	0.689980822	0.00017110264
37	0.779205024	0.753630012	0.00065408125
38	0.73917077	0.73781435	0.00000183988
39	0.790194819	0.783108612	0.00005021434

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pattern	Actual Output	Desire Output	Square Error
40	0.850139156	0.83750512	0.00015961888
41	0.899093699	0.894502473	0.00002107935
42	0.891529294	0.889214319	0.00000535911
43	0.787911225	0.774801434	0.00017186662
44	0.799257832	0.792640874	0.00004378413
45	0.76300578	0.756089106	0.00004784039
46	0.75865268	0.755250719	0.00001157333
47	0.784271748	0.774945417	0.00008698045
48	0.745522015	0.741919275	0.00001297974
49	0.822878755	0.803262852	0.00038478366
50	0.754799115	0.741920284	0.00016586429
51	0.853493185	0.829707742	0.00056574728
52	0.902162278	0.876471284	0.00066002716
53	0.90316135	0.895268551	0.00006229628
54	0.926996361	0.925176037	0.00000331358
55	0.822165132	0.813758722	0.00007066774
56	0.811746236	0.789329328	0.00050251777
57	0.792335688	0.77805752	0.00020386608
58	0.809248555	0.803390872	0.00003431244
59	0.818097481	0.809020076	0.00008239927
60	0.761364447	0.745784473	0.00024273559
61	0.823663741	0.814113778	0.00009120179
62	0.791051167	0.780768469	0.00010573388
63	0.86726611	0.841982624	0.00063925467
64	0.832655391	0.822551433	0.00010208997
65	0.924141868	0.920290311	0.00001483450
66	0.911938914	0.907571172	0.00001907717
67	0.818597017	0.816130653	0.00000608295

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pattern	Actual Output	Desire Output	Square Error
68	0.821808321	0.818882493	0.00000856047
69	0.792335688	0.767718622	0.00060599995
70	0.795190181	0.773331228	0.00047781382
71	0.804610005	0.787378565	0.00029692251
72	0.787411689	0.762258992	0.00063265815
73	0.833297652	0.813386055	0.00039647171
74	0.781845429	0.757756512	0.00058027595
75	0.855848141	0.833351837	0.00050608371
76	0.890530222	0.8869319	0.00001294792
77	0.933561693	0.909259946	0.00059057491
78	0.913794334	0.902416942	0.00012944505
79	0.856989938	0.847132206	0.00009717487
80	0.838364376	0.812171966	0.00068604233
81	0.807107686	0.778379504	0.00082530842
82	0.791336616	0.783549053	0.00006064614
83	0.852137301	0.837908132	0.00020246925
84	0.791265254	0.775104101	0.00026118287
85	0.846571041	0.825310912	0.00045199309
86	0.818597017	0.812597218	0.00003599759
87	0.8848926	0.875694659	0.00008460212
88	0.921501463	0.915735513	0.00003324618
89	0.945122386	0.929886335	0.00023213726
90	0.928280882	0.928104102	0.00000003125
91	0.849782345	0.849697811	0.00000000715
92	0.864768429	0.839991192	0.00061391149
93	0.853207736	0.850127311	0.00000948902
94	0.864197531	0.849574363	0.00021383705
95	0.843859274	0.832595264	0.00012687791

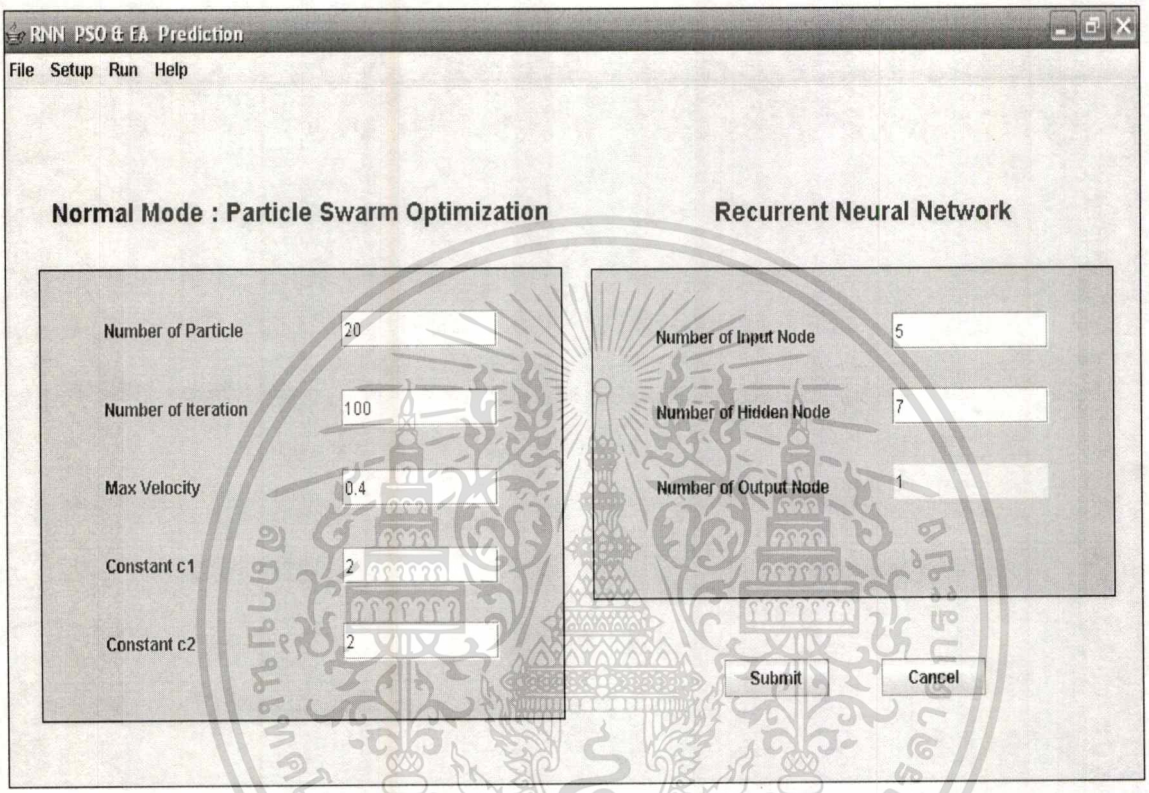
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pattern	Actual Output	Desire Output	Square Error
96	0.797188325	0.793640494	0.00001258711
97	0.866552487	0.850871233	0.00024590171
98	0.796617427	0.781642389	0.00022425175
99	0.88767573	0.872830391	0.00022038409
100	0.897381003	0.869276853	0.00078984329
101	0.946050096	0.941590532	0.00001988772
102	0.956468993	0.94555494	0.00011911655
103	0.870191965	0.863372515	0.00004650490
104	0.864268893	0.864087792	0.00000003280
105	0.84899736	0.826416622	0.00050988973
106	0.87376008	0.857108851	0.00027726341
107	0.866409762	0.853180434	0.00017501513
108	0.815671162	0.79127344	0.00059524886
109	0.889602512	0.866675594	0.00052564355
110	0.833939913	0.807625213	0.00069246345
111	0.922143724	0.900572298	0.00046532639
		MSE	0.00025458553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 การทดลองข้อมูลชุดที่ 2

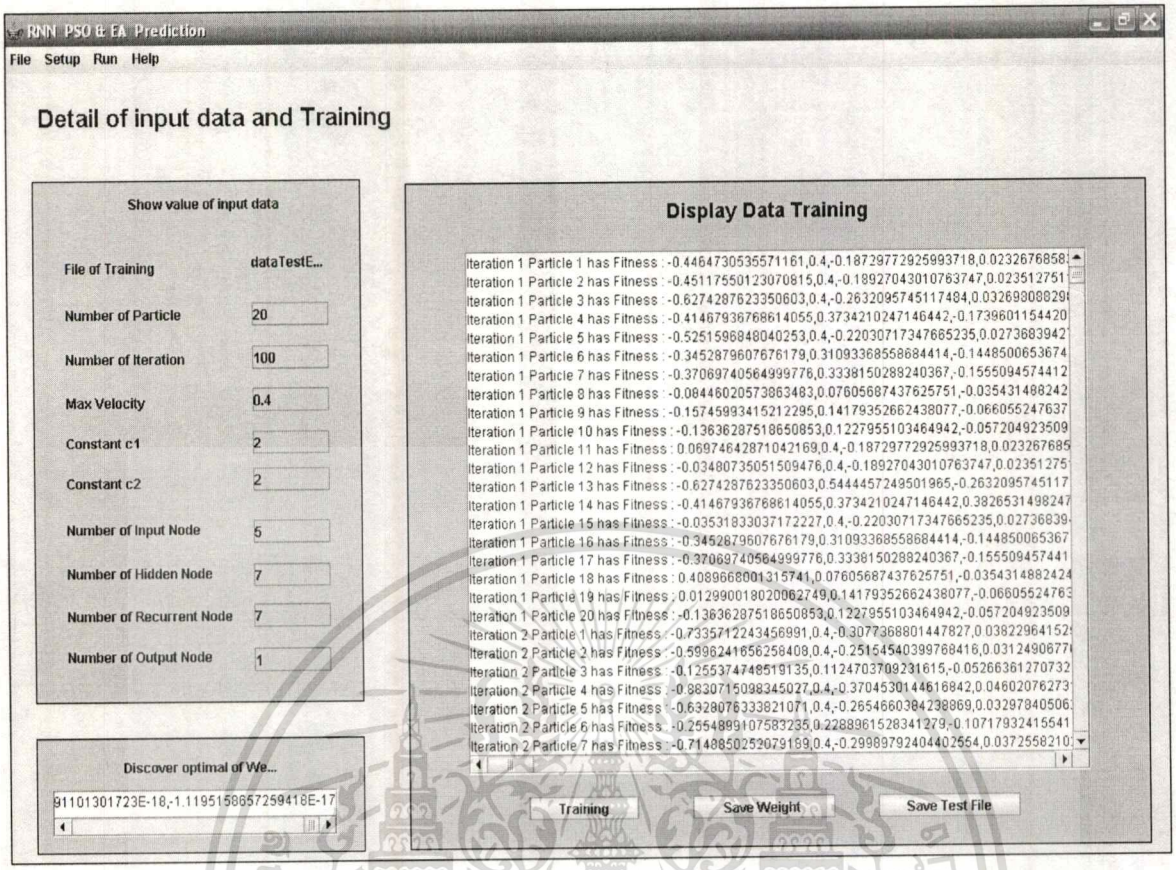
1. สำหรับการทำงานนั้น ก่อนอื่นจะต้องทำการเปิดไฟล์ข้อมูลเข้ามา ในที่นี้ทำการเปิดไฟล์ข้อมูลชุดที่ 2 เข้ามา จากนั้นก็ทำการปรับตั้งค่าโปรแกรมในเมนู Setup > Normal Mode Training ซึ่งจะทำการใส่ค่าดังแสดงในรูปที่ 4.15



รูปที่ 4.15 แสดงการปรับตั้งค่าของโปรแกรมเพื่อทำการทดลองกับข้อมูลชุดที่ 2

2. หลังจากที่ใส่ค่าของข้อมูลเรียบร้อยแล้ว ต่อจากนั้นก็ทำการ Run โปรแกรม ซึ่งในการฝึกหัดโครงข่ายประสาทเทียม ให้เลือกไปที่ Run > Run Training จะปรากฏหน้าจอของเมนู Run Training ขึ้นมา ซึ่งหน้าจอนี้ก็จะแสดงค่าของข้อมูลที่เราใส่เข้ามาให้ตรวจสอบดูว่าข้อมูลที่เราใส่นั้นถูกต้องหรือไม่ เพื่อที่จะทำให้ไม่ต้องเสียเวลาในการฝึกหัดโครงข่ายอีกครั้ง หากตรวจสอบแล้วว่าข้อมูลที่ใส่เข้าไปถูกต้องก็ทำการฝึกหัดโครงข่าย แสดงดังรูปที่ 4.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

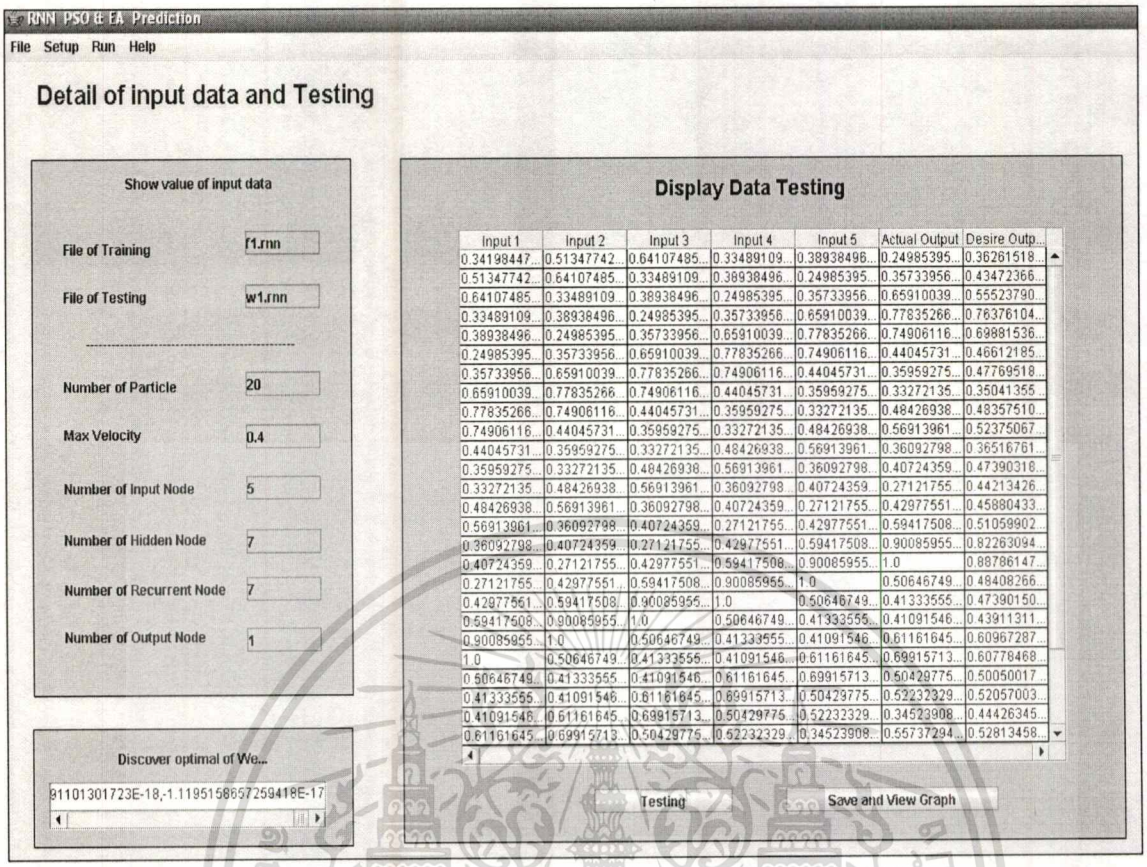


รูปที่ 4.16 แสดงการฝึกหัดโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 2

จากรูปที่ 4.16 เป็นการแสดงผลของการฝึกหัดโครงข่ายประสาทเทียม ซึ่งสามารถจะทำการบันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเพื่อเอาไว้ใช้ในส่วนของการทดสอบโครงข่ายได้ ทำให้ประหยัดเวลาในการทำงาน เนื่องจากไม่ต้องเสียเวลาในการฝึกหัดโครงข่าย สามารถเปิดไฟล์ที่บันทึกค่าถ่วงน้ำหนักมาใช้ทดสอบได้ทันที

3. ทำการทดสอบโครงข่ายประสาทเทียม โดยจะต้องเปิดไฟล์ข้อมูลที่เป็นไฟล์ที่ได้บันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเอาไว้ขึ้นมาทำการทดสอบ จากนั้นทำการทดสอบโครงข่ายประสาทเทียม โดยไปที่ Run > Run Testing แสดงดังรูปที่ 4.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

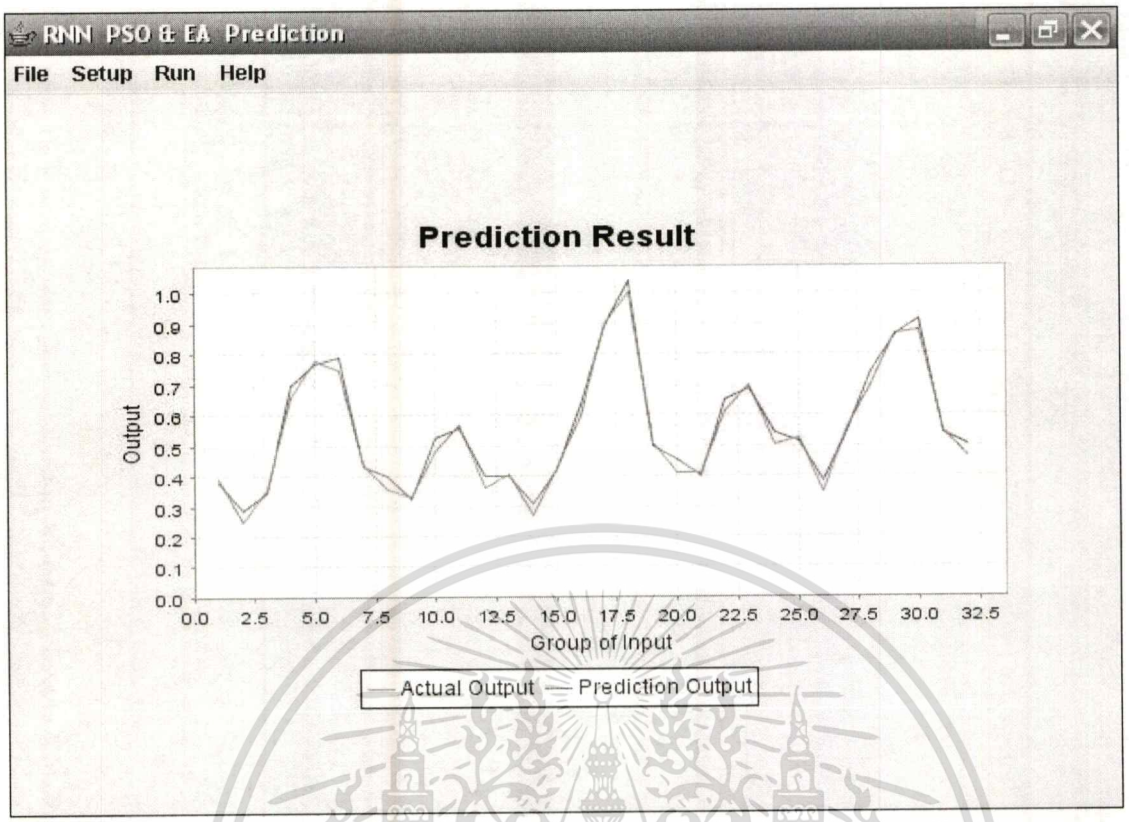


รูปที่ 4.17 แสดงการทดสอบโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 2

จากรูปที่ 4.17 แสดงหน้าจอของผลการทดลองในส่วนของการทดสอบโครงข่ายประสาทเทียม ซึ่งแสดงได้เป็นตารางการเปรียบเทียบของเอาต์พุตที่เป็นเอาต์พุตเป้าหมายที่เกิดขึ้นจริงในอดีต (Actual output) กับเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์ก (Prediction output)

4. สามารถนำข้อมูลเอาต์พุตมาแสดงเป็นกราฟได้ เพื่อให้การดูผลการทำนายของโปรแกรมที่เป็นการเปรียบเทียบเอาต์พุตที่เป็นเอาต์พุตเป้าหมายกับเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์กง่ายขึ้น แสดงดังรูปที่ 4.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 กราฟแสดงผลทดลองโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 2

5. นำค่าจากผลการทดสอบของข้อมูลชุดที่ 2 มาหาค่าความผิดพลาด (Sum Square Error) โดยค่าที่คำนวณได้มีดังนี้

ตารางที่ 4.2 การหาค่าความผิดพลาดของข้อมูลชุดที่ 2

Pattern	Actual Output	Desire Output	Square Error
1	0.24985396	0.362615185	0.0127150939
2	0.357339564	0.434723665	0.0059882990
3	0.659100392	0.555237904	0.0107874164
4	0.778352666	0.763761040	0.0002129155
5	0.749061170	0.698815369	0.0025246405
6	0.440457315	0.466121852	0.0006586685
7	0.359592756	0.477695182	0.0139481830
8	0.332721355	0.350413556	0.0003130140
9	0.484269382	0.483575106	0.0000004820
10	0.569139614	0.523750672	0.0020601561

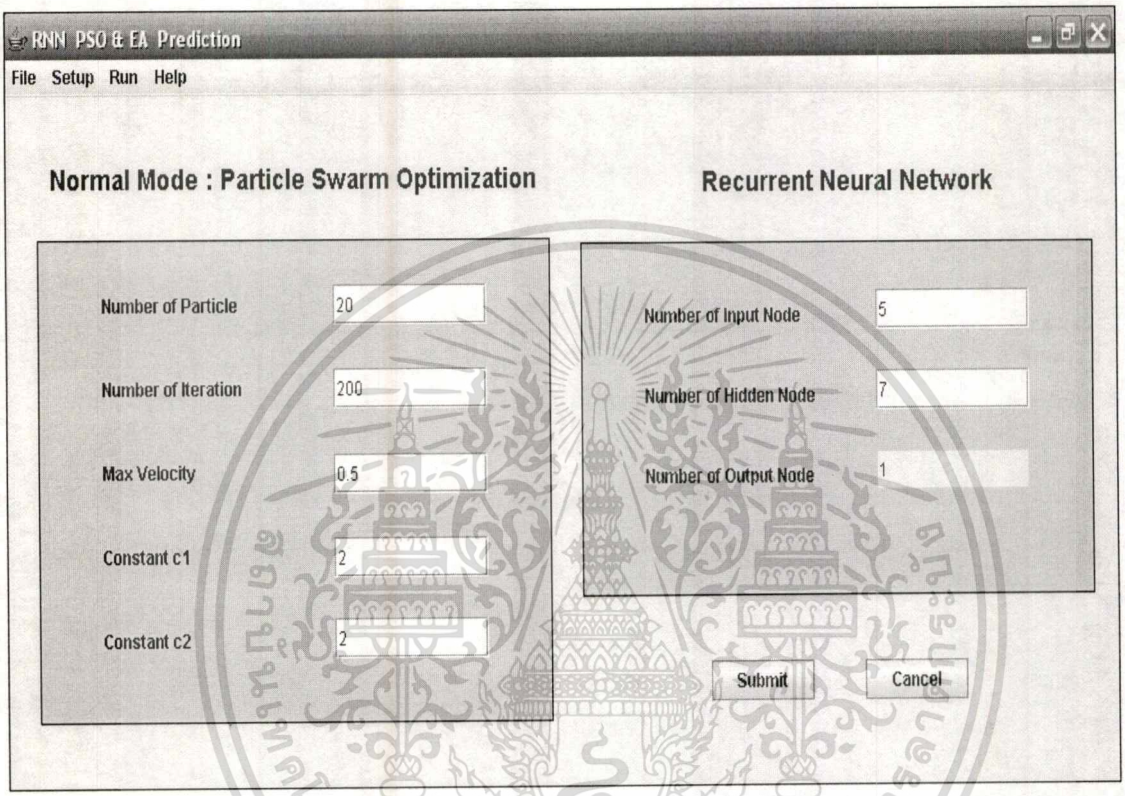
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pattern	Actual Output	Desire Output	Square Error
11	0.360927981	0.365167618	0.0000179745
12	0.407243595	0.473903185	0.0044435009
13	0.271217558	0.442134266	0.0292125209
14	0.429775515	0.458804331	0.0008426722
15	0.594175081	0.510599021	0.0069849579
16	0.900859551	0.822630942	0.0061197152
17	1.000000000	0.887861479	0.0125750480
18	0.506467496	0.484082668	0.0005010805
19	0.413335559	0.473901507	0.0036682341
20	0.410915464	0.439113118	0.0007951077
21	0.611616457	0.609672875	0.0000037775
22	0.699157139	0.607784681	0.0083489261
23	0.504297755	0.500500172	0.0000144216
24	0.522323291	0.520570035	0.0000030739
25	0.345239089	0.444263451	0.0098058243
26	0.557372945	0.528134584	0.0008548818
27	0.700242010	0.660559797	0.0015746780
28	0.868396896	0.799844692	0.0046994046
29	0.878577985	0.805190291	0.0053857537
30	0.549194692	0.519975200	0.0008537788
31	0.466661103	0.475794794	0.0000834243
		MSE	0.0047096008

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 การทดลองข้อมูลชุดที่ 3

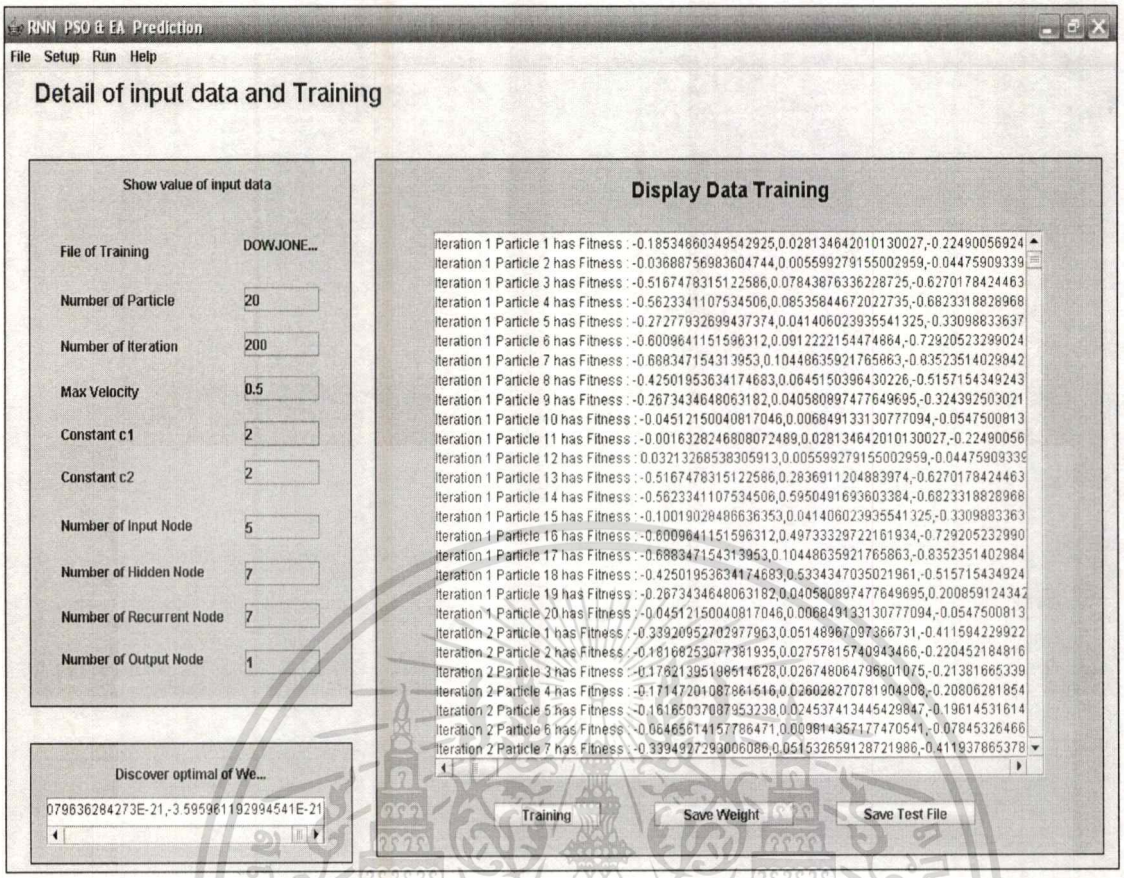
1. เริ่มแรกของการทำงานนั้นก่อนอื่นจะต้องทำการเปิดไฟล์ข้อมูลเข้ามา ในที่นี้ทำการเปิดไฟล์ข้อมูลชุดที่ 3 เข้ามา จากนั้นก็ทำการปรับตั้งค่าโปรแกรมในเมนู Setup > Normal Mode Training ซึ่งจะทำการใส่ค่าดังแสดงในรูปที่ 4.19



รูปที่ 4.19 แสดงการปรับตั้งค่าของโปรแกรมเพื่อทำการทดลองกับข้อมูลชุดที่ 3

2. หลังจากที่ใส่ค่าของข้อมูลเรียบร้อยแล้ว ต่อจากนั้นก็ทำการ Run โปรแกรม ซึ่งทำการฝึกหัดโครงข่ายประสาทเทียม โดยเลือกไปที่ Run > Run Training จะปรากฏหน้าจอของเมนู Run Training ขึ้นมา ซึ่งหน้าจอนี้ก็จะแสดงค่าของข้อมูลที่เราใส่เข้ามาให้ตรวจสอบดูว่าข้อมูลที่เราใส่นั้นถูกต้องหรือไม่ เพื่อที่จะทำให้ไม่ต้องเสียเวลาในการฝึกหัดโครงข่ายอีกครั้ง หากตรวจสอบแล้วว่าข้อมูลที่ใส่เข้าไปถูกต้องก็ทำการฝึกหัดโครงข่าย แสดงดังรูปที่ 4.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

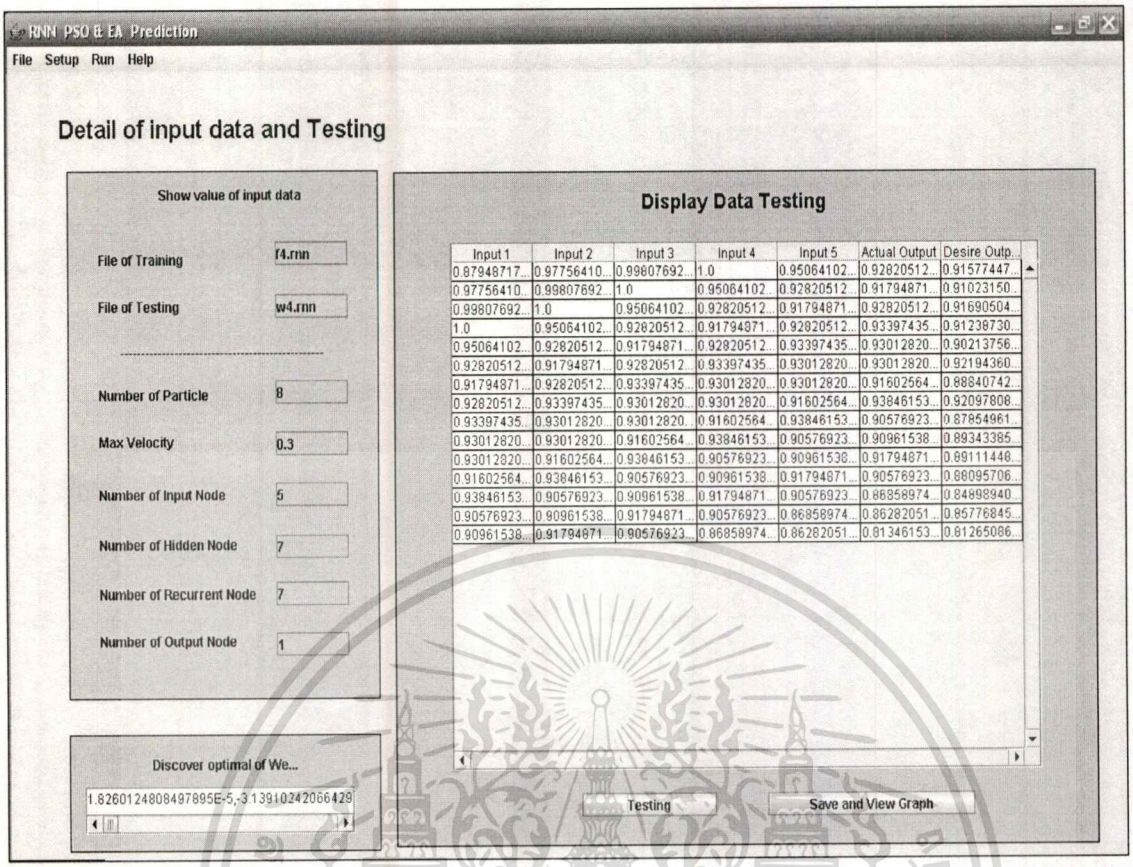


รูปที่ 4.20 แสดงการฝึกหัดโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 3

จากรูปที่ 4.20 เป็นการแสดงผลของการฝึกหัดโครงข่ายประสาทเทียม ซึ่งสามารถจะทำการบันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเพื่อเอาไว้ใช้ในส่วนของการทดสอบโครงข่ายได้ ทำให้ประหยัดเวลาในการทำงาน เนื่องจากไม่ต้องเสียเวลาในการฝึกหัดโครงข่าย สามารถเปิดไฟล์ที่บันทึกค่าถ่วงน้ำหนักมาใช้ทดสอบได้ทันที

3. ทำการทดสอบโครงข่ายประสาทเทียม โดยจะต้องเปิดไฟล์ข้อมูลที่เป็นไฟล์ที่ได้บันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเอาไว้ขึ้นมาทำการทดสอบ จากนั้นทำการทดสอบโครงข่ายประสาทเทียม โดยไปที่ Run > Run Testing แสดงดังรูปที่ 4.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

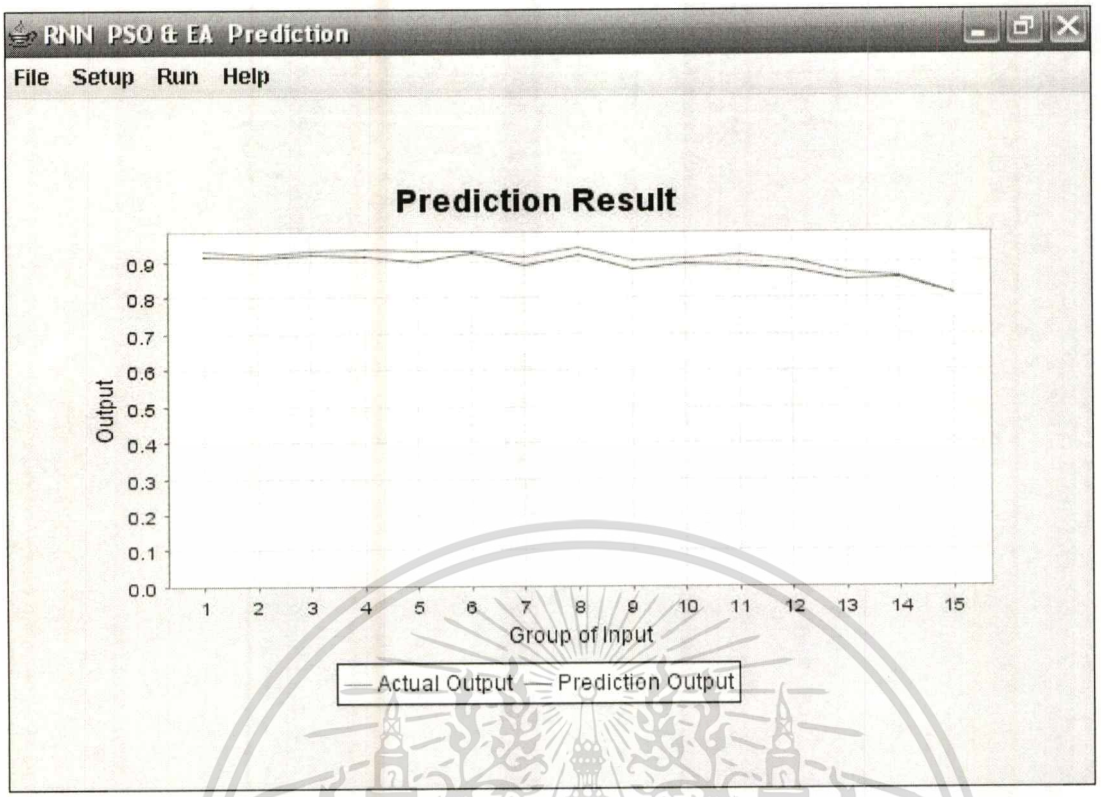


รูปที่ 4.21 แสดงการทดสอบโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 3

จากรูปที่ 4.21 แสดงหน้าจอของผลการทดลองในส่วนของการทดสอบโครงข่ายประสาทเทียม ซึ่งแสดงได้เป็นตารางการเปรียบเทียบของเอาต์พุตที่เป็นเอาต์พุตเป้าหมายที่เกิดขึ้นจริงในอดีต (Actual output) กับเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์ก (Prediction output)

4. สามารถนำข้อมูลเอาต์พุตมาแสดงเป็นกราฟได้ เพื่อให้การดูผลการทำนายของโปรแกรมที่เป็นการเปรียบเทียบเอาต์พุตที่เป็นเอาต์พุตเป้าหมายกับเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์กง่ายขึ้น แสดงดังรูปที่ 4.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 กราฟแสดงผลทดลองโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 3

5. นำค่าจากผลการทดสอบของข้อมูลชุดที่ 3 มาหาค่าความผิดพลาด (Sum Square Error) โดยค่าที่คำนวณได้มีดังนี้

ตารางที่ 4.3 การหาค่าความผิดพลาดของข้อมูลชุดที่ 3

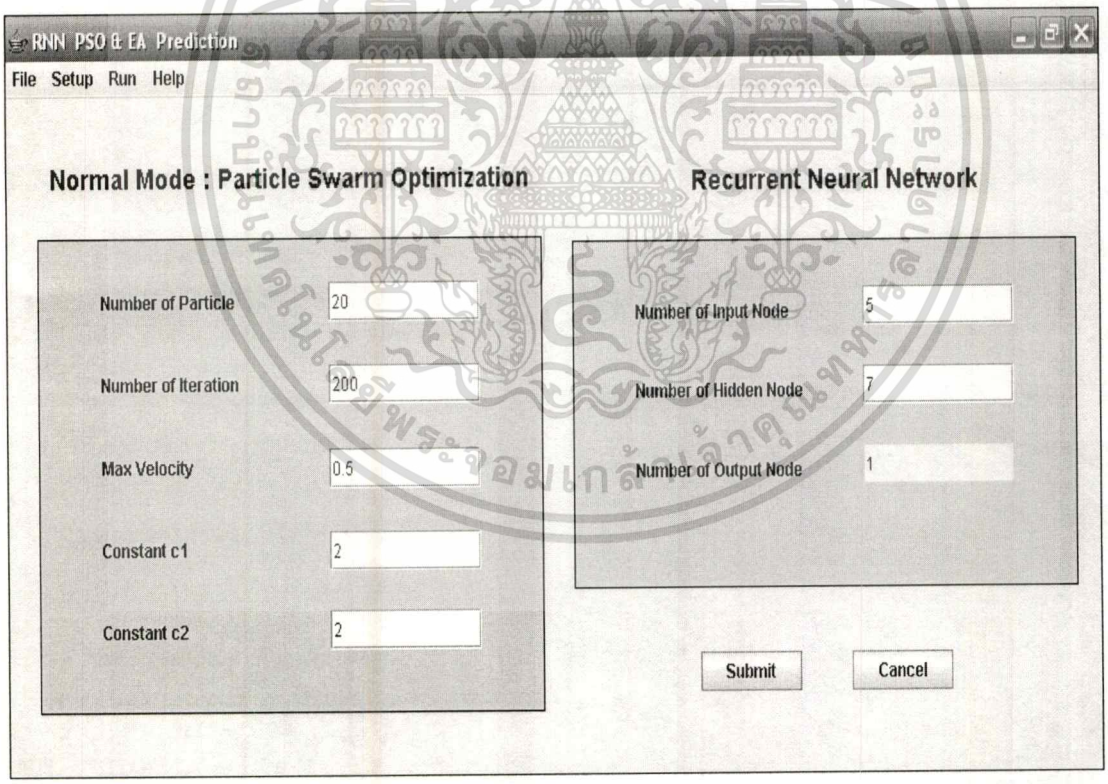
Pattern	Actual Output	Desire Output	Square Error
1	0.928205128	0.915774478	0.0001545211
2	0.917948718	0.910231506	0.0000595554
3	0.928205128	0.916905048	0.0001276918
4	0.933974359	0.912387305	0.0004660009
5	0.930128205	0.902137567	0.0007834758
6	0.930128205	0.921943604	0.0000669877
7	0.916025641	0.888407428	0.0007627657
8	0.938461538	0.920978082	0.0003056713
9	0.905769231	0.878549614	0.0007409075
10	0.909615385	0.893433851	0.0002618420

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pattern	Actual Output	Desire Output	Square Error
11	0.917948718	0.891114487	0.0007200760
12	0.905769231	0.880957064	0.0006156436
13	0.868589744	0.848989402	0.0003841734
14	0.862820513	0.857768452	0.0000255233
15	0.813461538	0.812650867	0.0000006572
		MSE	0.0003650328

4.2.4 การทดลองข้อมูลชุดที่ 4

1. สำหรับการทำงานนั้น ก่อนอื่นจะต้องทำการเปิดไฟล์ข้อมูลเข้ามา ในที่นี้ทำการเปิดไฟล์ข้อมูลชุดที่ 4 เข้ามา จากนั้นก็ทำการปรับตั้งค่าโปรแกรมในเมนู Setup > Normal Mode Training ซึ่งจะทำการใส่ค่าดังแสดงในรูปที่ 4.23

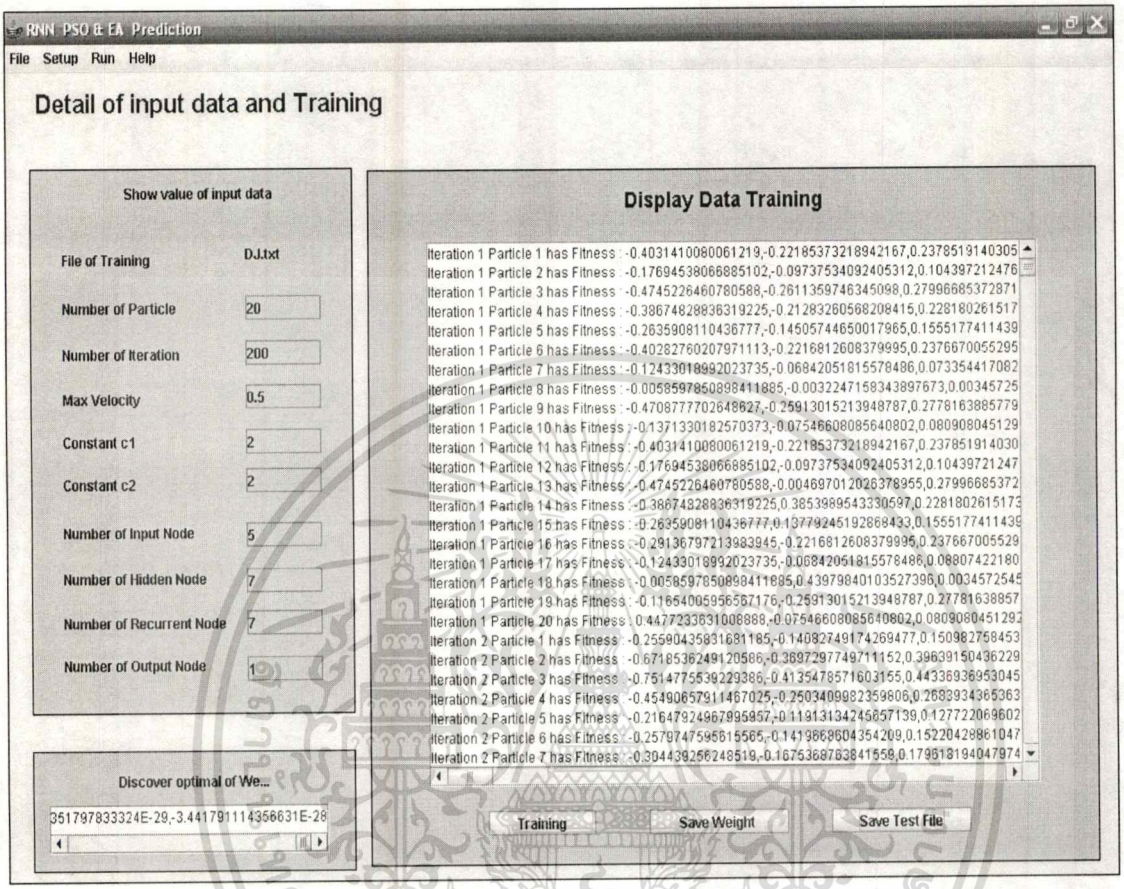


รูปที่ 4.23 แสดงการปรับตั้งค่าของโปรแกรมเพื่อทำการทดลองกับข้อมูลชุดที่ 4

2. หลังจากที่ใส่ค่าของข้อมูลเรียบร้อยแล้ว ต่อจากนั้นก็ทำการ Run โปรแกรม ซึ่งในการฝึกหัดโครงข่ายประสาทเทียม ให้เลือกไปที่ Run > Run Training จะปรากฏหน้าจอของเมนู Run Training ขึ้นมา ซึ่งหน้าจอนี้ก็จะแสดงค่าของข้อมูลที่เราใส่เข้ามาให้ตรวจสอบดูว่าข้อมูลที่เราใส่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั้นถูกต้องหรือไม่ เพื่อที่จะทำให้ไม่ต้องเสียเวลาในการฝึกหัดโครงข่ายอีกครั้ง หากตรวจสอบแล้วว่าข้อมูลที่ใส่เข้าไปถูกต้องก็ทำการฝึกหัดโครงข่าย แสดงดังรูปที่ 4.24

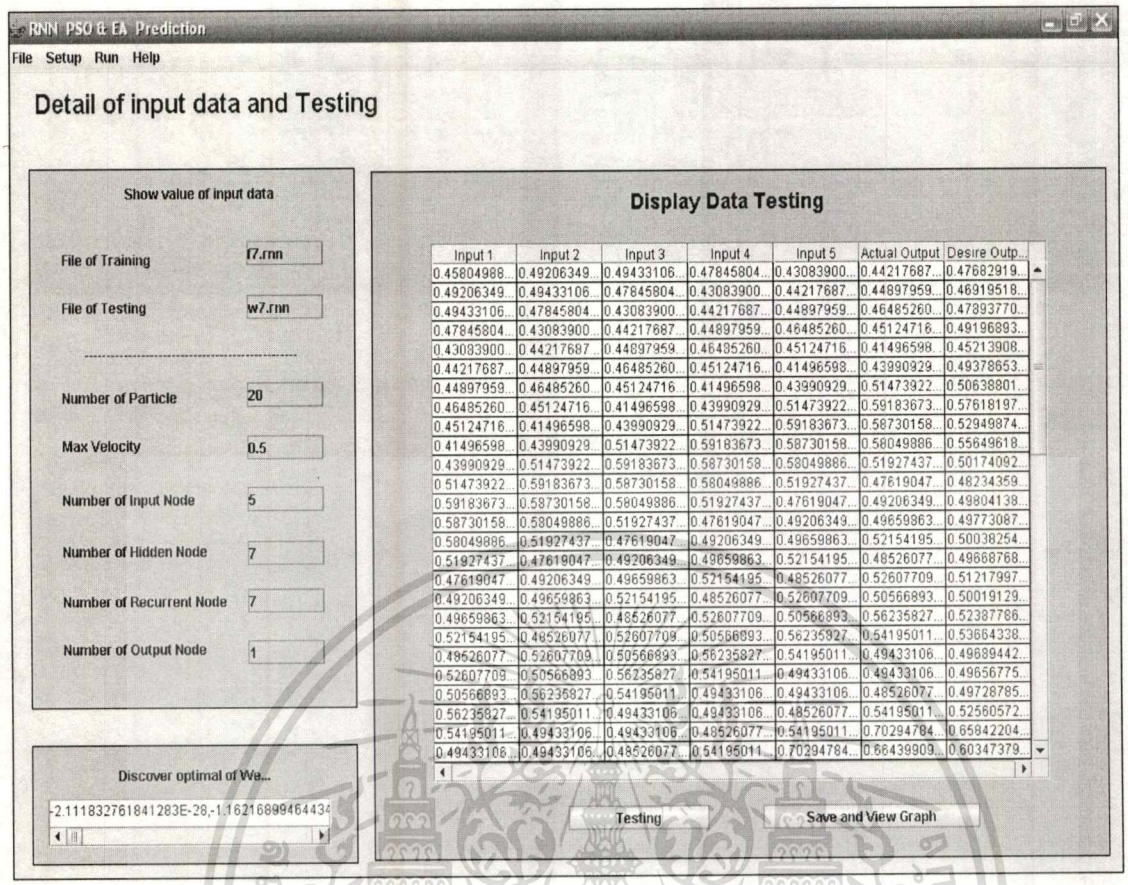


รูปที่ 4.24 แสดงการฝึกหัดโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 4

จากรูปที่ 4.24 เป็นการแสดงผลของการฝึกหัดโครงข่ายประสาทเทียม ซึ่งสามารถจะทำการบันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเพื่อเอาไว้ใช้ในส่วนของการทดสอบโครงข่ายได้ ทำให้ประหยัดเวลาในการทำงาน เนื่องจากไม่ต้องเสียเวลาในการฝึกหัดโครงข่าย สามารถเปิดไฟล์ที่บันทึกค่าถ่วงน้ำหนักมาใช้ทดสอบได้ทันที

3. ทำการทดสอบโครงข่ายประสาทเทียม โดยจะต้องเปิดไฟล์ข้อมูลที่เป็นไฟล์ที่ได้บันทึกค่าถ่วงน้ำหนักที่ดีที่สุดเอาไว้ขึ้นมาทำการทดสอบ จากนั้นทำการทดสอบโครงข่ายประสาทเทียม โดยไปที่ Run > Run Testing แสดงดังรูปที่ 4.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

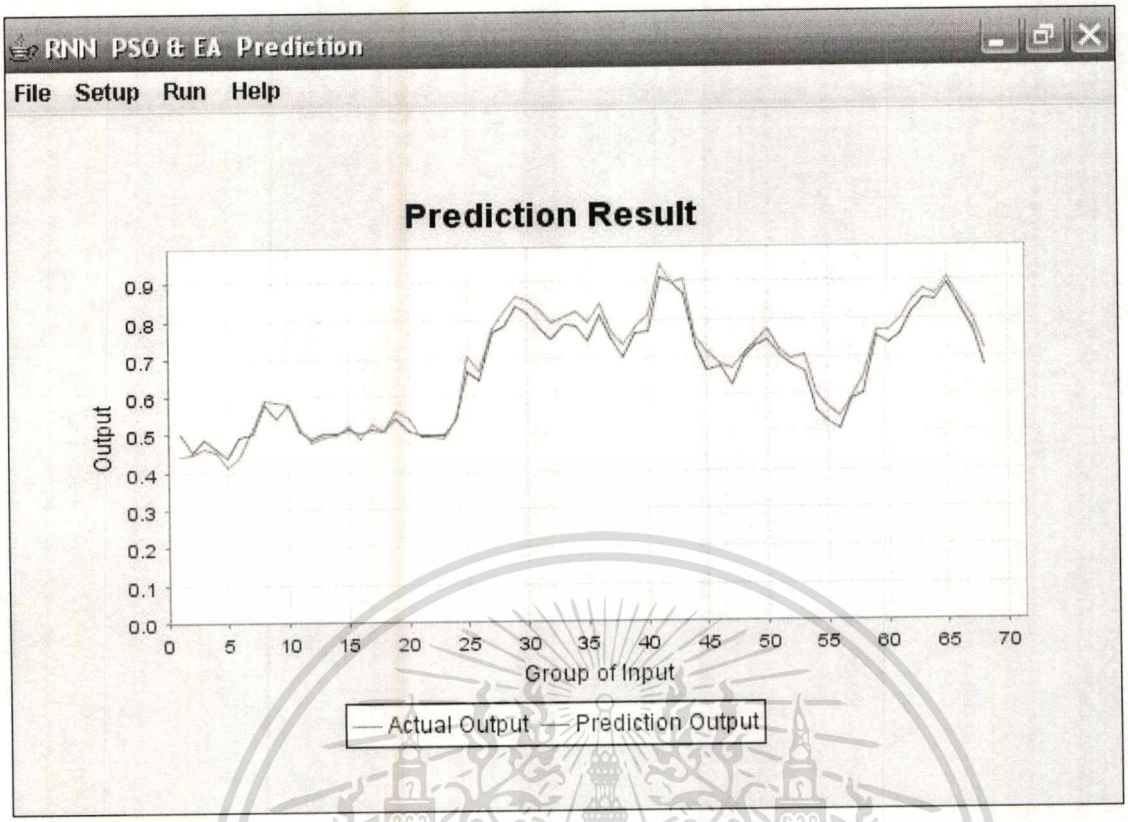


รูปที่ 4.25 แสดงการทดสอบโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 4

จากรูปที่ 4.25 แสดงหน้าจอของผลการทดลองในส่วนของการทดสอบโครงข่ายประสาทเทียม ซึ่งแสดงได้เป็นตารางการเปรียบเทียบของเอาต์พุตที่เป็นเอาต์พุตเป้าหมายที่เกิดขึ้นจริงในอดีต (Actual output) กับเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์ก (Prediction output)

4. สามารถนำข้อมูลเอาต์พุตมาแสดงเป็นกราฟได้ เพื่อให้การดูผลการทำนายของโปรแกรมที่เป็นการเปรียบเทียบเอาต์พุตที่เป็นเอาต์พุตเป้าหมายกับเอาต์พุตที่ได้จากแบบจำลองของนิวรอลเน็ตเวิร์กง่ายขึ้น แสดงดังรูปที่ 4.26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.26 กราฟแสดงผลทดลองโครงข่ายประสาทเทียมของโปรแกรมกับข้อมูลชุดที่ 4

5. นำค่าจากผลการทดสอบของข้อมูลชุดที่ 4 มาหาค่าความผิดพลาด (Sum Square Error) โดยค่าที่คำนวณได้มีดังนี้

ตารางที่ 4.4 การหาค่าความผิดพลาดของข้อมูลชุดที่ 4

Pattern	Actual Output	Desire Output	Square Error
1	0.442176871	0.448241193	0.0000367760
2	0.448979592	0.483822457	0.0012140252
3	0.464852608	0.46911977	0.0000182087
4	0.451247166	0.460550368	0.0000865496
5	0.414965986	0.478260092	0.0040061438
6	0.439909297	0.451271482	0.0001290992
7	0.514739229	0.498318394	0.0002696438
8	0.591836735	0.577978139	0.0001920607
9	0.587301587	0.583393253	0.0000152751
10	0.580498866	0.563512766	0.0002885276

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pattern	Actual Output	Desire Output	Square Error
11	0.519274376	0.517513506	0.0000031007
12	0.476190476	0.480863197	0.0000218343
13	0.492063492	0.492820463	0.0000005730
14	0.496598639	0.496182615	0.0000001731
15	0.52154195	0.519644604	0.0000035999
16	0.485260771	0.493117881	0.0000617342
17	0.526077098	0.517310845	0.0000768472
18	0.505668934	0.500340152	0.0000283959
19	0.562358277	0.549230175	0.0001723470
20	0.541950113	0.537755888	0.0000175915
21	0.494331066	0.495334457	0.0000010068
22	0.494331066	0.494861815	0.0000002817
23	0.485260771	0.495184625	0.0000984829
24	0.541950113	0.531685062	0.0001053713
25	0.702947846	0.69156786	0.0001295041
26	0.664399093	0.659996181	0.0000193856
27	0.780045351	0.764092008	0.0002545092
28	0.820861678	0.819441816	0.0000020160
29	0.861678005	0.848918982	0.0001627926
30	0.85260771	0.842329582	0.0001056399
31	0.825396825	0.817043406	0.0000697796
32	0.79138322	0.776299112	0.0002275303
33	0.806122449	0.799442103	0.0000446270
34	0.820861678	0.814109181	0.0000455962
35	0.79138322	0.779741756	0.0001355237
36	0.841269841	0.832034186	0.0000852973
37	0.766439909	0.762265169	0.0000174285
38	0.732426304	0.721852767	0.0001117997

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pattern	Actual Output	Desire Output	Square Error
39	0.777777778	0.774540556	0.0000104796
40	0.811791383	0.796129919	0.0002452815
41	0.945578231	0.936219765	0.0000875809
42	0.897959184	0.881173547	0.0002817576
43	0.907029478	0.906405953	0.0000003888
44	0.752834467	0.736349166	0.0002717651
45	0.714285714	0.705406822	0.0000788347
46	0.680272109	0.667065591	0.0001744121
47	0.66893424	0.655229319	0.0001878249
48	0.707482993	0.695111035	0.0001530653
49	0.739229025	0.722321542	0.0002858630
50	0.77324263	0.761528967	0.0001372099
51	0.721088435	0.720453035	0.0000004037
52	0.693877551	0.688096393	0.0000334218
53	0.702947846	0.701246671	0.0000028940
54	0.598639456	0.587737334	0.0001188563
55	0.566893424	0.555976673	0.0001191754
56	0.541950113	0.541071881	0.0000007713
		MSE	0.0001865904

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลและข้อเสนอแนะ

จากการศึกษาการทำนายอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA สำหรับสร้างโครงงานพัฒนาระบบนี้ สามารถสรุปผลการดำเนินงาน สรุปผลการทดลองและข้อเสนอแนะ ดังต่อไปนี้

5.1 สรุปผลการดำเนินงาน

จากการศึกษาในโครงงานนี้ สามารถสรุปผลการดำเนินงานในพัฒนาโปรแกรมซึ่งเป็นการประยุกต์การทำงานของโครงข่ายประสาทเทียมให้สามารถทำงานร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA ได้เป็นอย่างดี ซึ่งในการทำงานมีดังนี้

1. โครงข่ายประสาทเทียมเป็นลักษณะกระบวนการและแบบจำลองที่มีความสามารถในการใช้ตัวโครงข่ายประสาทเทียมเองแทนฟังก์ชันใดๆ ก็ได้ที่ต้องการและไม่สามารถมองเห็นว่าลักษณะฟังก์ชันเป็นอย่างไร

2. PSO เป็นอัลกอริทึมที่หาคำตอบที่เหมาะสมที่สุดของภายในกลุ่มประชากรนั้นๆ หลักการของการหาค่าความเหมาะสมแบบกลุ่มอนุภาค คือ เรียนรู้สถานการณ์และใช้ข้อมูลร่วมกัน ทำให้สามารถแก้ปัญหาเกี่ยวกับการหาคำตอบที่เหมาะสมได้

3. EA เป็นอัลกอริทึมที่ใช้ค้นหาคำตอบที่เหมาะสมหรือดีที่สุดของประชากรภายในกลุ่ม ซึ่งเป็นวิธีที่ใช้ในการหาคำตอบที่เหมาะสมที่สุดของประชากรนั้นจะสามารถค้นพบคำตอบที่ดีที่สุดที่มีประสิทธิภาพ

4. การทำ Mutation หรือการกลายพันธุ์ คือการเปลี่ยนแปลงลักษณะของยีนของประชากรไปจากเดิมที่ควรเป็นไปตามการถ่ายทอด ซึ่งเป็นสาเหตุของการเกิดลักษณะที่แปลกๆ อย่างมากมายของสิ่งมีชีวิต การกลายพันธุ์นั้นทุกลักษณะในแต่ละยีนย่อมมีโอกาสที่จะเกิดการเปลี่ยนแปลงได้พอกัน

5. ข้อมูลอินพุตที่จะนำมาใช้กับโครงข่ายประสาทเทียมนั้น จะต้องทำการนอร์มอลไลซ์ค่าให้อยู่ในช่วงที่โปรแกรมกำหนดก่อนเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 สรุปผลการทดลอง

จากการศึกษาในโครงการพัฒนาระบบงานนี้ สามารถสรุปผลการทดลองในการออกแบบการประยุกต์โครงข่ายประสาทเทียมให้ใช้งานในการทำนายร่วมกันอัลกอริทึมแบบผสมของ PSO และ EA ได้ดังนี้

1. การนำข้อมูลมาใช้ในการทำนายกับโครงข่ายประสาทเทียมนั้นมีความสำคัญเป็นอย่างยิ่ง หากเราใช้ข้อมูลปริมาณยิ่งมากเพียงใด ประสิทธิภาพของผลที่ได้จากการทำนายก็จะเพิ่มมากขึ้นตามไปด้วย

2. จำนวนค่าปริมาณของพาติเคิลที่ใช้ในการทดลองนั้นส่งผลบ้างต่อค่าความเหมาะสมที่ได้ แต่จะไม่ส่งผลมากนักต่อการค้นหาค่าที่เหมาะสมที่สุด

3. จำนวนค่าของรอบที่ใช้ในการทดลองนั้น ส่งผลค่อนข้างมากต่อการหาค่าความเหมาะสมที่สุด เนื่องจากยิ่งจำนวนรอบยิ่งมาก จะทำให้ผลการทำนายของโปรแกรมมีความแม่นยำมากยิ่งขึ้น

4. เวลาที่ใช้ในการฝึกหัดโครงข่ายจะแปรผันโดยตรงกับค่าของจำนวนพาติเคิลที่เรากำหนดขึ้น และขึ้นอยู่กับจำนวนรอบที่เราทำการฝึกหัดด้วยเช่นกัน

5. สำหรับการฝึกหัดโครงข่ายประสาทเทียมในแต่ละครั้งนั้น ถึงแม้ว่าเราจะกำหนดตัวแปรนำเข้าที่เอาไว้ทำการปรับตั้งค่าเหมือนกันหมด แต่ประสิทธิภาพในการทำนายของแต่ละครั้งจะได้ค่าไม่เท่ากัน ทั้งนี้เนื่องจากเป็นผลที่น่าจะมาจากการสุ่มค่าของโปรแกรม

6. ค่าของจำนวนโหนดในชั้นฮิดเดนนั้น จะส่งผลดีต่อการค้นหาค่าถ่วงน้ำหนักที่ถูกต้องที่สุดของโปรแกรม โดยยิ่งจำนวนโหนดยิ่งมากขึ้นเท่าไร ก็จะส่งผลให้ผลลัพธ์ที่ได้มีโอกาสที่จะได้ค่าถ่วงน้ำหนักที่มีค่าความคลาดเคลื่อนน้อยมากเท่านั้น

7. การนำ Evolutionary algorithm ในส่วนของการดำเนินการของการกลายพันธุ์ (Mutation) นั้น โดยที่ Mutation จะทำหน้าที่ป้องกันข้อผิดพลาดของวิธีการแก้ไขปัญหาทั้งหมดในการเข้าไปในเฉพาะปัญหา (Local minimum) ที่ดีที่สุดของการแก้ปัญหา

8. ค่าที่เหมาะสมสำหรับการปรับตั้งค่าให้กับโปรแกรมในส่วนของโครงข่ายประสาทเทียมและส่วนของ PSO นั้น จะมีประสิทธิภาพมากเพียงใดนั้น ขึ้นอยู่กับลักษณะของฟังก์ชันที่จะให้โครงข่ายประสาทเทียมเรียนรู้ด้วยเช่นกัน

9. ค่าถ่วงน้ำหนักที่ได้มาจากการสุ่ม มีผลต่อการทำงานของโปรแกรม รวมไปถึงผลที่จะได้จากการประมวลผลด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ข้อเสนอแนะ

จากการทดลองใช้งานโปรแกรมการทำนายข้อมูลอนุกรมเวลาโดยใช้หลักการรีเคอร์เร็นท์นิวรอลเน็ตเวิร์กร่วมกับอัลกอริทึมแบบผสมของ PSO และ EA นั้น สำหรับในส่วนการนำข้อมูลมาใช้กับอัลกอริทึมของ PSO และ EA นั้น ควรจะแบ่งข้อมูลออกเป็นสองส่วนโดยที่ไม่มีการตัดทิ้งข้อมูลที่มีค่าความเหมาะสมต่ำออกไปก่อน ควรที่จะนำข้อมูลมาแล้วนำมาดำเนินการทั้งส่วนของ PSO และ EA ในปริมาณที่ได้แบ่งออกเป็นสองส่วนแล้วเท่าๆ กัน จากนั้นจึงนำมาทำการหาค่าความเหมาะสมที่ดีที่สุดต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

กิตติ ภักดีวัฒนะกุล. 2547. **คัมภีร์ JAVA.** เล่ม 1. พิมพ์ครั้งที่ 4. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด.

วีรศักดิ์ ซึงถาวร. 2548. **JAVA PROGRAMMING**. Volume 1,2,3. กรุงเทพฯ : ซีเอ็ดยูเคชั่น จำกัด

Hiroyuki Mori, Toshiji Ogasawara. **A Recurrent Neural Network for Short-Term Load Forecasting.** [Online]. Available : http://ieeexplore.ieee.org/iel2/696/6640/ 00264315.pdf

Jakob Vesterstrøm and Jacques Riget. 2002. **Particle Swarms.** [Online]. Available : http://www.daimi.au.dk/~jve/Publications/MastersThesis.pdf

Jeff Riley and Victor B. Ciesielski. 2003. **An Evolutionary Approach to Training Feed-Forward and Recurrent Neural Networks.** [Online]. Available : http://www.rileys.id.au/ JeffsPublications/ KES98.pdf

John F. Kolen and Stefan C. Kremer. 2001. **Dynamical recurrent networks.** United States of America.

Kumar Chellapilla and David B. Fogel.2003. **Evolution, Neural Networks, Games, and Intelligence.**[Online]. Available : http://www.cs.nott.ac.uk/~gxk/courses/g5baim/papers/ evolve-nn-001/ ChellapillaAndFogelProcIEEEText.PDF

Larry Mecsker. 2000. **Recurrent neural network : Design and Application.** United States of America.

Matthew Settles. 2005. **An Introcuction to Particle Swarm Optimization.** [Online]. Available : http://www.cs.uidaho.edu/~tsoule/cs472fall05/particleswarm.pdf.

Mikael Boden.2003 **A guide to recurrent neural networks and backpropagation.** [Online]. Available :http://www2.hh.se/staff/mibo/publications/rn2.pdf

Rui Mendes. 2004. **Population Topologies and Their In_uence in Particle Swarm Performance**. [Online]. Available : http://www.di.uminho.pt/~rcm/publications/RuiPhD.pdf,

Stephan Blum and Romanas Puisa. 2005. **Adaptive Mutation Strategies for Evolutionary Algorithms.** [Online]. Available : http://www.dynardo.de/Paper/AdaptiveMutation.pdf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียนโครงงาน

ชื่อ-นามสกุล	นางสาวใกล้รุ่ง สันติผลธรรม
วัน เดือน ปีเกิด	4 มีนาคม 2523
ที่อยู่	ง.112/233 ถ.ดาวดึงส์ ซอยหิมพานต์ ต.ปากน้ำโพ อ.เมืองนครสวรรค์ จ.นครสวรรค์
ประวัติการศึกษา	2547 วิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏนครสวรรค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้