ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

การเปรียบเทียบอัลกอริทึมในการเรียนรู้แบบแพร่ย้อนกลับและอาร์ทแมพสำหรับการแบ่งกลุ่มข้อมูล

COMPARISON OF BACKPROPAGATION AND ARTMAP LEARNING ALGORITHMS FOR CLASSIFICATION

โดย

พินิจ เล้ารุ่งเรืองเดช

PHINIT LAORUNGRUANGDECH

H006099

อาจารย์ที่ปรึกษา

รศ.ดร. อาริต ธรรมโน

วิ.
พ186ก
2551

เลขหมู่ 06099
เลขทะเบียน
วัน,เดือน,ปี 14 มี.ค. 2553

b.19203907
i.

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่ 2 ปีการศึกษา 2551

COMPARISON OF BACKPROPAGATION AND ARTMAP LEARNING ALGORITHMS FOR CLASSIFICATION

PHINIT LAORUNGRUANGDECH

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

2/ 2008

COPYRIGHT 2009

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อ	การเปรียบเทียบอัลกอริทึมในการเรียนรู้แบบแพร่ย้อนกลับและอาร์ทแมพสำหรับการแบ่งกลุ่มข้อมูล
นักศึกษา	นายพินิจ เล้ารุ่งเรืองเดช
รหัสนักศึกษา	49066401
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2551
อาจารย์ที่ปรึกษา	รศ.ดร.อาริต ธรรมโน

บทคัดย่อ

การจำแนกหมวดหมู่ (Classification) เป็นกระบวนการหนึ่งในการทำ Data Mining เพื่อจำแนกหมวดหมู่ของข้อมูลออกตามกลุ่มที่ได้กำหนดไว้ การเลือกใช้วิธีการที่เหมาะสมจะช่วยให้การจำแนกหมวดหมู่ที่มีประสิทธิภาพมากยิ่งขึ้น โครงการพัฒนาระบบงานนี้จึงพัฒนาโปรแกรมขึ้นเพื่อเปรียบเทียบความสามารถในการจำแนกหมวดหมู่ข้อมูล โดยใช้โครงข่ายประสาทเทียมประเภท Backpropagation และ Simplified Fuzzy ARTMAP (SFAM) โดยศึกษาจากประสิทธิภาพด้านผลลัพธ์ที่ได้ เพื่อให้ได้อัลกอริทึมที่เหมาะสมที่สุดในการจำแนกหมวดหมู่

Title	Comparison of Backpropagation and Artmap Learning Algorithms for Classification
Student	Mr Phinit Laorungruangdech
Student ID.	49066401
Degree	Master of Science
Programme	Information Science
Academic Year	2008
Advisor	Associate. Professor Arit Thummano, Ph.D

ABSTRACT

Classification is once process in many process of data mining for classifies data into supervised class. Using a proper system will make us do this method effectively. So, An Application in this project is developed to compare between classification with using Backpropagation algorithm and Simplified Fuzzy ARTMAP (SFAM) by studying from result of both algorithm to get an algorithm which is appropriate for classification.

กิตติกรรมประกาศ

โครงการศึกษาพัฒนาระบบการเปรียบเทียบอัลกอริทึมการเรียนรู้แบบแพร่ย้อนกลับและอาร์ทแมพสำหรับการแบ่งกลุ่มข้อมูลจะสำเร็จลุล่วงไปไม่ได้เลย ถ้าไม่ได้รับการช่วยเหลือและแรงสนับสนุนจากบุคคลสำคัญหลายท่าน ดังต่อไปนี้

ข้าพเจ้าขอกราบขอบพระคุณ รศ.ดร.อาริต ธรรมโน ซึ่งเป็นอาจารย์ที่ปรึกษาโครงการนี้ ที่ให้ความกรุณาให้คำแนะนำ และปรึกษา ข้าพเจ้ารู้สึกซาบซึ้งและขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณเพื่อน ๆ พี่ ๆ น้อง ๆ ทุกคนที่ให้คำแนะนำ และคอยให้กำลังใจเสมอมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำโครงการพัฒนาระบบงานฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์จากโครงการพัฒนาระบบงานฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

พินิจ เล้ารุ่งเรืองเดช

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญรูป VII

บทที่ 1 บทนำ 1

1.1 ความเป็นมา 1

1.2 วัตถุประสงค์ 1

1.3 ขอบเขตการดำเนินงาน 2

1.4 ขั้นตอนการศึกษา 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

1.6 เครื่องมือที่ใช้ในการพัฒนาระบบ 3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 4

2.1 Data Mining 4

2.1.1 กระบวนการทำ Data Mining 5

2.1.2 Data Mining Operation 8

2.1.3 การประยุกต์ใช้ Data Mining 9

2.2 ทฤษฎีโครงข่ายประสาทเทียม 10

2.2.1 การเรียนรู้สำหรับโครงข่ายประสาทเทียม 14

2.3 โครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ(Backpropagation) 16

2.3.1 ขั้นตอนการฝึกหัดโครงข่ายประสาทเทียมแบบแพร่แบบย้อนกลับ 17

2.4 Simplified Fuzzy ARTMAP (SFAM) 19

บทที่ 3 วิธีการดำเนินงาน 22

3.1 ข้อมูลที่ใช้กับระบบ 22

3.2 ขั้นตอนการดำเนินงาน 24

สารบัญ (ต่อ)

หน้า

3.3 การพัฒนาโปรแกรมประยุกต์ 28
3.3.1 เครื่องมือที่ใช้ในการพัฒนาโปรแกรม 28
3.3.2 ตัวอย่างส่วนติดต่อผู้ใช้โปรแกรม 28
บทที่ 4 ผลการทดลอง 34
4.1 การใช้งานระบบ 34
4.2 ข้อมูลที่ใช้กับโปรแกรมประยุกต์ 51
4.3 ผลการทดลองและวิเคราะห์ 54
4.3.1 Iris Data 56
4.3.2 Lung Cancer Data 66
4.3.3 Wine Data 76
4.3.4 Breast Data 86
4.3.5 วิเคราะห์ผลการทดลอง 96
บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ 97
5.1 สรุปผลการศึกษา 97
5.2 ประโยชน์ที่ได้รับจากการศึกษาและพัฒนาระบบ 98
5.3 ข้อเสนอแนะ 98
บรรณานุกรม 99
ประวัติผู้เขียน 100

สารบัญตาราง

ตารางที่ หน้า

2.1 แสดงการเปรียบเทียบโครงข่ายประสาททางชีววิทยาและโครงข่ายประสาทเทียม 11

3.2 การคำนวณหามูลค่าปัจจุบันสุทธิของการพัฒนาระบบสนับสนุนการเบิกค่าใช้จ่าย..... 18

4.1 แสดงผลลัพธ์การทดลอง Iris ด้วยตัวอย่าง (Data Train) 100 ตัวอย่าง..... 56

4.2 แสดงผลลัพธ์การทดลอง Iris ด้วยตัวอย่าง (Data Test) 60 ตัวอย่าง 61

4.3 แสดงผลลัพธ์การทดลอง Lung Cancer ด้วยตัวอย่าง 24 ตัวอย่าง 66

4.4 แสดงผลลัพธ์การทดลอง Lung Cancer ด้วยตัวอย่าง (Data Test) 8 ตัวอย่าง..... 71

4.5 แสดงผลลัพธ์การทดลองWine Data ด้วยตัวอย่าง (Data Train) 100 ตัว..... 76

4.6 แสดงผลลัพธ์การทดลอง Wine Data ด้วยตัวอย่าง (Data Test) 78 ตัวอย่าง..... 81

4.7 แสดงผลลัพธ์การทดลอง Breast Data ด้วยตัวอย่าง 420 ตัวอย่าง 86

4.8 แสดงผลลัพธ์การทดลอง Breast Data ด้วยตัวอย่าง (Data Test) 279 ตัวอย่าง..... 91

สารบัญรูป

รูปที่ หน้า

2.1 กระบวนการทำ Data Mining 5
2.2 แสดง Model ของ Neuron ในสมองมนุษย์ 11
2.3 แสดง Model ของ Neuron ในคอมพิวเตอร์ 11
2.4 แสดงโครงสร้างโครง Neural Network 12
2.5 แสดงลักษณะของสมการ Threshold 13
2.6 แสดงกราฟของฟังก์ชันไบนารีซิกมอยด์ 14
2.7 แสดงกราฟของฟังก์ชันโพล่าซิกมอยด์ 14
2.8 แสดงการเรียนรู้แบบมีการสอน 15
2.9 แสดงการเรียนรู้แบบไม่มีการสอน 15
2.10 แสดงสถาปัตยกรรมของ MFNN 17
2.11 โครงสร้างของ SFAM 19
3.1 ตัวอย่างข้อมูลที่นำมาใช้ 23
3.2 ตัวอย่าง Text File ของ IRIS ก่อนแปลง 24
3.3 ตัวอย่าง Text File ของ IRIS หลังแปลงข้อมูลในตำแหน่งเป้าหมาย 25
3.4 ตัวอย่างหน้าจอหลักของโปรแกรม 29
3.5 ตัวอย่างหน้าต่าง Backpropagation 30
3.6 ตัวอย่างหน้าต่างโปรแกรม SFAM 32
4.1 หน้าจอเมนูหลักของโปรแกรมประยุกต์ Classification Application 34
4.2 หน้าต่างโปรแกรมแบคพรอพพาเกชันเน็ทเวิร์ค 35
4.3 แสดงส่วนหน้าจอของค่าที่ต้องทำการตั้งค่าก่อนการสอนโครงข่าย 36
4.4 แสดงปุ่มLoad ที่ทำการเลือกจาก ส่วนของการฝึกสอนโครงข่าย 37
4.5 Dialog Box ในการเลือก Text File 37
4.6 หน้าต่างแสดงข้อมูลที่ import มาจาก Text File 38
4.7 แสดงปุ่ม Normalize 39
4.8 หน้าต่างแสดงข้อมูลที่ทำการแปลงข้อมูลแล้ว 40
4.9 หน้าต่างโปรแกรมแสดงปุ่ม Train 40
4.10 แสดงน้ำหนักของแต่ละเส้นเชื่อมnode 41

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.11	แสดงปุ่ม Load ในส่วนของการทดสอบโครงข่าย	42
4.12	แสดงDialog Box เพื่อเลือกText File เพื่อนำมาทดสอบโครงข่าย	42
4.13	แสดงปุ่ม Normalize ในการ Test ระบบ	43
4.14	แสดงผลลัพธ์ของการทดสอบโครงข่าย (Test)	44
4.15	แสดงกราฟผลลัพธ์จากการทดสอบระบบBackpropagation	45
4.16	แสดงหน้าจอหลักของโปรแกรม SFAM	45
4.17	แสดง Dialog Box เพื่อเลือก Text File ในการสอนโครงข่าย SFAM	46
4.18	แสดงหน้าต่างSFAM หลังจากทำการอ่านข้อมูลจาก Text File	47
4.19	แสดงหน้าจอหลังจากแปลงข้อมูลใน SFAM	47
4.20	แสดงหน้าจอ SFAM หลังจากทำการฝึกสอนแล้ว	48
4.21	แสดง Dialog Box เพื่อเลือก Text File นำมาใช้ในการทดสอบโครงข่าย SFAM	49
4.22	แสดงผลลัพธ์จากการทดสอบโครงข่าย SFAM	49
4.23	แสดงผลลัพธ์จากการทดสอบโครงข่ายSFAMเทียบกับคำตอบ	50
4.24	แสดงผลการเปรียบเทียบผลลัพธ์จาก 2 อัลอริทึมและคำตอบของข้อมูลนั้น	51
4.25	ข้อมูล Iris Data	52
4.26	ข้อมูล Lung Cancer	53
4.27	ข้อมูล Wine Data	54
4.28	ข้อมูล Breast Data	55
4.29	ผลการทดลองของ Iris Data 100 ตัวอย่าง (1รอบ)	57
4.30	ผลการทดลองของ Iris Data 100 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล	57
4.31	ผลการทดลองของ Iris Data 100 ตัวอย่าง(10รอบ)	58
4.32	ผลการทดลองของ Iris Data 100 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล	58
4.33	ผลการทดลองของ Iris Data Train 100 ตัวอย่าง(100รอบ)	59
4.34	ผลการทดลองของ Iris Data 100 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล	59
4.35	ผลการทดลองของ Iris Data 100 ตัวอย่าง (1000รอบ)	60
4.36	ผลการทดลองของ Iris Data 100 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล	60
4.37	ผลการทดลองของ Iris Data 50 ตัวอย่าง (1รอบ)	62
4.38	ผลการทดลองของ Iris Data 50 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล	62

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.39 ผลการทดลองของ Iris Data 50 ตัวอย่าง (10รอบ) 63

4.40 ผลการทดลองของ Iris Data 50 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล 63

4.41 ผลการทดลองของ Iris Data 50 ตัวอย่าง (100รอบ) 64

4.42 ผลการทดลองของ Iris Data 50 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล 64

4.43 ผลการทดลองของ Iris Data 50 ตัวอย่าง (1000รอบ) 65

4.44 ผลการทดลองของ Iris Data 50 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล 65

4.45 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (1รอบ) 67

4.46 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล 67

4.47 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (10รอบ) 68

4.48 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล 68

4.49 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (100รอบ) 69

4.50 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล 69

4.51 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (1000รอบ) 70

4.52 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล 70

4.53 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (1รอบ) 72

4.54 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล 72

4.55 ผลการทดลองของ Lung Cancer Data Test Backpropagation (10รอบ) 73

4.56 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล 73

4.57 ผลการทดลองของ Lung Cancer Data Test Backpropagation (100รอบ) 74

4.58 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล 74

4.59 ผลการทดลองของ Lung Cancer Data Test Backpropagation (1000รอบ) 75

4.60 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล 75

4.61 ผลการทดลองของ Wine Data จำนวน 100 ตัวอย่าง (1รอบ) 77

4.62 ผลการทดลองของ Wine Data 100 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล 77

4.63 ผลการทดลองของ Wine Data จำนวน 100 ตัวอย่าง (10รอบ) 78

4.64 ผลการทดลองของ Wine Data 100 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล 78

4.65 ผลการทดลองของ Wine Data จำนวน 100 ตัวอย่าง (100รอบ) 79

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.66 ผลการทดลองของ Wine Data 100 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล 79

4.67 ผลการทดลองของ Wine Data จำนวน 100 ตัวอย่าง (1000รอบ)..... 80

4.68 ผลการทดลองของ Wine Data 100 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล 80

4.69 ผลการทดลองของ Wine Data จำนวน 78 ตัวอย่าง (1รอบ)..... 82

4.70 ผลการทดลองของ Wine Data 78 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล 82

4.71 ผลการทดลองของ Wine Data จำนวน 78 ตัวอย่าง (10รอบ)..... 83

4.72 ผลการทดลองของ Wine Data 78 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล 83

4.73 ผลการทดลองของ Wine Data จำนวน 78 ตัวอย่าง (100รอบ)..... 84

4.74 ผลการทดลองของ Wine Data 78 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล 84

4.75 ผลการทดลองของ Wine Data จำนวน 78 ตัวอย่าง (1000รอบ)..... 85

4.76 ผลการทดลองของ Wine Data 78 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล 85

4.77 ผลการทดลองของ Breast Data จำนวน 420 ตัวอย่าง (1รอบ)..... 87

4.78 ผลการทดลองของ Breast Data 420 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล..... 87

4.79 ผลการทดลองของ Breast Data จำนวน 420 ตัวอย่าง (10รอบ)..... 88

4.80 ผลการทดลองของ Breast Data 420 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล..... 88

4.81 ผลการทดลองของ Breast Data จำนวน 420 ตัวอย่าง (100รอบ)..... 89

4.82 ผลการทดลองของ Breast Data 420 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล..... 89

4.83 ผลการทดลองของ Breast Data จำนวน 420 ตัวอย่าง (1000รอบ)..... 90

4.84 ผลการทดลองของ Breast Data 420 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล..... 90

4.85 ผลการทดลองของ Breast Data จำนวน 279 ตัวอย่าง (1รอบ)..... 92

4.86 ผลการทดลองของ Breast Data 279 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล..... 92

4.87 ผลการทดลองของ Breast Data จำนวน 279 ตัวอย่าง (10รอบ)..... 93

4.88 ผลการทดลองของ Breast Data 279 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล..... 93

4.89 ผลการทดลองของ Breast Data จำนวน 279 ตัวอย่าง (100รอบ)..... 94

4.90 ผลการทดลองของ Breast Data 279 ตัวอย่าง (100 รอบ) โดยเทียบกลุ่มข้อมูล..... 94

4.91 ผลการทดลองของ Breast Data จำนวน 279 ตัวอย่าง (1000รอบ)..... 95

4.92 ผลการทดลองของ Breast Data 279 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล..... 95

บทที่ 1

บทนำ

1.1 ความเป็นมา

ในปัจจุบันความก้าวหน้าทางเทคโนโลยีด้านสารสนเทศได้รับการพัฒนาขึ้นอย่างมาก ทำให้มีการเก็บข้อมูลในรูปแบบต่างๆ อย่างต่อเนื่อง เพื่อที่จะนำข้อมูลเหล่านั้นมาใช้ประโยชน์อย่างเต็มที่ โดยกระบวนการในการวิเคราะห์ข้อมูลในการทำการจำแนกกลุ่มข้อมูล ได้แก่ การทำงานโครงข่ายประสาทเทียม (Neural Network) แบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation network) และ ซิลพลิไฟล์ฟัซซี่อาร์ทแมป (Simplified Fuzzy ARTMAP) ที่สามารถนำผลการการจำแนกกลุ่มไปใช้ประโยชน์ในการจำแนกกลุ่มข้อมูลได้ทั้งสองรูปแบบ จึงเกิดความคิดว่าข้อมูลชุดเดียวกัน อัลกอริทึมใดสามารถทำการจำแนกกลุ่มข้อมูลได้มีประสิทธิภาพได้มากกว่ากัน โดยการพัฒนาโปรแกรมการเปรียบเทียบโครงสร้าง โครงข่ายประสาทเทียมสำหรับการจำแนกกลุ่มข้อมูลโดยการใช้ข้อมูลชุดเดียวกันในจำแนกกลุ่มข้อมูลผ่านกระบวนการของทั้ง แบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation network) และ ซิลพลิไฟล์ฟัซซี่อาร์ทแมป (Simplified Fuzzy ARTMAP)

การพัฒนาโปรแกรมเปรียบเทียบการจำแนกกลุ่มข้อมูลด้วยรูปแบบโครงข่ายประสาทเทียม (Neural Network) ที่มีการทำงานแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และการทำงานแบบฟัซซี่อาร์ทแมป (Simplified Fuzzy ARTMAP) ในการจำแนกกลุ่มข้อมูลเพื่อนำมาเปรียบเทียบอัลกอริทึมในการทำงานในการจำแนกกลุ่มข้อมูล

1.2 วัตถุประสงค์

โครงการพัฒนาระบบงานเรื่องการศึกษาเปรียบเทียบอัลกอริทึมในการจำแนกกลุ่มข้อมูลมีวัตถุประสงค์ดังนี้

1. เพื่อศึกษาระบบโครงข่ายประสาทเทียม (Neural Network) แบบที่มีการทำงานแบบแบคพรอพพาเกชั่นเน็ทเวิร์ค (Back-propagation) และ การทำงานแบบฟัซซี่อาร์ทแมป (Simplified Fuzzy ARTMAP)

2. เพื่อพัฒนาโปรแกรมแบบจำลองด้วยรูปแบบโครงข่ายประสาทเทียม (Neural Network) แบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และการทำงานแบบซิมพลิไฟล์ฟัซซี่อาร์ทแมป (Simplified Fuzzy ARTMAP)

3. เพื่อเปรียบเทียบประสิทธิภาพทางด้านความแม่นยำในการพยากรณ์ ระหว่างวิธีแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และการทำงานแบบซิลพลิไฟล์ฟัชซี่อาร์ทแมป (Simplified Fuzzy ARTMAP)

1.3 ขอบเขตการดำเนินงาน

การศึกษาโครงงานนี้ได้กำหนดขอบเขตในการศึกษาเป็นการพัฒนาระบบโครงข่ายประสาทเทียมในการจำแนกกลุ่มข้อมูลโดยการใช้โครงสร้างแบบแบคพรอพพาเกชันเน็ทเวิร์กและแบบซิมพลิไฟล์ฟัชซี่อาร์ทแมป เพื่อเปรียบเทียบประสิทธิภาพผลลัพธ์ในการจำแนกกลุ่มข้อมูลของผลจากการทำการทดลอง โดยใช้ข้อมูลชุดเดียวกันเพื่อนำมาเปรียบเทียบผมลัพธ์ที่ได้จากการจำแนกกลุ่มข้อมูลของแต่ละอัลกอริทึมนำมาวิเคราะห์ว่าแบบใดให้ผลลัพธ์ได้ถูกต้องกว่า ซึ่งมีขอบเขตการศึกษาหลักๆ ดังต่อไปนี้

- เทคนิคการฝึกสอนแบบแบคพรอพพาเกชัน (Back-propagation)
- เทคนิคการฝึกสอนแบบฟัชซี่อาร์ทแมป(Fuzzy ARTMAP)

1.4 ขั้นตอนและวิธีการดำเนินงาน

เพื่อให้การศึกษาเป็นไปตามวัตถุประสงค์ และขอบเขตที่กำหนด จึงได้กำหนดขั้นตอนในการดำเนินงานไว้ ดังนี้

1. ศึกษาทฤษฎีที่เกี่ยวข้องกับการจำแนกกลุ่มข้อมูล classification
2. ศึกษาทฤษฎีโครงข่ายประสาทเทียม (Neural Network)
3. ศึกษาแบคพรอพพาเกชันอัลกอริทึม (Back-propagation)
4. ศึกษาซิมพลิไฟล์ฟัชซี่อาร์ทแมปอัลกอริทึม (Fuzzy ARTMAP)
5. กำหนดวัตถุประสงค์ในการพัฒนาโปรแกรม
6. พัฒนาโปรแกรมเตรียมข้อมูลเพื่อการจำแนกกลุ่มข้อมูล
7. ทดสอบการใช้งานโปรแกรม
8. สรุปผลการศึกษาและข้อเสนอแนะ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ทำให้ทราบวิธีการพัฒนาระบบที่มีการนำกระบวนการการจำแนกกลุ่มข้อมูล (Classification) โดยใช้รูปแบบโครงข่ายประสาทเทียม (Neural Network) ที่มีการทำงานแบบแบคพรอพพาเกชัน (Back-propagation) และการทำงานแบบซิมพลิไฟล์ฟัชซี่อาร์ทแมปอัลกอริทึม (Fuzzy ARTMAP) มาประยุกต์ใช้กับการทำนายข้อมูลในการจำแนกกลุ่มข้อมูลว่าระบบได้มี

ประสิทธิภาพในการจำแนกกลุ่มที่แม่นยำและมีความเหมาะสมที่จะนำรูปแบบใดรูปแบบหนึ่งไปใช้เป็นเครื่องมือในการจำแนกกลุ่มข้อมูล

1.6 เครื่องมือที่ใช้ในการพัฒนาระบบ

Visual Basic.NET หรือ VB.NET เป็นเครื่องมือที่ใช้พัฒนาโปรแกรม Visual Programming บนระบบปฏิบัติการ Windows ซึ่งได้รับการพัฒนามาจากภาษา BASIC (Beginners All Purpose Symbolic Instruction Code) ซึ่งเป็นภาษาโปรแกรมที่ได้รับความนิยมอย่างแพร่หลายสำหรับผู้เริ่มต้นหัดเขียนโปรแกรมคอมพิวเตอร์ เนื่องจากภาษา BASIC เป็นโปรแกรมที่สามารถทำความเข้าใจได้ง่าย

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

ในปัจจุบันมีความนิยมนำวิธีการจำแนกหมวดหมู่ (Classification) ไปประยุกต์ใช้งานด้านต่างๆ มากมาย ทำให้เกิดทฤษฎีและแนวคิดใหม่ๆขึ้น เพื่อใช้ในการจำแนก เช่น ทฤษฎีเบย์เซียลดีซิชั่น (Bayesian Decision Theory), แบบจำลองฮิทเดนมาคอฟ (Hidden Markov Model) และโครงข่ายปราสาทเทียม (Neural Network) เป็นต้น ซึ่งงานวิจัยจำนวนมากได้ให้ความสนใจในการออกแบบแนวคิดในการจำแนก เนื่องจากคุณภาพการจำแนกแบ่งกลุ่มที่ดีและเหมาะสมนั้นมีความสำคัญกับงานในด้านต่างๆ ที่ต้องการระบุความน่าจะเป็นในแต่ละรูปแบบที่เป็นไปได้ เช่น การรู้จำ(Recognition),การพยากรณ์ (Forecasting) เป็นต้น ซึ่งความยากง่ายในการจำแนกกลุ่มนั้นขึ้นอยู่กับข้อมูล (Data) ในแต่ละกลุ่ม เนื่องจากข้อมูลในกลุ่มหนึ่งอาจไปสัมพันธ์กับข้อมูลอีกกลุ่มหนึ่งหรือข้อมูลในกลุ่มเดียวกันอาจเป็นข้อมูลที่ซับซ้อน (Complexity) หรืออาจจะเป็นสิ่งรบกวร (Noise) ก็ได้

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้อง ได้แก่ พื้นฐานการทำ Data Mining โครงข่ายปราสาทเทียมแบบต่างๆ รวมทั้งสถาปัตยกรรมและระบบการทำงานของอัลกอริทึมที่ใช้ในการทำ Classification ซึ่งเนื้อหาทั้งหมดนี้จำเป็นสำหรับการศึกษา และพัฒนาโปรแกรมประยุกต์ต่อไป

2.1 Data Mining

Data Mining เป็นกระบวนการในการกลั่นกรองข้อมูลเพื่อดึงเอาสารสนเทศออกมาจากฐานข้อมูลขนาดใหญ่ ซึ่งสารสนเทศที่ได้จะมีลักษณะเป็นความรู้ใหม่ๆ ที่ไม่เคยรู้มาก่อน(Unknow) มีความถูกต้อง (Valid) และสามารถนำมาใช้ประโยชน์ในทางปฏิบัติได้ (Actionable) เพื่อนำมาใช้สนับสนุนการตัดสินใจต่างๆ Data Mining จึงถือเป็นส่วนหนึ่งของกิจกรรมในการค้นหาความรู้ในฐานข้อมูล (Knowledge Discovery in Database: KDD)

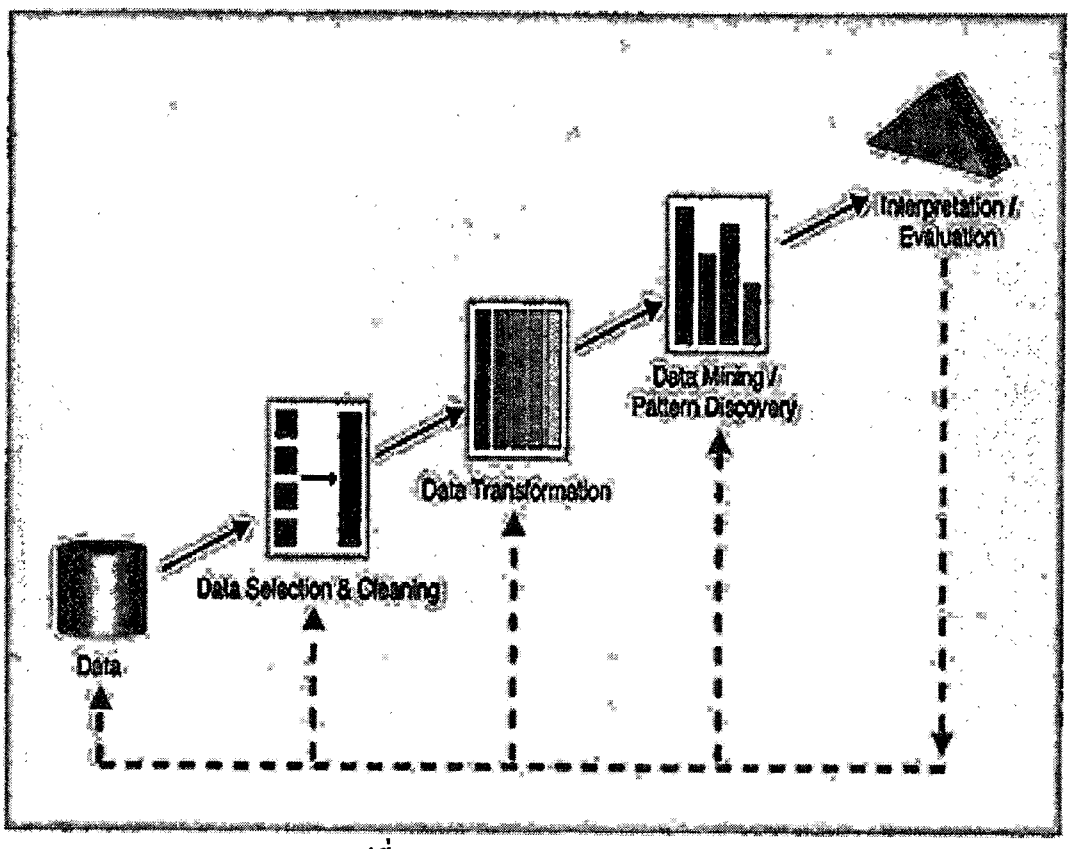
Data mining แตกต่างไปจากเครื่องมือที่ใช้ในงานวิเคราะห์ข้อมูลอื่นๆ คือจะทำการค้นหาความรู้คดยสามารถมองความสัมพันธ์ของข้อมูลจากฐานข้อมูลขนาดใหญ่ในหลายมิติได้ โดยไม่ต้องตั้งสมมติฐานไว้ล่วงหน้า แต่จะนำความรู้ที่ได้นั้นมาทดสอบสมมติฐานในภายหลัง ส่วนเครื่องมืออื่นๆ เช่น SQL Query หรือ OLAP (On Line Analytical Processing) นั้นจะต้องสร้าง

สมมติฐานเกี่ยวกับความสัมพันธ์ของข้อมูลขึ้นก่อน จากนั้นจึงใช้เครื่องมือเพื่อพิสูจน์ว่าเป็นจริงหรือไม่ตามสมมติฐานที่สร้างขึ้น

2.1.1 กระบวนการทำ Data Mining

Data Mining เป็นการรวบรวมเทคนิคจากงานต่างๆ เช่นการจดจำรูปแบบ (Recognition) การเรียนรู้ของเครื่องจักร (Machine Learning) หลักสถิติ (Statistics) และฐานข้อมูล (Database) เป็นต้น เพื่อนำมาค้นหารูปแบบความสัมพันธ์ของข้อมูล และวิเคราะห์หาข้อมูลที่เป็นประโยชน์ที่อาจจะซ่อนอยู่ภายใต้ฐานข้อมูลขนาดใหญ่ เพื่อให้ได้ข้อมูลที่สามารถนำไปใช้งานได้จริงและเป็นประโยชน์ โดยทั่วไปเมื่อกล่าวถึง Data Mining ส่วนใหญ่จะคำนึงถึงและให้ความสำคัญกับขั้นตอนการ Mining หรือการค้นหาลักษณะทิศทางของข้อมูล แต่ความเป็นจริงแล้วการทำ Data Mining เป็นเพียงหนึ่งในกระบวนการทำ Data Mining เท่านั้น

กระบวนการทำ Data Mining เป็นขั้นตอนการสร้างแบบจำลอง (Model) โดยจะสร้างแบบจำลองของกลุ่มข้อมูล (Data Set) เพื่อสร้างความเข้าใจในแนวโน้ม รูปแบบ ความเกี่ยวข้องสัมพันธ์กันของกลุ่มข้อมูลและสามารถทำนายข้อมูลนั้นๆ ซึ่งประกอบไปด้วย 5 ขั้นตอน ดังรูปที่ 2.1



รูปที่ 2.1 กระบวนการทำ Data Mining

2.1.1.1 การกำหนดวัตถุประสงค์ทางธุรกิจ (Business Object Determination)

การกำหนดวัตถุประสงค์และปัญหาที่ชัดเจนจะเป็นตัวกำหนดทิศทางในการทำ Data Mining ดังนั้นในการกำหนดวัตถุประสงค์ของงานจะต้องเข้าใจถึงปัญหาและความต้องการของงานนั้นๆ รวมทั้งต้องศึกษาความเป็นไปได้ว่าวิธีการ Data Mining นั้นเหมาะกับการหาคำตอบของปัญหาหรือไม่ เพราะ Data Mining อาจไม่ใช่เทคนิคที่ดีที่สุดในการหาคำตอบของปัญหาได้ ดังนั้นจึงต้องมีการศึกษาความต้องการในการวิเคราะห์ ซึ่งก็คือการกำหนดปัญหาและวัตถุประสงค์ทางธุรกิจหรือองค์กรให้ชัดเจน โดยขั้นตอนนั้นประกอบด้วยการวิเคราะห์ข้อมูลเบื้องต้นว่ามีข้อมูลใดอยู่บ้าง ต้องการสารสนเทศใดจากแหล่งข้อมูลนั้นและเป็นการกำหนดว่าเมื่อไหร่ควรจะใช้ Data Mining กับการแก้ปัญหา

2.1.1.2 การเตรียมข้อมูล (Data Preparation)

การเตรียมข้อมูลเป็นขั้นตอนที่สำคัญและใช้เวลามากที่สุด ประมาณ 60% ของเวลาทั้งหมด การเตรียมข้อมูลนั้นประกอบไปด้วย 3 ขั้นตอนย่อย คือ

- <u>การคัดเลือกข้อมูล</u> (Data Selection) มีจุดประสงค์หลักคือการระบุลักษณะและคัดเลือกข้อมูลที่ต้องการและนำข้อมูลที่ไม่ต้องการออกไปซึ่งการเลือกข้อมูลนี้ขึ้นอยู่กับวัตถุประสงค์ทางธุรกิจที่ได้กำหนดไว้ในขั้นตอนก่อนหน้า และจะต้องเข้าใจถึงแหล่งที่มาของข้อมูล ความหมายของข้อมูล ประเภทของข้อมูล ค่าที่สามารถเป็นไปได้ของข้อมูล และลักษณะอื่นๆ เช่น ความทันสมัยของข้อมูล เนื่องจากการเปลี่ยนแปลงของเวลาและสถานการณ์ภายนอกมีอยู่ตลอดเวลาและมีผลทำให้ประสิทธิภาพของข้อมูลลดลง นอกจากนี้ยังมีประเด็นที่สำคัญที่ต้องพิจารณาอีก 4 ประเด็นคือ

-ระดับของข้อมูลโดยขึ้นอยู่กับวัตถุประสงค์ในการทำว่าเป็นข้อมูลที่สรุปแล้วหรือว่าเป็นข้อมูลระดับรายการ (Item)

-ลักษณะการจัดเก็บของข้อมูล ซึ่งขึ้นอยู่กับภาษาคอมพิวเตอร์และระบบปฏิบัติการที่ใช้

-ความแตกต่างของข้อมูลในแต่ละแหล่งทั้งรูปแบบความหมายและลักษณะการจัดเก็บ

-ประเภทข้อความซึ่งค่าของข้อมูลที่จัดเก็บอาจแตกต่างกันซึ่งอาจเกิดจากความผิดพลาดในขั้นตอนการบันทึกข้อมูล (Data Entry) หรืออื่นๆ ขณะที่หมายถึงสิ่งเดียวกัน เช่น “NO” และ “No” เป็นต้น

- การกรองข้อมูล (Data Preprocessing) เป็นกระบวนการนำข้อมูลที่ผ่านขั้นตอนการคัดเลือกข้อมูลมาแล้ว มาทำการกลั่นกรองข้อมูล (Data Cleaning) เพื่อให้สามารถมั่นใจได้ว่าข้อมูลที่เลือกมีความถูกต้องเหมาะสม โดยพิจารณาใน 2 ประเด็น คือ Noisy Data คือ ข้อมูลมีลักษณะแตกต่างจากที่ควรจะเป็นหรือที่คาดการณ์ไว้ ซึ่งอาจเป็นค่าจริงที่เกิดขึ้น หรือเกิดจากความผิดพลาดในการบันทึกข้อมูล และ missing value คือ ข้อมูลที่ไม่ปรากฏในข้อมูลที่ผ่านการคัดเลือก นั่นคือมีข้อมูลบางส่วนขาดหายไป ซึ่งอาจจะเกิดจากความไม่สมบูรณ์ของข้อมูล หรือความผิดพลาดในการบันทึกข้อมูล ในการตรวจสอบแก้ไขให้ถูกต้องนั้นต้องอาศัยวิธีการต่างๆ เข้าช่วยเพื่อทำ Noise Detection ดังกล่าวหรือไม่ อาจจะไม่นำข้อมูลนั้นมาวิเคราะห์
- การแปลงข้อมูล (Data Transformation) เป็นการแปลงข้อมูลให้อยู่ในรูปแบบที่เหมาะสมที่พร้อมจะนำไปวิเคราะห์ตามอัลกอริทึมที่จะนำมาใช้ในการทำ Data Mining ซึ่งจะมีลักษณะแตกต่างกันไป เช่น การแปลงข้อมูลให้เป็นช่วงเพื่อใช้กับ Decision Tree หรือการปรับอัตราส่วนตัวเลขให้อยู่ในช่วง 0 – 1 เพื่อใช้กับอัลกอริทึมใน Neural Network

2.1.1.3 การทำ Mining ข้อมูล (Data Mining)

การทำ Data Mining นั้นจะมีการเลือกอัลกอริทึมที่จะนำมาใช้ให้เหมาะสมกับลักษณะปัญหาและลักษณะของข้อมูลที่ได้ดำเนินการไว้ในขั้นตอนที่ผ่านมา แล้วทำการประมวลผลข้อมูลตามอัลกอริทึมที่เลือกไว้ ซึ่งในขั้นตอนนี้อาจต้องย้อนกลับไปทำขั้นตอนที่เกี่ยวข้องใหม่ โดยจะพัฒนากระบวนการในขั้นตอนต่างๆ เพื่อให้ได้ผลที่มีประสิทธิภาพและมีความถูกต้องมากขึ้น

2.1.1.4 การวิเคราะห์ผลลัพธ์ที่ได้ (Analysis of Result)

การวิเคราะห์ผลลัพธ์ที่ได้เป็นขั้นตอนที่มีความสำคัญมาก โดยจะทำการแปลความหมายและประเมินผลลัพธ์ที่ได้จากขั้นตอนการทำ Data Mining ทั้งนี้จะต้องอาศัยทักษะทั้งในส่วนของการวิเคราะห์ข้อมูลและการวิเคราะห์ทางธุรกิจประกอบกัน รวมถึงเครื่องมือด้าน Graphical Visualization ซึ่งช่วยให้การวิเคราะห์ข้อมูลทำได้อย่างสะดวกและรวดเร็วขึ้น นอกจากนี้ยังพิจารณาสารสนเทศที่ได้มาว่ามีความน่าสนใจ ความสมบูรณ์และสามารถนำไปใช้ได้หรือไม่ หากผลที่ออกมาไม่เป็นเช่นนั้นอาจต้องวนกลับไปพิจารณาขั้นตอนต่างๆ เพื่อแก้ไข Model ใหม่

2.1.1.5 การนำสารสนเทศที่ได้ไปใช้ประโยชน์ (Assimilation of Knowledge)

การนำสารสนเทศที่ได้ไปใช้ประโยชน์ โดยการรวบรวมความเข้าใจทางธุรกิจและผลจากขั้นตอนด้านต้น ซึ่งเกี่ยวข้องโดยตรงกับวัตถุประสงค์ทางธุรกิจ ทั้งนี้มีหลักที่ต้องพิจารณา 2 ประการคือ การนำเสนอแนวคิดทางธุรกิจที่ค้นพบใหม่ และหาแนวทางในการที่จะใช้ความรู้ใหม่ที่ค้นพบเพื่อให้เกิดประโยชน์สูงสุด

2.1.2 Data Mining Operation

การทำ Data Mining นั้นมีการดำเนินการอยู่หลายรูปแบบ ซึ่งการเลือกใช้จะขึ้นอยู่กับวัตถุประสงค์ในการทำ Data Mining นั้นๆ Data Mining Operation สามารถจำแนกออกได้เป็น 4 ประเภทหลักๆ ได้แก่

2.1.2.1 Database Segmentation หรือ Database Clustering

ใช้ในการจัดกลุ่มข้อมูล โดยการแบ่งกลุ่มข้อมูลจะแบ่งตามลักษณะที่เหมือนกัน ของข้อมูล เช่น การแบ่งตามอายุ รายได้ ซึ่งในการแบ่งกลุ่มข้อมูลนั้นไม่สามารถที่จะกำหนดได้ ว่าข้อมูลควรที่จะอยู่ในกลุ่มใด แต่จะเป็นการกำหนดกลุ่มจากธรรมชาติของข้อมูลเองมากกว่า โดยไม่มีการใช้อคติหรือประสบการณ์มาช่วยในการตัดสินใจ

การจัดกลุ่มข้อมูล คือ การรวบรวมข้อมูลที่มีความคล้ายคลึงกันไว้ในกลุ่มเดียวกันและข้อมูลที่ไม่คล้ายคลึงกันจะอยู่ในกลุ่มอื่น ซึ่งข้อมูลแต่ละตัวจะต้องอยู่ในกลุ่มใดกลุ่มหนึ่งเพียงกลุ่มเดียว โดยข้อมูลแต่ละกลุ่มที่ได้นั้นจะเรียกว่า "Cluster" นั่นคือลักษณะการแบ่งแยกข้อมูลบนพื้นฐานของความคล้ายคลึงกันในตัวเอง วัตถุประสงค์ของการจัดกลุ่มคือ เพื่อจัดกลุ่มข้อมูลซึ่งเป็นประโยชน์ในงานด้านต่างๆ เช่น การตลาด การแพทย์ การปกครอง เป็นต้น

2.1.2.2 Link Analysis

ใช้วิเคราะห์ตรวจหาสิ่งเชื่อมโยงหรือความสัมพันธ์ (Association) ระหว่าง Record หรือกลุ่มของ Record ใน Database เพื่อที่จะได้ดูว่าแต่ละกลุ่มของข้อมูลมีความสัมพันธ์กันในลักษณะใด ซึ่งเป็นแบบจำลองที่ได้รับความนิยมในการหาความสัมพันธ์ระหว่างลูกค้ากับสินค้า

2.1.2.3 Deviation Detection

ใช้ในการตรวจจับสิ่งผิดปกติต่างๆ จากการหาค่าเบี่ยงเบนไปจากค่ามาตรฐานหรือค่าที่คาดคิดไว้ โดยทั่วไปมักใช้วิธีการทางสถิติหรืออาศัยการวาดกราฟหรือแสดงผลในรูปกราฟฟิคที่ทำให้เห็นภาพรวมมากขึ้น แล้วดูการกระจายข้อมูลว่ามีการกระจายออกไปจากกลุ่มหรือไม่ ซึ่งเทคนิคนี้ถูกใช้ในงานด้านการตรวจสอบการปลอมแปลงบัตรเครดิตหรือการจับการโกงเป็นต้น

2.1.2.4 Predictive Modeling

ใช้ในการสร้างแบบจำลองพยากรณ์เพื่อทำนายหาค่าความเป็นไปได้ โดยสังเกตุจากข้อมูลที่มีอยู่ ซึ่งข้อมูลต้องมีความถูกต้องและสมบูรณ์ จะทำให้แบบจำลองสามารถทำนายผลได้อย่างถูกต้อง มี 2 ลักษณะ คือ

- Value Prediction การพยากรณ์หรือการประมาณค่าข้อมูลซึ่งมีความสัมพันธ์กันใน Database Record
- Classification เป็นการจำแนกหมวดหมู่ของข้อมูล โดยพิจารณาข้อมูลตามหมวดหมู่ที่ได้กำหนดไว้แล้ว

ซึ่งในกระบวนการจะประกอบไปด้วย 2 ขั้นตอนใหญ่ๆ คือ

การเรียนรู้ (Learning) เป็นการนำเอากลุ่มของข้อมูล (Training Dataset) ที่จะนำมาทำการศึกษา โดยใช้อัลกอริทึมของการจำแนกข้อมูล มาทำการเรียนรู้กลุ่มของข้อมูลนั้น เพื่อจะทำการสร้าง โมเดล(Model) ที่สามารถอธิบายถึงลักษณะเฉพาะที่ซ่อนอยู่ภายใต้ข้อมูล ซึ่งโมเดลนี้จะมีการแจกแจงกลุ่มของข้อมูลออกเป็นคลาส (Class) ต่างๆ ด้วยกฎการจำแนก (Classification rule) และคลาสแต่ละคลาสนี้จะมีลักษณะเฉพาะกลุ่มที่สามารถสรุปออกมาได้เป็นความสัมพันธ์จากชุดของคุณลักษณะไปเป็นกลุ่มที่สามารถระบุได้ เช่น การแบ่งประเภทลูกค้าว่าเชื่อถือได้หรือไม่โดยใช้ Model Construction (Learning) ซึ่งเป็นการสร้างแบบจำลอง โดยการเรียนรู้จากข้อมูลที่ได้กำหนดไว้เรียบร้อยแล้ว (Training data) จากข้อมูลในอดีต ซึ่งจะใช้ข้อมูลปริมาณมาก

การทดสอบข้อมูล (Testing data) จะถูกนำมาใช้เพื่อวัดถึงความถูกต้องของกฎการจำแนกที่ถูกสร้างขึ้นมาจากขั้นตอนการเรียนรู้ โดยความถูกต้องนี้อาจจะเป็นตัวพิจารณาว่ากฎการจำแนกที่สร้างขึ้นสามารถนำมาใช้กับกลุ่มข้อมูลใหม่ๆได้หรือไม่

ในการจำแนกหมวดหมู่ของข้อมูลนั้น นอกจากกระบวนการหลักที่ใช้แล้วยังสามารถนำเทคนิคแบบต่างๆ มาใช้ประกอบกันได้ เช่น

- แบบต้นไม้(Decision Tree) มีลักษณะการแบ่งกลุ่มของข้อมูลเป็นโครงสร้างต้นไม้
- แบบโครงข่ายปราสาทเทียม (Neural Network) เป็นแบบจำลองที่สร้างขึ้นเพื่อเลียนแบบการทำงานของโครงข่ายประสาทสิ่งมีชีวิต (Biological Neural) โดยเป็นชุดการเชื่อมต่อของ Input และ Output โดยจะมีค่าน้ำหนัก (Weight) สำหรับแต่ละเส้นทางการเชื่อมต่อซึ่งในขั้นตอนการเรียนรู้ค่าน้ำหนักจะถูกปรับให้มีค่าที่เหมาะสมที่สุด

2.1.3 การประยุกต์ใช้ Data Mining

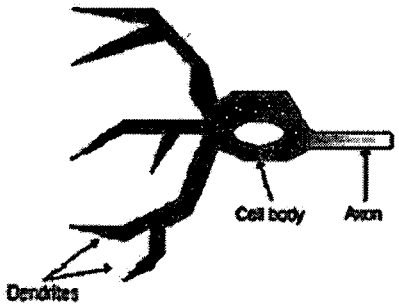
ลักษณะการประยุกต์ใช้งาน Application ในการทำ Data Mining มีอยู่หลายรูปแบบ เช่น

1. ทางการตลาด เช่น การนำข้อมูลลูกค้าจากการทำบัตรสมาชิกมาเป็นข้อมูลในการทำงาน ตัวอย่างเช่น การทำ Cross-Selling,Purchasing Pattern over time ,Customer Profiling, Direct Mail Campaign, Customer Relation Management (Forecasting)
2. ทางด้านการวิเคราะห์ความเสี่ยงต่างๆ เช่น การพยากรณ์ (Forecasting), การทำ Credit Scoring , การทำ Profile of attrition
3. ด้านการตรวจจับการโกง
4. ด้านการป้องกันการฟอกเงิน
5. การป้องกันการโกงประกันภัย
6. Text Mining
7. Web Mining

ในการศึกษาโครงการนี้ จะประยุกต์การใช้งาน Data Mining ในด้าน Classification (การจำแนกหมวดหมู่) ซึ่งแบบจำลองที่ใช้จะอยู่ในรูปของโครงข่ายประสาทเทียม

2.2ทฤษฎีโครงข่ายประสาทเทียม (Artificial Neural Network)

โครงข่ายประสาทเทียม (Artificial Neural Network) คือโมเดลทางคณิตศาสตร์ สำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบคอนเนคชันนิสต์ (Connectionist) เพื่อจำลองการทำงานของเครือข่ายประสาทในสมองมนุษย์ สถาปัตยกรรมโครงข่าย แบ่งเป็น 4 แบบ คือ Feed forward Network, Feedback Network, Network Layer และ Perceptrons ด้วยวัตถุประสงค์ที่จะสร้างเครื่องมือซึ่งมีความสามารถในการเรียนรู้การจดจำรูปแบบ (Pattern Recognition) และการอนุมานความรู้ (Knowledge deduction) เช่นเดียวกับความสามารถที่มีในสมองมนุษย์ แนวคิดเริ่มต้นของเทคนิคนี้ได้มาจากการศึกษาข่ายงานไฟฟ้าชีวภาพ (Bioelectric Network) ในสมอง ซึ่งประกอบด้วย เซลล์ประสาทหรือนิวรอน (Neurons) และจุดประสานประสาท (Synapses) แต่ละเซลล์ประสาทประกอบด้วยปลายในการรับกระแสประสาท เรียกว่า เดนไดรท์ (Dendrite) ซึ่งเป็นอินพุต (Input) และปลายในการส่งกระแสประสาทเรียกว่า แอคซอน (Axon) ซึ่งเป็นเหมือนเอาต์พุต (Output) ของเซลล์ เซลล์เหล่านี้ทำงานด้วยปฏิกิริยาไฟฟ้าเคมี เมื่อมีการกระตุ้นด้วยสิ่งเร้าภายนอกหรือกระตุ้นด้วยเซลล์ด้วยกัน กระแสประสาทจะวิ่งผ่านเดนไดรท์เข้าสู่นิวเคลียสซึ่งจะเป็นตัวตัดสินว่าต้องกระตุ้นเซลล์อื่น ๆ ต่อหรือไม่ ถ้ากระแสประสาทแรงพอนิวเคลียสก็จะกระตุ้นเซลล์อื่น ๆ ต่อไปผ่านทางแอคซอนของมัน ตามโมเดลนี้ข่ายงานประสาทเกิดจากการเชื่อมต่อระหว่างเซลล์ประสาทจนเป็นเครือข่ายที่ทำงานร่วมกัน

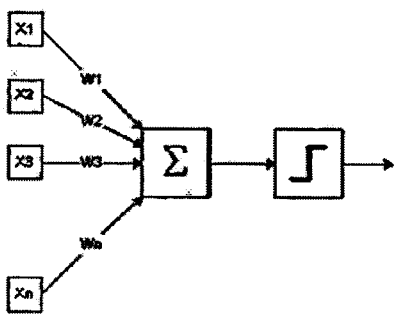


รูปที่ 2.2 แสดง Model ของ Neuron ในสมองมนุษย์

ตารางที่ 2.1 แสดงการเปรียบเทียบโครงข่ายประสาททางชีววิทยาและโครงข่ายประสาทเทียม

Biological neural network	**Artificial neural network**
Soma	Neuron
Dendrite	Input
Axon	Output
Synapse	Weight

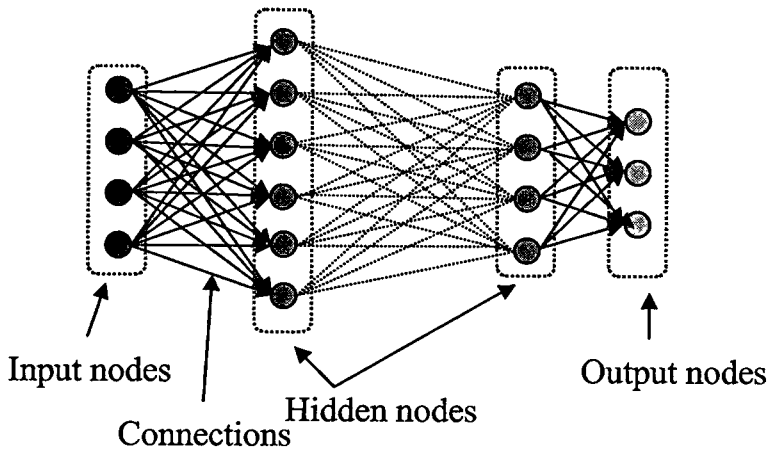
ในปัจจุบันพบว่าโครงข่ายประสาทเทียมมีโครงสร้างแตกต่างจากโครงข่ายในสมอง แต่ก็ยังเหมือนสมอง ในแง่ที่ว่าโครงข่ายประสาทเทียม คือการรวมกลุ่มแบบขนานของหน่วยประมวลผลย่อย ๆ และการเชื่อมต่อนี้เป็นส่วนสำคัญที่ทำให้เกิดสติปัญญาของโครงข่าย เมื่อพิจารณาขนาดแล้วสมองมีขนาดใหญ่กว่าโครงข่ายประสาทเทียมอย่างมาก รวมทั้งเซลล์ประสาทยังมีความซับซ้อนกว่าหน่วยย่อยของโครงข่ายประสาทเทียม อย่างไรก็ดีหน้าที่สำคัญของสมอง เช่น การเรียนรู้ยังคงสามารถถูกจำลองขึ้นอย่างง่ายด้วยโครงข่ายประสาทนี้



รูปที่ 2.3 แสดง Model ของ Neuron ในคอมพิวเตอร์

สำหรับในคอมพิวเตอร์นิวรอน (Neurons) ประกอบด้วยอินพุต (Input) และเอาต์พุต (Output) เหมือนกัน โดยจำลองให้อินพุตแต่ละอันมี Weight เป็นตัวกำหนดค่าน้ำหนักของอินพุต โดยนิวรอน (Neurons) แต่ละหน่วยจะมีค่าเทรโชด์ (Threshold) เป็นตัวกำหนดว่าค่าน้ำหนักรวมของอินพุตต้องมากขนาดไหนจึงจะสามารถส่งเอาต์พุตไปยัง นิวรอน (Neurons) ตัวอื่นได้ เมื่อนำนิวรอนแต่ละหน่วยมาต่อกันให้ทำงานร่วมกันการทำงานนี้ในทางตรรกะแล้วก็จะเหมือนกับปฏิกิริยาเคมีที่เกิดในสมอง เพียงแต่ในคอมพิวเตอร์ทุกอย่างเป็นตัวเลขเท่านั้นเอง

การทำงานของโครงข่ายประสาทเทียม (Neural Networks) คือเมื่อมีอินพุตเข้ามายังโครงข่าย (Network) ก็เอาอินพุตมาคูณกับค่าน้ำหนัก (Weight) ของแต่ละขา ผลที่ได้จากอินพุตทุก ๆ ขาของนิวรอน (Neuron) จะเอามารวมกันแล้วก็เอามาเทียบกับค่าเทรโชด์ (Threshold) ที่กำหนดไว้ ถ้าผลรวมมีค่ามากกว่าค่าเทรโชด์แล้วนิวรอนก็จะส่งเอาต์พุตออกไป เอาต์พุตนี้ก็จะถูกส่งไปยังอินพุตของนิวรอนอื่น ๆ ที่เชื่อมกันใน โครงข่ายถ้าค่าน้อยกว่าค่าเทรโชด์ก็จะไม่เกิดเอาต์พุต



รูปที่ 2.4 แสดงโครงสร้างโครง Neural Network

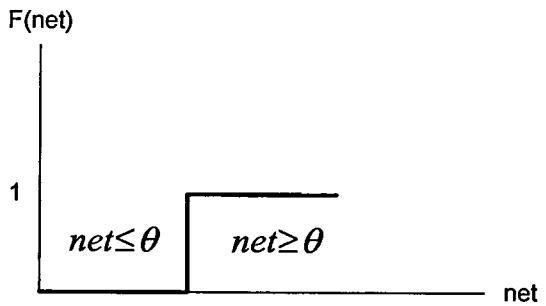
ส่วนของเอาต์พุต (Output) ที่ออกมาจะถูกกระทำไปโดยส่วนที่เรียกว่า ฟังก์ชันกระตุ้น (Activation Function) ค่าที่ได้จากฟังก์ชันกระตุ้นนี้จะทำให้เซลล์ประสาทเทียมมีลักษณะเหมือนเซลล์ประสาททางชีววิทยา กล่าวคือ ค่าผลลัพธ์ที่ได้จากฟังก์ชันกระตุ้นเป็นค่าที่อยู่ในช่วง 0 ถึง 1 ถ้าค่าสัญญาณที่ออกจากเซลล์ทางชีววิทยามีลักษณะเป็นสถานะของการกระตุ้นจะสามารถแทนความหมายของสถานการณ์กระตุ้นของเซลล์ประสาทด้วยค่า 1 และการไม่กระตุ้นด้วยค่า 0 รูปแบบของฟังก์ชันมีหลายรูปแบบดังนี้

1. Step Function เป็นฟังก์ชันที่ใช้ในโครงข่ายแบบชั้นเดียว (Single Layer) ซึ่งใช้ในการแปลงค่า Net Input ที่อยู่ในรูปของค่าต่อเนื่องให้อยู่ในรูปของไบนารี (Binary) คือ 1 และ 0 หรือไบโพลล่า (Bipolar) คือ 1 และ -1 ฟังก์ชันนี้จะใช้ค่าเทรโชด์ (Threshold) ในการกำหนดการแปลงค่า ซึ่งเรียกฟังก์ชันนี้ว่า Threshold Function หรือ Heavised Function ดังสมการ (2.1) และรูปที่ 2.4

$$f(net)=\begin{cases}1 & If \quad net \geq \theta \\ 0 & If \quad net \leq \theta\end{cases} \tag{2.1}$$

เมื่อ *net* เป็นค่า net input ของเซลล์ประสาทเทียม

θ เป็นค่าเทรโชด์ (Threshold)



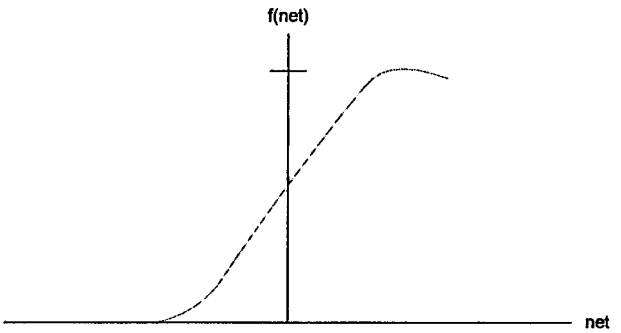
รูปที่ 2.5 แสดงลักษณะของสมการ Threshold

2. Sigmoid Function เป็นฟังก์ชันที่ใช้ในการแปลงค่า Net Input ให้อยู่ในช่วง 0 ถึง 1 ว่า Binary Sigmoid Function หรืออยู่ในช่วง 1 ถึง -1 ซึ่งเรียกว่า Bipolar Sigmoid Function ดังแสดงในสมการ (2.2) และรูปที่ 2.5

$$f(net)=\frac{1}{(1+e^{-net})} \tag{2.2}$$

เมื่อ *net* เป็นค่า Net Input ของเซลล์ประสาทเทียม

e เป็นฟังก์ชันเอกซ์โปเนนเชียล

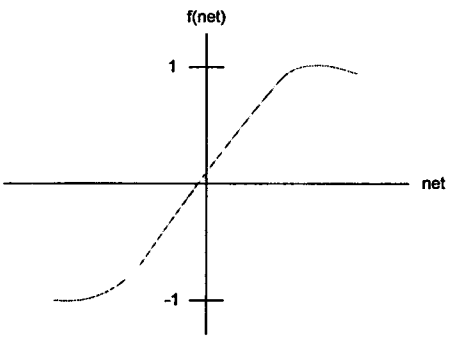


รูปที่ 2.6 แสดงกราฟของฟังก์ชันไบนารีซิกมอยด์

$$f(net) = \frac{(e^{net} - e^{-net})}{(e^{net} + e^{-net})} \tag{2.3}$$

เมื่อ *net* เป็นค่า Net Input ของเซลล์ประสาทเทียม

e เป็นฟังก์ชันเอกซ์โปแนนเชียล

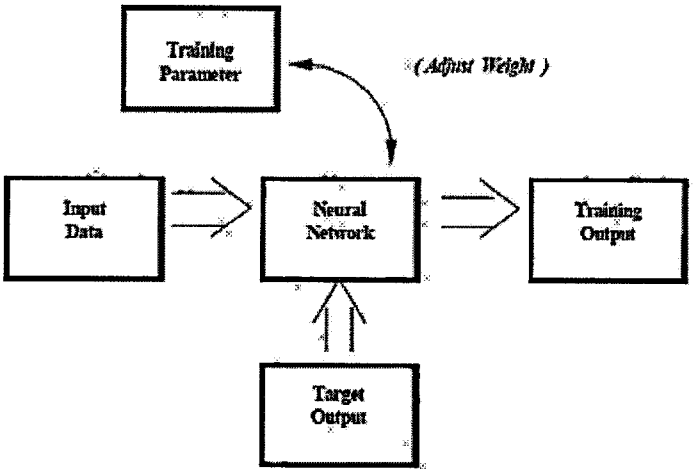


รูปที่ 2.7 แสดงกราฟของฟังก์ชันโพล่าซิกมอยด์

2.2.1 การเรียนรู้สำหรับโครงข่ายประสาทเทียม (Neural Network)

2.2.1.1 Supervised Learning การเรียนแบบมีการสอน

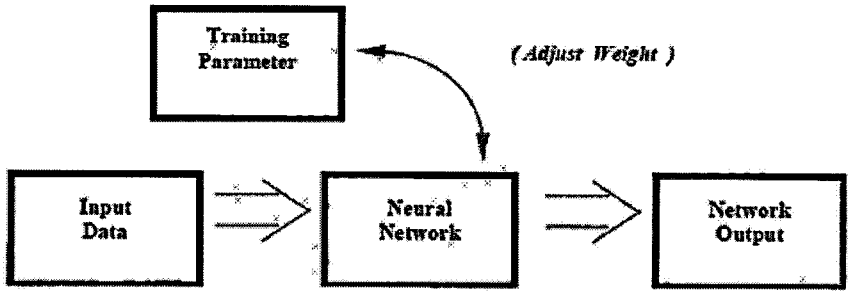
เป็นการเรียนแบบที่มีการตรวจคำตอบเพื่อให้โครงข่ายปรับตัว ชุดข้อมูลที่ใช้สอนโครงข่ายจะมีคำตอบไว้คอยตรวจดูว่าโครงข่ายให้คำตอบที่ถูกหรือไม่ ถ้าตอบไม่ถูก โครงข่ายก็จะปรับตัวเองเพื่อให้ได้คำตอบที่ดีขึ้น (เปรียบเทียบกับคน เหมือนกับการสอนนักเรียนโดยมีครูผู้สอนคอยแนะนำ)



รูปที่ 2.8 แสดงการเรียนรู้แบบมีการสอน (Supervised Learning)

2.2.1.2 Unsupervised Learning การเรียนแบบไม่มีการสอน

เป็นการเรียนแบบไม่มีผู้แนะนำ ไม่มีการตรวจคำตอบว่าถูกหรือผิด โครงข่ายจะจัดเรียงโครงสร้างด้วยตัวเองตามลักษณะของข้อมูล ผลลัพธ์ที่ได้ โครงข่ายจะสามารถจัดหมวดหมู่ของข้อมูลได้ (เปรียบเทียบกับคน เช่น การที่เราสามารถแยกแยะพันธุ์พืช พันธุ์สัตว์ตามลักษณะรูปร่างของมันได้เองโดยไม่มีใครสอน)



รูปที่ 2.9 แสดงการเรียนรู้แบบไม่มีการสอน (Unsupervised Learning)

การสอนโครงข่ายเซลล์ประสาทเป็นการหาฟังก์ชันการแปลง และ ฟังก์ชันการแปลงที่ได้จะมีคุณสมบัติไม่เป็นเชิงเส้น ซึ่งฟังก์ชันการแปลงของโครงข่ายประสาทเทียมในที่นี้ คือ เซตของค่าถ่วงน้ำหนักของโครงข่าย ดังนั้นฟังก์ชันการแปลงจะมีศักยภาพมากน้อยเพียงใดนั้นจะขึ้นอยู่กับค่าถ่วงน้ำหนักของโครงข่ายนั้นๆ ว่ามีเสถียรภาพมากน้อยเพียงใดแบะค่าถ่วงน้ำหนัก คำนวณได้จาก

การสอนโครงข่าย ซึ่งการสอนโครงข่ายมีหลายแบบด้วยกัน เช่น กฎการสอนของ Hebb (Hebbian Learning), กฎการสอนแบบเดลต้า (Delta Rule or Error-Correction Learning), กฎการเรียนรู้แบบแข่งขัน(Competitive Learning)

2.3 โครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ (Multilayered Feedforword Neural Network หรือ Backpropagation Algorithm)

สถาปัตยกรรมของโครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ แสดงในรูปที่ 2.4 ประกอบไปด้วยชั้นอินพุต (Input Layer) ชั้นเอาต์พุต (Output Layer) และชั้นซ่อน (Hidden Layer) ที่อยู่ระหว่างชั้นอินพุตและชั้นเอาต์พุตของนิวรอน ซึ่งข้อมูลจะไหลจากซ้ายไปขวา โดยอินพุต x จะติดต่อกับชั้นซ่อนและชั้นเอาต์พุตของนิวรอนผ่านทางค่าน้ำหนัก (Weight) โดยน้ำหนักที่ชั้นซ่อนของโหนดอินพุต i กับนิวรอน j คือ w_{ij} ส่วนน้ำหนักที่ชั้นซ่อนของนิวรอน j กับเอาต์พุตของนิวรอน k แทนด้วย v_{jk}

จากการคำนวณค่าน้ำหนักที่อินพุตและเอาต์พุตจะเป็นฟังก์ชันซิกมอยด์ โดยฟังก์ชันซิกมอยด์จะเป็นตัวกำหนดคุณสมบัติให้กับโครงข่าย สำหรับสมการหาค่าเอาต์พุตที่นิวรอน j^{th} ในชั้นซ่อนคือ

$$net_j^h = \sum_{i=1}^{N} w_{ji} x_i \tag{2.4}$$

$$y_j = f(net_j^h) \tag{2.5}$$

เมื่อ N เป็นจำนวนอินพุต x จากชั้นอินพุต
ส่วนสมการหาค่าเอาต์พุตที่นิวรอน k^{th} ในชั้นเอาต์พุตคือ

$$net_k^o = \sum_{j=1}^{J} v_{kj} y_j \tag{2.6}$$

$$z_k = f(net_k^o) \tag{2.7}$$

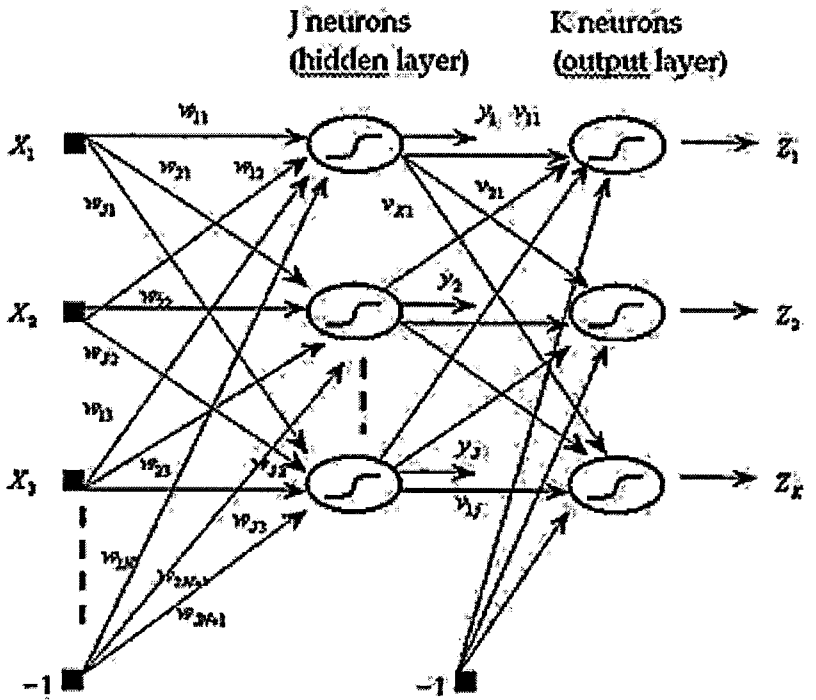
เมื่อ J เป็นจำนวนนิวรอนในชั้นซ่อน
ส่วนฟังก์ชันซิกมอยด์ (ฟังก์ชันกระตุ้น) คือ

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

$$f(net)=\frac{1}{1+e^{-\lambda net}} \tag{2.8}$$

เป็นฟังก์ชันที่อยู่ในการแปลงค่า Net Inputให้อยู่ในช่วง 0 ถึง 1ซึ่งค่าเอาต์พุตที่ได้จากฟังก์ชันจะอยู่ระหว่าง 0 ถึง 1 เมื่อ λ เป็นค่าความชัน และ e เป็นฟังก์ชันเอกซ์โพแนนเชียล

สำหรับเอาต์พุต z_k จะนำมาเปรียบเทียบกับค่าเป้าหมาย d_k ซึ่งค่าผิดพลาดที่ได้จะนำไปปรับแต่งค่าน้ำหนัก (Weight) ในชั้นซ่อนและชั้นเอาต์พุตของนิวรอน โดยจะศึกษาการฝึกหัดโครงข่ายประสาทเทียมโดยวิธีการแพร่ย้อนกลับ (Backpropagation Algorithm) ซึ่ง Back - Propagation เป็นอัลกอริทึมที่ใช้ในการเรียนรู้ของโครงข่ายประสาทเทียมวิธีหนึ่งที่นิยมใช้ใน Multilayer Perceptron เพื่อปรับค่าน้ำหนักในเส้นเชื่อมต่อระหว่างโหนดให้เหมาะสม โดยการปรับค่านี้จะขึ้นกับความแตกต่างของค่าเอาต์พุตที่คำนวณได้กับค่าเอาต์พุตที่ต้องการ



รูปที่ 2.10 แสดงสถาปัตยกรรมของ MFNN

2.3.1 ขั้นตอนการฝึกหัดโครงข่ายประสาทเทียมแบบแพร่แบบย้อนกลับ

ขั้นที่หนึ่ง : กำหนดอินพุต x จากชั้นอินพุต โดยเซตค่า $E=0$

ขั้นที่สอง : คำนวณหาค่าเอาต์พุตของแต่ละนิวรอนในชั้นซ่อน

06099

$$net_j^h = \sum_{i=1}^{N+1} w_{ji} x_i \tag{2.9}$$

$$y_j = f(net_j^h) \tag{2.10}$$

เมื่อ N เป็นจำนวนอินพุต x จากชั้นอินพุต
ฟังก์ชันกระตุ้น เป็น

$$f(net) = \frac{1}{1+e^{-\lambda net}} \tag{2.11}$$

ขั้นที่สาม : คำนวณหาค่าเอาต์พุตของแต่ละนิวรอนในชั้นเอาต์พุต

$$net_k^o = \sum_{i=1}^{J+1} v_{kj} y_j \tag{2.12}$$

$$y_j = f(net_k^o) \tag{2.13}$$

เมื่อ J เป็นจำนวนนิวรอนในชั้นซ่อน
ฟังก์ชันกระตุ้น เป็น

$$f(net) = \frac{1}{1+e^{-\lambda net}} \tag{2.14}$$

ขั้นที่สี่ : ปรับแต่งค่าน้ำหนักที่ชั้นเอาต์พุต

$$v_{kj} \leftarrow v_{kj} + c\lambda(d_k - z_k) z_k (1 - z_k) y_j \tag{2.15}$$

ขั้นที่ห้า : ปรับแต่งค่าน้ำหนักที่ชั้นซ่อน

$$w_{ji} \leftarrow w_{ji} + c\lambda^2 y_j (1 - y_j) x_i \bullet (\sum_{k=1}^{K} (d_k - z_k) z_k (1 - z_k) v_{kj}) \tag{2.16}$$

ขั้นที่หก : คำนวณเทอมค่าผิดพลาด

$$E \leftarrow E + \sum_{k=1}^{K} (d_k - z_k)^2 \qquad (2.17)$$

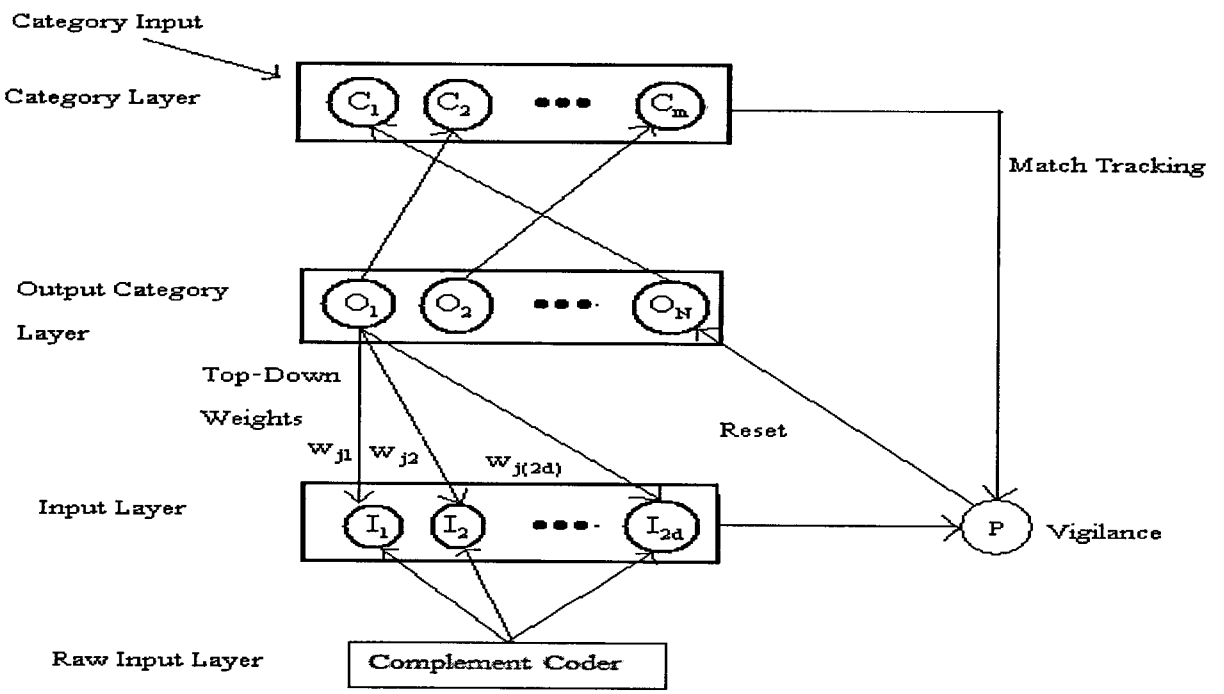
และทำซ้ำตั้งแต่ขั้นที่หนึ่ง

ขั้นที่เจ็ด : ถ้า E มีค่าต่ำกว่า 0.000001 จึงหยุด

2.4 Simplified Fuzzy ARTMAP (SFAM)

SFAM จะประกอบด้วย 2 input layer และ output layer และเส้นเชื่อมที่มีค่าน้ำหนัก ข้อดีของ SFAM คือถ้าโครงข่ายมีจำนวนตัวแปรน้อยจะช่วยลดค่าใช้จ่ายในการคำนวณ และความซ้ำซ้อนของอัลกอริทึมมีไม่มาก

Input ที่รับเข้ามาใน SFAM จะต้องมีค่าอยู่ระหว่าง 0 ถึง 1 โดย input จะถูกใส่เข้าไปในโครงข่ายและถูกส่งไปยัง "Black Box"(complement coder) และจะถูกส่งเข้าไปยังชั้น input layer ต่อ ค่าน้ำหนักที่แต่ละoutput โหนดจะมีผลต่อการเลือก outputโหนด ที่เหมาะสมจึงเรียกว่า "top-down weight"



รูปที่ 2.11 โครงสร้างของ SFAM

การทำ complement coder

$-\overline{a_i} = 1 - a_i$ (2.18)

$-I = (a, \overline{a}) = (a_1, ..., a_d, \overline{a_1}, ..., \overline{a_d})$

ให้ $\overline{a_i}$ คือ input ที่ได้จาก complement coder

a_i คือ input ที่ได้รับเข้ามา

d คือ จำนวนของ input

Activation function ใช้ในการหาว่า output โหนดตัวใดใน output layer ที่มีความคล้ายคลึงกับรูปแบบ input ที่รับเข้ามามากที่สุด โดยพิจารณาจาก output ที่มีค่า activation สูงที่สุด

$$T_j(I) = \frac{|I \wedge W_j|}{\alpha + |W_j|} \quad (2.19)$$

ให้ T_j = output activation

$\wedge$ = AND operation นิยามโดย

$$(p \wedge q)_i \equiv \min(p_i, q_i) \quad (2.20)$$

W_j = ค่าน้ำหนัก

$|\bullet|$ = ค่า Norm ของเวกเตอร์ นิยามโดย

$$|p| = \sum_{i=1}^{d} p_i \quad (2.21)$$

Vigilance (ρ) เป็นค่าที่ใช้ในการตัดสินว่ารูปแบบ input สามารถรวมกลุ่มกับ output โหนดที่ชนะได้หรือไม่ เนื่องจากค่า ρ มีค่าระหว่าง 0 ถึง 1 ดังนั้นถ้า ρ มีค่าสูงหมายความว่ารูปแบบ input และ output โหนดมีความคล้ายคลึงกันอย่างมากจึงรวมเป็นกลุ่มกันได้

Match tracking คือขั้นตอนที่ทำการตัดสินใจว่าการจัดกลุ่มระหว่างรูปแบบ input กับ output โหนดที่ชนะมีความเหมาะสมหรือไม่

$$\text{Match function} = \frac{|I \wedge W_j|}{I} \qquad (2.22)$$

ถ้า Match function $\geq \rho$ เรียกว่า สถานะ Resonance

หมายความว่าoutput โหนดที่ชนะเหมาะสมกับรูปแบบinput ที่เข้ามาและให้พิจารณา 2 กรณี

1. ถ้ารูปแบบ input กับ output อยู่ในประเภทเดียวกัน สามารถปรับค่าน้ำหนักได้ โดยสมการที่ใช้ในการปรับค่าน้ำหนักคือ

$$W_j^{new} = \beta(I \wedge W_j^{old}) + (1-\beta)W_j^{old} \qquad (2.23)$$

โดย W_j^{new} คือ ค่าน้ำหนักตัวใหม่

W_j^{old} คือ ค่าน้ำหนักตัวเก่า

β คือ อัตราการเรียนรู้

2. ถ้ารูปแบบ input กับ output อยู่คนละประเภทกันให้ทำการปรับค่า ρ ให้สูงขึ้นเพื่อไม่ให้ output โหนดตัวนั้นเป็นผู้ชนะ และทำการเลือก output โหนดตัวใหม่มาแทน

Match function $< \rho$ จะเรียกว่า สถานะ Mismatch Reset มีความหมายว่า output โหนดที่ชนะไม่เหมาะสมกับรูปแบบ

input ที่เข้ามา จึงไม่สามารถปรับค่าน้ำหนักได้ และให้ปรับค่า ρ ให้สูงขึ้นเพื่อกันไม่ให้ output โหนดตัวนั้นเป็นผู้ชนะ และเลือก output โหนดใหม่มาแทน

บทที่ 3

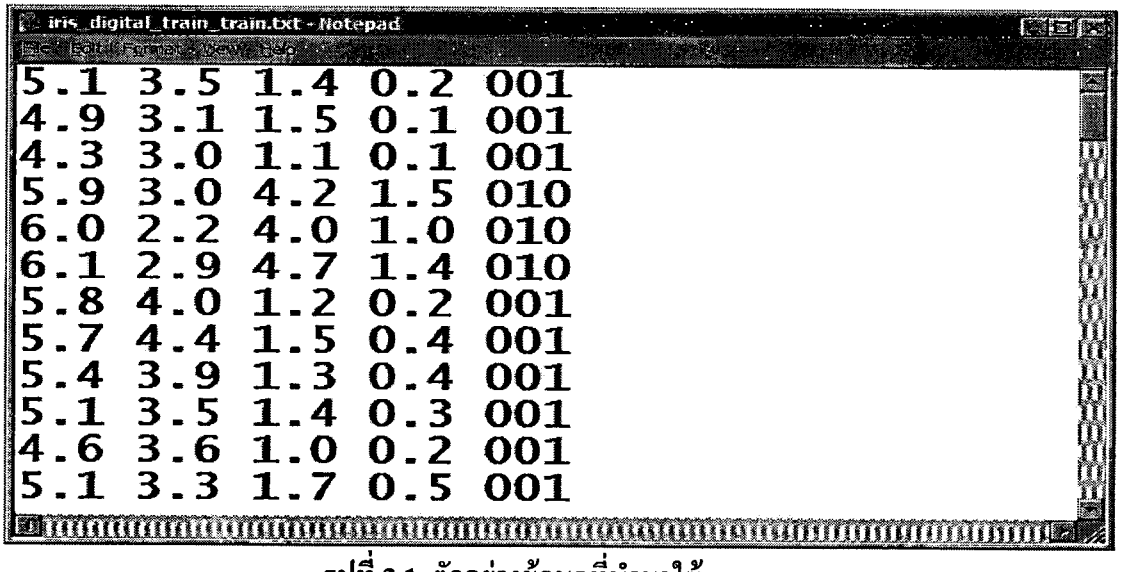
วิธีการดำเนินการศึกษา

ในการศึกษาโครงงานนี้ เป็นการนำทฤษฎีโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และ ซิมพลิไฟล์ฟัซซี่อาร์ทแมป (Simplified Fuzzy ARTMAP:SFAM) มาพัฒนาเป็นโปรแกรมคอมพิวเตอร์ เพื่อประยุกต์ใช้ในการจำแนกข้อมูล และเปรียบเทียบผลลัพธ์การทำงานของทั้งสองอัลกอริทึมเพื่อใช้ในการเปรียบเทียบประสิทธิภาพของรูปแบบทั้งสอง โดยแบ่งการดำเนินการศึกษาออกเป็น 3 ส่วน ดังนี้

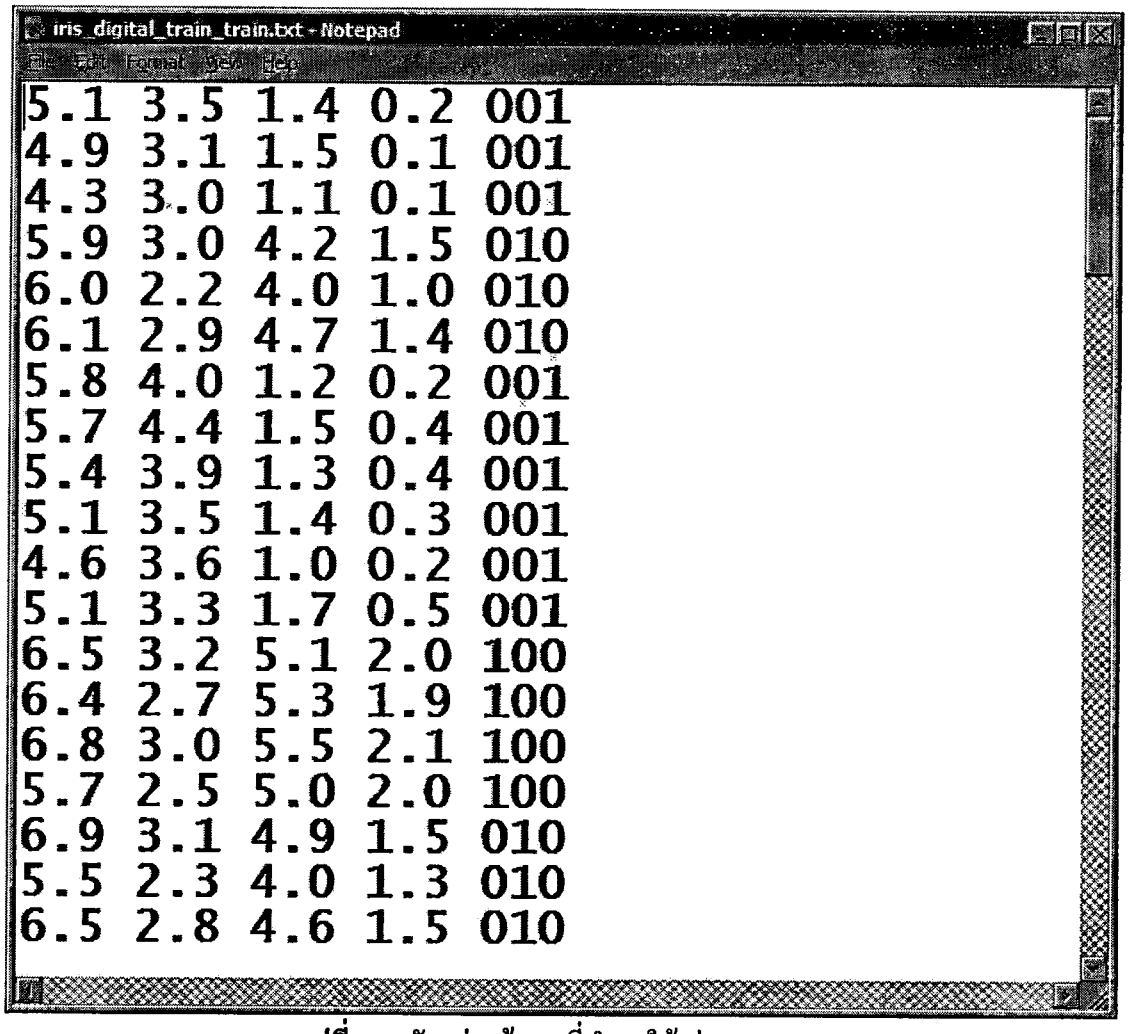
1. ข้อมูลที่ใช้กับระบบ
2. ขั้นตอนการดำเนินงาน
3. การพัฒนาโปรแกรมประยุกต์

3.1 ข้อมูลที่ใช้กับระบบ

โปรแกรมในโครงการนี้จะรับข้อมูลเข้าในลักษณะของ Text File ซึ่งประกอบไปด้วยคุณลักษณะต่างๆ ของชุดข้อมูลที่จะนำมาใช้ และค่าสุดท้ายของแต่ละชุดข้อมูลจะบอกถึงกลุ่มที่ชุดของข้อมูลนั้นๆ ถูกจัดให้อยู่ โดยแต่ละคุณลักษณะจะเป็นข้อมูลชนิดตัวเลขที่ถูกคั่นด้วยเครื่องหมาย ช่องว่าง ( ) ดังรูปที่ 3.1



รูปที่ 3.1 ตัวอย่างข้อมูลที่นำมาใช้



รูปที่ 3.1 ตัวอย่างข้อมูลที่นำมาใช้(ต่อ)

สามารถสรุปลักษณะของข้อมูลที่นำมาใช้ได้ ดังนี้

- ในแต่ละบรรทัด คือ Input 1 รูปแบบ
- Input แต่ละรูปแบบจะมีลักษณะเหมือนกันคือ ประกอบไปด้วยคุณลักษณะแบบเดียวกัน คั่นข้อมูลแต่ละคุณลักษณะด้วยเครื่องหมายช่องว่าง ( )
- ข้อมูลในแต่ละคุณลักษณะของ Input มีค่าเป็นตัวเลข

เนื่องจากต้องการให้ตรวจสอบระดับความถูกต้องของการเรียนรู้ของระบบได้ ดังนั้นจึงได้ทำการแบ่งข้อมูลทั้งหมดออกเป็น 2 ส่วน คือ

- ข้อมูลฝึกฝน (Train Data) สำหรับใช้ในการสอนการเรียนรู้ของระบบ
- ข้อมูลทดสอบ (Test Data) สำหรับใช้ทดสอบระบบที่ผ่านการเรียนรู้เสร็จ

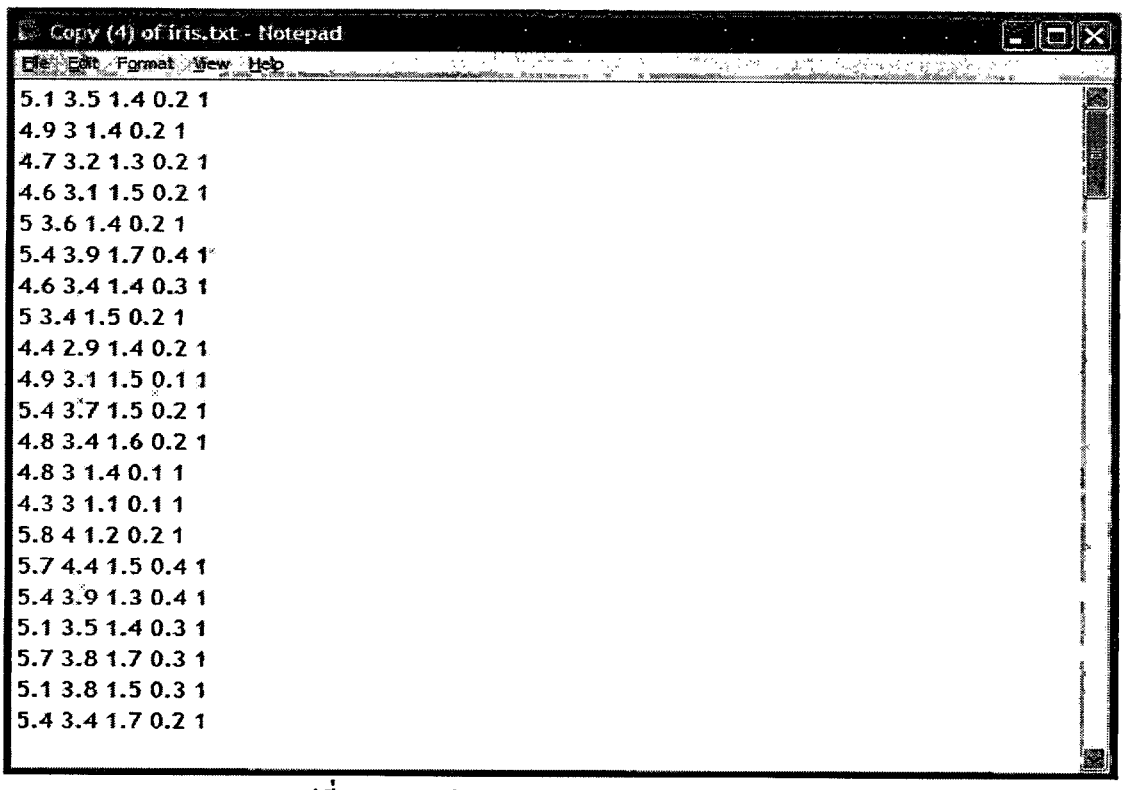
โดยข้อมูลฝึกฝน (Train Data) จะถูกนำเข้าสู่ระบบการจำแนกข้อมูล เมื่อเสร็จสิ้นขั้นตอนการเรียนรู้ (Learning) แล้ว จึงจะนำข้อมูลทดสอบ (Test Data) เข้าสู่ระบบเพื่อทดสอบประสิทธิภาพของระบบที่ได้จากการเรียนรู้

3.2 ขั้นตอนการดำเนินงาน

3.2.1 ขั้นตอนการดำเนินงาน

การศึกษาโครงงานนี้จะดำเนินการศึกษาโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และ ซิมพลิไฟล์ฟัซซีอาร์ทแมป (Simplified Fuzzy ARTMAP (SFAM)) โดยทำการพัฒนาระบบสร้างเป็นโปรแกรมคอมพิวเตอร์ เพื่อประยุกต์ใช้ในการศึกษาเปรียบเทียบประสิทธิภาพแบบจำลองในการจำแนกกลุ่มข้อมูล ทั้ง 2 แบบ ในการจำแนกหมวดหมู่ข้อมูล โดยมีขั้นตอนการดำเนินงานดังนี้

1. ขั้นตอนการจัดเตรียมข้อมูล โดยในขั้นตอนนี้จะมีการปรับปรุงข้อมูล เพื่อจัดเตรียมเป็น input vector และ output vector ที่เหมาะสมเพื่อป้อนให้กับโครงข่ายโดยที่เราจะสร้างขึ้น โดยจะทำการแปลงข้อมูลที่มีอยู่ให้อยู่ในรูป Text File ดังต่อไปนี้



รูปที่ 3.2 ตัวอย่าง Text File ของ IRIS ก่อนแปลง

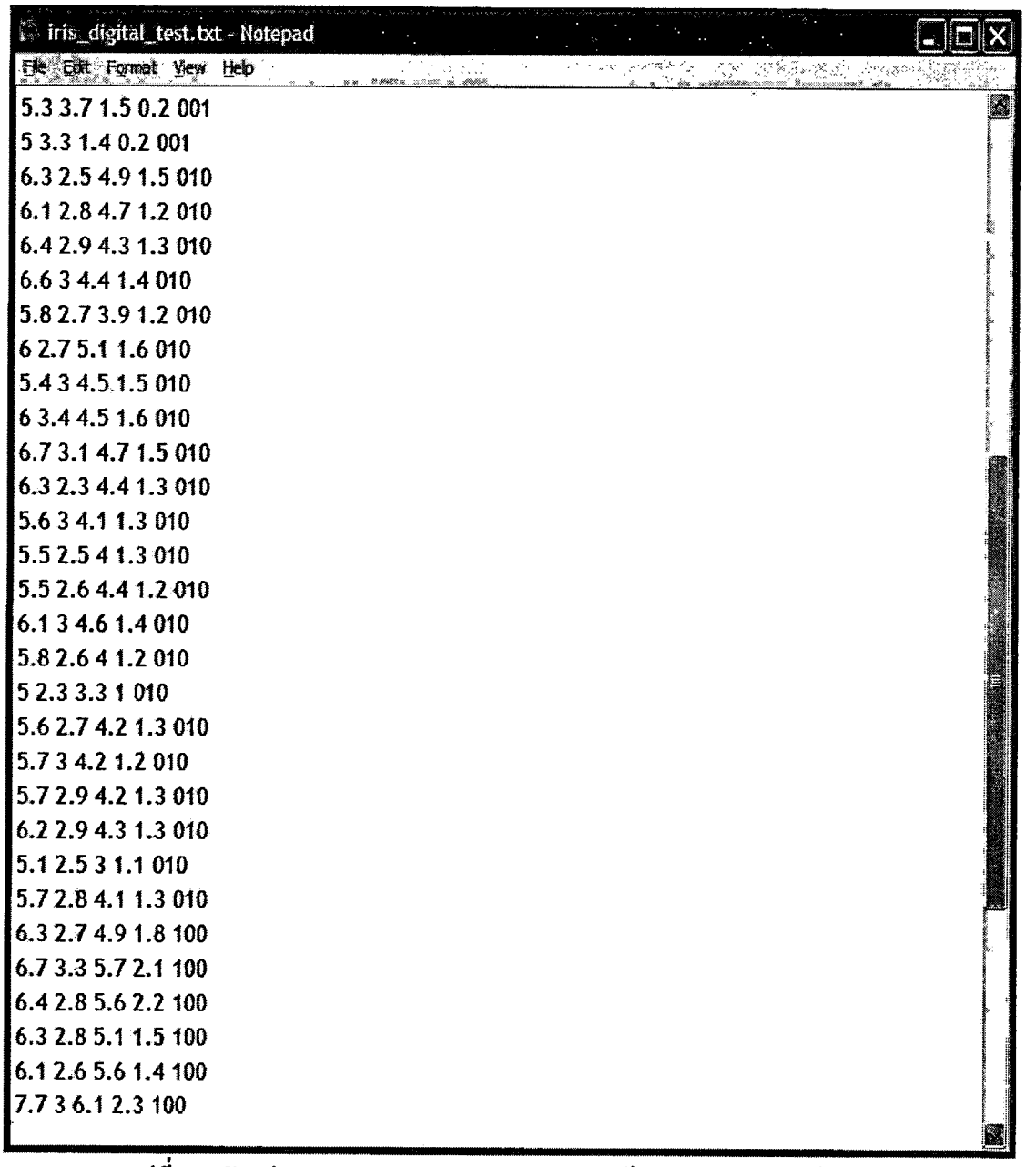
ทำการแปลงข้อมูลในส่วนของ Target ในตำแหน่งสุดท้ายเป็นแบบ bit

โดยกำหนดให้ ตัวอย่างข้อมูล IRIS มีทั้งหมด 3 กลุ่ม

-กลุ่มที่ 1 = 001

-กลุ่มที่ 2 = 010

-กลุ่มที่ 3 = 100



รูปที่ 3.3 ตัวอย่าง Text File ของ IRIS หลังแปลงข้อมูลในตำแหน่งเป้าหมาย

2. สุ่มเลือกข้อมูลสำหรับ

- ทำการฝึกสอนโครงข่าย (Train Data) โดยการเลือกข้อมูลจะต้องสุ่มเลือกให้ครอบคลุมทุกลักษณะของ Output
- ทำการทดสอบโครงข่าย (Test Data) โดยข้อมูลชุดทดสอบต้องไม่เป็นชุดเดียวกันกับชุดฝึกสอนโครงข่าย

3. นำชุดข้อมูลสำหรับฝึกสอน (Train Data) มาใส่เป็นข้อมูลชุด Input ในส่วนของการฝึกสอนโครงข่ายของโปรแกรมประยุกต์ เมื่อโปรแกรมทำการเรียนรู้ (Learning) เรียบร้อยแล้วจะมีการเก็บค่าต่างๆไว้สำหรับทดสอบข้อมูล

4. นำชุดข้อมูลสำหรับทดสอบ (Test Data) มาป้อนเป็นชุด Input ในส่วนของการทดสอบโครงข่ายโปรแกรมประยุกต์ โดยนำค่าที่ได้จากการเรียนรู้ (Learning) มาตั้งค่าการทดสอบโครงข่าย โดยจะแสดงออกมาเป็นเปอร์เซ็นความถูกต้องและกราฟสรุปภาพรวม

5. ดำเนินการดังขั้นต้นให้กับทั้ง 2 โครงข่าย และทดลองกับข้อมูลชุดอื่น

6. นำผลที่ได้จากการทดสอบโครงข่ายจาก 2 โครงข่ายมาเปรียบเทียบ วิเคราะห์การทำงานและสรุปผล

3.2.2 การออกแบบโปรแกรม

โครงการนี้จะต้องทำการจำลองการทำงานของโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และ ซิมพลิไฟล์ฟัซซีอาร์ทแมป(Simplified FuzzyARTMAP (SFAM)) โดยจะมีการแบ่งการทำงานออกเป็นโมดูลย่อยๆดังนี้

โมดูลการจัดการนำข้อมูลนำเข้า(Import Text File)

เป็นโมดูลที่จัดการการรับข้อมูล Text File เข้าสู่ระบบผ่านทาง I/O โดยเราสามารถเลือกตำแหน่งText File (File Path) ได้

โมดูลการนอร์มอลไลเซชันข้อมูล

1. เป็นการนำข้อมูลที่ได้จากขั้นตอนที่ 1 มาปรับเปลี่ยนให้ข้อมูลมีค่าอยู่ในช่วงที่กำหนด เพื่อให้ข้อมูลสามารถนำไปใช้งานกับอัลกอริทึมแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และ ซิมพลิไฟล์ฟัซซีอาร์ทแมป(Simplified Fuzzy ARTMAP (SFAM)) ได้ และให้สอดคล้องกับฟังก์ชันที่นำมาใช้ (Sigmoid function) การนอร์มอลไลเซชันทำได้ดังสมการ 3.1

$$normalize\ data = [\frac{data - \min}{\max - \min}] \times [\max_s - \min_s] + \min_s \qquad (3.1)$$

โดย max คือ ข้อมูลที่มีค่ามากสุด

min คือ ข้อมูลที่มีค่าน้อยที่สุด

max_s คือ ค่าข้อมูลที่ต้องการให้มากที่สุด

min_s คือ ค่าข้อมูลที่ต้องการให้น้อยที่สุด

โดยอัลกอริทึมแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และ ซิมพลิไฟล์ฟัซซีอาร์ทแมป (Simplified Fuzzy ARTMAP (SFAM)) ต้องแปลงข้อมูลให้อยู่ในช่วง 0 ถึง 1

โมดูลการจัดเตรียมค่าสำหรับโครงข่าย

เป็นโมดูลการตั้งค่าต่างๆให้กับโครงข่าย ซึ่งจะเป็นการรับค่าตัวแปรจากผู้ใช้ โดยค่าที่ต้องตั้งค่ามีดังต่อไปนี้

แบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation)

- จำนวน Input Node Layer
- จำนวน Hidden Node Layer
- จำนวน Output Node Layer
- อัตราการเรียนรู้ของโครงข่าย Learning Rate
- จำนวนรอบของการฝึกสอนโครงข่าย

ซิมพลิไฟล์ฟัซซีอาร์ทแมป (Simplified Fuzzy ARTMAP (SFAM))

- จำนวน Input Node Layer
- จำนวน Output Node Layer
- อัตราการเรียนรู้ของโครงข่าย Learning Rate
- ค่า Vigilance (ρ)
- ค่า Sum square error

โมดูลการสอนโครงข่าย

เป็นโมดูลการสอนโครงข่ายตามอัลกอริทึมของแต่ละโครงข่าย โดยใช้ข้อมูลฝึกฝน(Train Data) นำมาฝึกสอนโครงข่าย

โมดูลบันทึกข้อมูล

เป็นโมดูลที่ใช้ในการบันทึก Output ที่ได้จากการทำนายเทียบกับ target ที่แท้จริงของแต่ละ record นั้น รวมไปถึงน้ำหนักที่ได้จากการฝึกสอน

โมดูลการทดสอบโครงข่าย

เป็นโมดูลที่ใช้ทดสอบโครงข่ายที่เราสร้างขึ้นหลังจากที่โครงข่ายผ่านการเรียนรู้โดยข้อมูลชุดฝึกฝน(Train Data)มาแล้ว โดยโมดูลการทดสอบโครงข่ายนี้จะใช้ข้อมูลชุดทดสอบ(Test data) ในการทดสอบ โดยจะมีการประเมินความถูกต้องของข้อมูลด้วย

โมดูลแสดงผลลัพธ์

เป็นโมดูลที่ใช้แสดงผลลัพธ์การทำงานโดยในส่วนของการฝึกสอนจะมีการคำนวนค่าความผิดพลาดของแต่ละรอบ แสดงจำนวนรอบ และในส่วนของการทดสอบจะมีส่วนที่แสดง % ความถูกต้องของการจำแนกข้อมูลด้วย โดยสามารถแสดงเป็นกราฟและบันทึกเป็น file ได้

3.3 การพัฒนาโปรแกรมประยุกต์

3.3.1 เครื่องมือที่ใช้ในการพัฒนาโปรแกรม

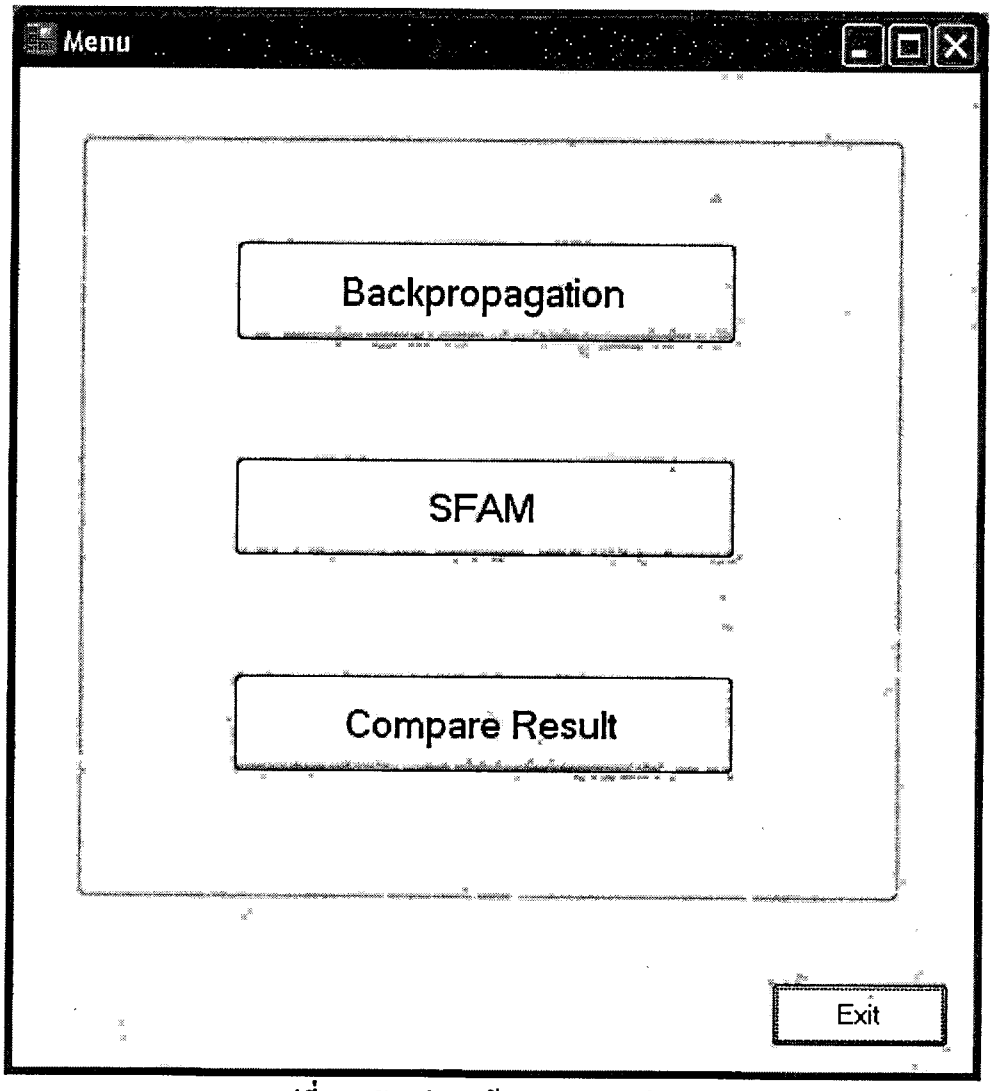
ในการศึกษาอัลกอริทึมแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และ ซิมพลิไฟล์ ฟัซซีอาร์ทแมป (Simplified Fuzzy ARTMAP (SFAM)) มีเครื่องมือดังนี้

- Desktop PC Pentium 4 2.4Ghz
- RAM 1 GB
- ระบบปฏิบัติการ Windows XP
- Harddisk Storage 120GB
- โปรแกรมไมโครซอฟท์วิชวลเบสิกดอทเน็ต (Microsoft Visual Studio.NET 2005) โดยเลือกใช้ ในส่วนของ VB.NET ในการพัฒนาโปรแกรมประยุกต์

3.3.2 ตัวอย่างส่วนติดต่อผู้ใช้โปรแกรม

ในการออกแบบโปรแกรมประยุกต์ เพื่อความสะดวกในการใช้งานโปรแกรม จึงได้ทำการออกแบบ GUI(Graphic User Interface) เพียง 3 หน้าจอหลักเท่านั้น

หน้าจอหลัก (Main Menu)

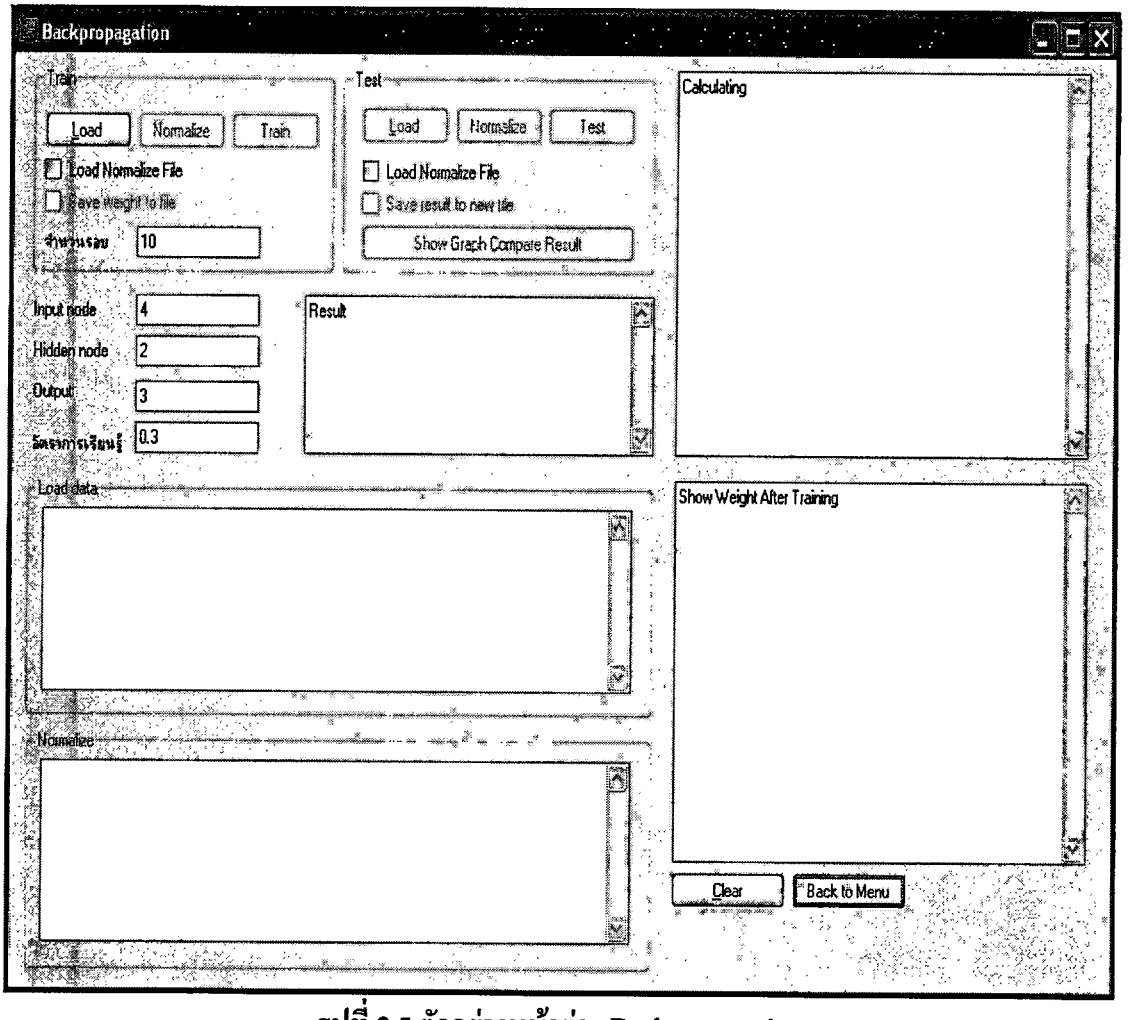


รูปที่ 3.4 ตัวอย่างหน้าจอหลักของโปรแกรม

หน้าต่างหลักของโปรแกรมประกอบด้วย

- Backpropagation : เป็นส่วนที่นำเข้าสู่โปรแกรมการจำแนกข้อมูลด้วยอัลกอริทึมแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation)
- SFAM : เป็นส่วนที่นำเข้าสู่โปรแกรมการจำแนกข้อมูลด้วยซิมพลิไฟล์ฟิชซีอาร์ทแมป (SFAM)
- Compare Result : เป็นส่วนที่นำเข้าสู่การแสดงกราฟเพื่อการเปรียบเทียบผลการทำนายของทั้ง 2 อัลกอริทึม เมื่อเทียบกับ Output ที่แท้จริง
- Exit : เป็นส่วนที่ออกจากโปรแกรม

หน้าต่างในส่วนของ Backpropagation



รูปที่ 3.5 ตัวอย่างหน้าต่าง Backpropagation

เมื่อกดปุ่ม Backpropagation แล้วโปรแกรมจะนำเข้ามาสู่หน้าต่าง Backpropagation หน้าต่างจะแสดงดังรูปที่ 3.5 โดยจะแบ่งการทำงานเป็น 3 ส่วน

- ตั้งค่าให้กับโครงข่าย
  - Input Node ใส่จำนวน Input ของข้อมูล
  - Hidden Node ใส่จำนวน Hidden node ที่ต้องการ
  - Output Node ใส่จำนวน Target ของข้อมูล
  - จำนวนรอบ ใส่จำนวนรอบของการเทรนข้อมูล
  - อัตราการเรียนรู้ กำหนดค่าอัตราการเรียนรู้ของโครงข่าย

- Train Data
    - ปุ่ม Load ใช้เลือก Text File เข้าสู่ระบบเพื่อทำการ Train โครงข่าย
    - ปุ่ม Normalize ทำการนอร์มอลไลเซชันข้อมูลที่โปรแกรมimport เข้ามา โดยจะแสดงข้อมูลในช่อง Load Data และจะแสดงข้อมูลหลังจากนอร์มอลไลเซชันเสร็จแล้วในช่อง Normalize
    - Load Normalize File ใช้เมื่อเลือก file ที่ทำการNormalize แล้ว
    - ปุ่ม Train ทำการสอนโครงข่ายให้ทำการเรียนรู้
    - Calculating จะแสดงค่า Error ของการฝึกสอนโครงข่าย
    - Weight จะแสดงค่าน้ำหนักของแต่ละโหนด
- Test Data

    -ปุ่ม Load ใช้เลือก Text File เข้าสู่ระบบเพื่อทำการ Test โครงข่าย

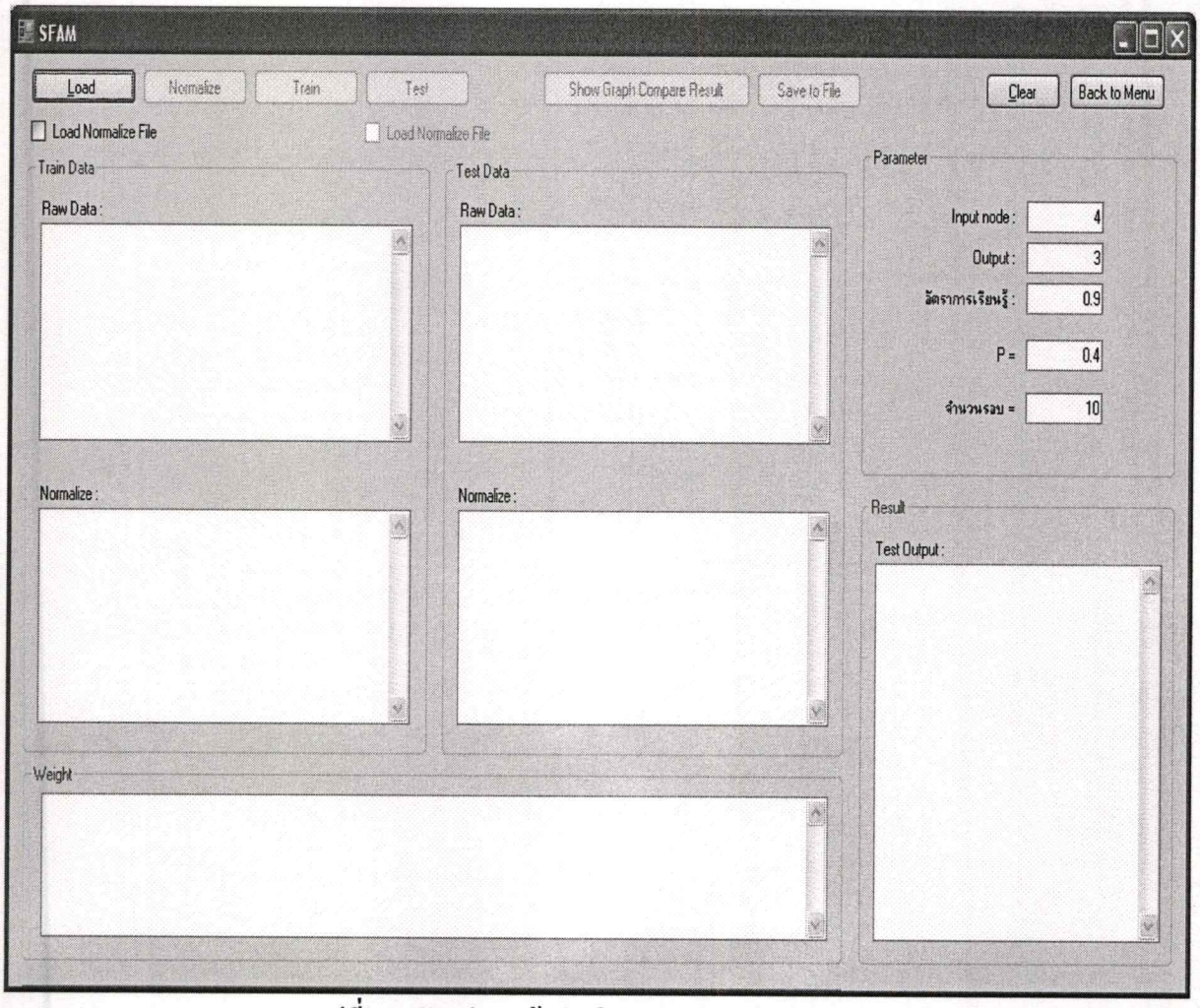
    -ปุ่ม Normalize ทำการนอร์มอลไลเซชันข้อมูลที่โปรแกรมimport เข้ามา โดยจะแสดงข้อมูลในช่อง Load Data และจะแสดงข้อมูลหลังจากนอร์มอลไลเซชันเสร็จแล้วในช่อง Normalize

    - Load Normalize File ใช้เมื่อเลือก file ที่ทำการNormalize แล้ว

    -ปุ่ม Test ทำการทดสอบโครงข่าย โดย ส่วนของ Result จะแสดงผลลัพธ์ของการทดสอบเป็นเปอร์เซนต์ความถูกต้อง

    - ปุ่ม Show Graph Compare จะแสดงผลลัพธ์ในรูปแบบของกราฟเพื่อดูค่ารวมของการทดสอบโครงข่าย

หน้าต่างในส่วนของ SFAM



รูปที่ 3.6 ตัวอย่างหน้าต่างโปรแกรม SFAM

เมื่อกดปุ่ม SFAM แล้วโปรแกรมจะนำเข้ามาสู้หน้าต่าง Backpropagation หน้าต่างจะแสดงดังรูปที่ 3.5 โดยจะแบ่งการทำงานเป็น 3 ส่วน

- ตั้งค่าให้กับโครงข่าย
  - Input Node ใส่จำนวน Input ของข้อมูล
  - Output Node ใส่จำนวน Target ของข้อมูล
  - อัตราการเรียนรู้ กำหนดค่าอัตราการเรียนรู้ของโครงข่าย
  - กำหนดค่า Vigilance (ρ)
  - จำนวนรอบในการฝึกสอนโครงข่าย

- Train Data
    - ปุ่ม Load ใช้เลือก Text File เข้าสู่ระบบเพื่อทำการ Train โครงข่าย
    - ปุ่ม Normalize ทำการนอร์มอลไลเซชันข้อมูลที่โปรแกรมimport เข้ามา โดยจะแสดงข้อมูลในช่อง Raw Data และจะแสดงข้อมูลหลังจากนอร์มอลไลเซชันเสร็จแล้วในช่อง Normalize
    - ปุ่ม Train ทำการสอนโครงข่ายให้ทำการเรียนรู้
    - Weight จะแสดงค่าน้ำหนักของแต่ละโหนด
- Test Data

- -ปุ่ม Test ลือก Text File เข้าสู่ระบบเพื่อทำการ Test โครงข่าย
  โปรแกรมจะทำการ นอร์มอลไลเซชันข้อมูลที่โปรแกรมimport เข้ามา โดยจะแสดงข้อมูลในช่อง Load Data และจะแสดงข้อมูลหลังจากนอร์มอลไลเซชันเสร็จแล้วในช่อง Normalize
  -ปุ่ม Test ทำการทดสอบโครงข่าย โดย ส่วนของ Result จะแสดงผลลัพธ์ของการทดสอบเป็นเปอร์เซนต์ความถูกต้อง
  - ปุ่ม Show Graph Compare จะแสดงผลลัพธ์ในรูปแบบของกราฟเพื่อดูค่ารวมของการทดสอบโครงข่าย
  - ปุ่ม Save to file จะทำการ save ค่าที่ทำนายได้ลงไปใน file

บทที่ 4

ผลการทดลอง

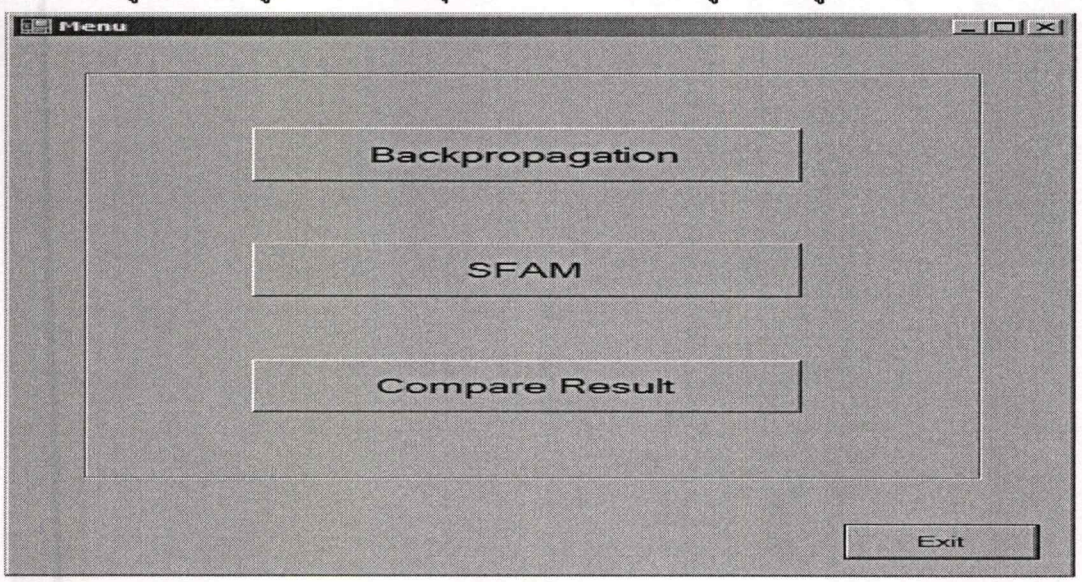
ในบทนี้จะกล่าวถึงรายละเอียดของขั้นตอนการใช้งานโปรแกรมประยุกต์ การแบ่งข้อมูลที่จะนำมาฝึกสอนระบบและทดสอบระบบ โดย โดยแบ่งการอธิบายออกเป็นส่วนต่าง ๆ ดังนี้

1. การใช้งานระบบ
2. ข้อมูลที่ใช้กับโปรแกรมประยุกต์
3. ผลการทดลองและการวิเคราะห์

4.1 การใช้งานระบบ

การทำงานของโปรแกรมประยุกต์การจำแนกกลุ่มข้อมูล มีหน้าจอการทำงานที่เรียบง่าย แต่ครบถ้วนในเรื่องการตั้งค่าต่างๆให้โครงข่าย รวมทั้งการแสดงผลในส่วนต่างๆไม่ว่าจะเป็นข้อมูลที่อ่านจากText File ข้อมูลที่ผ่านการแปลงข้อมูลแล้ว รวมไปถึงเปอร์เซ็นต์ความถูกต้องอื่นของการทำจำแนกข้อมูลด้วย อีกทั้งถ้าผู้ใช้ต้องการบันทึกข้อมูลลง Text File ก็สามารถเลือกได้ในหน้าจอเดียวกัน ทั้งหมดออกแบบเพื่อความสะดวกและเรียบง่ายในการใช้งาน โดยจะมีขั้นตอนการทำงานดังนี้

เมื่อผู้ใช้งานเข้าสู่โปรแกรมประยุกต์ จะเริ่มที่หน้าต่างเมนูหลัก ดังรูปที่ 4.1



รูปที่ 4.1 หน้าจอเมนูหลักของโปรแกรมประยุกต์ Classification Application

4.1.1 โปรแกรม Backpropagation

จากหน้าจอหลักผู้ใช้สามารถเริ่มการทำงานของโปรแกรมประยุกต์ โดยการเลือกอัลกอริทึมที่ผู้ใช้ต้องการ โดยโปรแกรมจะมีให้เลือก 2 อัลกริทึม คือ แบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และ ฟัชซี่อาร์ทแมป (Simplified Fuzzy ARTMAP:SFAM)

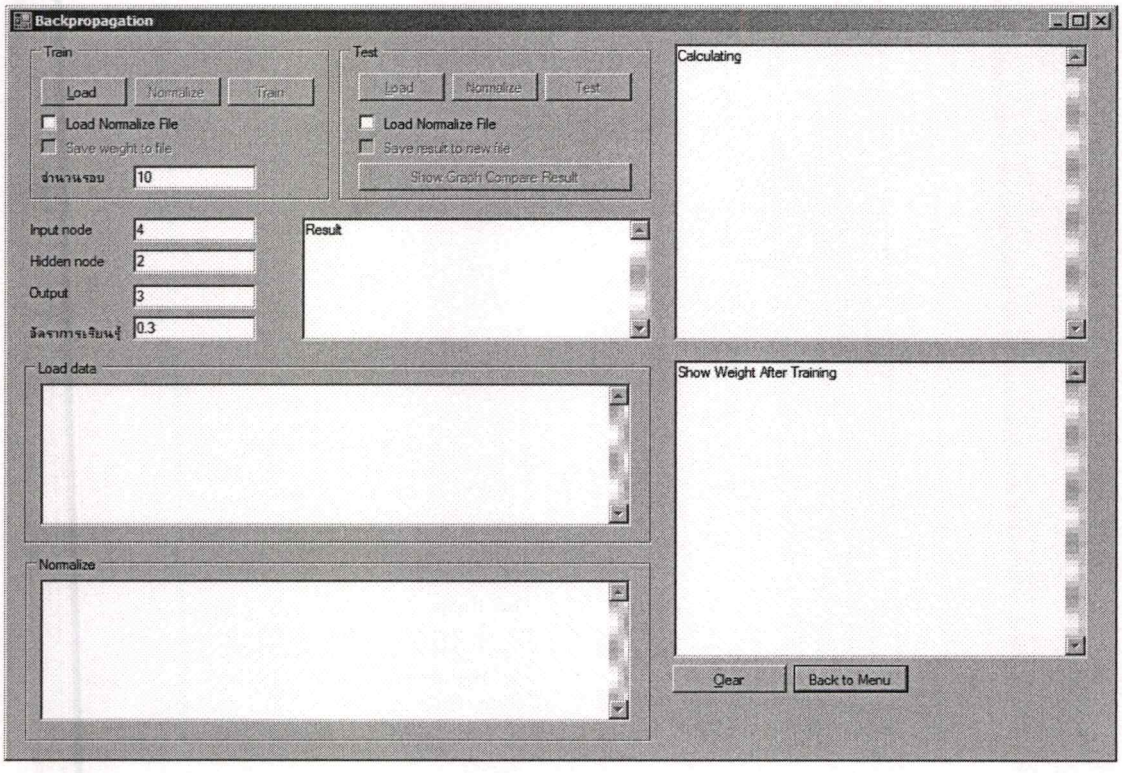
-ปุ่ม 

จะเข้าสู่กระบวนการจำแนกข้อมูลด้วยอัลกอริทึมแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation)

-ปุ่ม 

จะเข้าสู่กระบวนการจำแนกข้อมูลด้วยอัลกอริทึมฟัชซี่อาร์ทแมป (Simplified Fuzzy ARTMAP:SFAM)

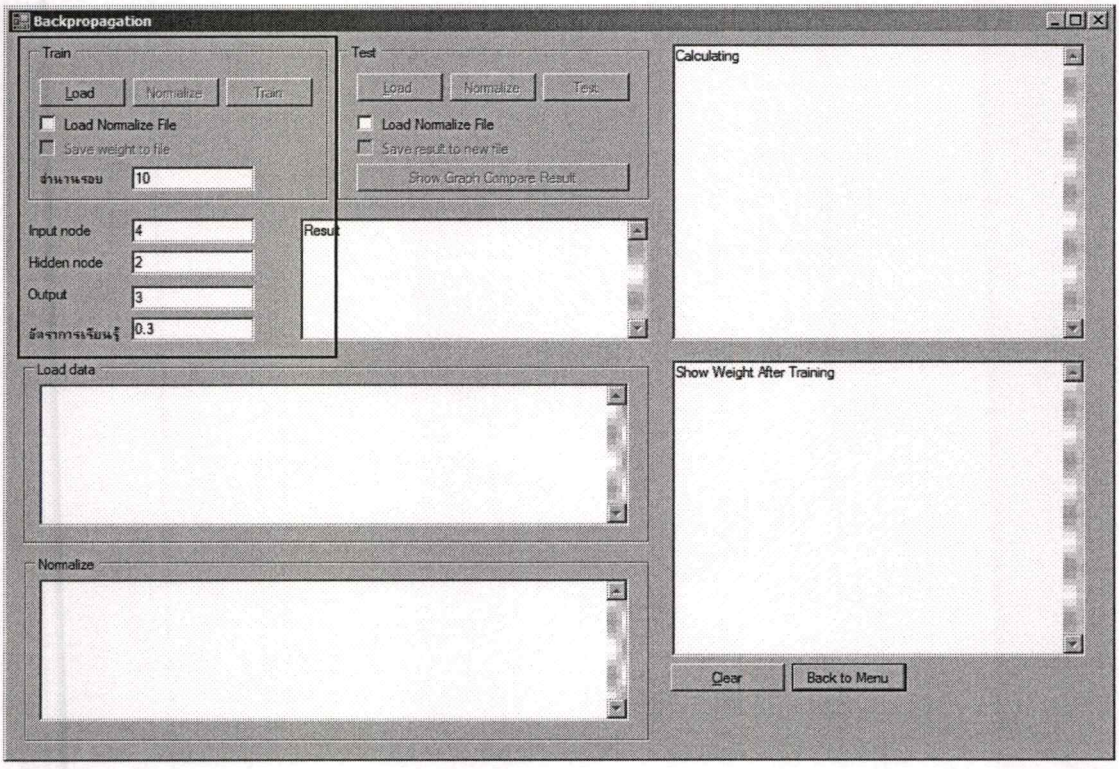
จากหน้าต่างเมนูหลัก เมื่อทำการเลือก Backpropagation จะนำเข้าสู่หน้าต่างการฝึกสอนและทดสอบโดยอัลกอริทึม แบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) ดังรูปที่ 4.2



รูปที่ 4.2 หน้าต่างโปรแกรมแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation)

เมื่ออยู่ในหน้าต่าง Backpropagation ผู้ใช้สามารถทำการฝึกสอนโครงข่ายและทดสอบโครงข่ายได้ในหน้าต่างเดียว ซึ่งในหน้าต่าง Backpropagation จะแบ่งการใช้งานเป็น 3 ส่วน คือ การฝึกสอนโครงข่าย(Train) และการทดสอบโครงข่าย(Test) โดยมีขั้นตอนการใช้งานตามส่วนงาน ดังนี้

4.1.1.1 ส่วนของการฝึกสอน(Train) อัลกอริทึม Backpropagation



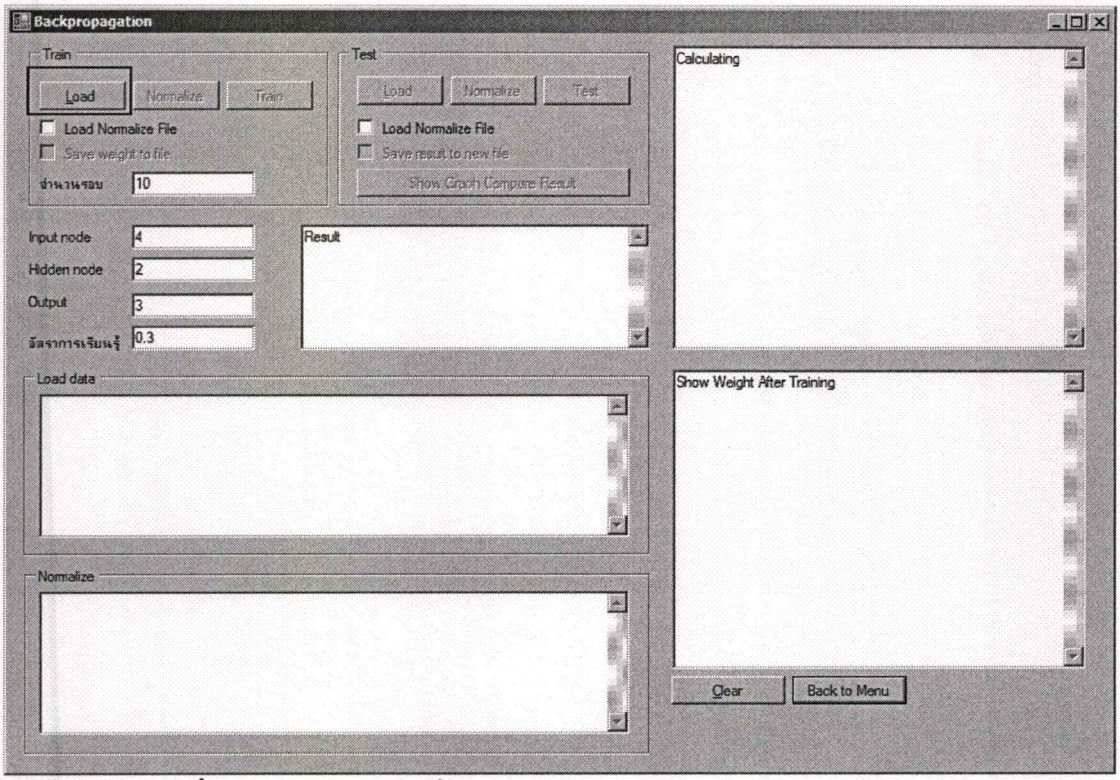
รูปที่ 4.3 แสดงส่วนหน้าจอของค่าที่ต้องทำการตั้งค่าก่อนการสอนโครงข่าย

ขั้นตอนที่ 1

- ทำการกำหนดตัวแปรต่างๆให้ครบตามที่โปรแกรมกำหนด ดังรูปที่ 4.3

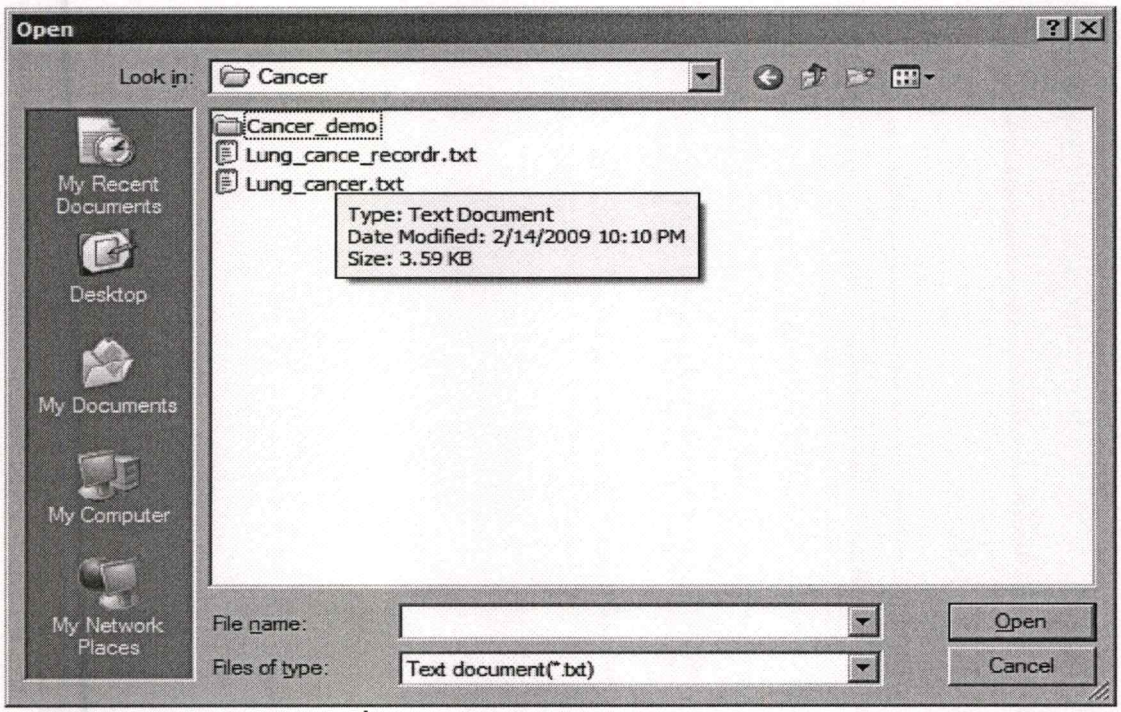
ขั้นตอนที่ 2

- เลือกปุ่ม Load เพื่อทำการเลือก Text File ที่ต้องการนำเข้ามาสอนโครงข่าย ถ้าข้อมูลที่เลือกเป็นข้อมูลที่ Normalize แล้ว ให้ทำการเช็คที่ช่อง Normalize File ดังรูปที่ 4.4



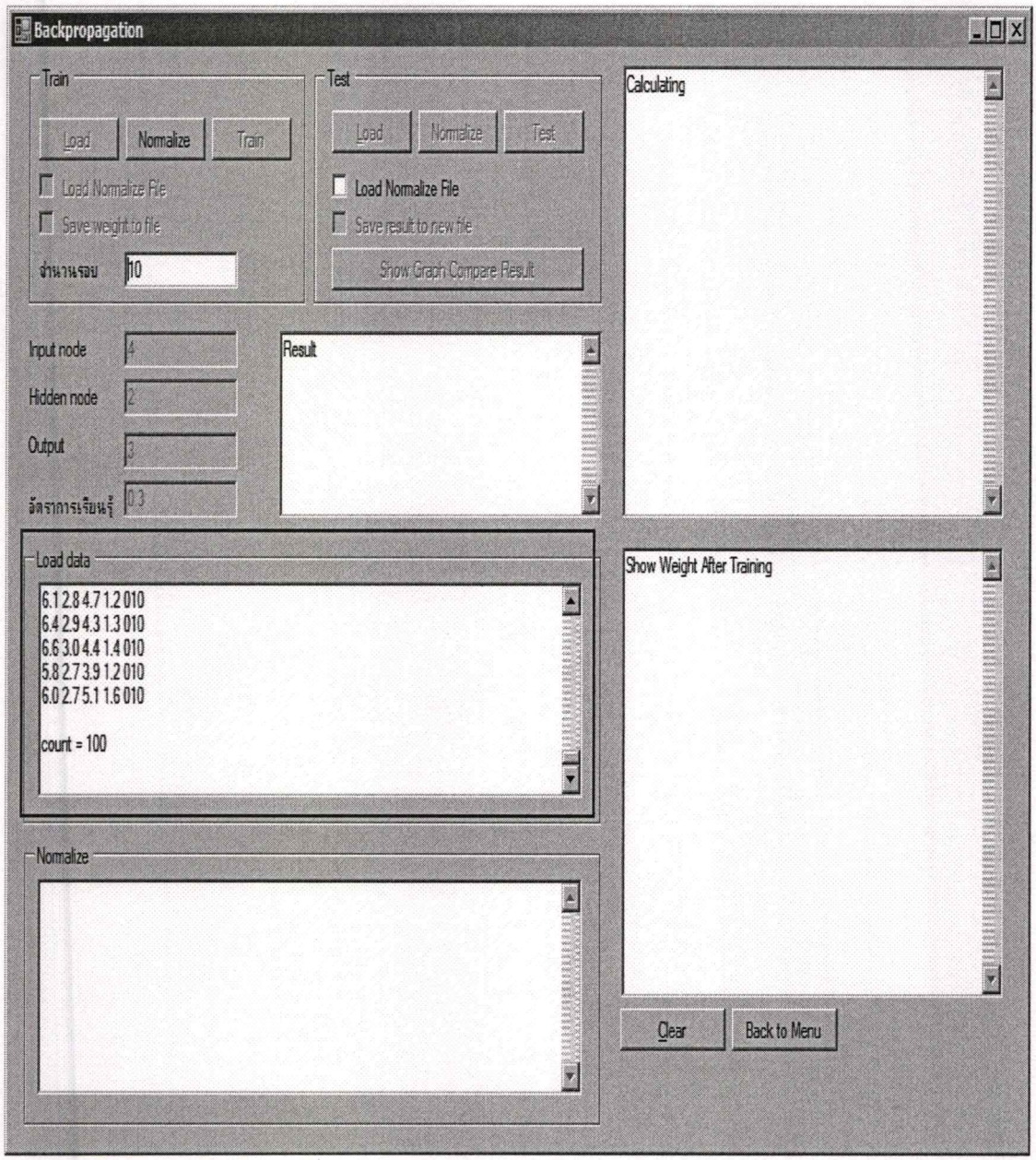
รูปที่ 4.4 แสดงปุ่มLoad ที่ทำการเลือกจาก ส่วนของการฝึกสอนโครงข่าย

หลังจากเลือกปุ่ม Load แล้ว โปรแกรมจะแสดง Dialog Box ให้เลือก Text File ที่เราต้องการนำมาฝึกสอนโครงข่าย ดังรูปที่ 4.5



รูปที่ 4.5 Dialog Box ในการเลือก Text File

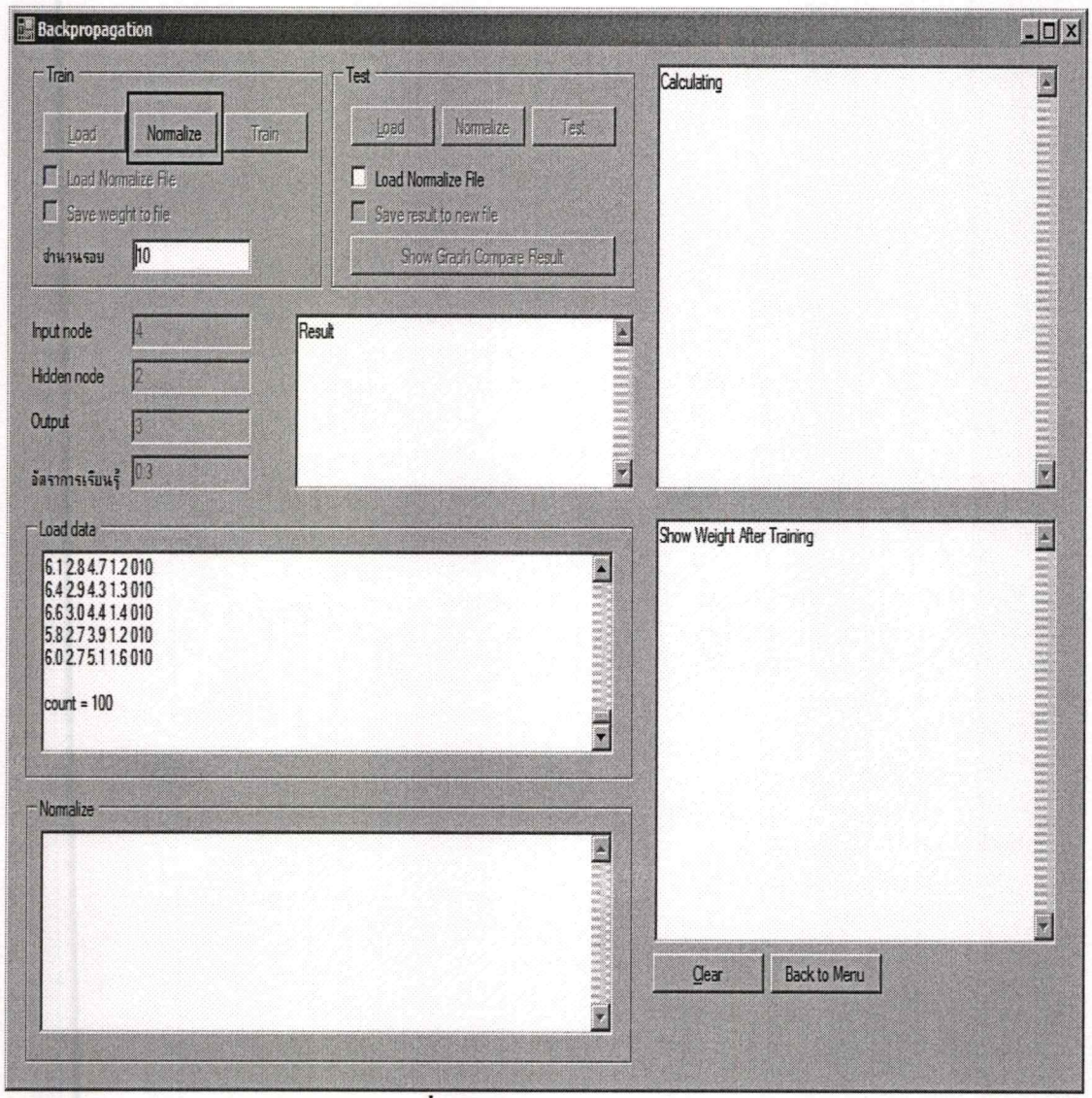
เมื่อทำการเลือก Text File ที่ต้องการจะใช้ในการฝึกสอนโครงข่ายเรียบร้อยแล้ว โปรแกรมประยุกต์จะแสดงข้อมูลที่import จาก Text File เข้ามาโดยแสดงในส่วนของ Load Data ดังรูปที่ 4.6



รูปที่ 4.6 หน้าต่างแสดงข้อมูลที่ import มาจาก Text File

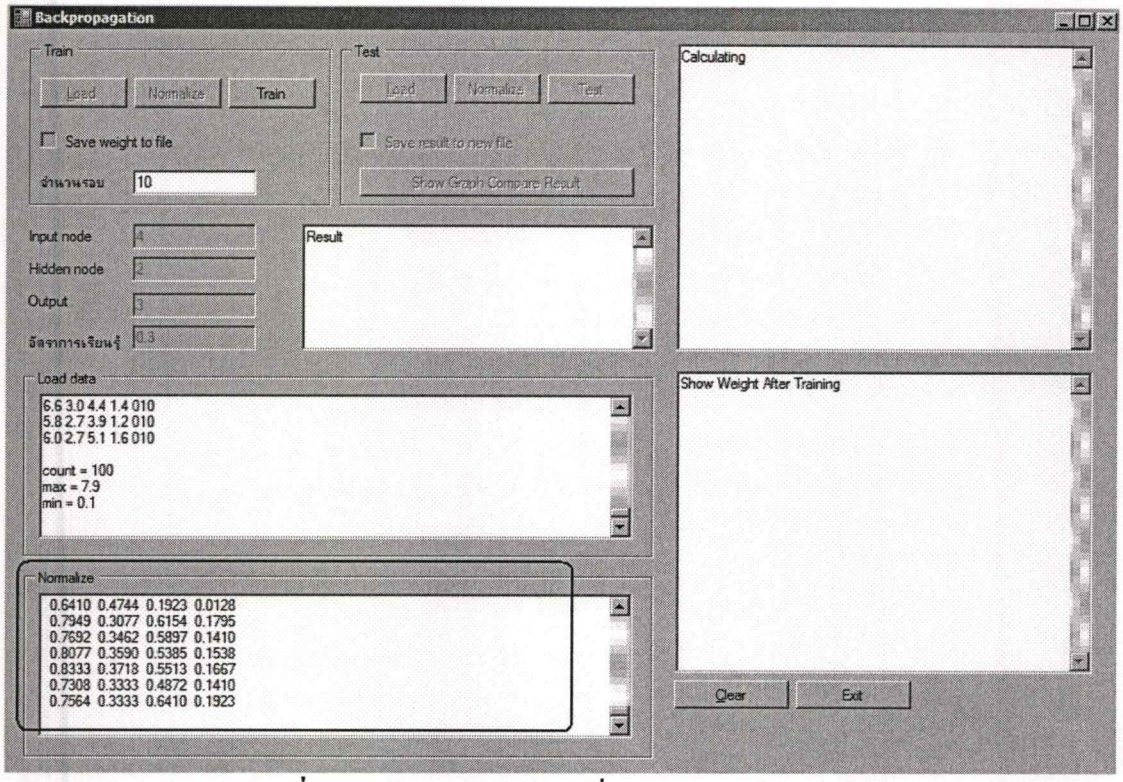
ขั้นตอนที่ 3

- ทำการกดปุ่ม Normalize เพื่อทำการNormalize ดังรูปที่ 4.7ข้อมูลตามที่กล่าวไว้ในบทที่ 3 ซึ่งอัลกอริทึม Backpropagation ต้องทำการแปลงข้อมูลให้อยู่ในช่วง 0 ถึง 1



รูปที่ 4.7 แสดงปุ่ม Normalize

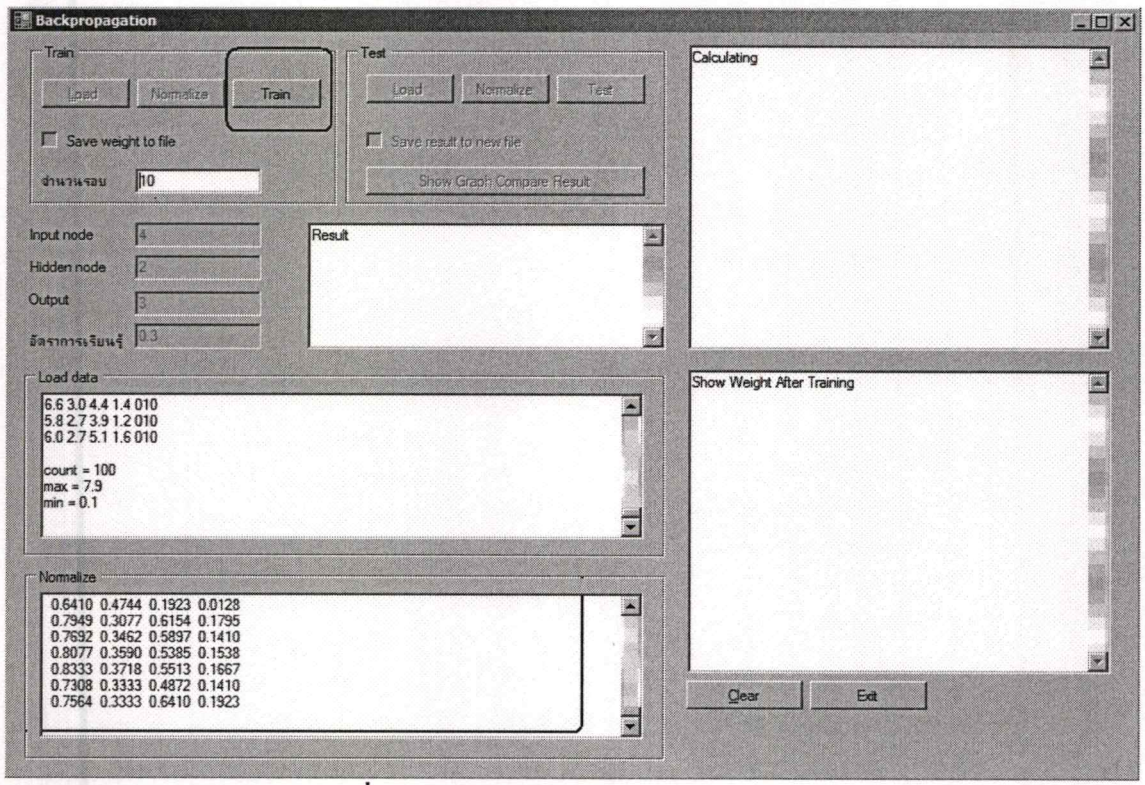
หลังจากทำการกดปุ่ม Normalize แล้วจะแสดงค่าที่ผ่านการแปลงข้อมูลแล้วในส่วนของ Normal ดังรูปที่ 4.8



รูปที่ 4.8 หน้าต่างแสดงข้อมูลที่ทำการแปลงข้อมูลแล้ว

ขั้นที่ 4

- ทำการกดปุ่ม Train เพื่อทำการฝึกสอนโครงข่าย ดังรูปที่ 4.9

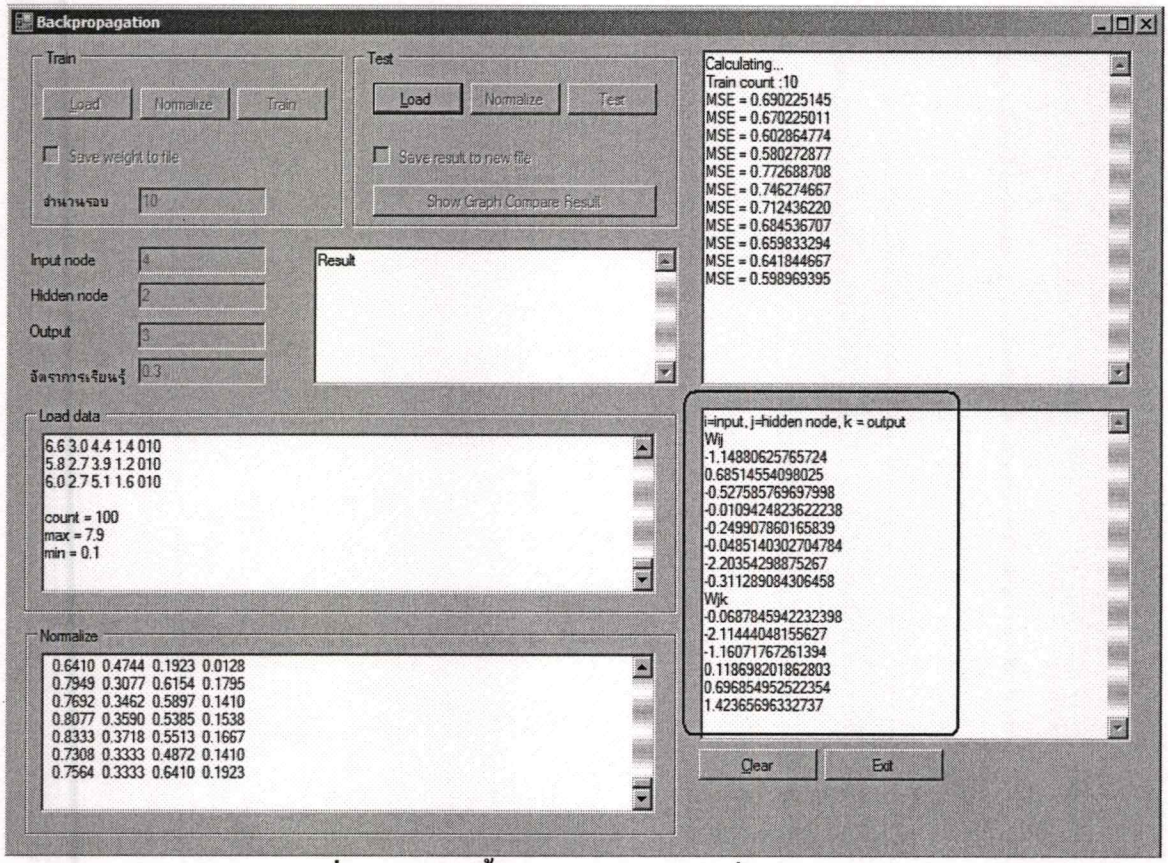


รูปที่ 4.9 หน้าต่างโปรแกรมแสดงปุ่ม Train

หลังจากที่ผู้ใช้กดปุ่ม Train แล้ว ผลลัพธ์ที่ได้จากการฝึกสอนโครงข่ายจะแสดง โดยจะแสดงผลดังนี้

- จะแสดง Weight ของแต่ละเส้น ที่เชื่อมจากnode ไป node

ดังรูปที่ 4.10



รูปที่ 4.10 แสดงน้ำหนักของแต่ละเส้นเชื่อมnode

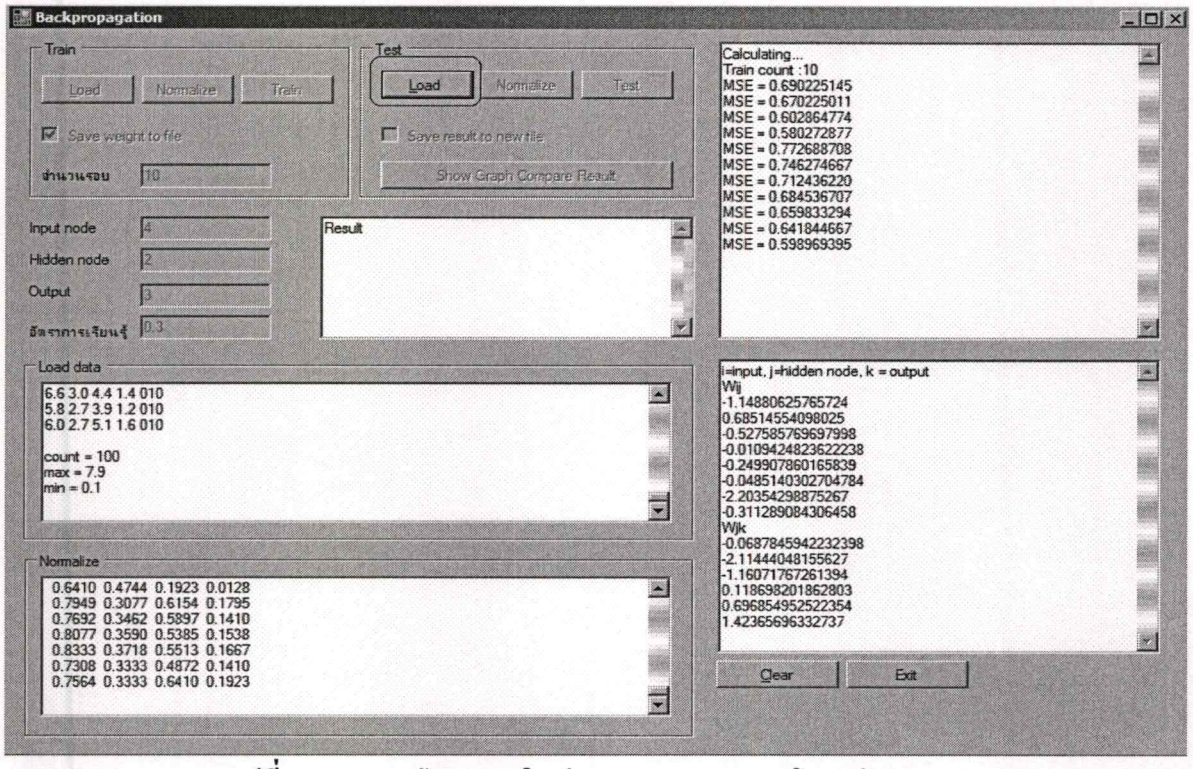
โปรแกรม Classification Applicationส่วนของ Backpropagation ในส่วนของการฝึกสอนโครงข่าย (Train) จะต้องทำการกดปุ่มตามลำดับของอัลกอริทึม ถ้าไม่ทำตามลำดับของอัลกอริทึมโปรแกรมจะไม่อนุญาตให้กดปุ่ม จะต้องทำตามลำดับเท่านั้น เพื่อความถูกต้องในการฝึกสอนโครงข่าย

4.1.1.2 ส่วนของการทดสอบ (Test) อัลกอริทึม Backpropagation

ก่อนเราจะทำการใช้งานส่วนของการทดสอบโครงข่ายได้จะต้องผ่านกระบวนการขั้นตอนการฝึกสอนโครงข่าย(Train) ก่อนที่จะเข้ากระบวนการการทดสอบโครงข่าย (Test) เนื่องจากการะบวนการทดสอบโครงข่าย(Test) จะใช้น้ำหนัก(Weight) ที่ได้จากการฝึกสอนโครงข่าย โดยจากหน้าต่างโปรแกรมBackpropagation มีขั้นตอนการใช้งานดังนี้

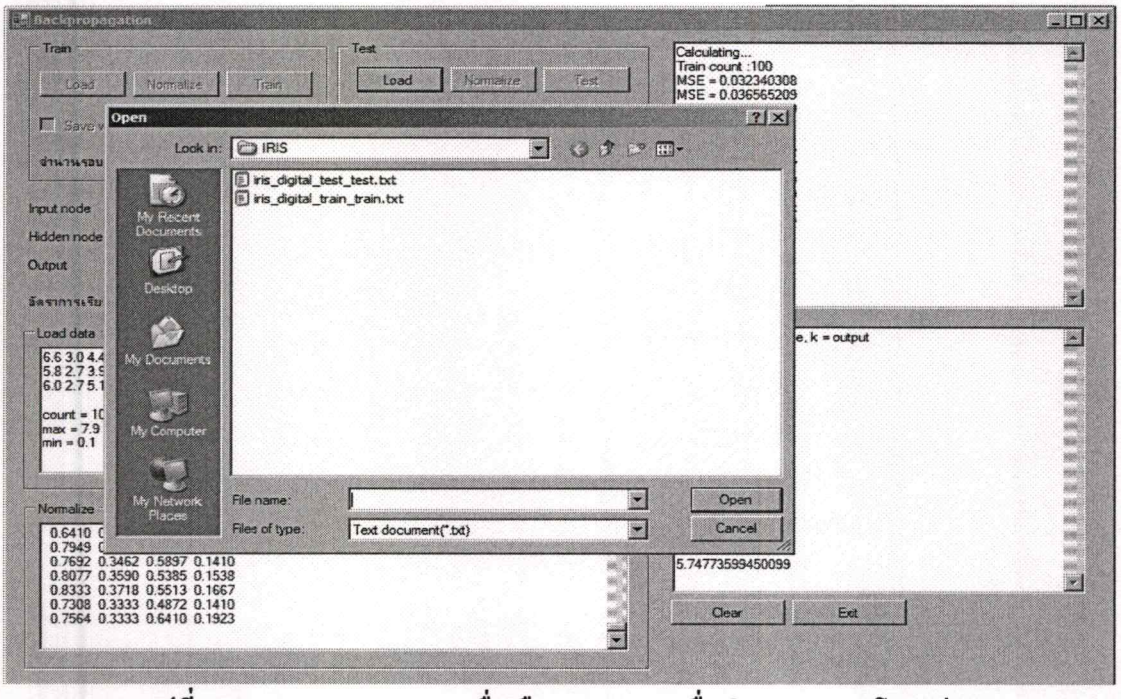
ขั้นตอนที่ 1

-ทำการเลือกปุ่ม Load เพื่อเลือก Text File ที่จะนำมาทดสอบ ดังรูปที่ 4.11



รูปที่ 4.11 แสดงปุ่ม Load ในส่วนของการทดสอบโครงข่าย

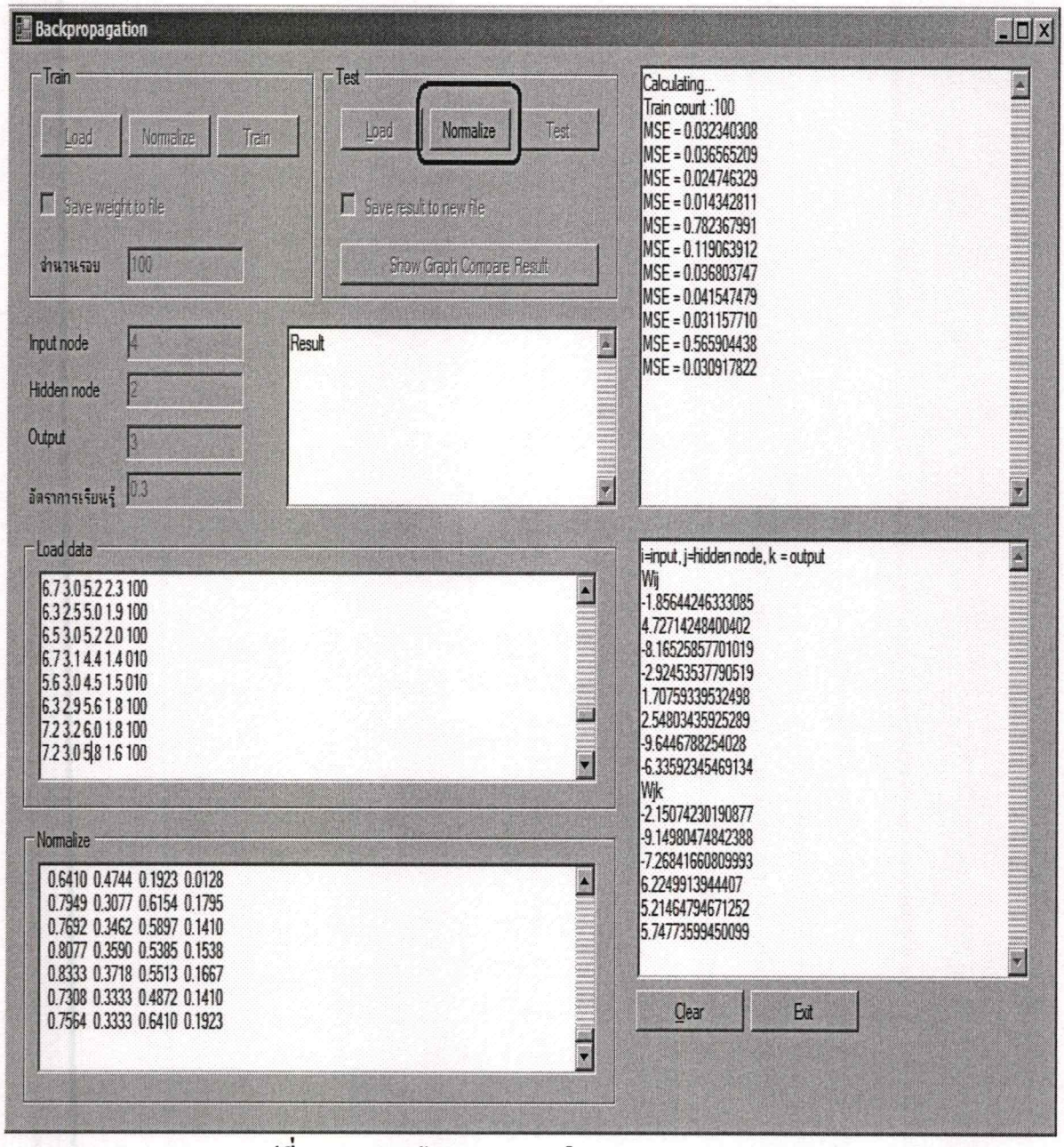
หลังจากทำการ Load โปรแกรมจะแสดง Dialog Box เพื่อทำการเลือก Text File ที่ต้องการจะนำมาทดลองโครงข่าย (Train) ดังรูปที่ 4.12



รูปที่ 4.12 แสดงDialog Box เพื่อเลือกText File เพื่อนำมาทดสอบโครงข่าย

ขั้นตอนที่ 2

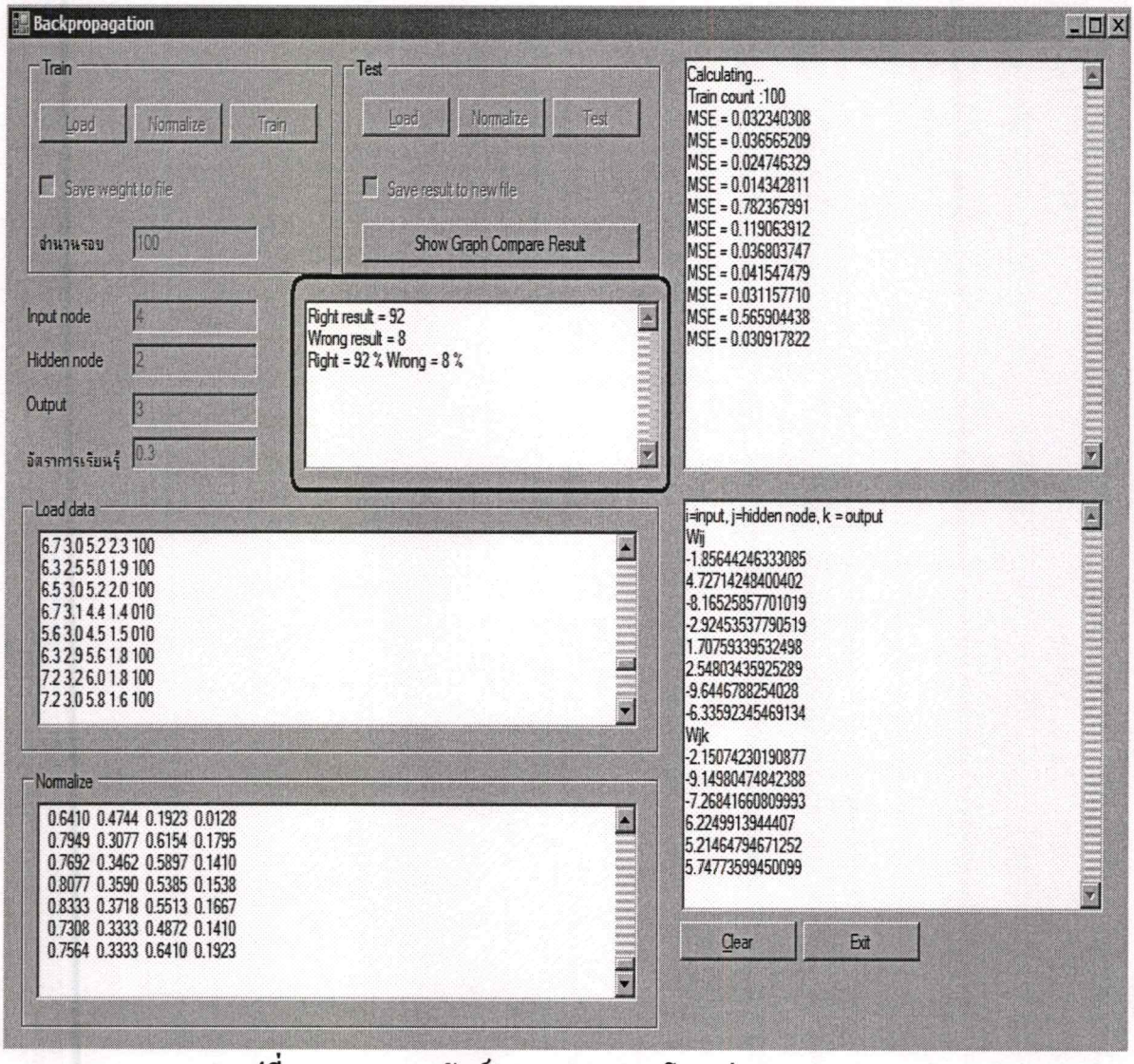
หลังจากที่เลือก Text File ที่ต้องการทดสอบโครงข่ายเข้าสู่โปรแกรมแล้ว โปรแกรมจะให้กดปุ่ม Normalize เพื่อทำการแปลงข้อมูล ดังรูปที่ 4.13



รูปที่ 4.13 แสดงปุ่ม Normalize ในการ Test ระบบ

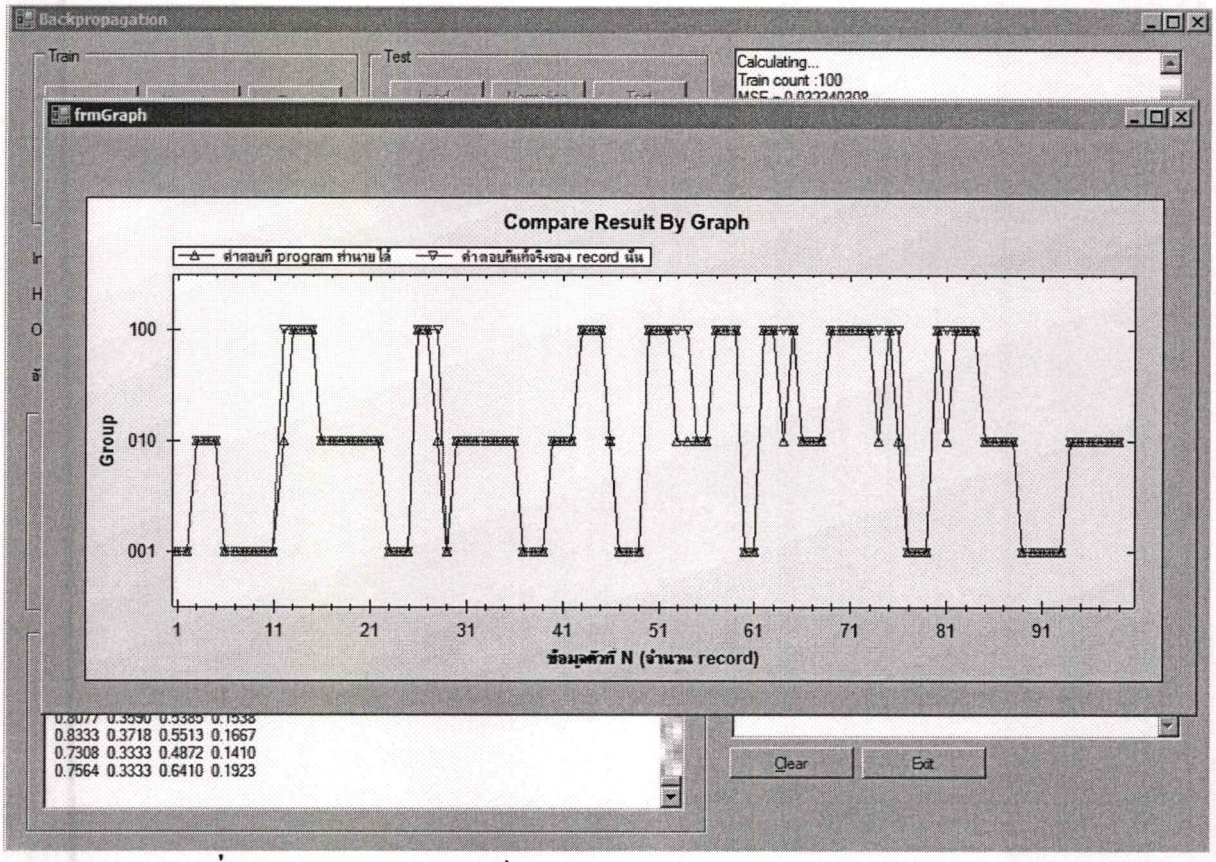
ขั้นตอนที่ 3

หลังจากที่ทำการแปลงข้อมูลที่ใช้ในการทดสอบโครงข่าย(Test) แล้ว ผู้ใช้จะกดปุ่ม Test เพื่อทำการทดสอบโครงข่ายจาก Text File ที่ได้เลือกเข้ามาทดสอบ โปรแกรมจะแสดงผลลัพธ์ที่ได้เป็นเปอร์เซนต์ความถูกต้อง ดังรูปที่ 4.14



รูปที่ 4.14 แสดงผลลัพธ์ของการทดสอบโครงข่าย (Test)

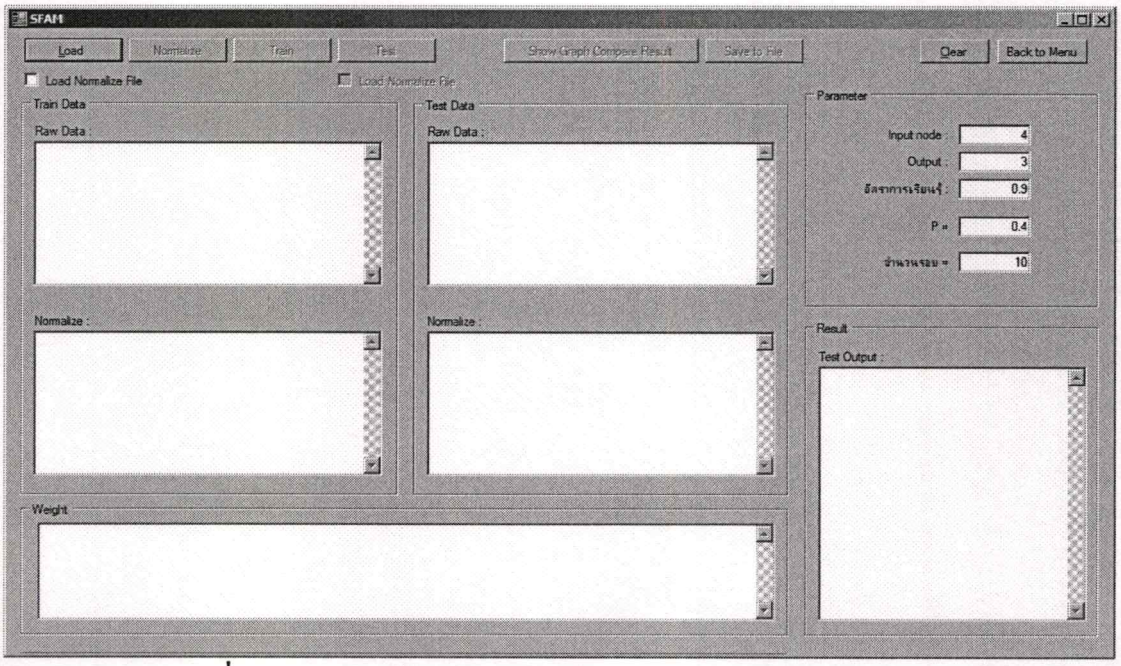
ในส่วนของการทดสอบโครงข่าย หลังจากที่การทดสอบเสร็จสิ้น ผู้ใช้สามารถดูผลลัพธ์ที่โครงข่ายทำจำแนกหมวดหมู่ เปรียบเทียบกับคำตอบที่แท้จริงได้ โดยการกดปุ่ม Show Graph Compare Result ดังรูปที่ 4.15



รูปที่ 4.15 แสดงกราฟผลลัพธ์จากการทดสอบระบบBackpropagation

4.1.2 โปรแกรม (Simplified Fuzzy ARTMAP: SFAM)

จากหน้าจอหลักเมื่อผู้ใช้ ทำการกดปุ่ม SFAM จะปรากฏหน้าจอหลักของ SFAM ขึ้นมาดังรูปที่ 4.16



รูปที่ 4.16 แสดงหน้าจอหลักของโปรแกรม SFAM

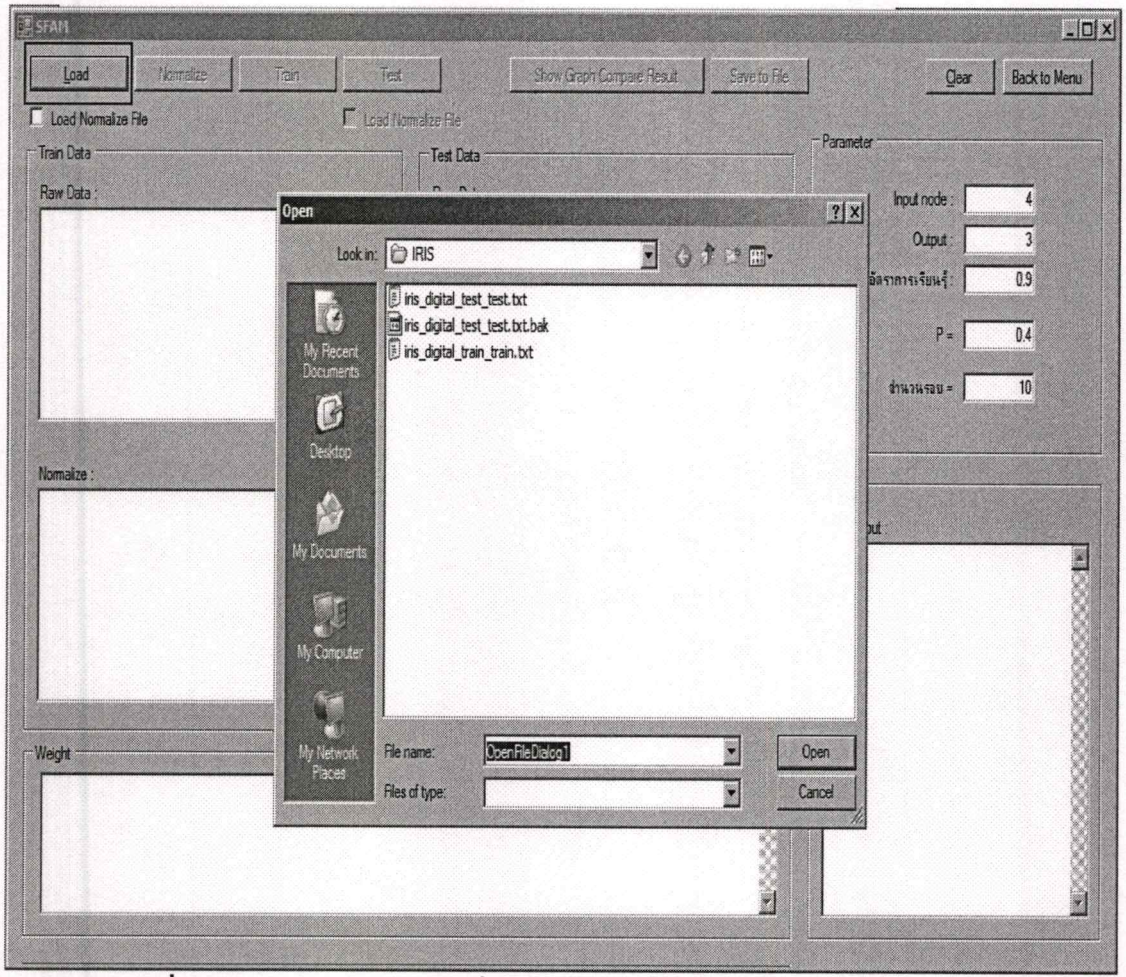
ส่วนของการฝึกสอน(Train) โครงข่าย SFAM

ขั้นตอนที่ 1

- ทำการตั้งค่า เริ่มต้นของโครงข่าย SFAM ก่อน โดยกำหนดค่าดังนี้

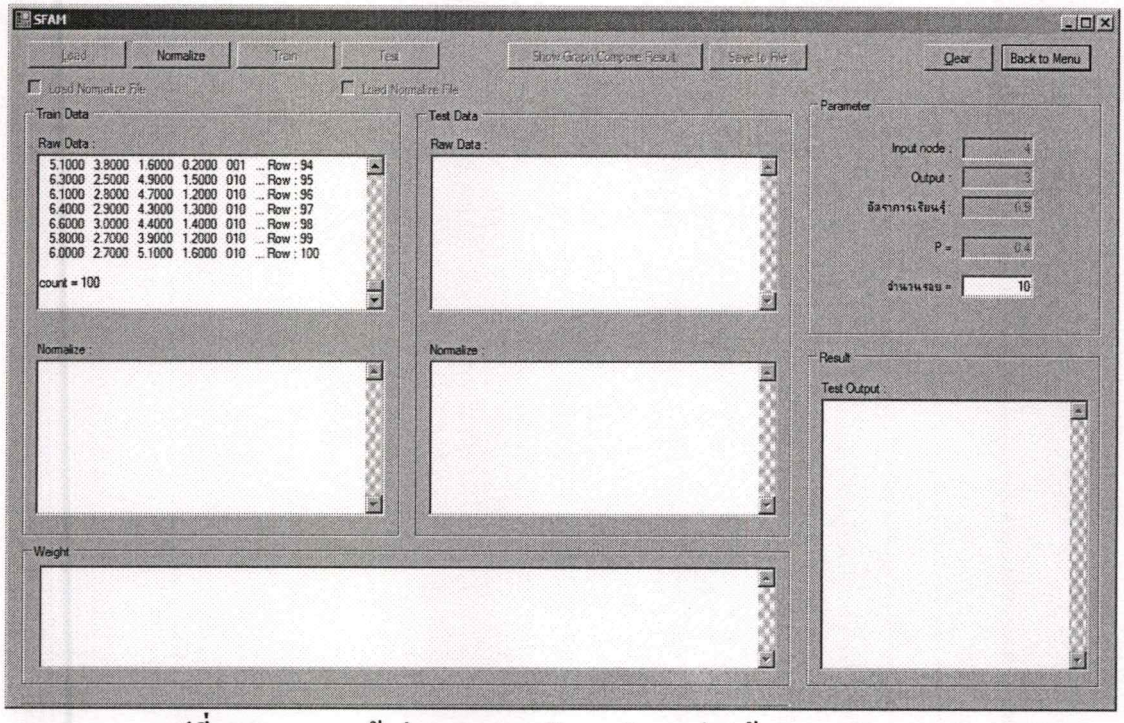
  - input node: จำนวนnode ของ input ข้อมูล
  - output node: จำนวนnode ของกลุ่มเป้าหมาย
  - อัตราการเรียนรู้
  - ค่ากำหนดค่า Vigilance (ρ)

ทำการกดปุ่ม Load ในหน้าจอ SFAM หน้าต่างโปรแกรมจะแสดง Dialog Box เพื่อให้เลือก Text File ที่ต้องการจะทำการนำมาฝึกสอน(Train) โครงข่ายดังรูปที่ 4.17



รูปที่ 4.17 แสดง Dialog Box เพื่อเลือก Text File ในการสอนโครงข่าย SFAM

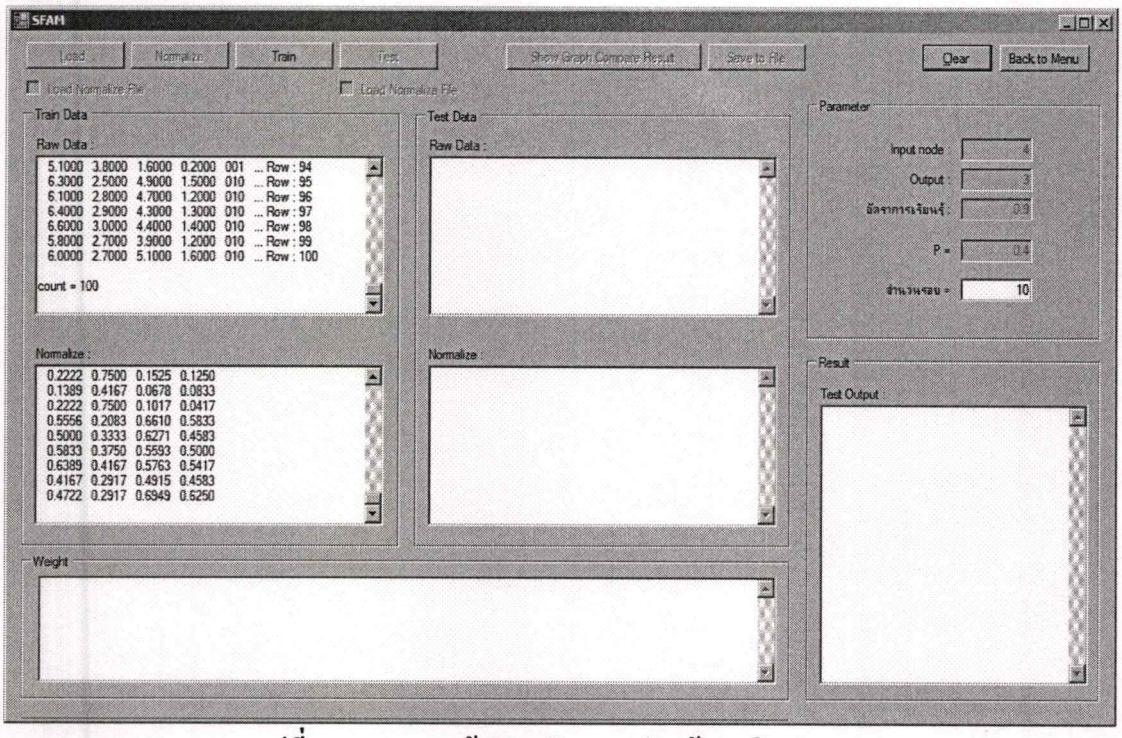
ข้อมูลที่อ่านจาก Text File จะแสดงดังรูปที่ 4.18



รูปที่ 4.18 แสดงหน้าต่างSFAM หลังจากทำการอ่านข้อมูลจาก Text File

ขั้นตอนที่ 2

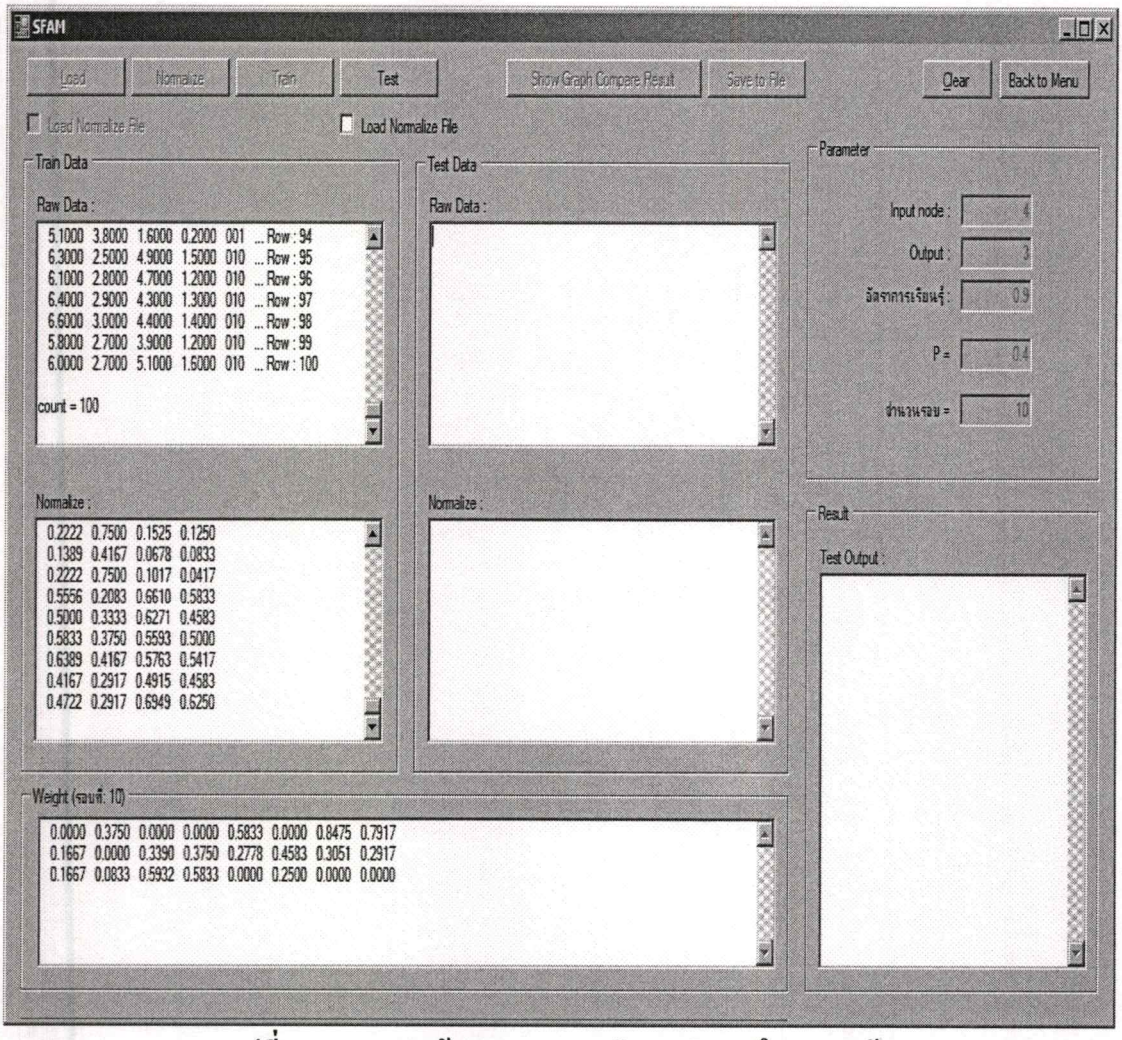
- ทำการ Normalize ข้อมูลโดยการกดปุ่ม Normalize เพื่อแปลงข้อมูลให้อยู่ในช่วง 0 ถึง 1 โดยข้อมูลที่ถูกแปลงแล้วจะแสดงใน



รูปที่ 4.19 แสดงหน้าจอหลังจากแปลงข้อมูลใน SFAM

ขั้นตอนที่ 3

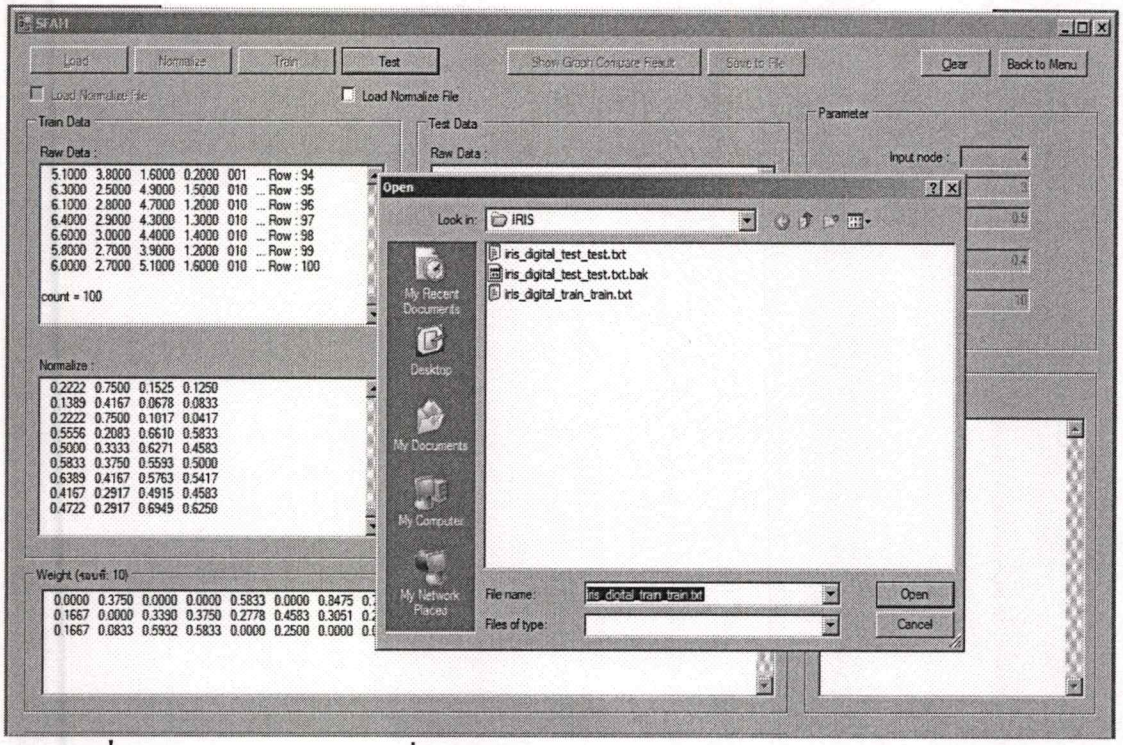
-ทำการกดปุ่ม Train เพื่อฝึกสอนโครงข่าย เมื่อทำการฝึกสอน แล้วจะได้ weight มาทำใช้สำหรับทำการทดสอบโครงข่ายต่อไป



รูปที่ 4.20 แสดงหน้าจอ SFAM หลังจากทำการฝึกสอนแล้ว

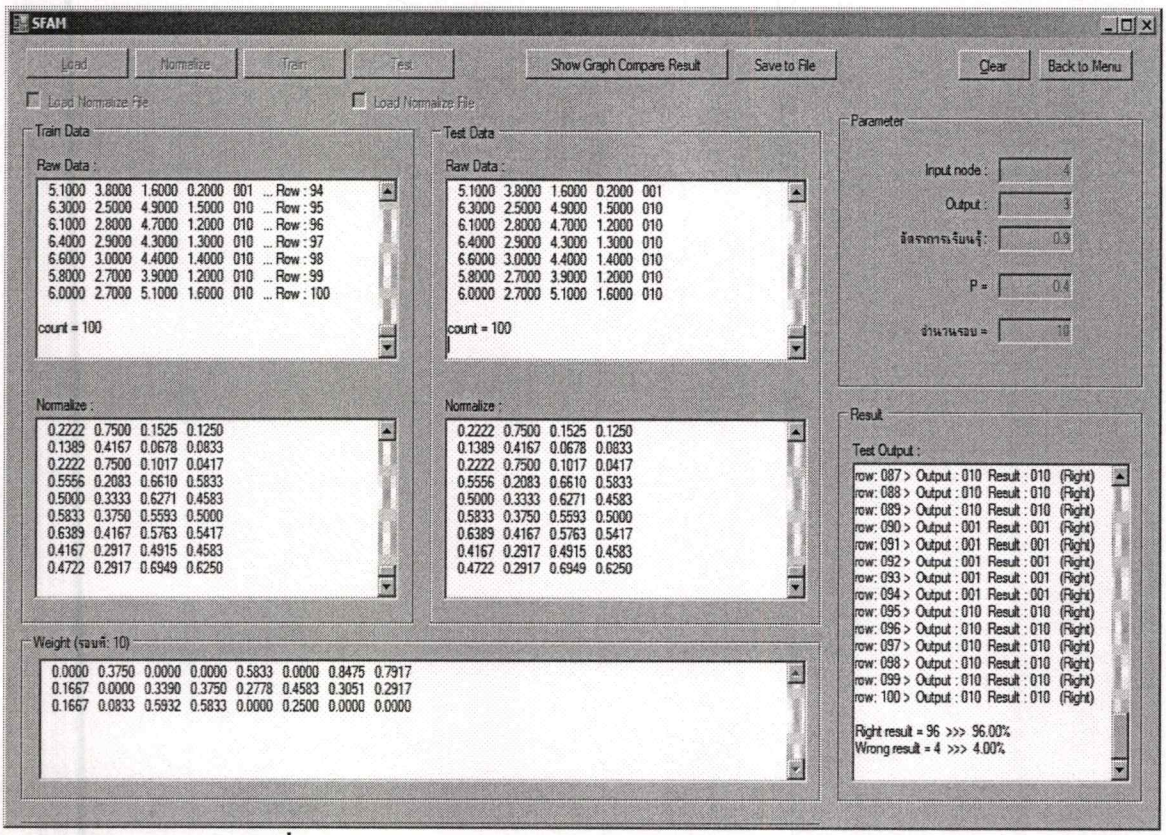
ส่วนของการทดสอบ(Test) โครงข่าย SFAM

เมื่อผู้ใช้ได้ดำเนินฝึกสอนโครงข่าย SFAM แล้วตามลำดับ ผู้ใช้สามารถใช้ส่วนของการทดสอบ(Test)โครงข่ายSFAM โดยการ เลือกที่ปุ่ม Test เมื่อทำการเลือกที่ปุ่ม Test แล้ว โปรแกรมจะแสดง Dialog Box เพื่อให้ผู้ใช้เลือก Text File ที่ต้องการนำมาทดสอบโครงข่าย ดังรูปที่ 4.21



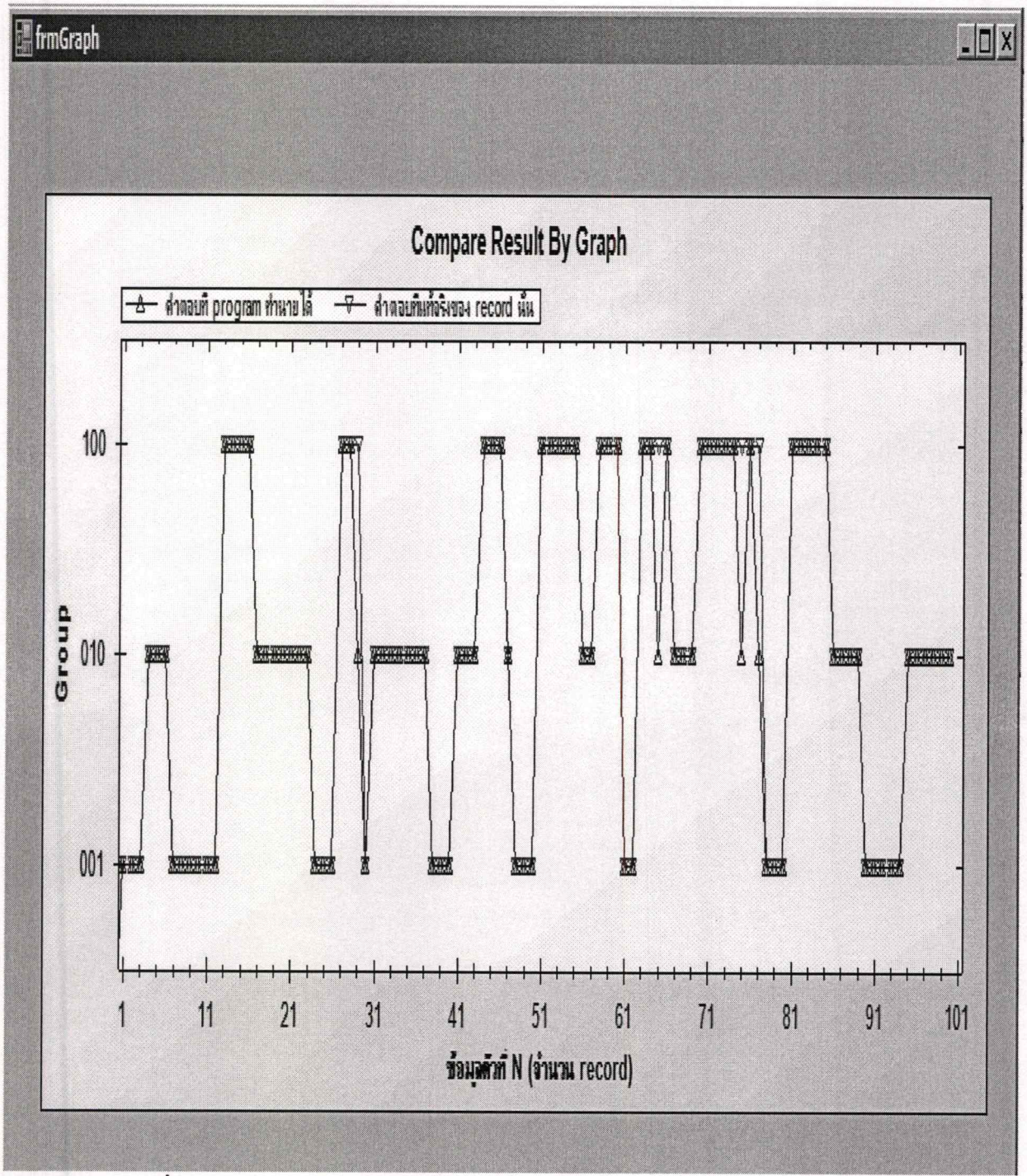
รูปที่ 4.21 แสดง Dialog Box เพื่อเลือก Text File นำมาใช้ในการทดสอบโครงข่าย SFAM

หลังจากทำการกดปุ่ม Test แล้วเลือก File ที่ต้องการนำมา Test เสร็จ ระบบจะแสดงผลลัพธ์ที่ได้ดังรูปที่ 4.22



รูปที่ 4.22 แสดงผลลัพธ์จากการทดสอบโครงข่าย SFAM

ผู้ใช้สามารถแสดงผลลัพธ์จากการทดสอบเป็นรูปแบบกราฟได้ โดยการกดปุ่ม Show Graph Compare result โดยกราฟจะแสดงผลที่โครงข่ายแบ่งได้เทียบกับกลุ่มคำตอบของข้อมูลนั้น ดังรูปที่ 4.23

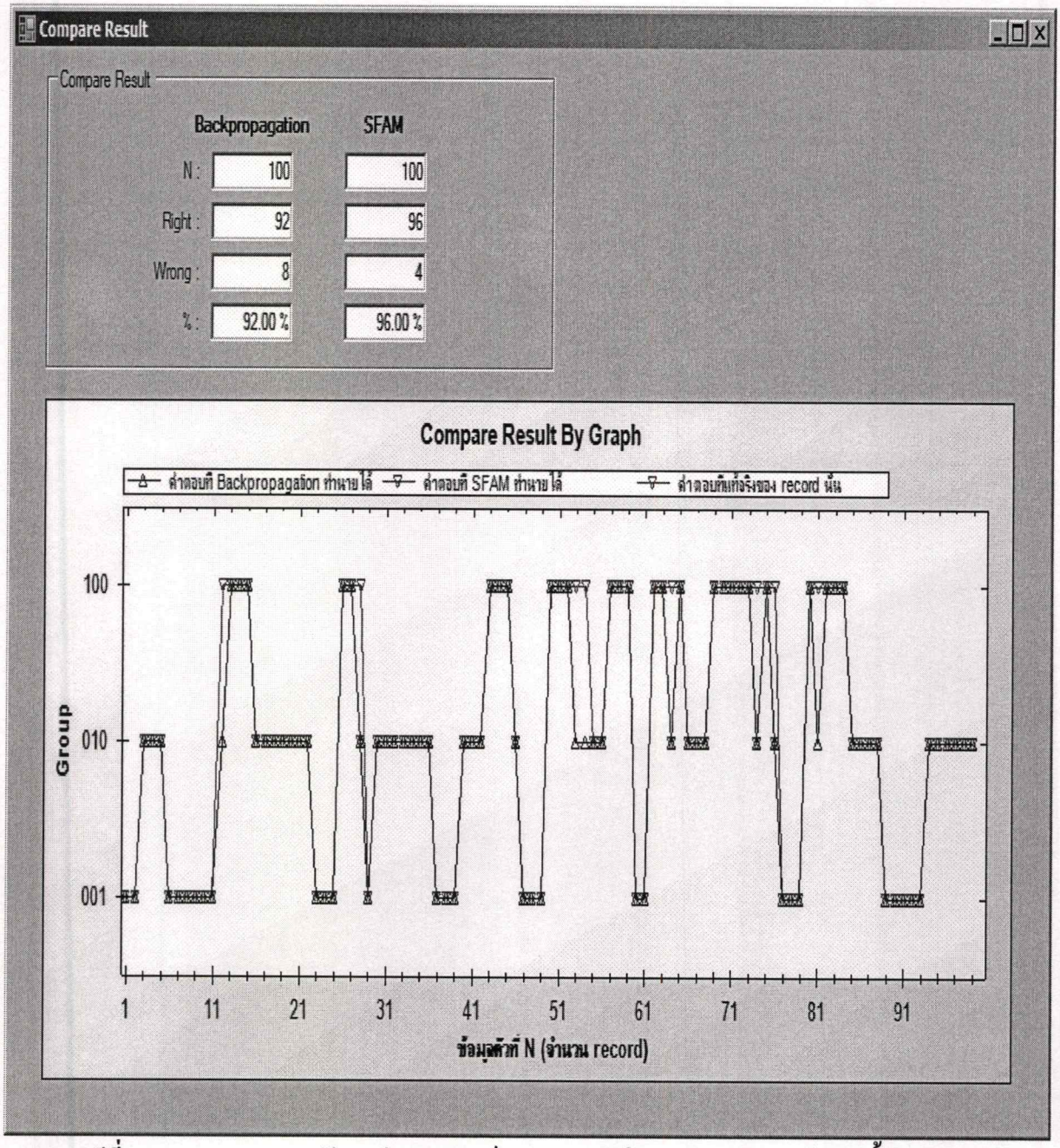


รูปที่ 4.23 แสดงผลลัพธ์จากการทดสอบโครงข่ายSFAMเทียบกับคำตอบ

4.1.3 โปรแกรมเปรียบเทียบผลลัพธ์ Backpropagation และ SFAM

จากหน้าจอเมนูหลักของโปรแกรมผู้ใช้สามารถเลือกที่ปุ่ม Compare Result เพื่อเปรียบเทียบดังนี้

- คำตอบที่แท้จริงของข้อมูลนั้น
- คำตอบที่ได้จากการทดสอบจากโครงข่าย Backpropagation
- คำตอบที่ได้จากการทอสอบจากโครงข่าย SFAM



รูปที่ 4.24 แสดงผลการเปรียบเทียบผลลัพธ์จาก 2 อัลอริทึมและคำตอบของข้อมูลนั้น

4.2 ข้อมูลที่ใช้กับโปรแกรมประยุกต์

ข้อมูลที่นำมาใช้ในการทดสอบกับโปรแกรมประยุกต์ จะถูกจัดเก็บอยู่ในรูปของ Text File

Iris Data

ข้อมูลดอกไม้ Iris มีลักษณะข้อมูลทั้งหมด 4 คุณลักษณะที่แตกต่างกัน คือ

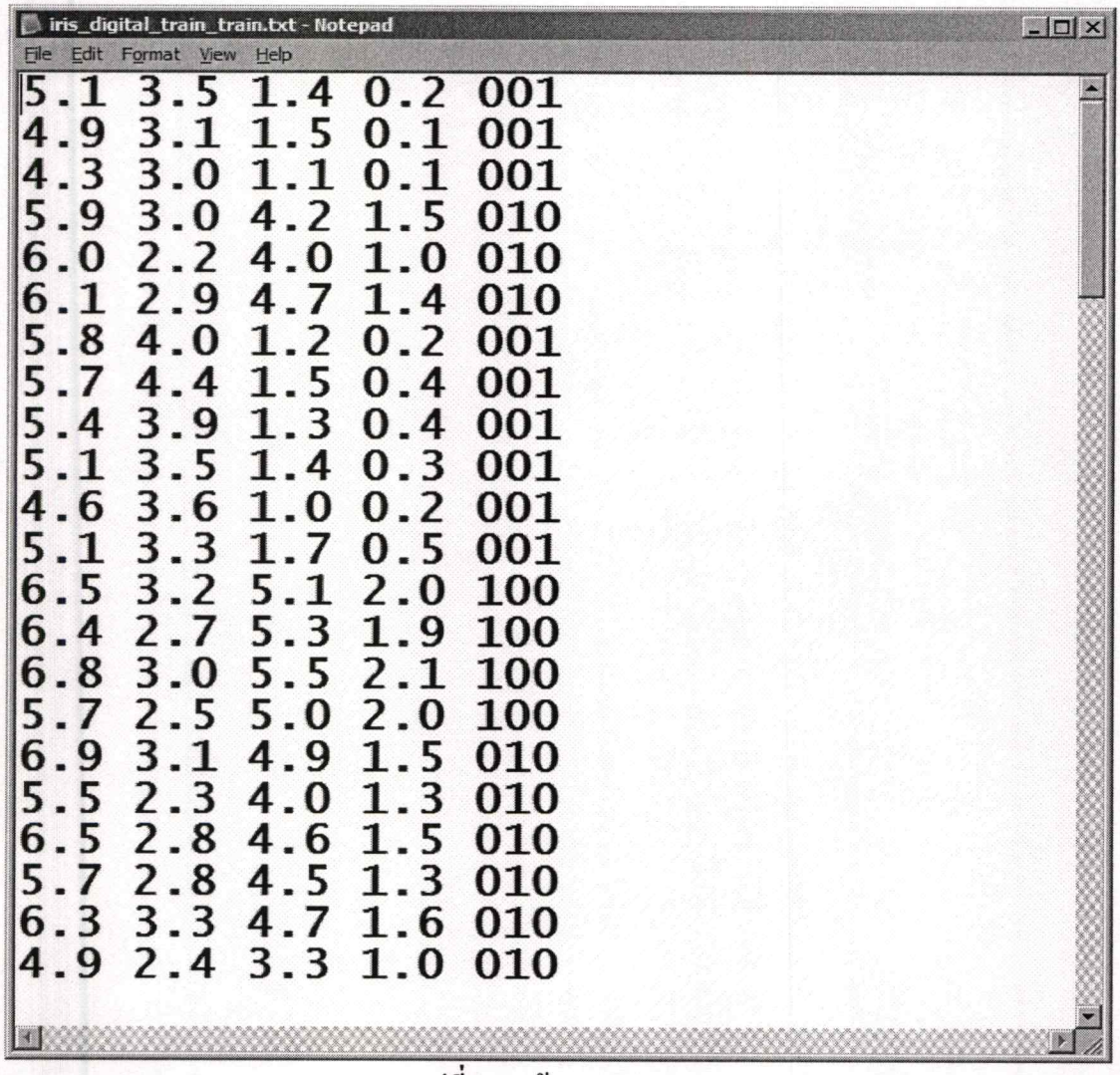
-ความยาวกลีบเลี้ยงดอกไม้

-ค่าความกว้างกลีบเลี้ยงดอกไม้

-ค่าความยาวของกลีบดอกไม้

-ค่าความกว้างของกลีบดอกไม้

โดยชนิดของดอกไม้ Iris จะมีทั้งหมด 3 ชนิด โดยมีจำนวนข้อมูลทั้งหมด 150 ตัวอย่าง โดยแบ่งเป็นชนิดละ 50 ตัวอย่าง โดยจะทำการแบ่งข้อมูลเป็นข้อมูลสำหรับฝึกสอน 100 ตัวอย่าง และข้อมูลสำหรับทดสอบ 50 ตัวอย่าง



รูปที่ 4.25 ข้อมูล Iris Data

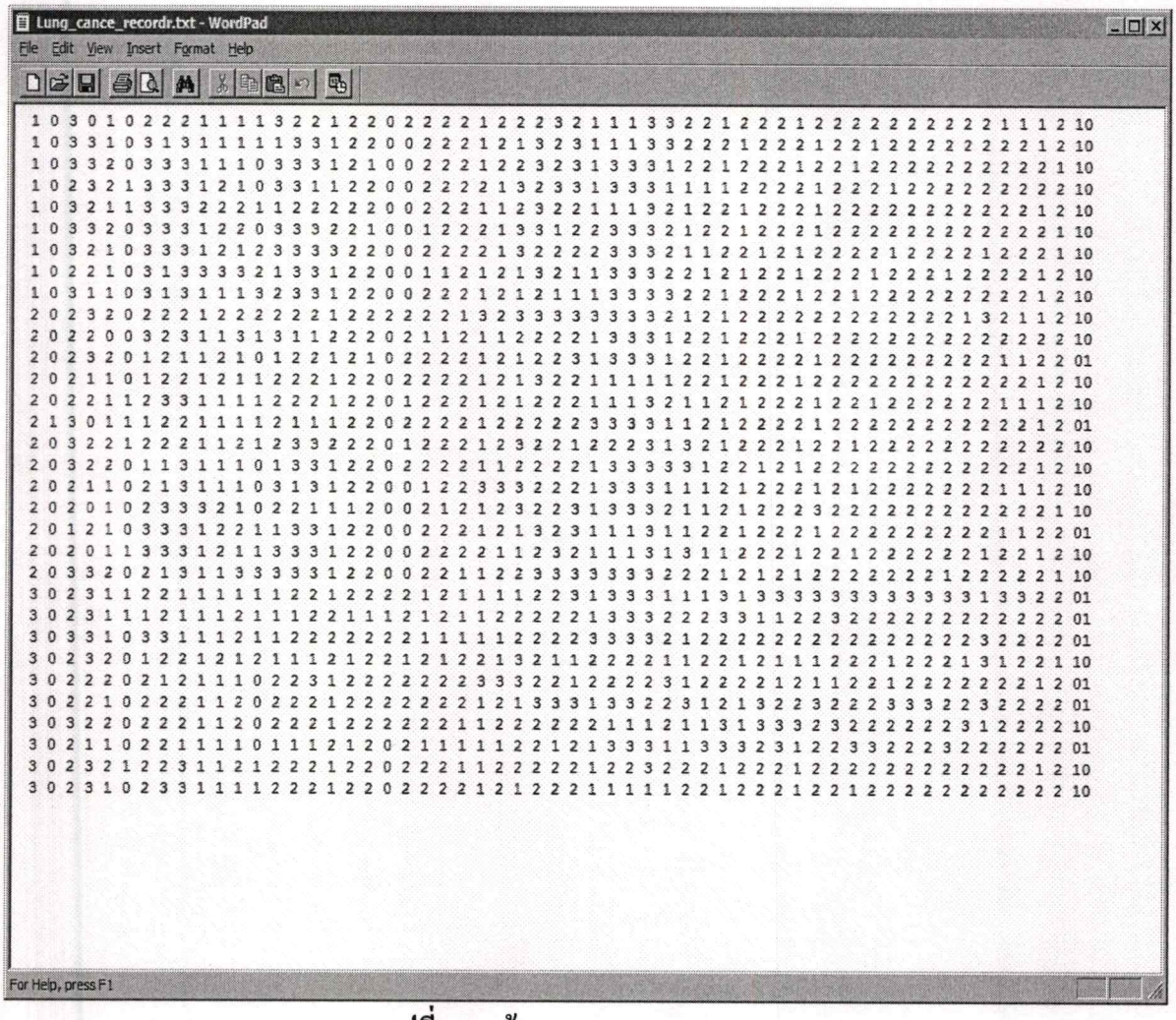
Lung Cancer

ข้อมูลผู้ป่วยมะเร็งปอด จะประกอบด้วยข้อมูลคุณลักษณะ 56 ลักษณะ ซึ่งแต่ละลักษณะจะแตกต่างกัน ซึ่งข้อมูลจะถูกจัดได้เป็น 2 กลุ่ม

กลุ่มที่ 1 จำนวน 9 ตัวอย่าง คือ ผู้ป่วยที่สามารถดำรงชีวิตได้เกิน 5 ปี

กลุ่มที่ 2 จำนวน 23 ตัวอย่าง คือ ผู้ป่วยที่เสียชีวิตภายใน 5 ปีหลังจากเป็นโรคมะเร็ง

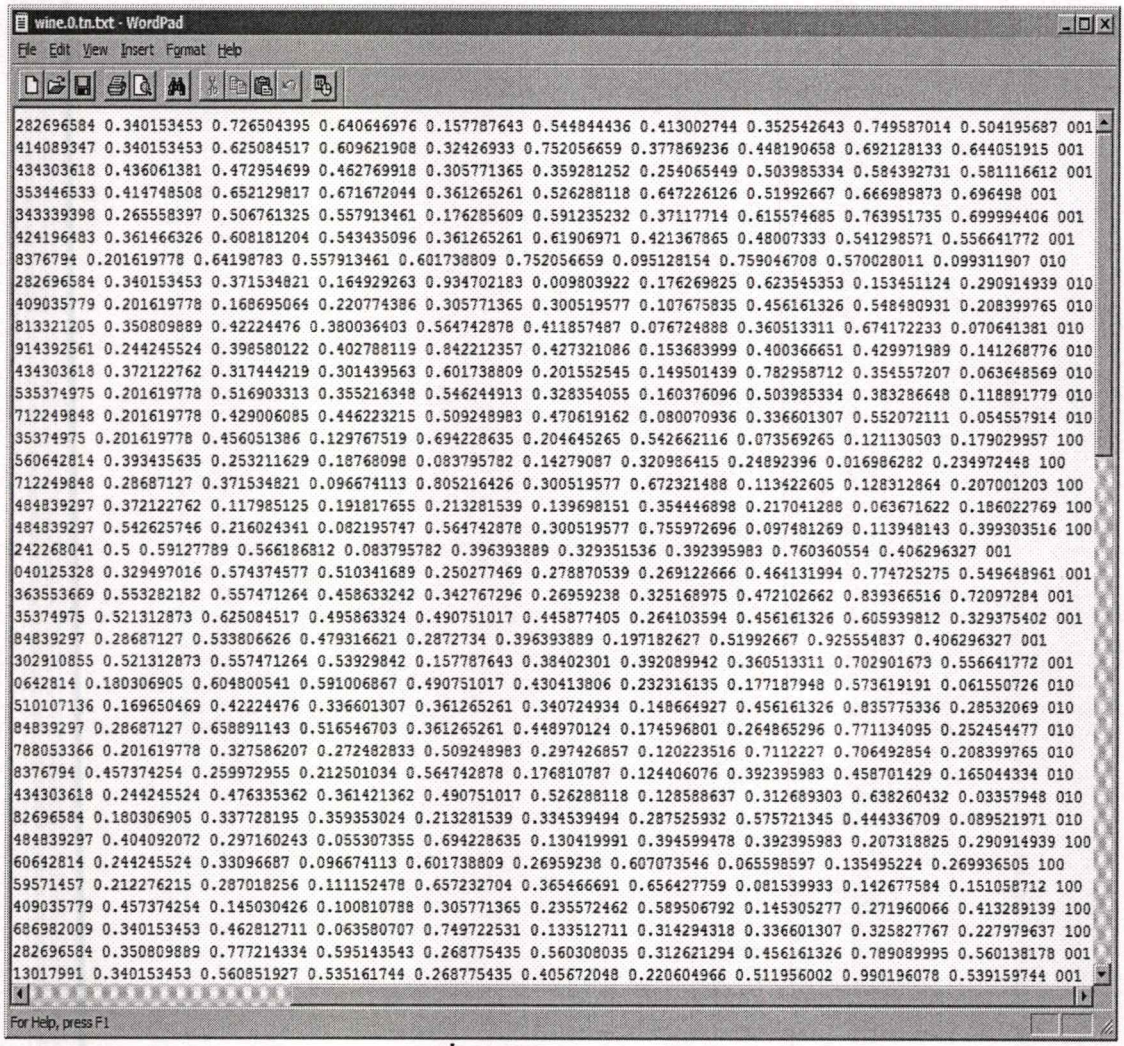
รวมทั้งหมด 32 ตัวอย่าง โดยทำการแบ่งเป็น ข้อมูลฝึกสอนโครงข่าย 24 ตัวอย่าง และข้อมูลสำหรับทดสอบโครงข่าย 8 ตัวอย่าง



รูปที่ 4.26 ข้อมูล Lung Cancer

Wine Data

Wine Data ประกอบด้วยลักษณะ 13 อย่าง โดยแบ่งข้อมูลออกเป็น 3 กลุ่ม คือ กลุ่มที่ 1 จำนวน 59 ตัวอย่าง กลุ่มที่ 2 จำนวน 71 ตัวอย่าง กลุ่มที่ 3 จำนวน 48 ตัวอย่าง รวมทั้งหมด 178 ตัวอย่าง โดยจะแบ่งเป็นข้อมูลสำหรับฝึกสอนจำนวน 100 ตัวอย่าง และข้อมูลสำหรับทดสอบ จำนวน 78 ตัวอย่าง


wine.0.tn.txt - WordPad

File Edit View Insert Format Help

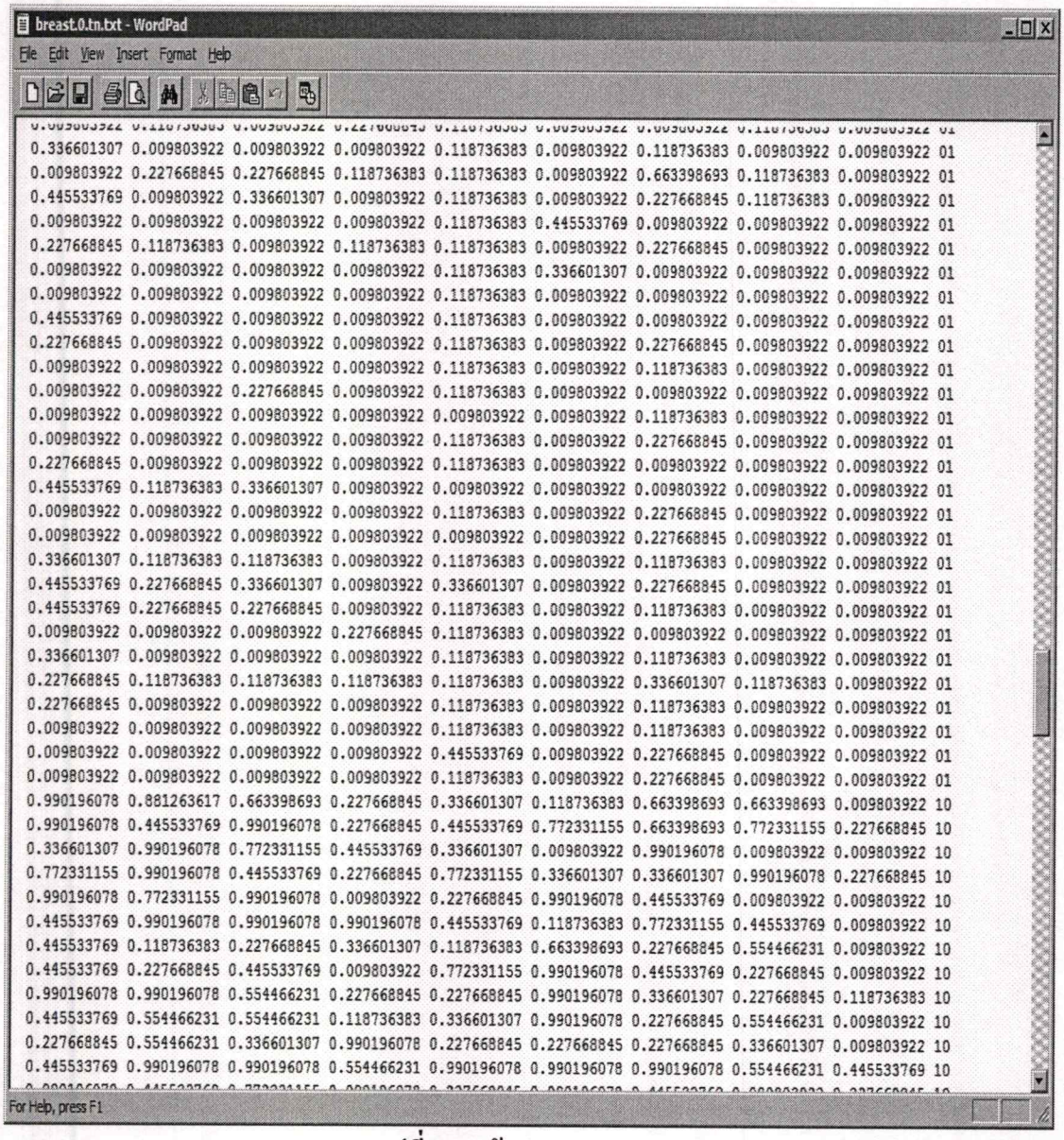
282696584 0.340153453 0.726504395 0.640646976 0.157787643 0.544844436 0.413002744 0.352542643 0.749587014 0.504195687 001
414089347 0.340153453 0.625084517 0.609621908 0.32426933 0.752056659 0.377869236 0.448190658 0.692128133 0.644051915 001
434303618 0.436061381 0.472954699 0.462769918 0.305771365 0.359281252 0.254065449 0.503985334 0.584392731 0.581116612 001
353446533 0.414748508 0.652129817 0.671672044 0.361265261 0.526288118 0.647226126 0.51992667 0.666989873 0.696498 001
343339398 0.265558397 0.506761325 0.557913461 0.176285609 0.591235232 0.37117714 0.615574685 0.763951735 0.699994406 001
424196483 0.361466326 0.608181204 0.543435096 0.361265261 0.61906971 0.421367865 0.48007333 0.541298571 0.556641772 001
8376794 0.201619778 0.64198783 0.557913461 0.601738809 0.752056659 0.095128154 0.759046708 0.570028011 0.099311907 010
282696584 0.340153453 0.371534821 0.164929263 0.934702183 0.009803922 0.176269825 0.623545353 0.153451124 0.290914939 010
409035779 0.201619778 0.168695064 0.220774386 0.305771365 0.300519577 0.107675835 0.456161326 0.548480931 0.208399765 010
813321205 0.350809889 0.42224476 0.380036403 0.564742878 0.411857487 0.076724888 0.360513311 0.674172233 0.070641381 010
914392561 0.244245524 0.398580122 0.402788119 0.842212357 0.427321086 0.153683999 0.400366651 0.429971989 0.141268776 010
434303618 0.372122762 0.317444219 0.301439563 0.601738809 0.201552545 0.149501439 0.782958712 0.354557207 0.063648569 010
535374975 0.201619778 0.516903313 0.355216348 0.546244913 0.328354055 0.160376096 0.503985334 0.383286648 0.118891779 010
712249848 0.201619778 0.429006085 0.446223215 0.509248983 0.470619162 0.080070936 0.336601307 0.552072111 0.054557914 010
35374975 0.201619778 0.456051386 0.129767519 0.694228635 0.204645265 0.542662116 0.073569265 0.121130503 0.179029957 100
560642814 0.393435635 0.253211629 0.18768098 0.083795782 0.14279087 0.320986415 0.24892396 0.016986282 0.234972448 100
712249848 0.28687127 0.371534821 0.096674113 0.805216426 0.300519577 0.672321488 0.113422605 0.128312864 0.207001203 100
484839297 0.372122762 0.117985125 0.191817655 0.213281539 0.139698151 0.354446898 0.217041288 0.063671622 0.186022769 100
484839297 0.542625746 0.216024341 0.082195747 0.564742878 0.300519577 0.755972696 0.097481269 0.113948143 0.399303516 100
242268041 0.5 0.59127789 0.566186812 0.083795782 0.396393889 0.329351536 0.392395983 0.760360554 0.406296327 001
040125328 0.329497016 0.574374577 0.510341689 0.250277469 0.278870539 0.269122666 0.464131994 0.774725275 0.549648961 001
363553669 0.553282182 0.557471264 0.458633242 0.342767296 0.26959238 0.325168975 0.472102662 0.839366516 0.72097284 001
35374975 0.521312873 0.625084517 0.495863324 0.490751017 0.445877405 0.264103594 0.456161326 0.605939812 0.329375402 001
84839297 0.28687127 0.533806626 0.479316621 0.2872734 0.396393889 0.197182627 0.51992667 0.925554837 0.406296327 001
302910855 0.521312873 0.557471264 0.53929842 0.157787643 0.38402301 0.392089942 0.360513311 0.702901673 0.556641772 001
0642814 0.180306905 0.604800541 0.591006867 0.490751017 0.430413806 0.232316135 0.177187948 0.573619191 0.061550726 010
510107136 0.169650469 0.42224476 0.336601307 0.361265261 0.340724934 0.148664927 0.456161326 0.835775336 0.28532069 010
84839297 0.28687127 0.658891143 0.516546703 0.361265261 0.448970124 0.174596801 0.264865296 0.771134095 0.252454477 010
788053366 0.201619778 0.327586207 0.272482833 0.509248983 0.297426857 0.120223516 0.7112227 0.706492854 0.208399765 010
8376794 0.457374254 0.259972955 0.212501034 0.564742878 0.176810787 0.124406076 0.392395983 0.458701429 0.165044334 010
434303618 0.244245524 0.476335362 0.361421362 0.490751017 0.526288118 0.128588637 0.312689303 0.638260432 0.03357948 010
82696584 0.180306905 0.337728195 0.359353024 0.213281539 0.334539494 0.287525932 0.575721345 0.444336709 0.089521971 010
484839297 0.404092072 0.297160243 0.055307355 0.694228635 0.130419991 0.394599478 0.392395983 0.207318825 0.290914939 100
60642814 0.244245524 0.33096687 0.096674113 0.601738809 0.26959238 0.607073546 0.065598597 0.135495224 0.269936505 100
59571457 0.212276215 0.287018256 0.111152478 0.657232704 0.365466691 0.656427759 0.081539933 0.142677584 0.151058712 100
409035779 0.457374254 0.145030426 0.100810788 0.305771365 0.235572462 0.589506792 0.145305277 0.271960066 0.413289139 100
686982009 0.340153453 0.462812711 0.063580707 0.749722531 0.133512711 0.314294318 0.336601307 0.325827767 0.227979637 100
282696584 0.350809889 0.777214334 0.595143543 0.268775435 0.560308035 0.312621294 0.456161326 0.789089995 0.560138178 001
13017991 0.340153453 0.560851927 0.535161744 0.268775435 0.405672048 0.220604966 0.511956002 0.990196078 0.539159744 001

For Help, press F1

รูปที่ 4.27 ข้อมูล Wine Data

Breast Data

Breast Data ประกอบด้วยข้อมูล 9 ลักษณะ และแบ่งเป็น 2 กลุ่มข้อมูล โดยทำการแบ่งเป็นข้อมูลสำหรับฝึกสอนโครงข่ายจำนวน 420 ตัวอย่าง และข้อมูลสำหรับทดสอบโครงข่ายจำนวน 279 ตัวอย่าง



รูปที่ 4.28 ข้อมูล Breast Data

4.3 ผลการทดลองและวิเคราะห์

4.3.1 Iris Data

ทำการเปรียบเทียบความถูกต้องข้อมูล Iris จำนวน 100 ตัวอย่าง ระหว่าง Backpropagation และ SFAM

โดยกำหนด

-จำนวนตัวอย่างฝึกสอนโครงข่าย 100 ตัวอย่าง (Data Train)

-จำนวนตัวอย่างที่ใช้ทดสอบโครงข่าย 100 ตัวอย่าง (Data Train)

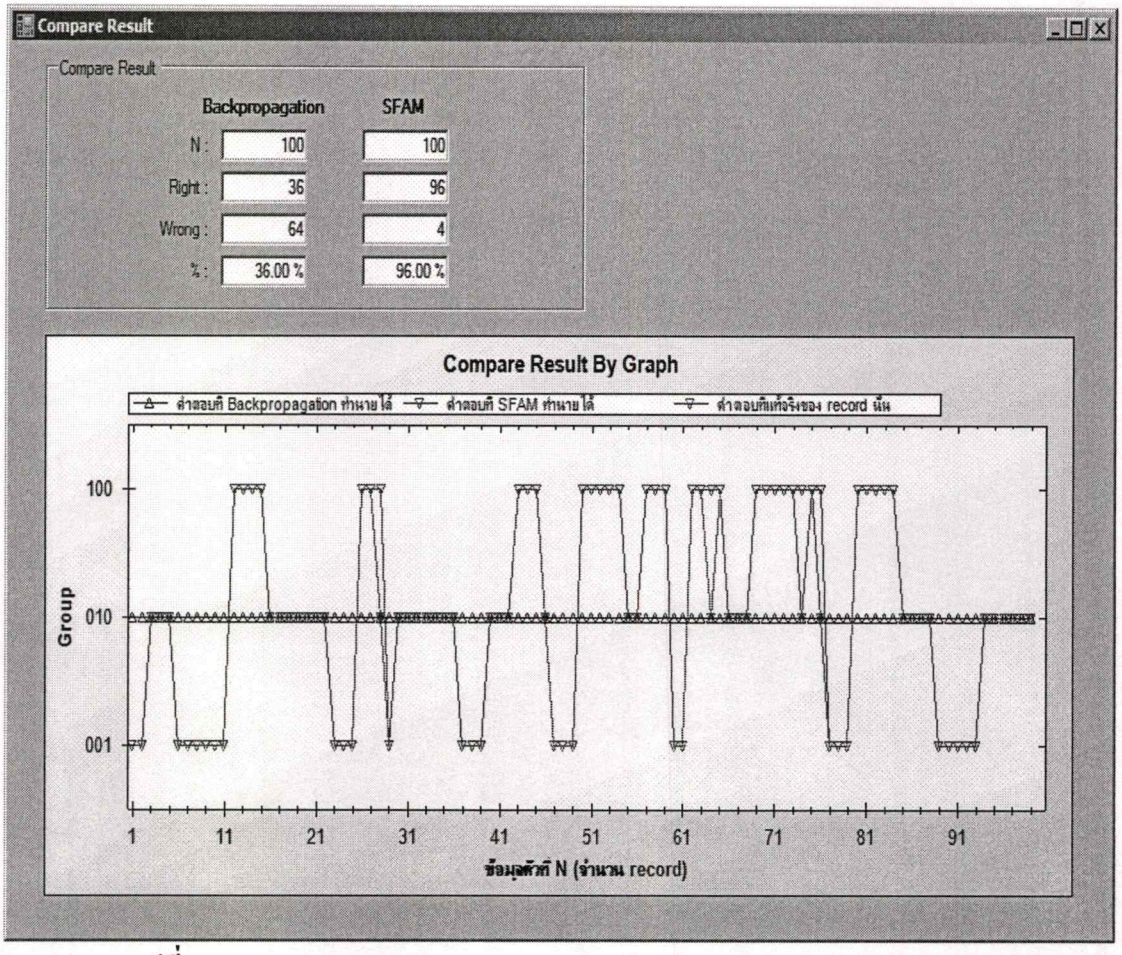
-Backpropagation ทำการวนที่ 1รอบ,10รอบ,100 รอบ,1000รอบ

-SFAM กำหนดค่า Vigilance (ρ) = 0.4

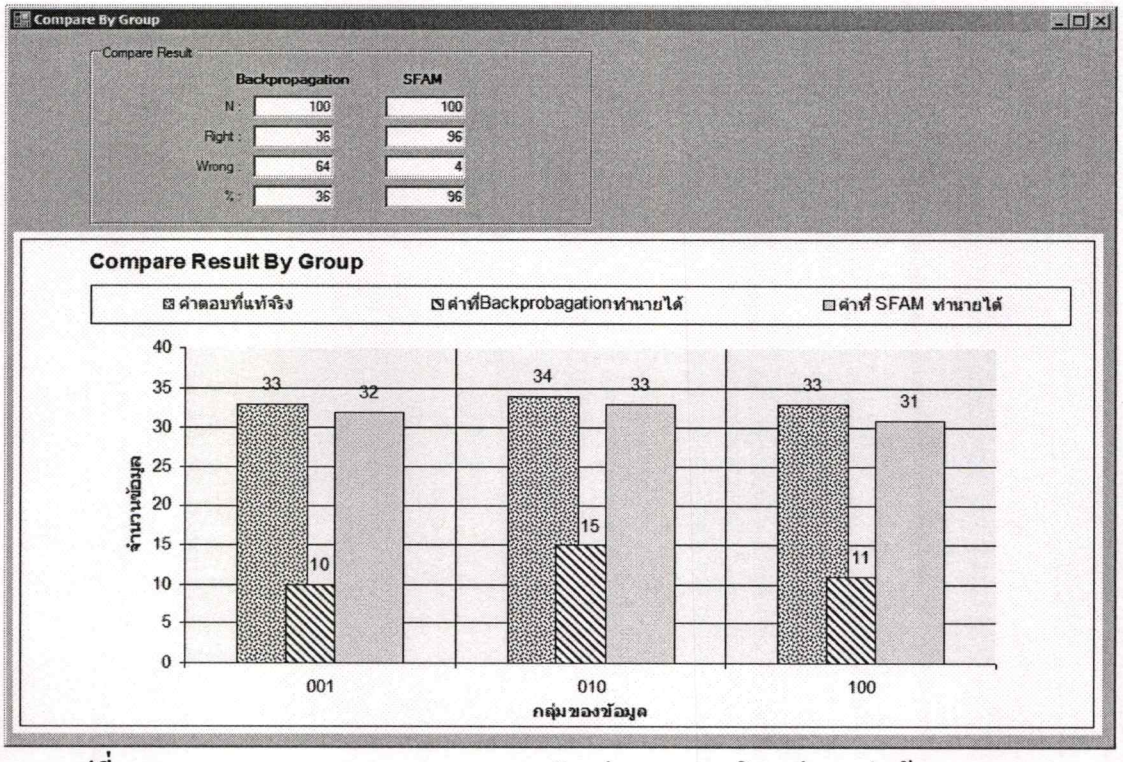
ผลที่ได้จากการจำแนกข้อมูล Iris คือ

ตารางที่ 4.1 แสดงผลลัพธ์การทดลอง Iris ด้วยตัวอย่าง (Data Train) 100 ตัว

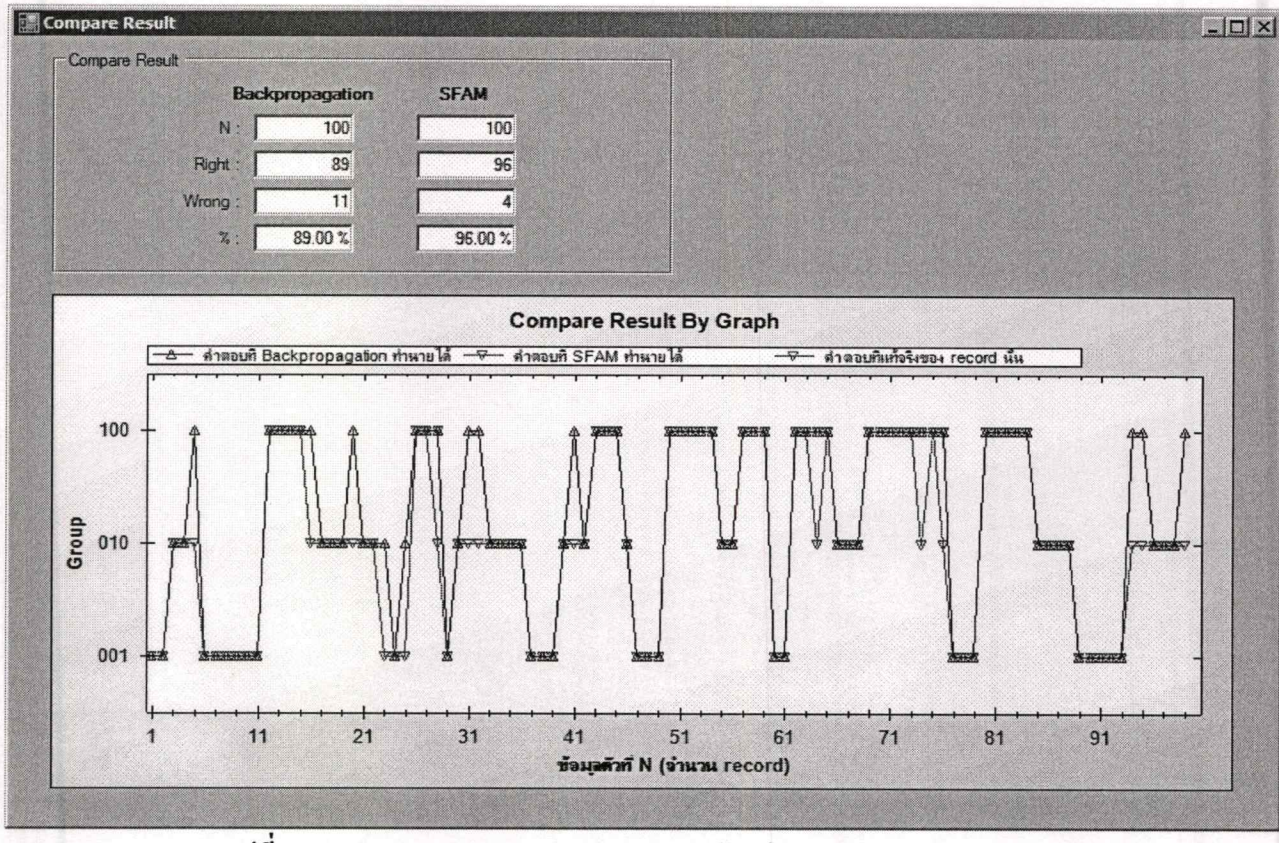
อัลกอริทึม	Backpropagation (1รอบ)	Backpropagation (10รอบ)	Backpropagation (100รอบ)	Backpropagation (1000รอบ)	SFAM
จำนวนตัวอย่าง	100	100	100	100	100
จำแนกถูก	36	89	92	96	96
จำแนกผิด	64	11	8	4	4
เปอร์เซ็นความถูกต้อง	36%	89%	92%	96%	96%



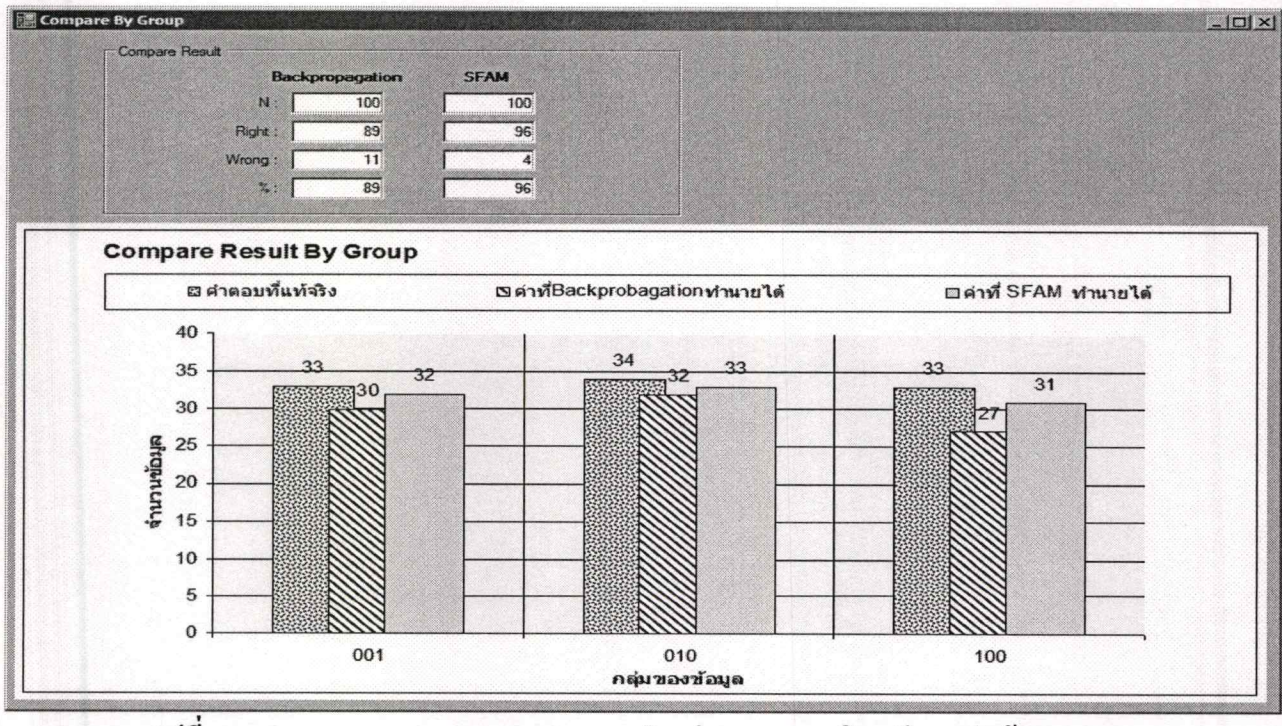
รูปที่ 4.29 ผลการทดลองของ Iris Data 100 ตัวอย่าง (1รอบ)



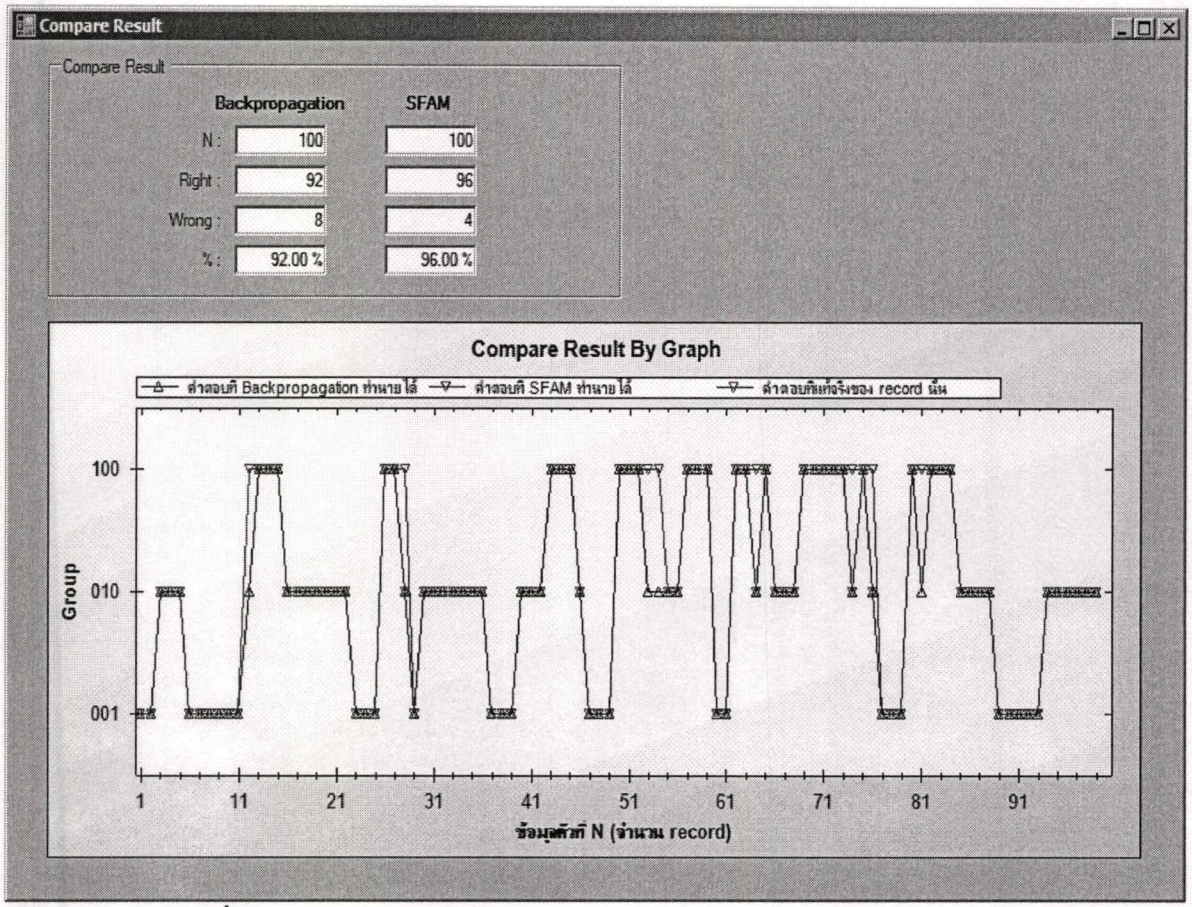
รูปที่ 4.30 ผลการทดลองของ Iris Data 100 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล



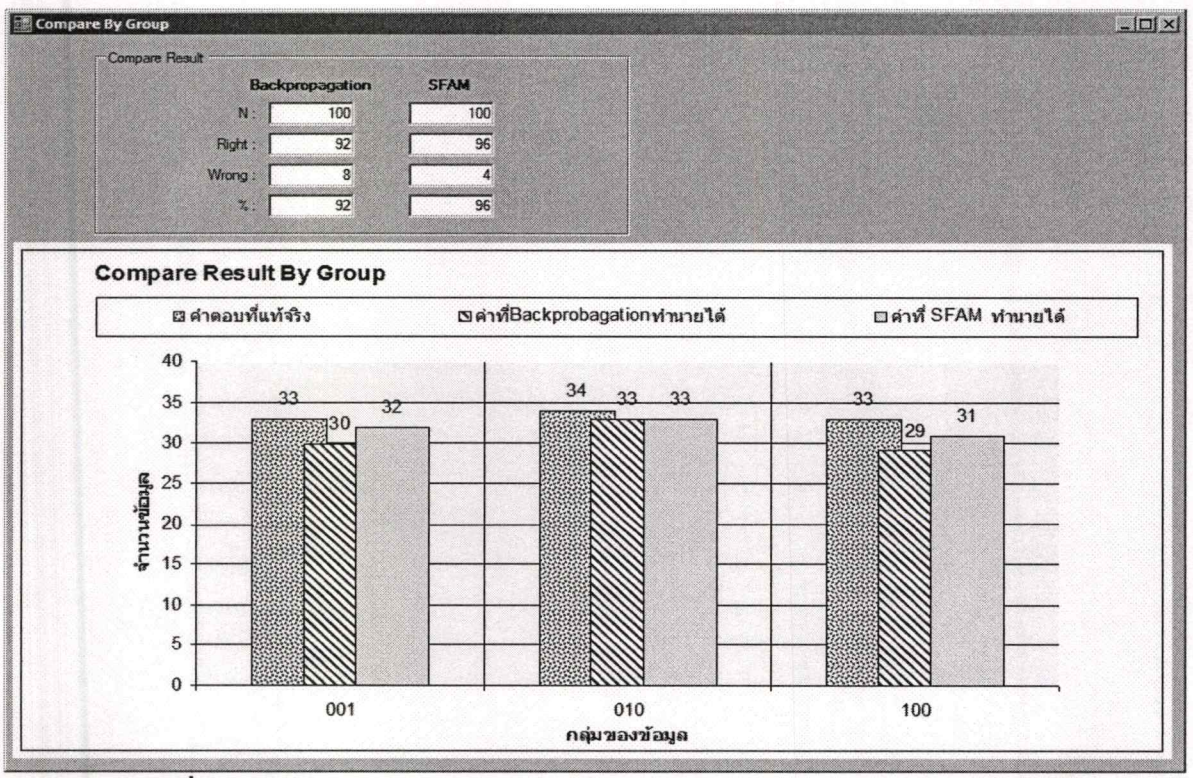
รูปที่ 4.31 ผลการทดลองของ Iris Data 100 ตัวอย่าง(10รอบ)



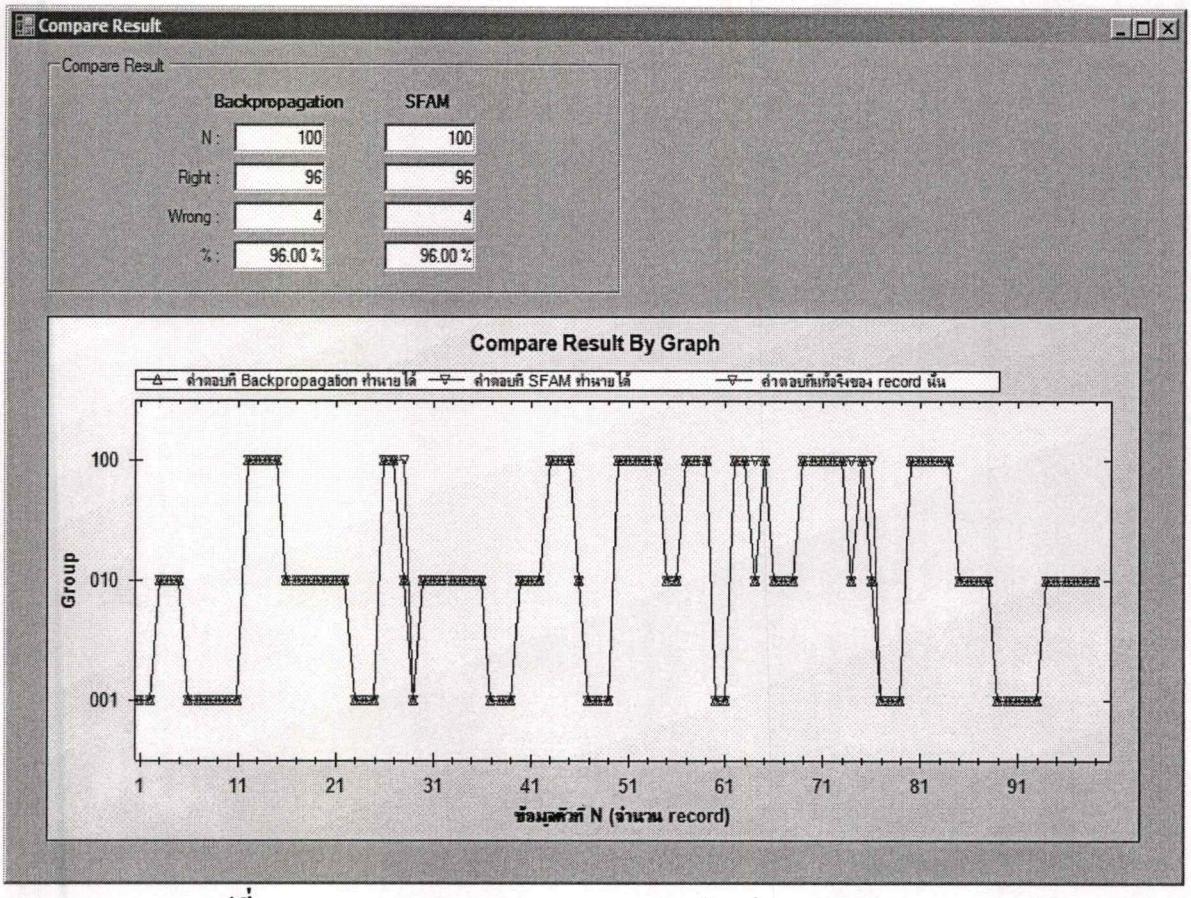
รูปที่ 4.32 ผลการทดลองของ Iris Data 100 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล



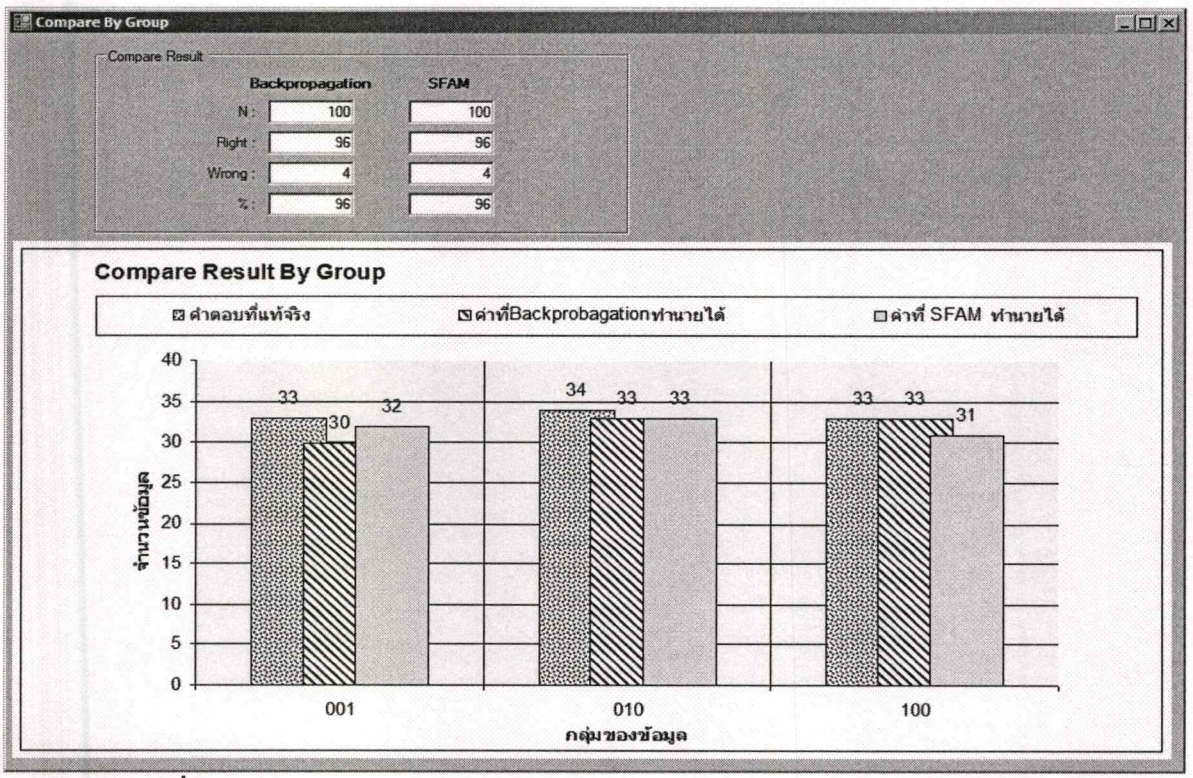
รูปที่ 4.33 ผลการทดลองของ Iris Data Train 100 ตัวอย่าง(100รอบ)



รูปที่ 4.34 ผลการทดลองของ Iris Data 100 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล



รูปที่ 4.35 ผลการทดลองของ Iris Data 100 ตัวอย่าง (1000รอบ)



รูปที่ 4.36 ผลการทดลองของ Iris Data 100 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล

ต่อมานำข้อมูลทดสอบโครงข่าย(Test Data)จำนวน 50 ตัวอย่างมาทดสอบ โดยใช้จำนวนข้อมูลในการฝึกสอนโครงข่าย 100 ตัวอย่าง เมื่อนำมาฝึกสอนโครงข่ายแล้ว นำมาทดสอบ

โดยกำหนด

-จำนวนตัวอย่างฝึกสอนโครงข่าย 100 ตัวอย่าง (Data Train)

-จำนวนตัวอย่างที่ใช้ทดสอบโครงข่าย 50 ตัวอย่าง (Data Test)

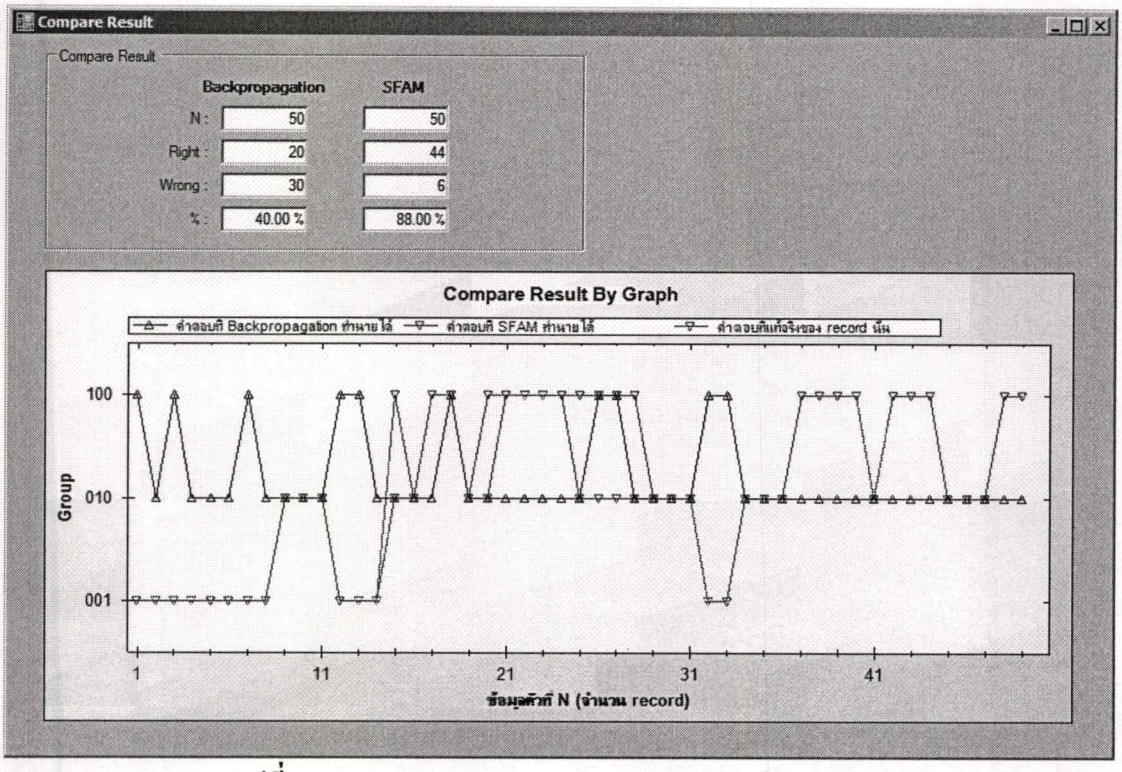
-Backpropagation ทำการวนที่ 1รอบ, 10รอบ, 100 รอบ, 1000รอบ

-SFAM กำหนดค่า Vigilance (ρ) = 0.4

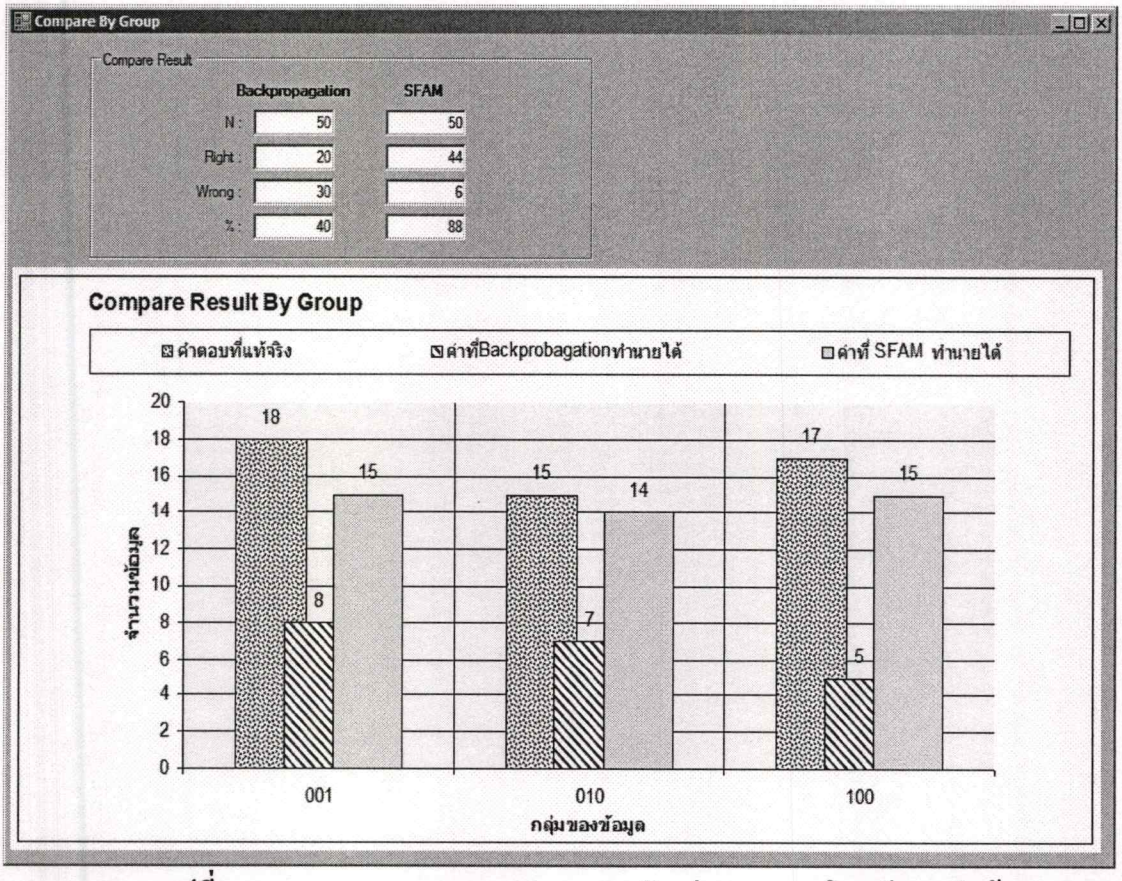
ได้ผลการทดลองดังนี้

ตารางที่ 4.2 แสดงผลลัพธ์การทดลอง Iris ด้วยตัวอย่าง (Data Test) 50 ตัวอย่าง

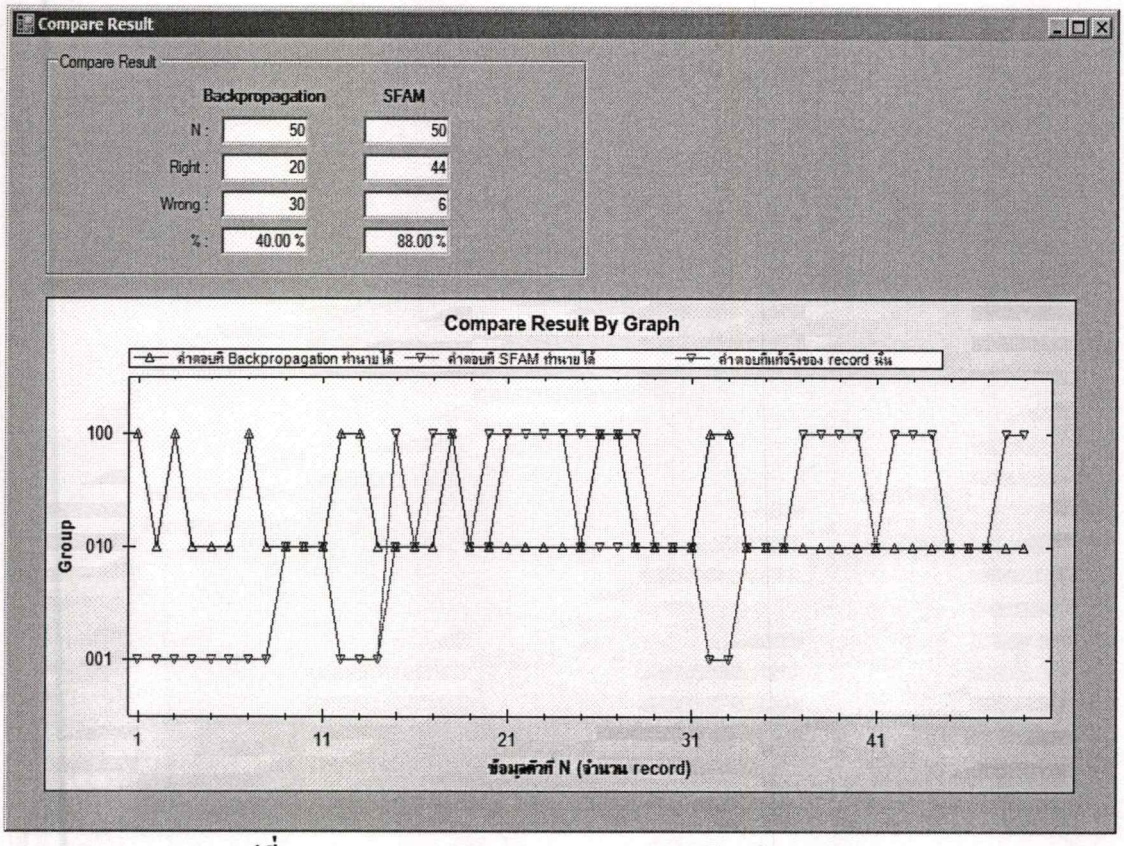
อัลกอริทึม	Backpropagation (1รอบ)	Backpropagation (10รอบ)	Backpropagation (100รอบ)	Backpropagation (100รอบ)	SFAM
จำนวนตัวอย่าง	50	50	50	50	50
จำนวนถูก	20	53	47	48	44
จำแนกผิด	30	7	3	2	6
เปอร์เซ็นความถูกต้อง	40 %	88.33%	94%	96%	88 %



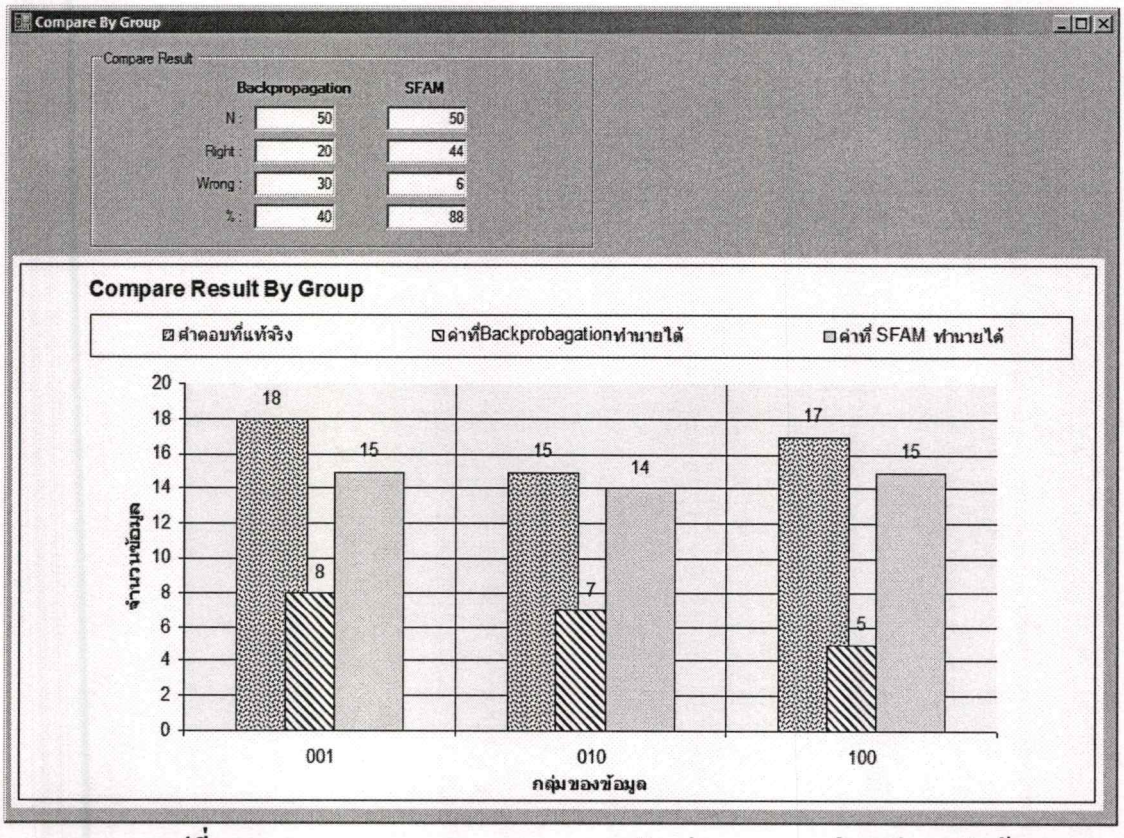
รูปที่ 4.37 ผลการทดลองของ Iris Data 50 ตัวอย่าง (1รอบ)



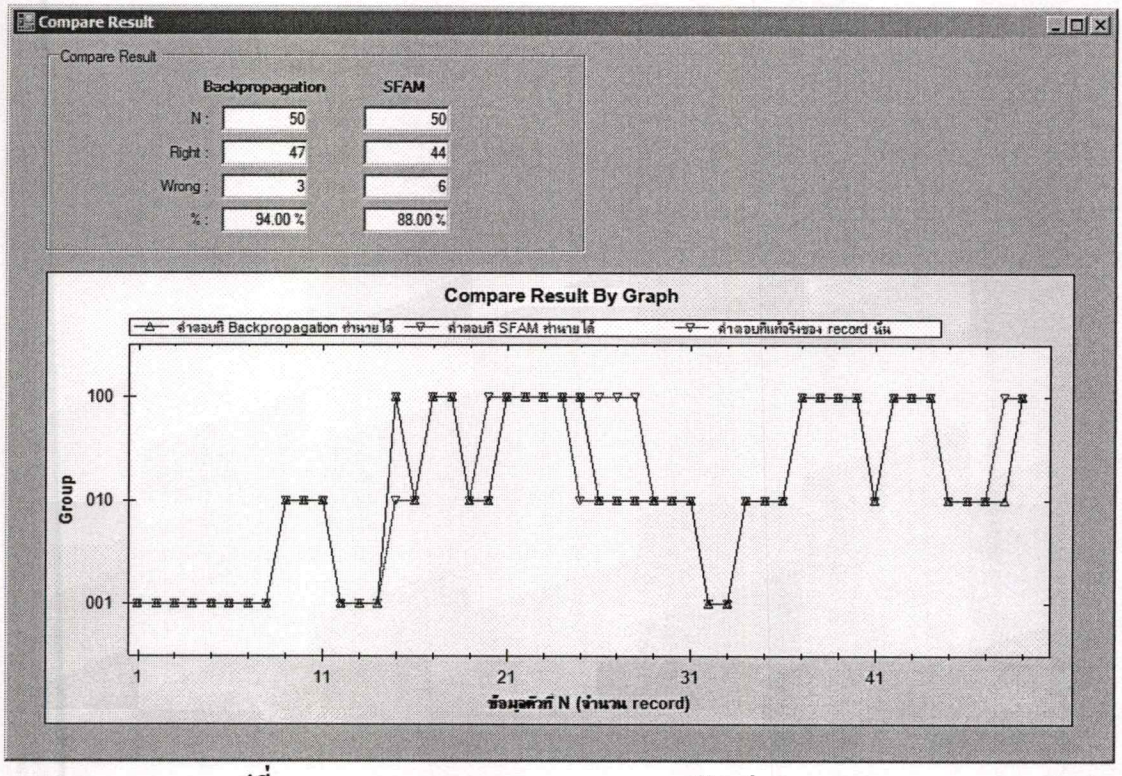
รูปที่ 4.38 ผลการทดลองของ Iris Data 50 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล



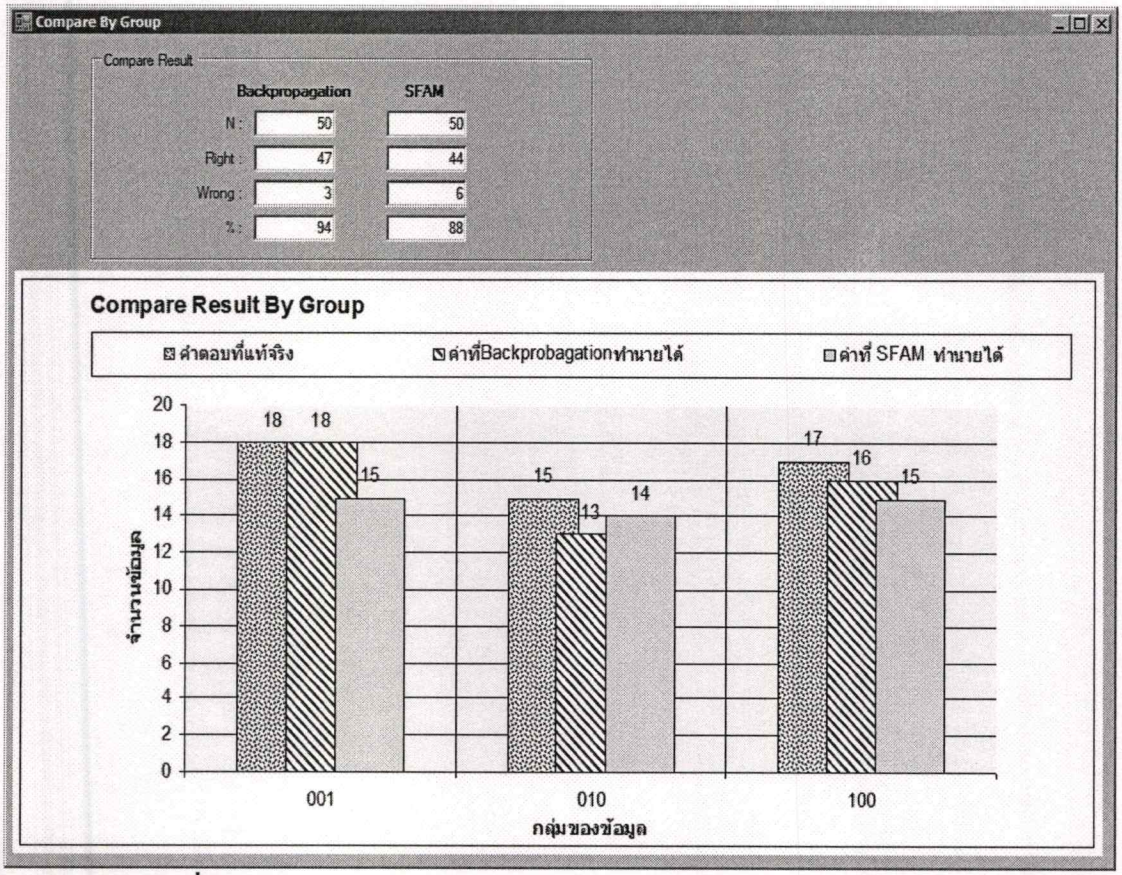
รูปที่ 4.39 ผลการทดลองของ Iris Data 50 ตัวอย่าง (10รอบ)



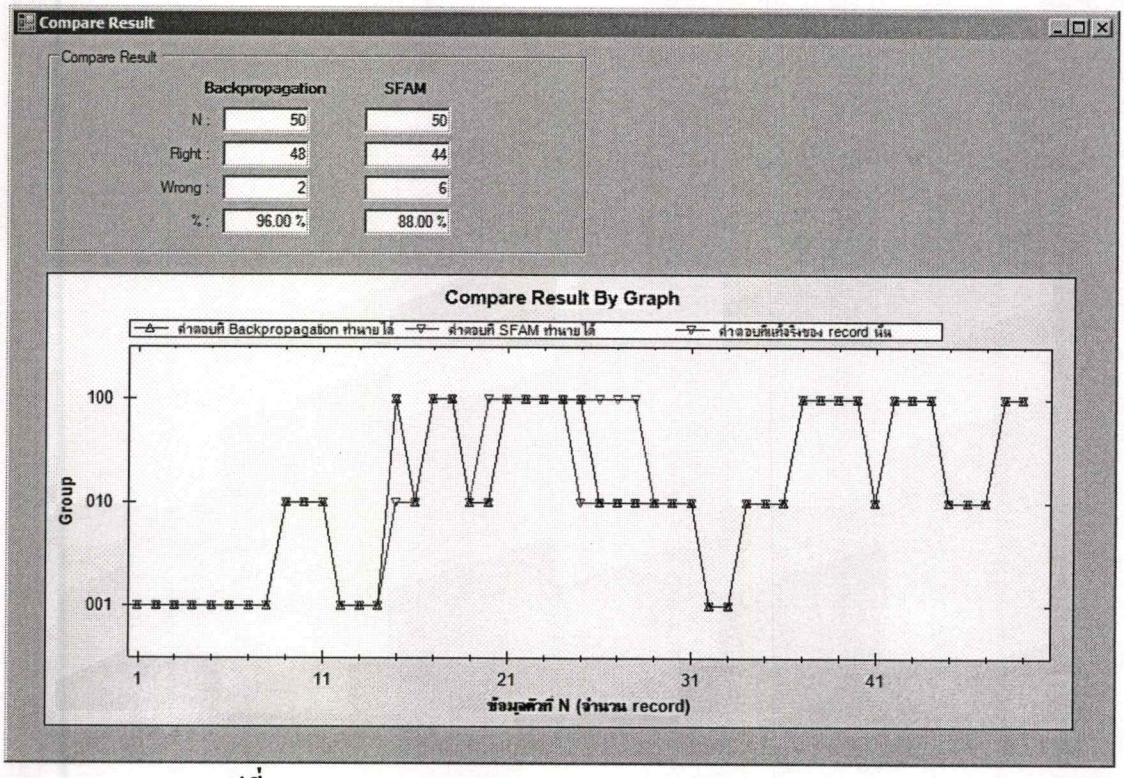
รูปที่ 4.40 ผลการทดลองของ Iris Data 50 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล



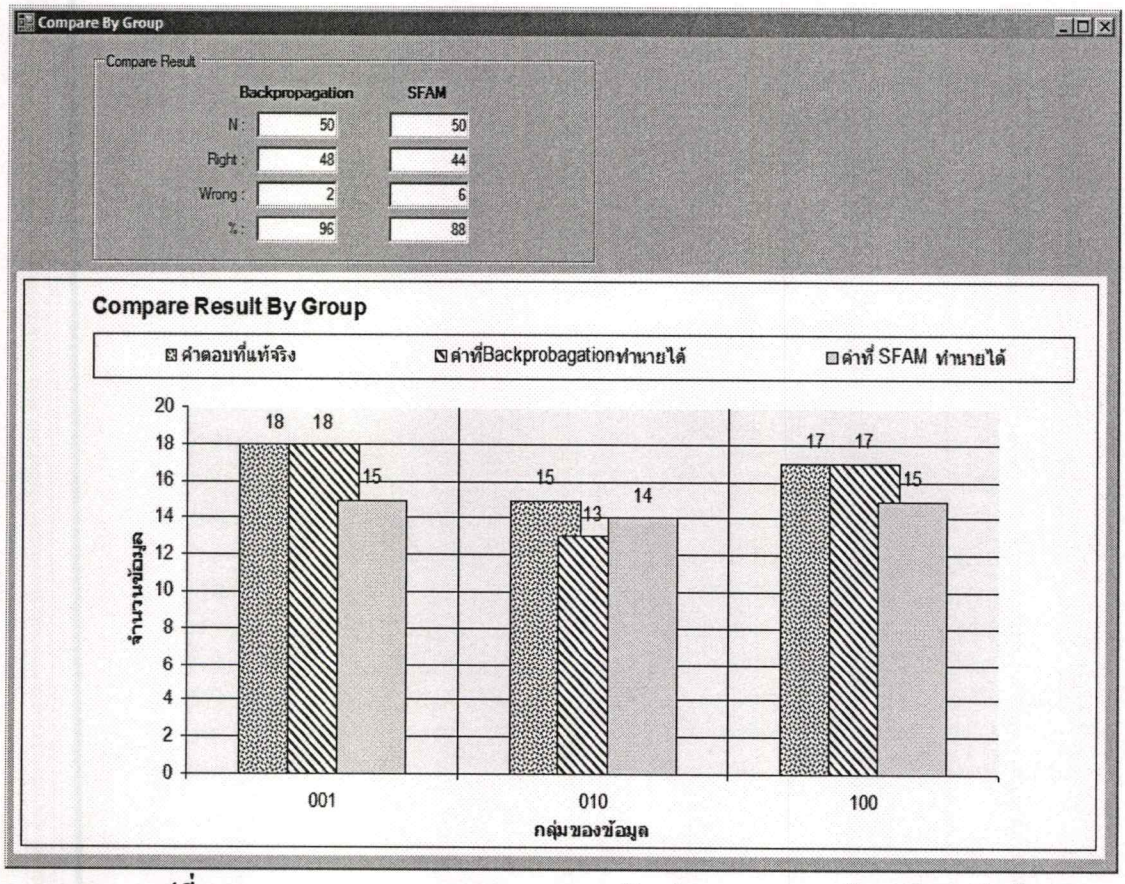
รูปที่ 4.41 ผลการทดลองของ Iris Data 50 ตัวอย่าง (100รอบ)



รูปที่ 4.42 ผลการทดลองของ Iris Data 50 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล



รูปที่ 4.43 ผลการทดลองของ Iris Data 50 ตัวอย่าง (1000รอบ)



รูปที่ 4.44 ผลการทดลองของ Iris Data 50 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล

4.3.2 Lung Cancer Data

ทำการเปรียบเทียบความถูกต้องข้อมูล Lung Cancer จำนวน 24 ตัวอย่าง ระหว่าง Backpropagation และ SFAM

โดยกำหนด

-จำนวนตัวอย่างฝึกสอนโครงข่าย 24 ตัวอย่าง (Data Train)

-จำนวนตัวอย่างที่ใช้ทดสอบโครงข่าย 24 ตัวอย่าง (Data Train)

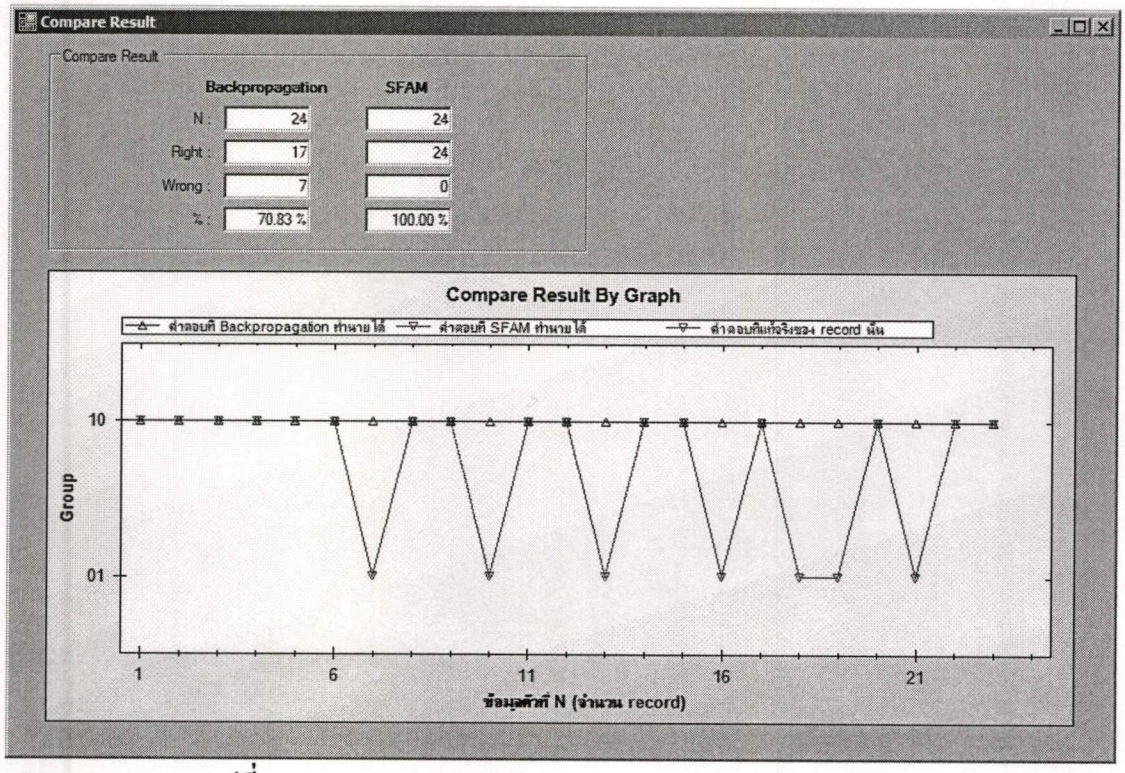
-Backpropagation ทำการวนที่ 1รอบ,10รอบ,100 รอบ,1000รอบ

-SFAM กำหนดค่า Vigilance (ρ) = 0.4

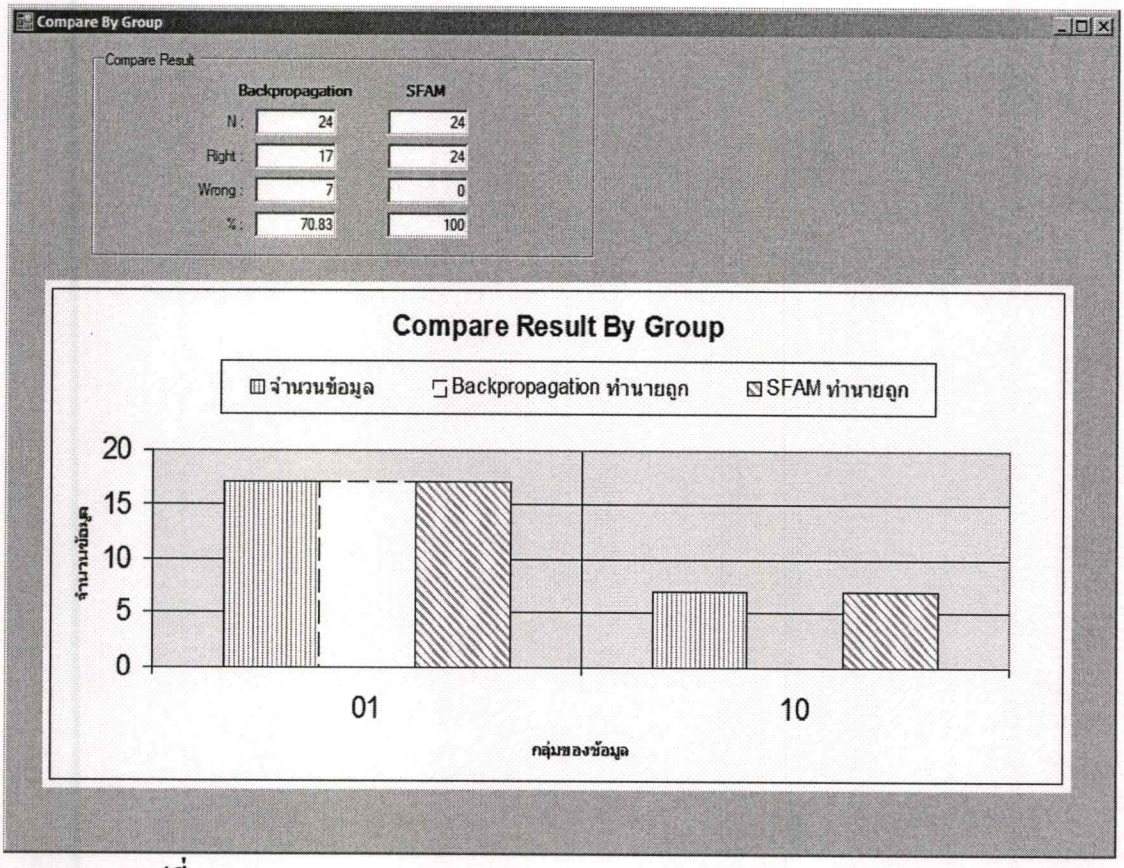
ได้ผลการทดลองดังนี้

ตารางที่ 4.3 แสดงผลลัพธ์การทดลอง Lung Cancer ด้วยจำนวน 24 ตัวอย่าง

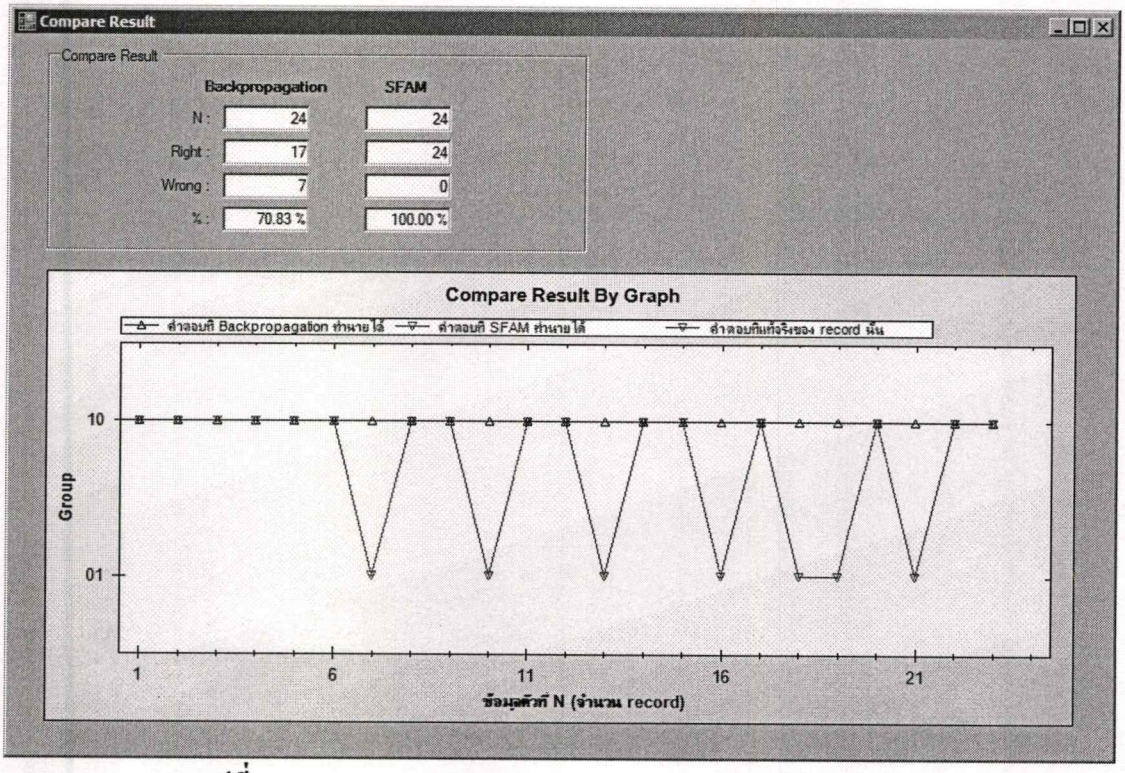
อัลกอริทึม	Backpropagation (1รอบ)	Backpropagation (10รอบ)	Backpropagation (100รอบ)	Backpropagation (1000รอบ)	SFAM
จำนวนตัวอย่าง	24	24	24	24	24
จำนวนถูก	17	17	24	24	24
จำแนกผิด	7	7	0	0	0
เปอร์เซ็นความถูกต้อง	70.88%	70.88%	100%	100%	100%



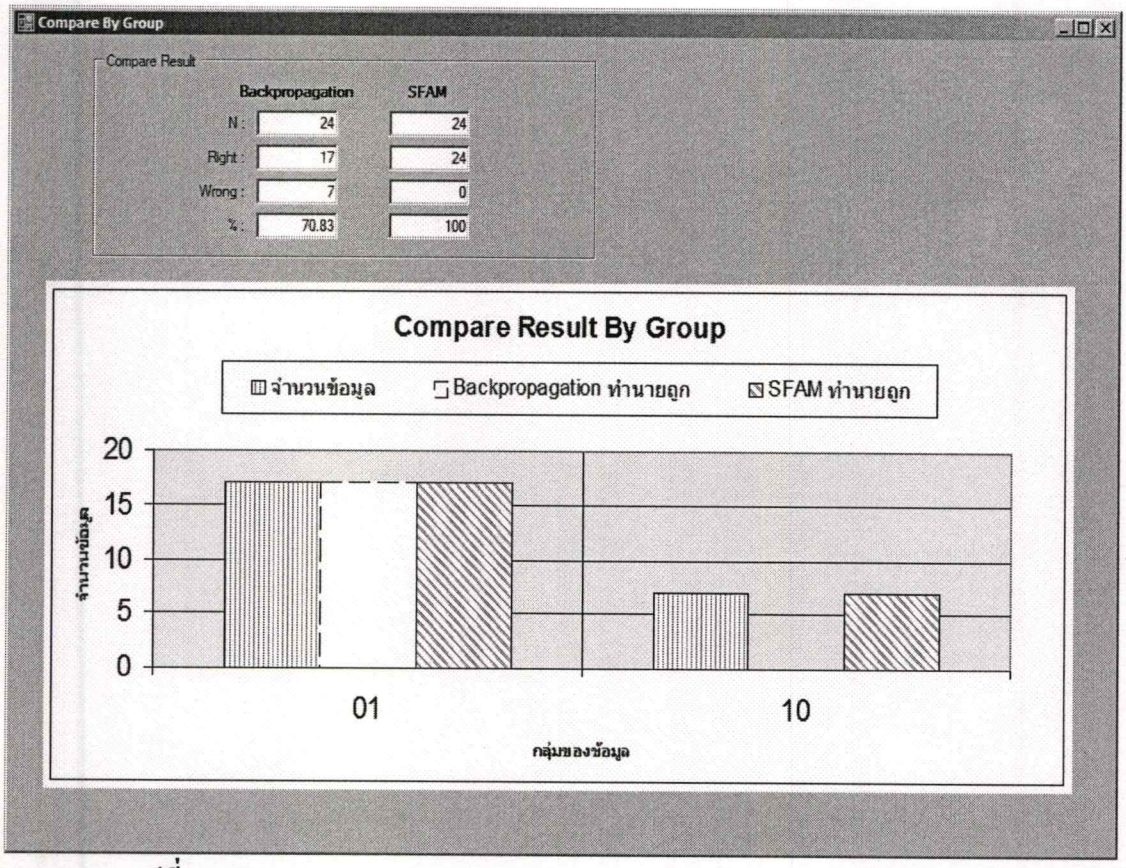
รูปที่ 4.45 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (1รอบ)



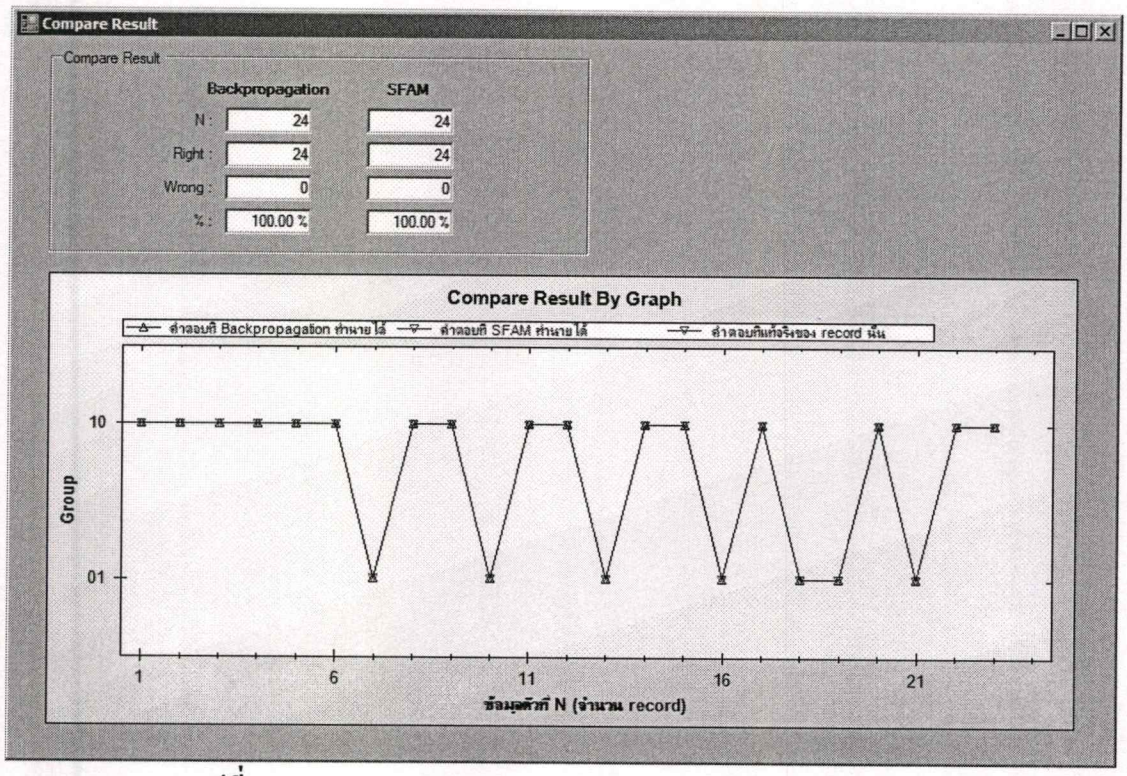
รูปที่ 4.46 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล



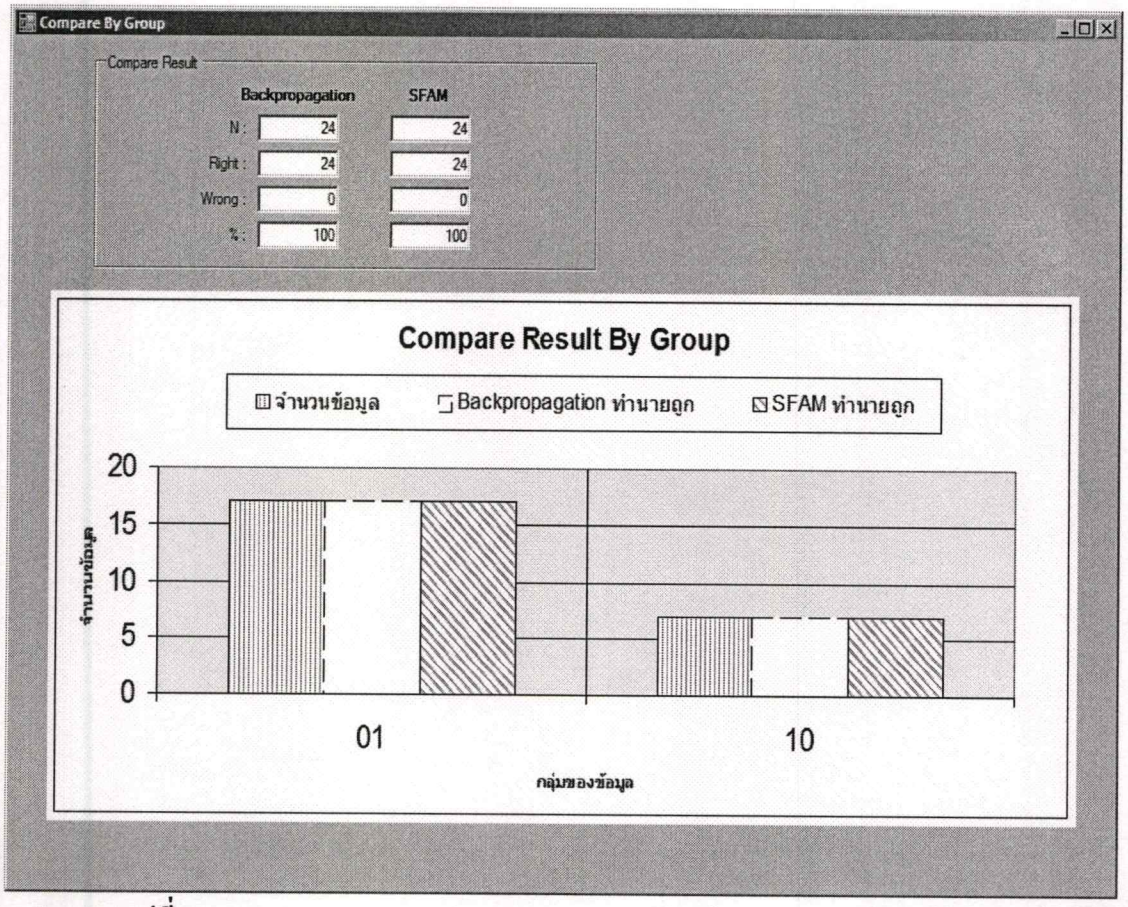
รูปที่ 4.47 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (10รอบ)



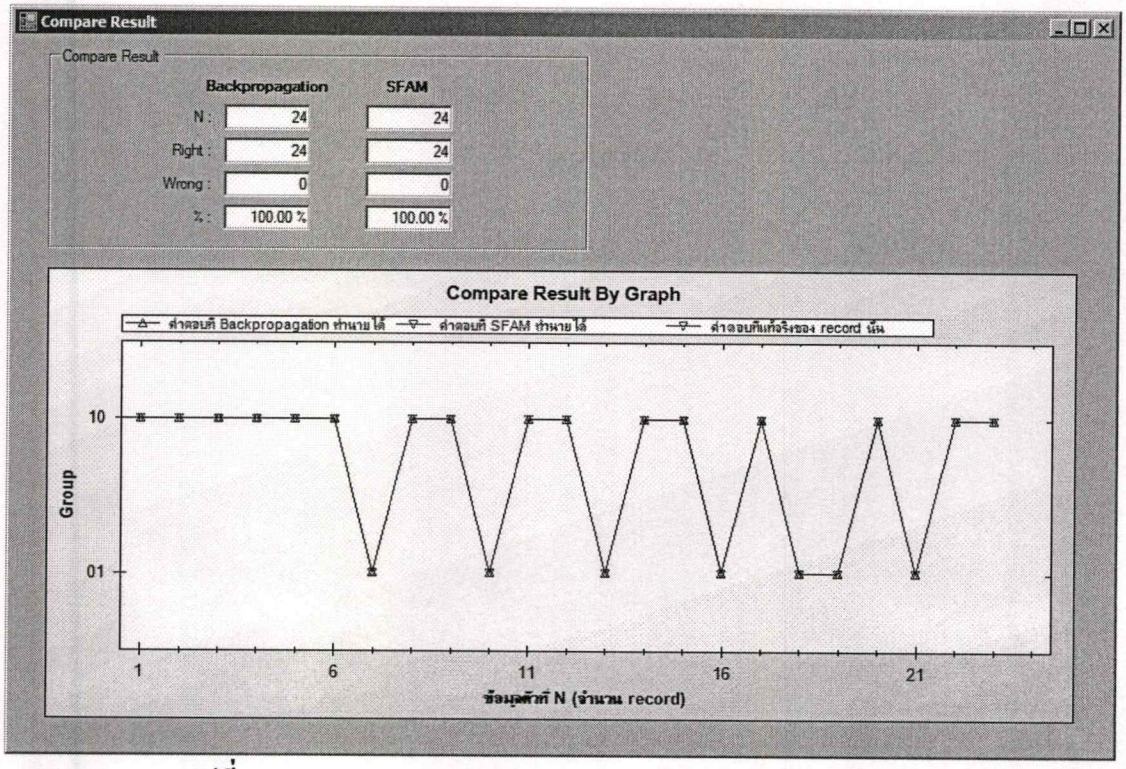
รูปที่ 4.48 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล



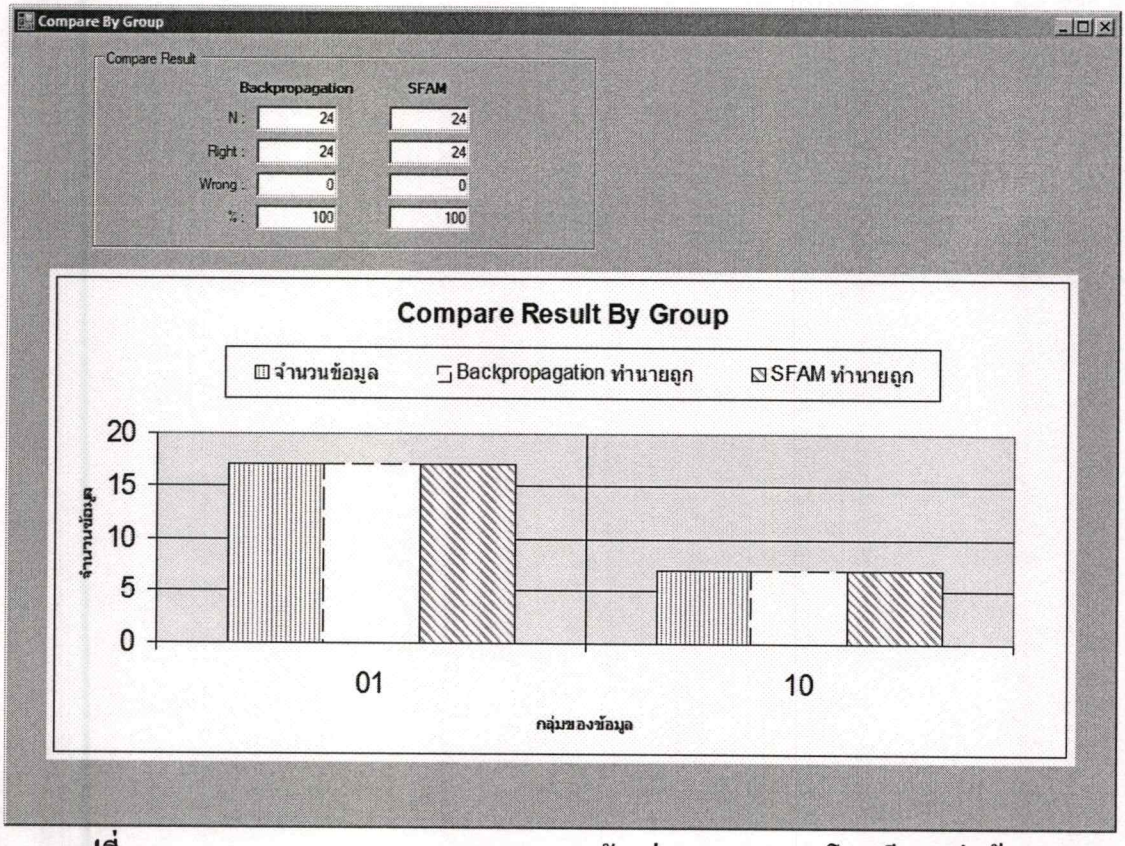
รูปที่ 4.49 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (100รอบ)



รูปที่ 4.50 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล



รูปที่ 4.51 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (1000รอบ)



รูปที่ 4.52 ผลการทดลองของ Lung Cancer 24 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล

ต่อมานำข้อมูลLung Cancer มาทดสอบโครงข่าย (Test Data)จำนวน 8 ตัวอย่างมาทดสอบ โดยใช้จำนวนข้อมูลในการฝึกสอนโครงข่าย 24 ตัวอย่าง เมื่อนำมาฝึกสอนโครงข่ายแล้ว นำมาทดสอบ

โดยกำหนด

-จำนวนตัวอย่างฝึกสอนโครงข่าย 24ตัวอย่าง (Data Train)

-จำนวนตัวอย่างที่ใช้ทดสอบโครงข่าย 8 ตัวอย่าง (Data Test)

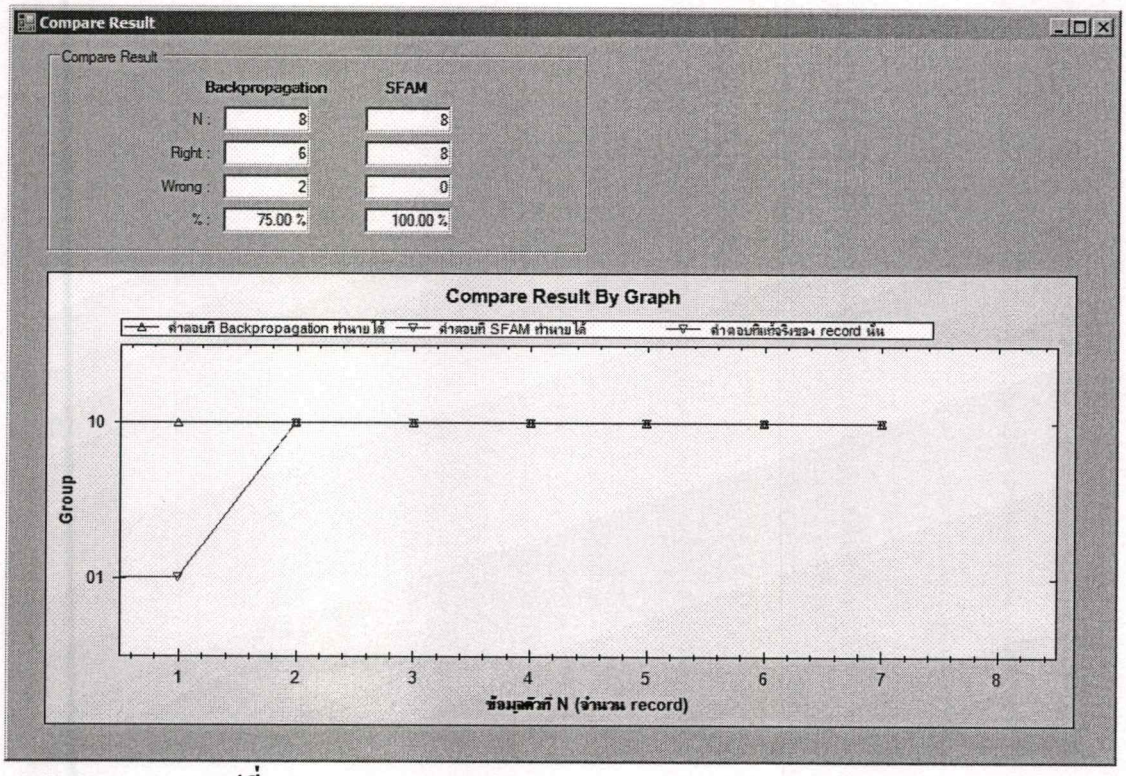
-Backpropagation ทำการวนที่ 1 รอบ, 10 รอบ, 100 รอบ, 1000 รอบ

-SFAM กำหนดค่า Vigilance (ρ) = 0.4

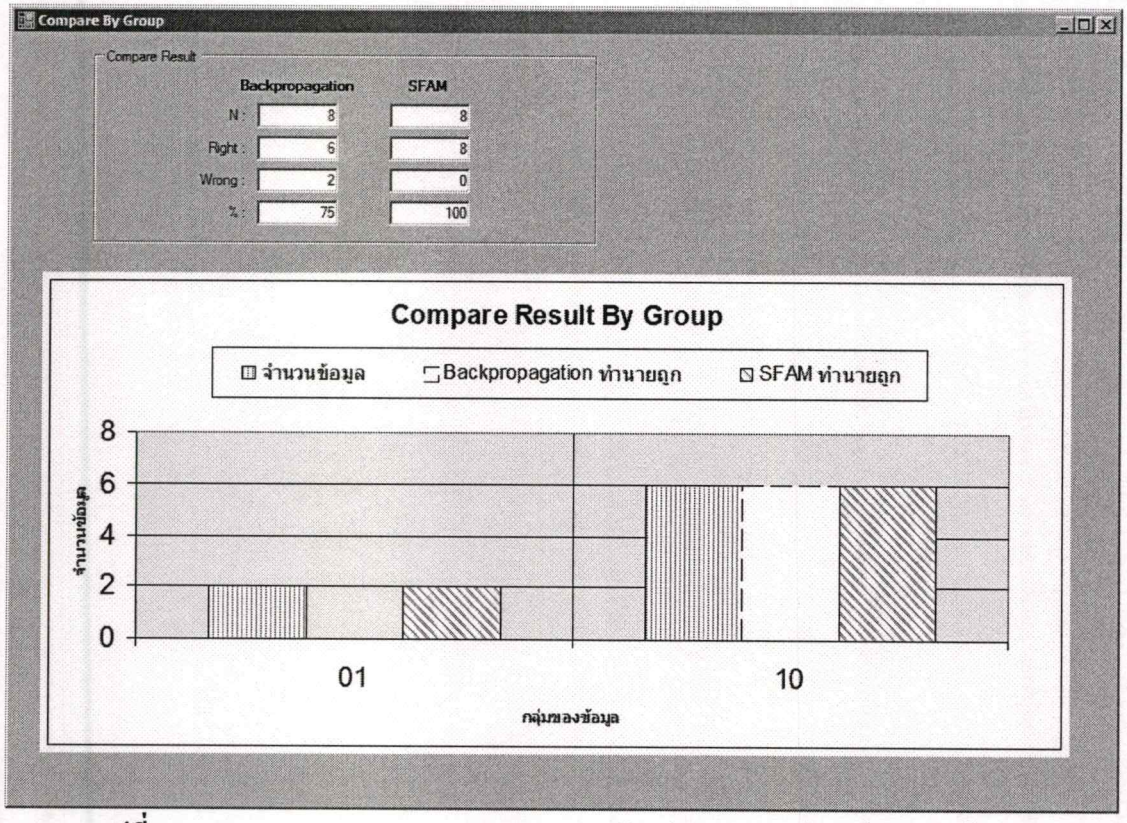
ได้ผลการทดลองดังนี้

ตารางที่ 4.4 แสดงผลลัพธ์การทดลอง Lung Cancer ด้วยตัวอย่าง (Data Test) 8 ตัวอย่าง

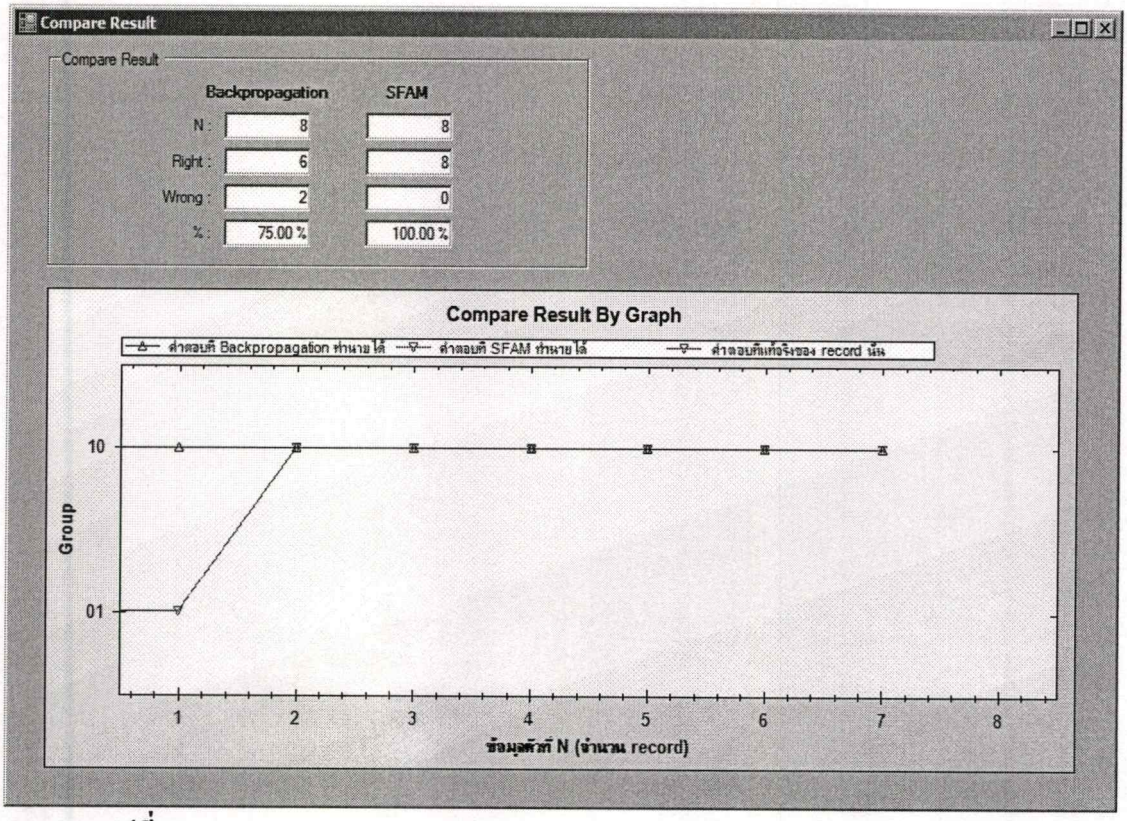
อัลกอริทึม	Backpropagation (1รอบ)	Backpropagation (10รอบ)	Backpropagation (100รอบ)	Backpropagation (1000รอบ)	SFAM
จำนวนตัวอย่าง	8	8	8	8	8
จำนวนถูก	6	6	7	7	8
จำแนกผิด	2	2	1	1	0
เปอร์เซ็นความถูกต้อง	75%	75%	87.5%	87.5%	100%



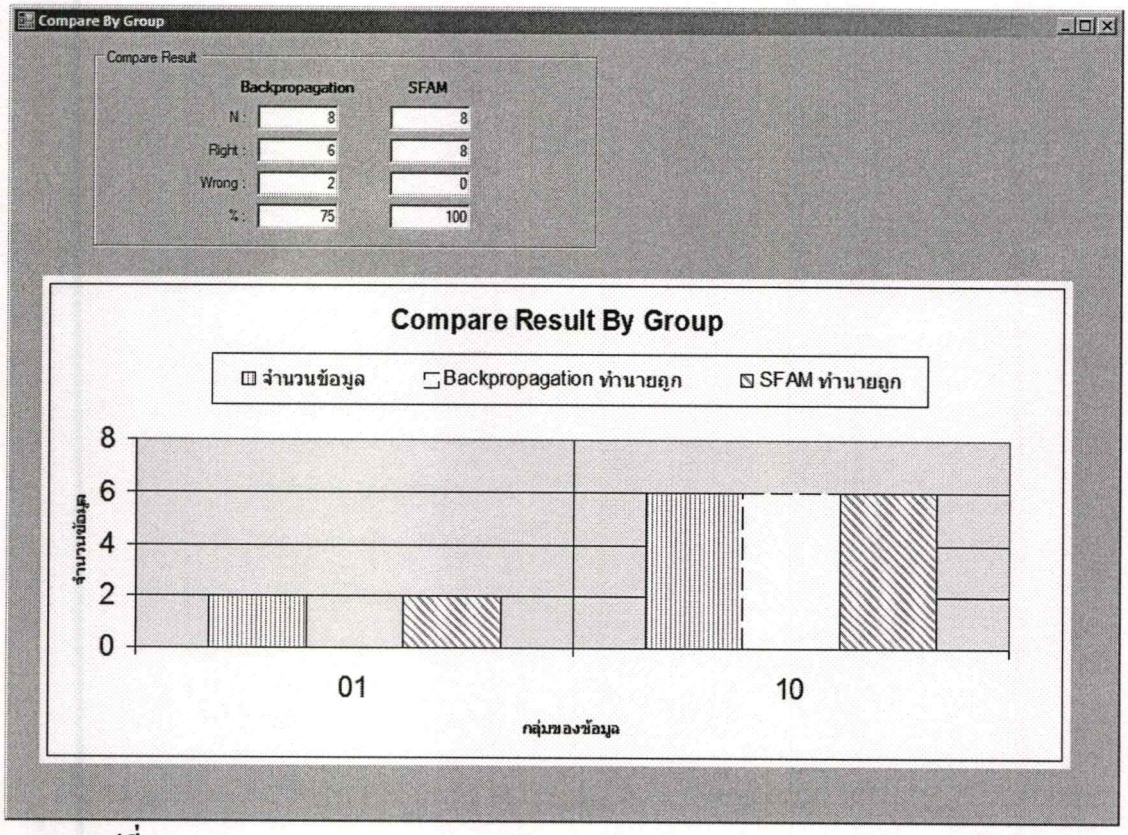
รูปที่ 4.53ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (1รอบ)



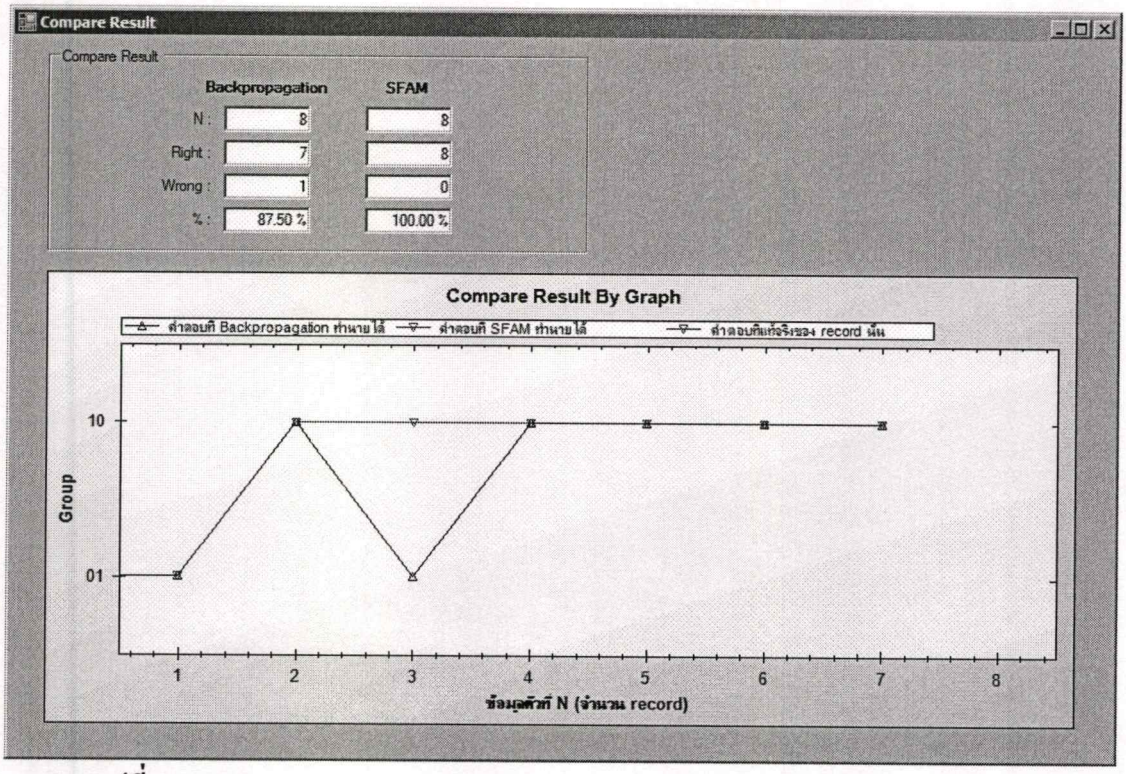
รูปที่ 4.54 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล



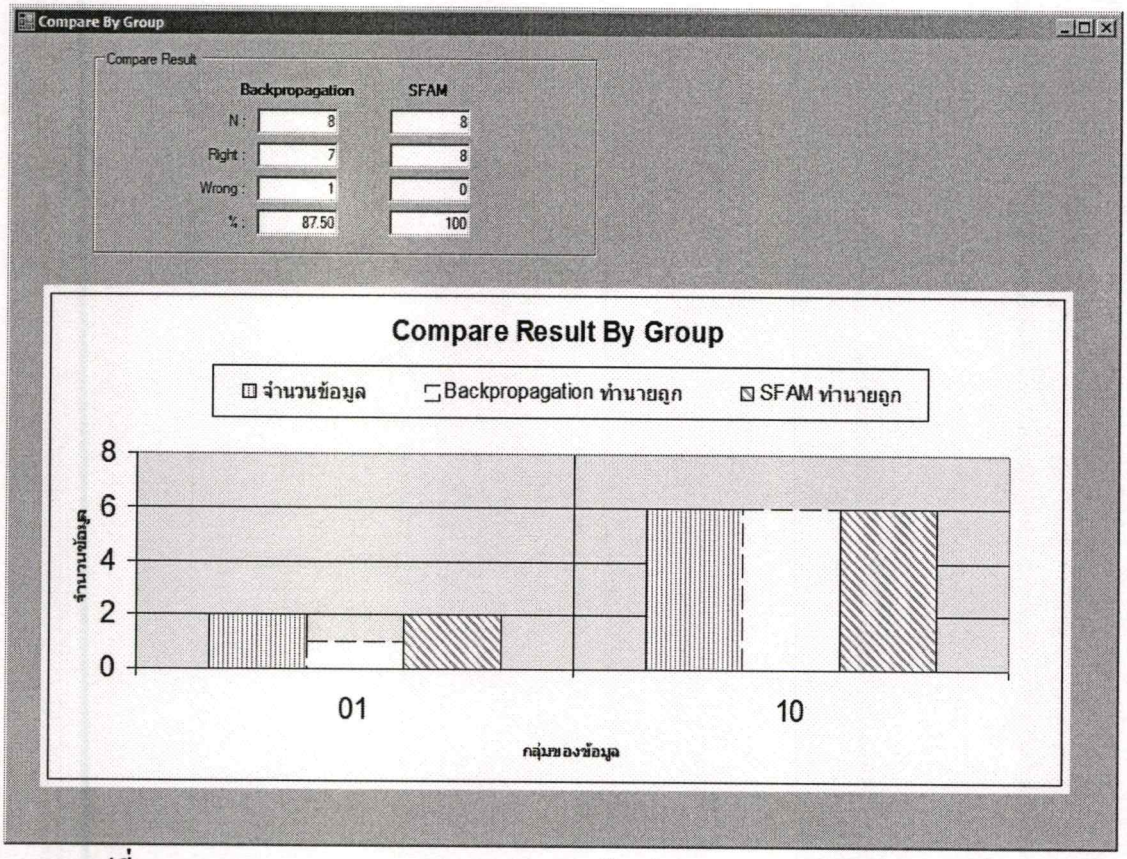
รูปที่ 4.55 ผลการทดลองของ Lung Cancer Data Test Backpropagation (10รอบ)



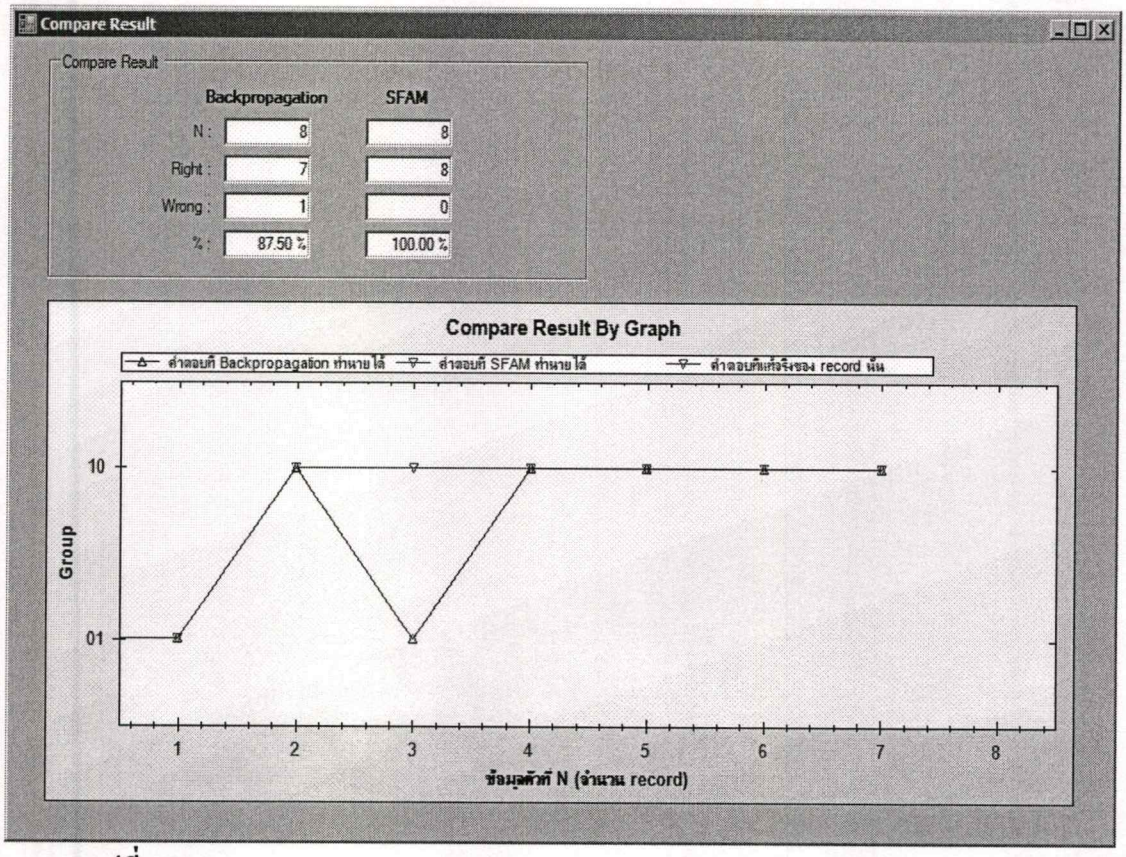
รูปที่ 4.56 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล



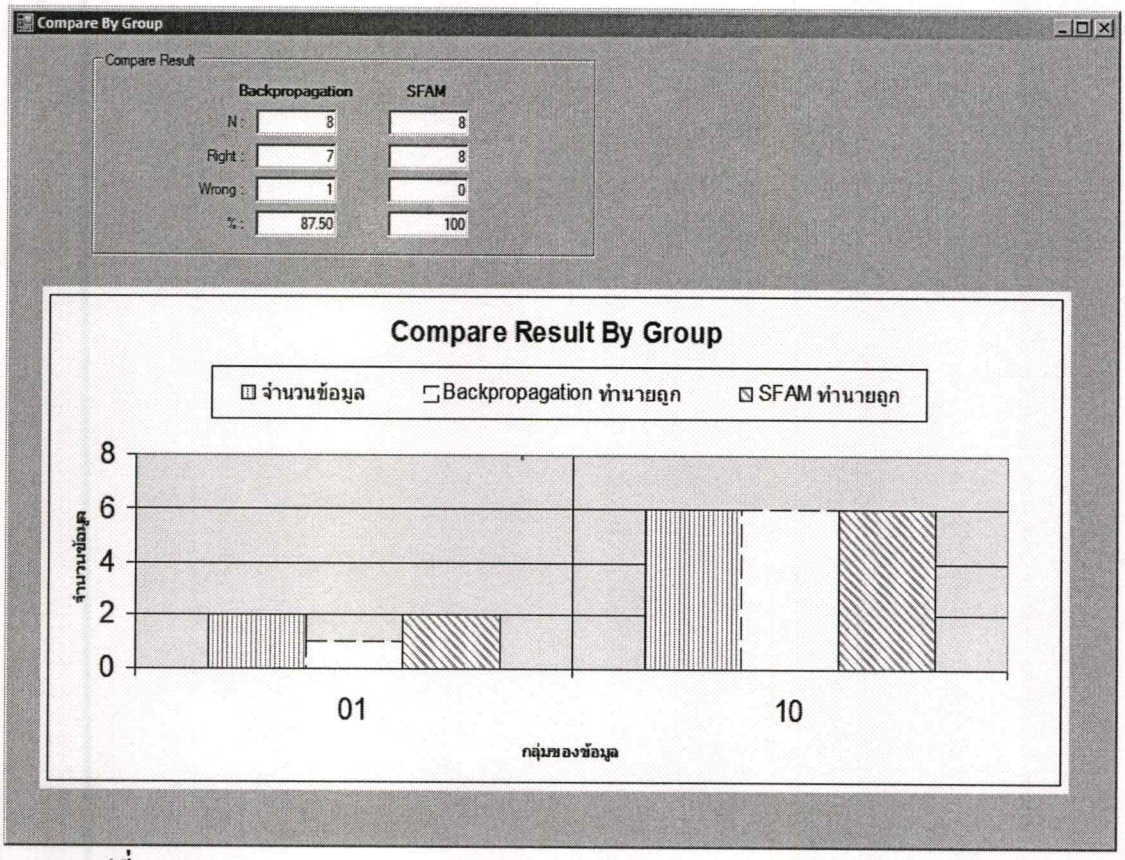
รูปที่ 4.57 ผลการทดลองของ Lung Cancer Data Test Backpropagation (100รอบ)



รูปที่ 4.58 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล



รูปที่ 4.59 ผลการทดลองของ Lung Cancer Data Test Backpropagation (1000รอบ)



รูปที่ 4.60 ผลการทดลองของ Lung Cancer 8 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล

4.3.3 Wine Data

ทำการเปรียบเทียบความถูกต้องข้อมูล Wine Data จำนวน 100 ตัวอย่าง ระหว่าง Backpropagation และ SFAM

โดยกำหนด

-จำนวนตัวอย่างฝึกสอนโครงข่าย 100 ตัวอย่าง (Data Train)

-จำนวนตัวอย่างที่ใช้ทดสอบโครงข่าย 100 ตัวอย่าง (Data Train)

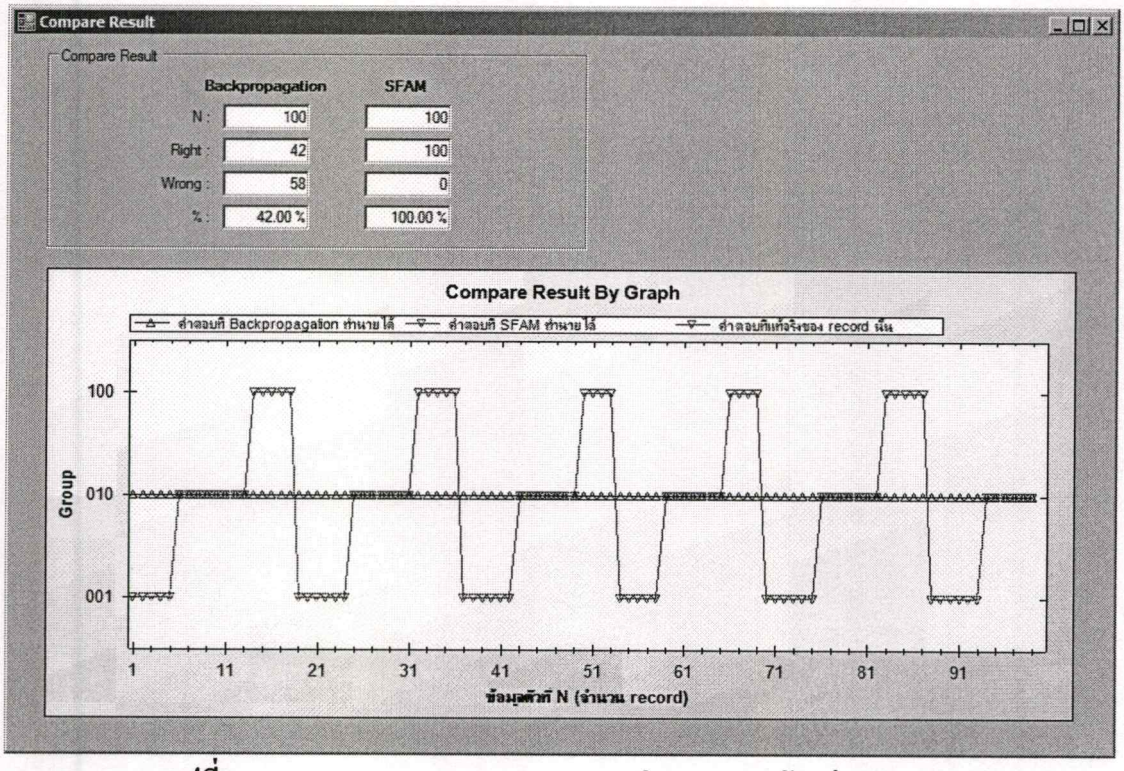
-Backpropagation ทำการวนที่ 1รอบ,10รอบ,100 รอบ,1000รอบ

-SFAM กำหนดค่า Vigilance (ρ) = 0.4

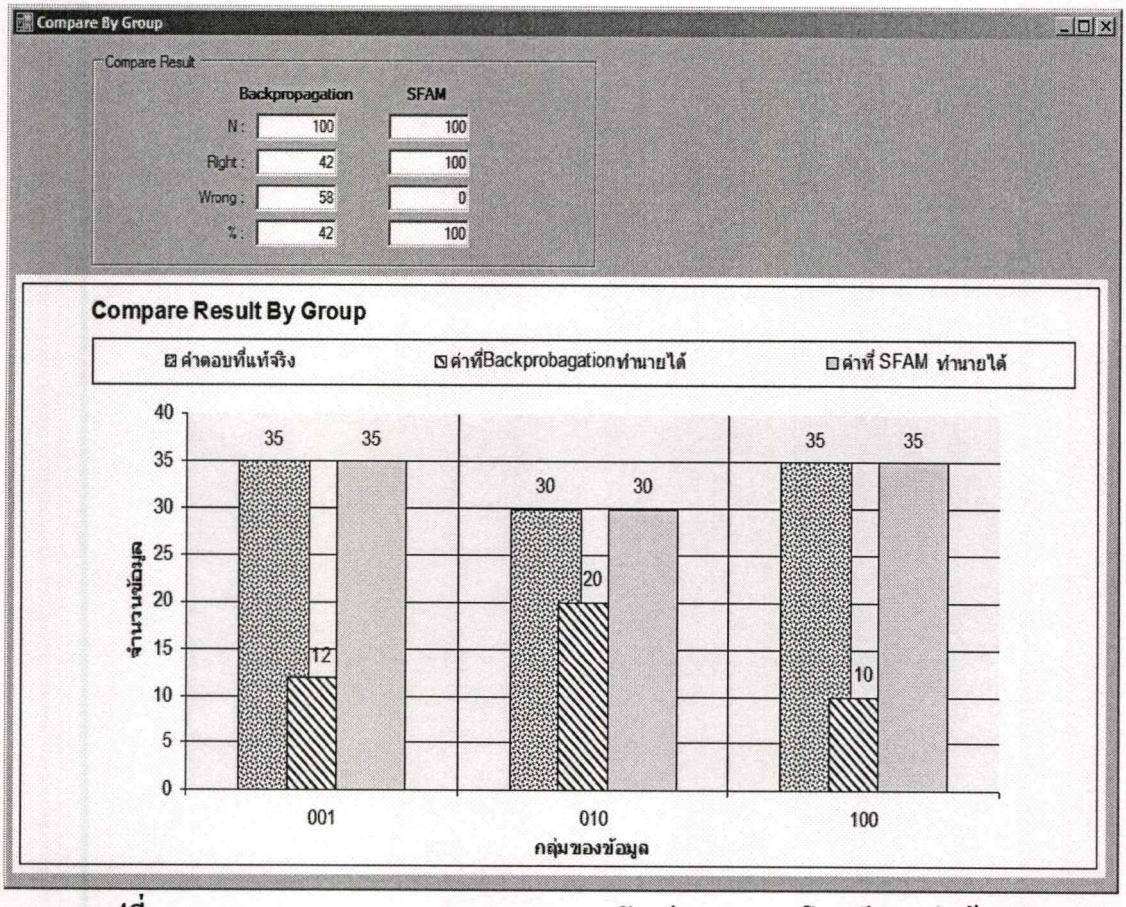
ได้ผลการทดลองดังนี้

ตารางที่ 4.5 แสดงผลลัพธ์การทดลอง Wine Data ด้วยตัวอย่าง 100 ตัว

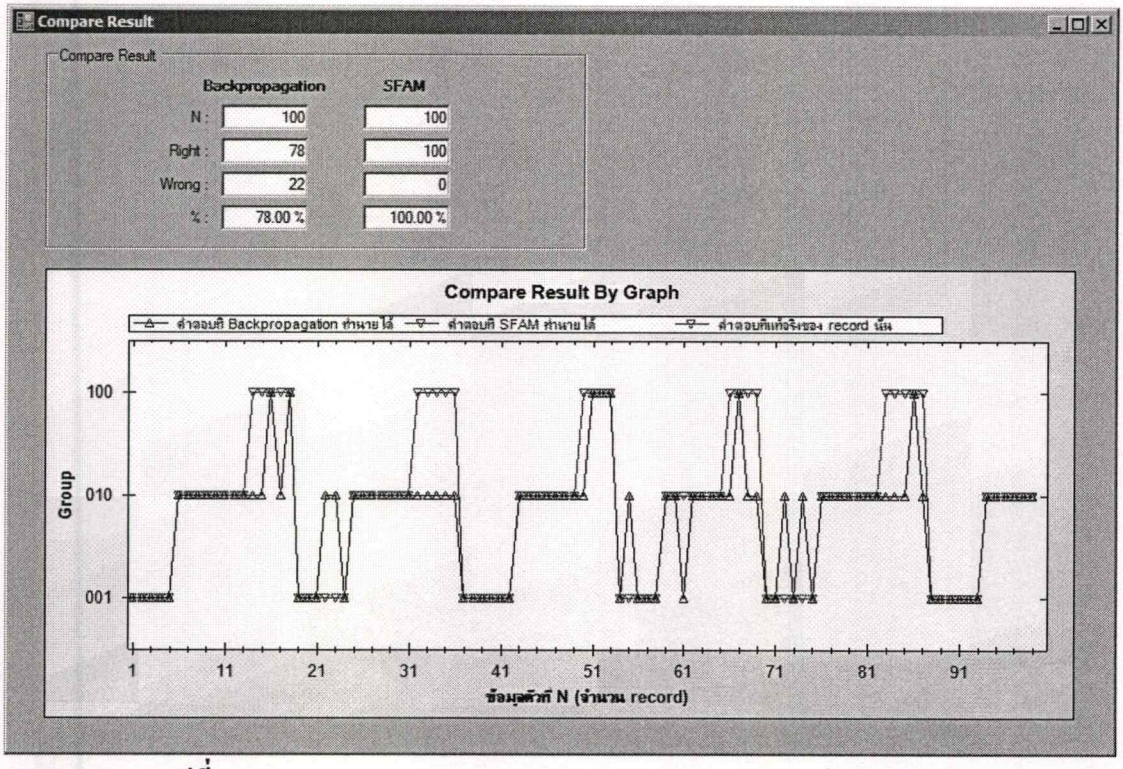
อัลกอริทึม	Backpropagation (1รอบ)	Backpropagation (10รอบ)	Backpropagation (100รอบ)	Backpropagation (1000รอบ)	SFAM
จำนวนตัวอย่าง	100	100	100	100	100
จำนวนถูก	42	78	100	100	100
จำแนกผิด	58	22	0	0	100
เปอร์เซ็นความถูกต้อง	42%	78%	100%	100%	100%



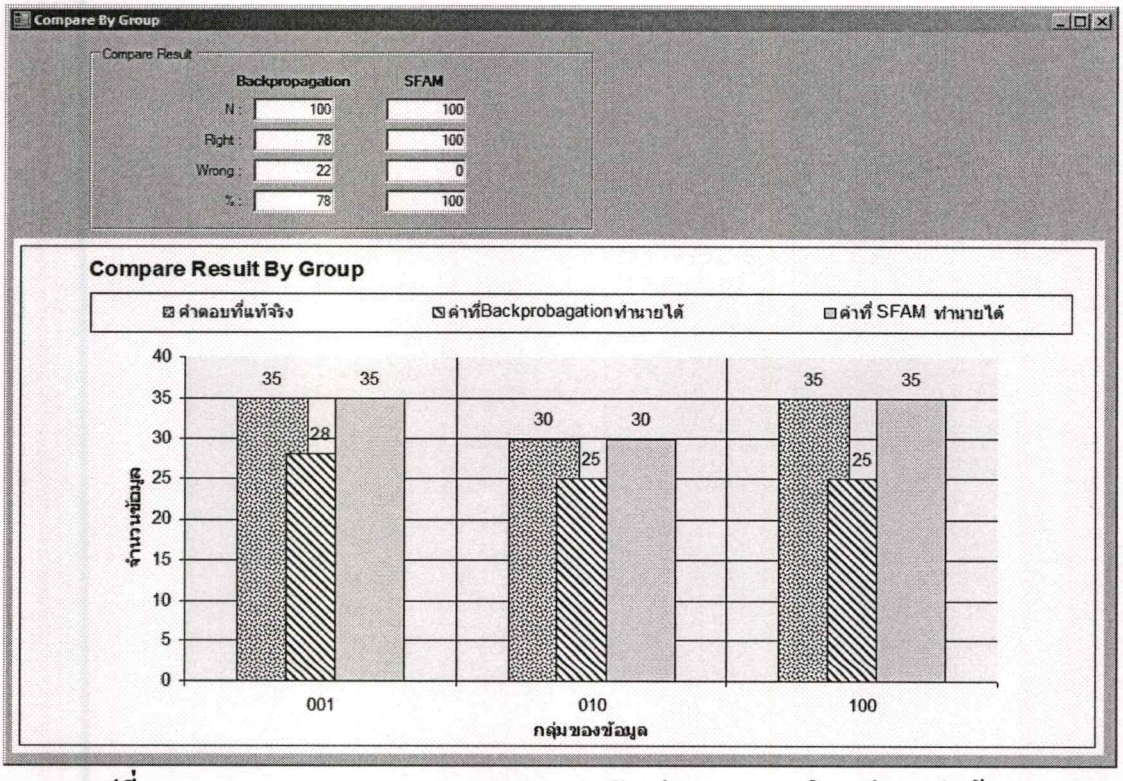
รูปที่ 4.61 ผลการทดลองของ Wine Data จำนวน 100 ตัวอย่าง (1รอบ)



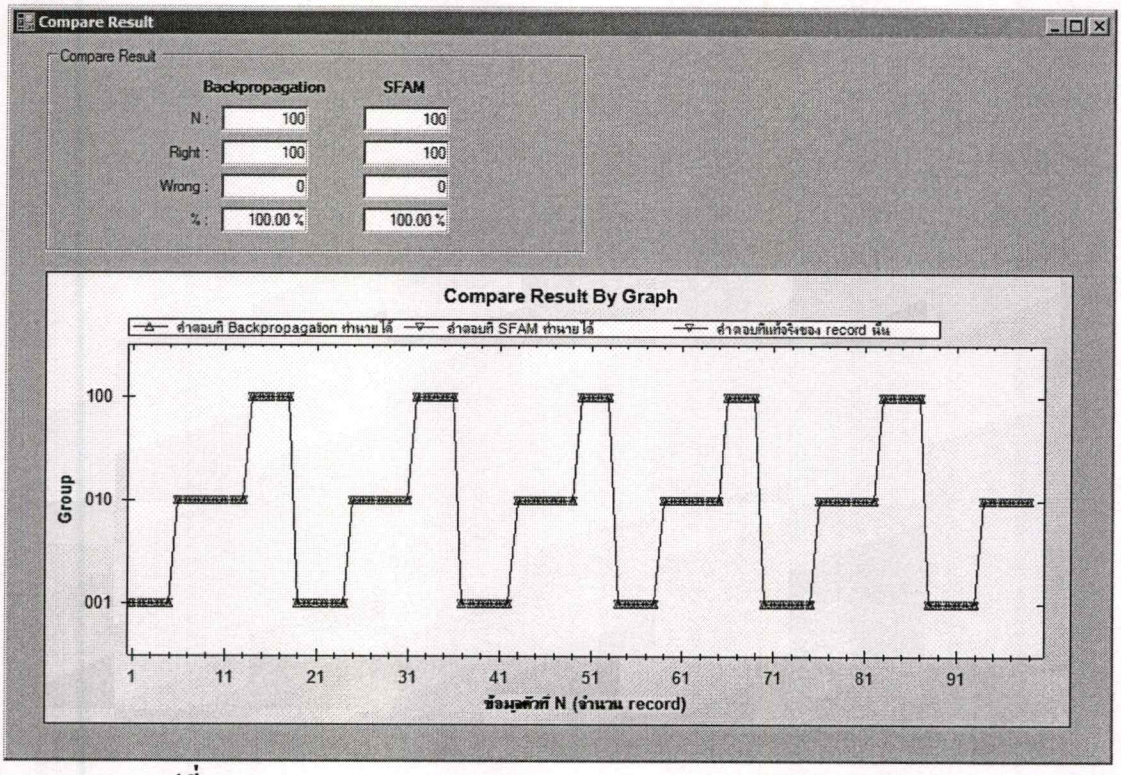
รูปที่ 4.62 ผลการทดลองของ Wine Data 100 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล



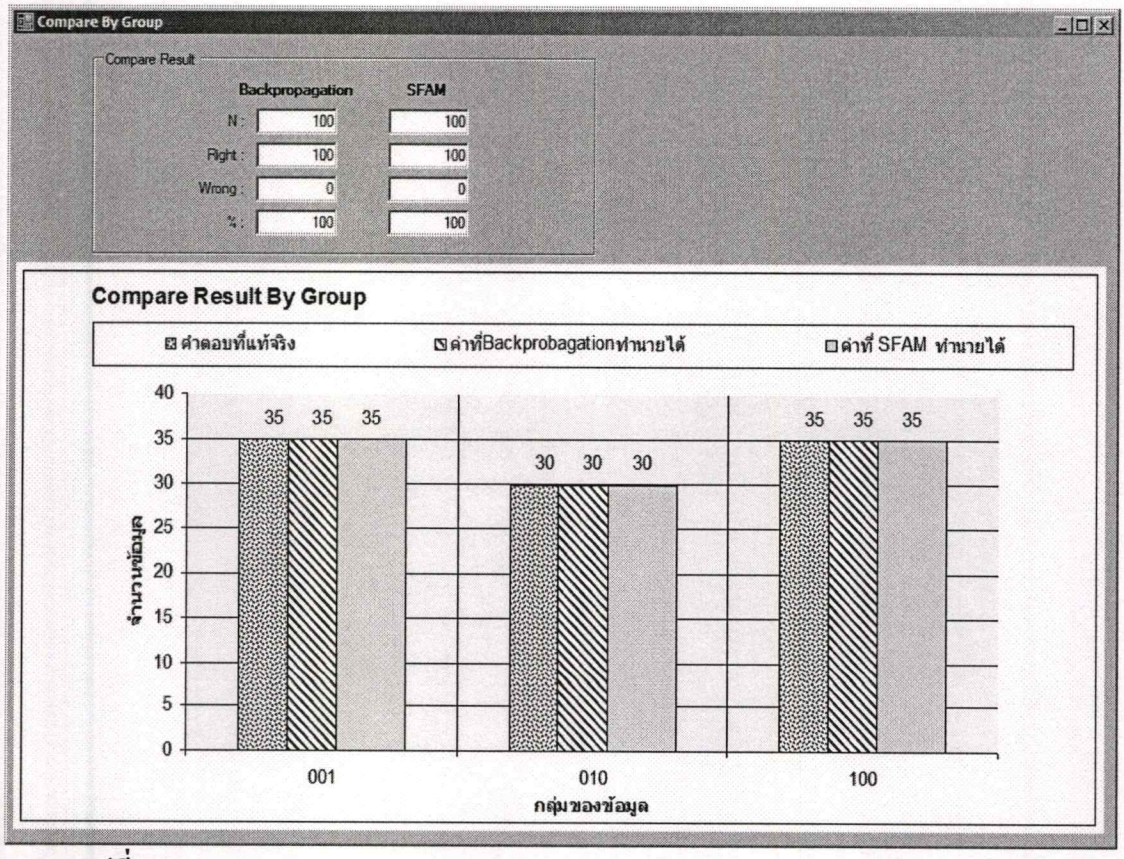
รูปที่ 4.63 ผลการทดลองของ Wine Data จำนวน 100 ตัวอย่าง (10รอบ)



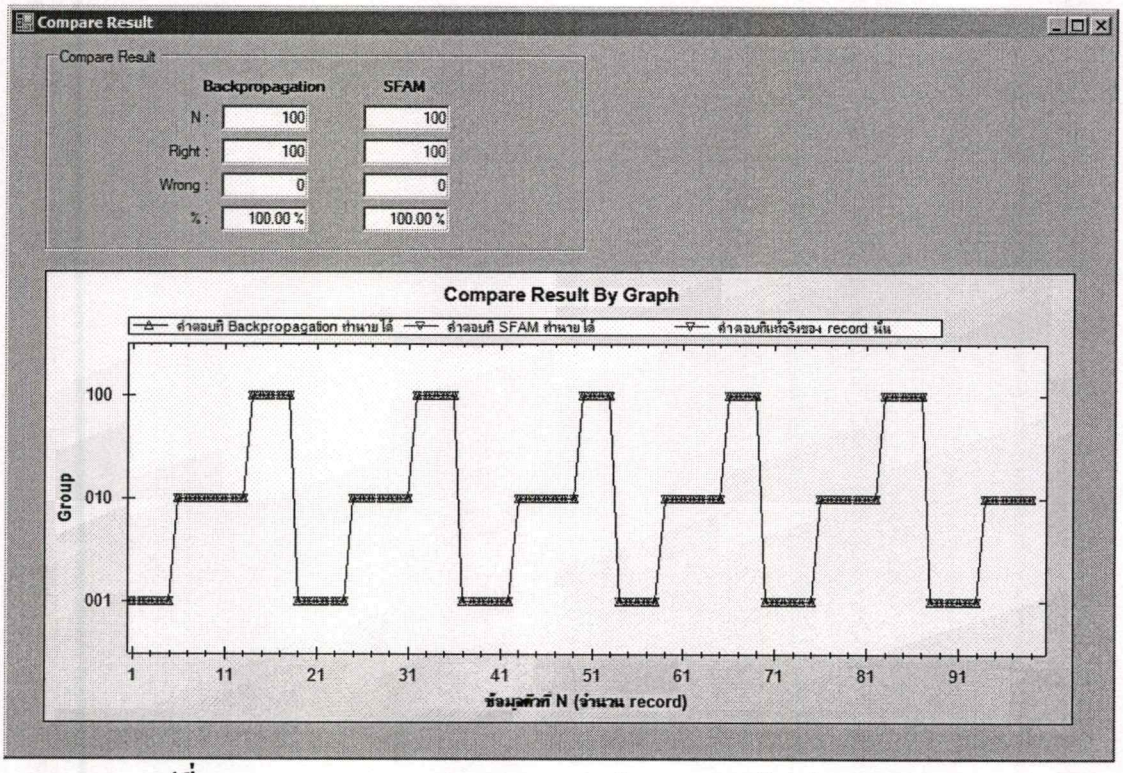
รูปที่ 4.64 ผลการทดลองของ Wine Data 100 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล



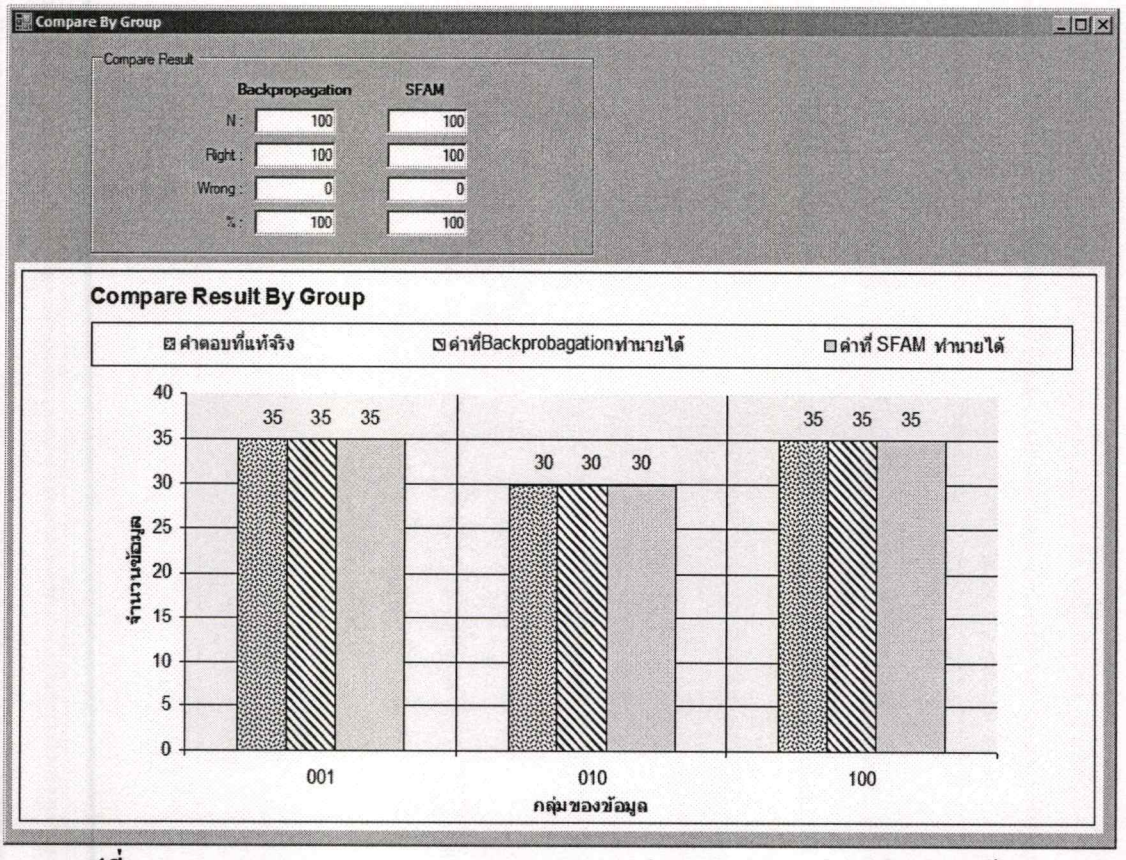
รูปที่ 4.65 ผลการทดลองของ Wine Data จำนวน 100 ตัวอย่าง (100รอบ)



รูปที่ 4.66 ผลการทดลองของ Wine Data 100 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล



รูปที่ 4.67 ผลการทดลองของ Wine Data จำนวน 100 ตัวอย่าง (1000รอบ)



รูปที่ 4.68 ผลการทดลองของ Wine Data 100 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล

ต่อมานำข้อมูลWine มาทดสอบโครงข่าย (Test Data)จำนวน 78 ตัวอย่างมาทดสอบ โดยใช้จำนวนข้อมูลในการฝึกสอนโครงข่าย 100 ตัวอย่าง เมื่อนำมาฝึกสอนโครงข่ายแล้ว นำมาทดสอบ โดยกำหนด

-จำนวนตัวอย่างฝึกสอนโครงข่าย 100ตัวอย่าง (Data Train)

-จำนวนตัวอย่างที่ใช้ทดสอบโครงข่าย 78 ตัวอย่าง (Data Test)

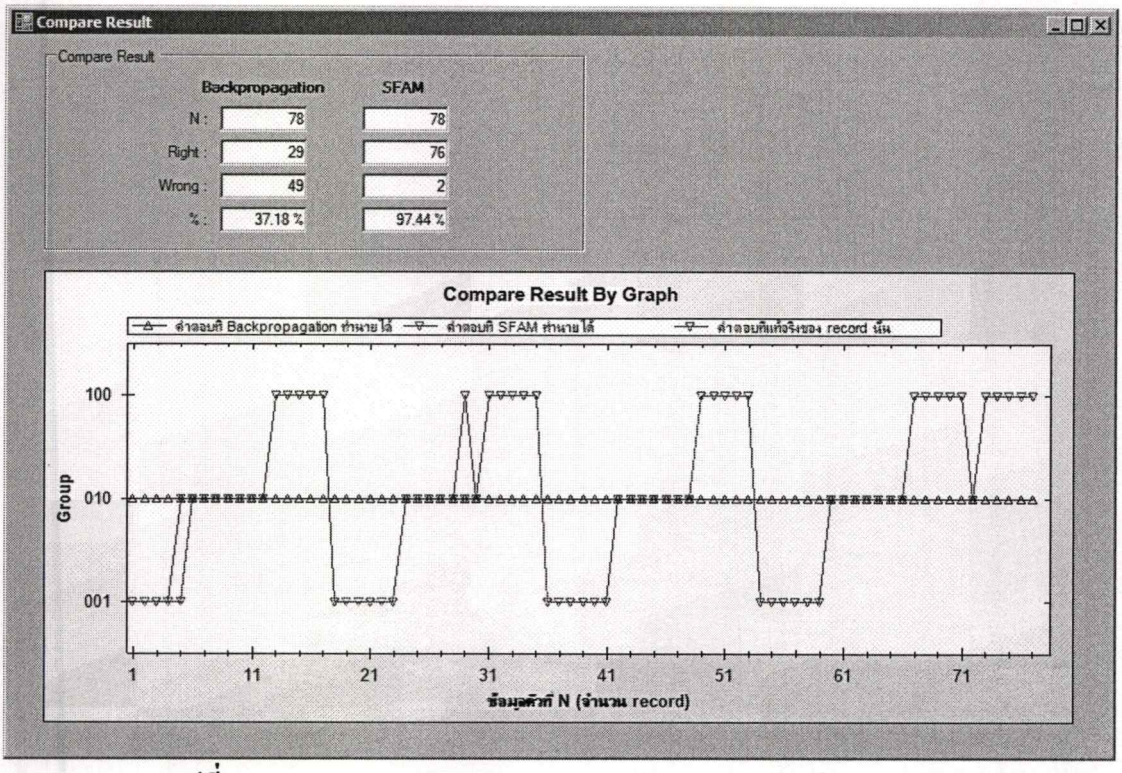
-Backpropagation ทำการวนที่ 1 รอบ, 10 รอบ, 100 รอบ, 1000 รอบ

-SFAM กำหนดค่า Vigilance (ρ) = 0.4

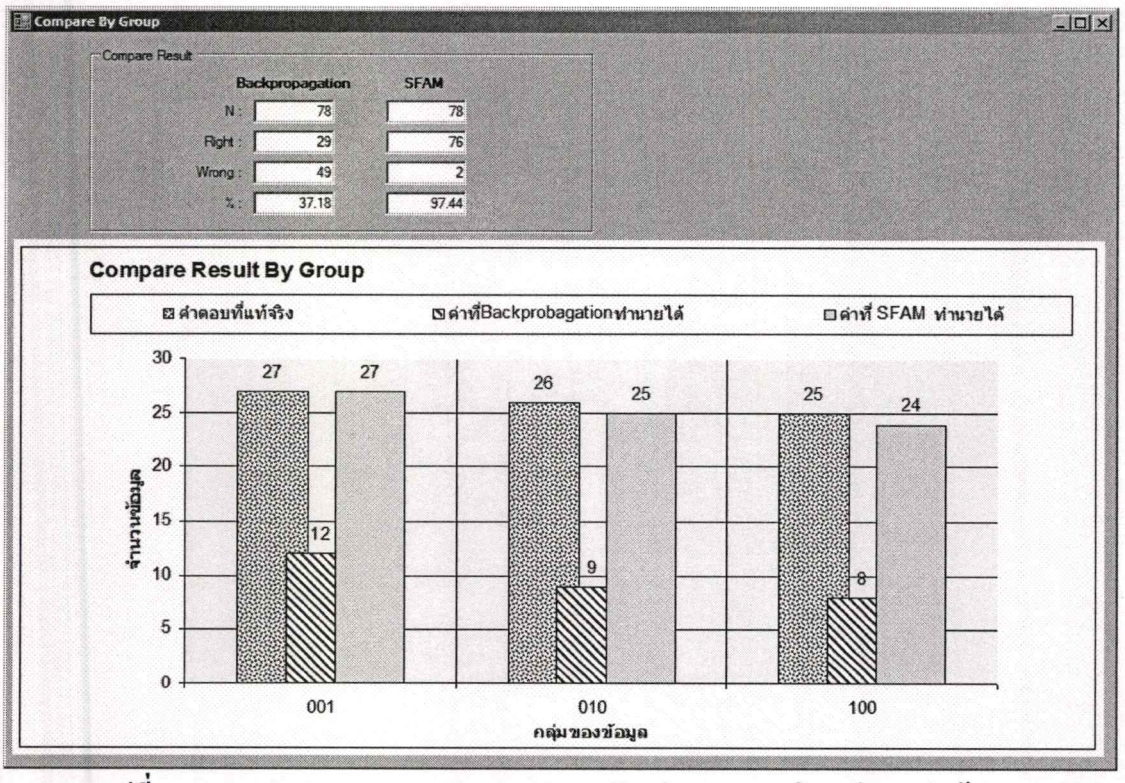
ได้ผลการทดลองดังนี้

ตารางที่ 4.6 แสดงผลลัพธ์การทดลอง Wine Data ด้วยตัวอย่าง (Data Test) 78 ตัวอย่าง

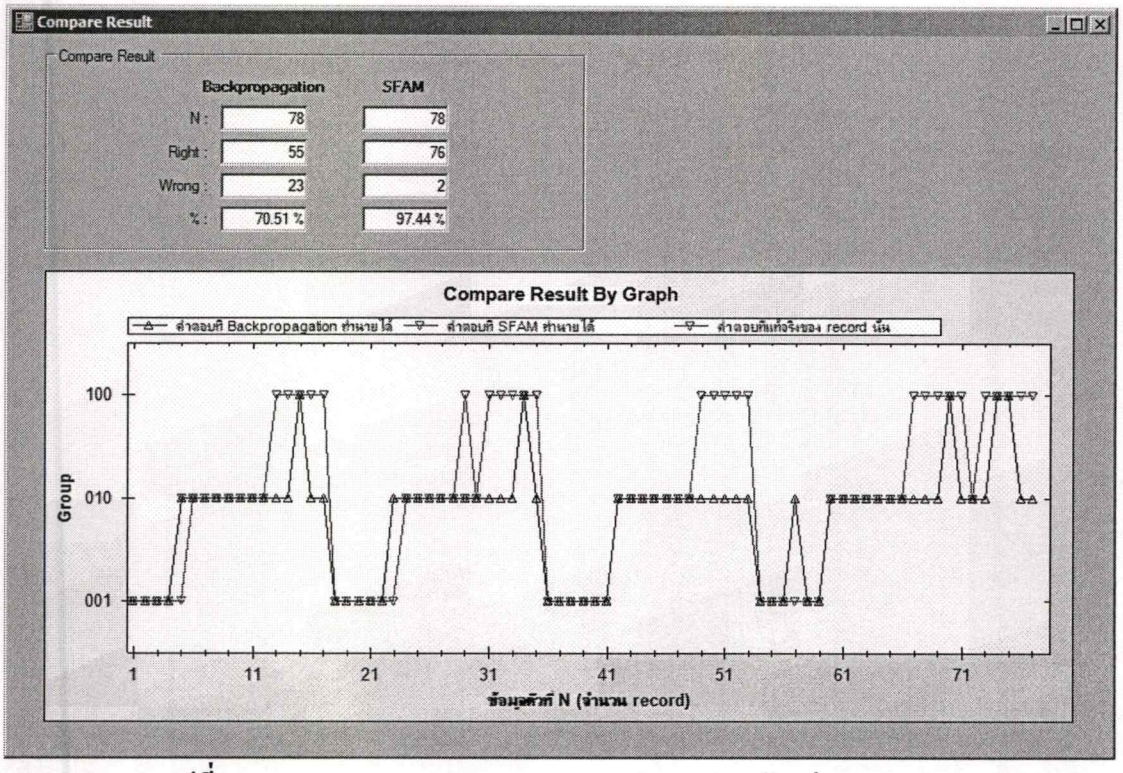
อัลกอริทึม	Backpropagation (1รอบ)	Backpropagation (10รอบ)	Backpropagation (100รอบ)	Backpropagation (1000รอบ)	SFAM
จำนวนตัวอย่าง	78	78	78	78	78
จำนวนถูก	29	55	76	76	76
จำแนกผิด	49	23	2	2	2
เปอร์เซ็นความถูกต้อง	37.18%	70.51%	97.44%	97.44%	97.44%



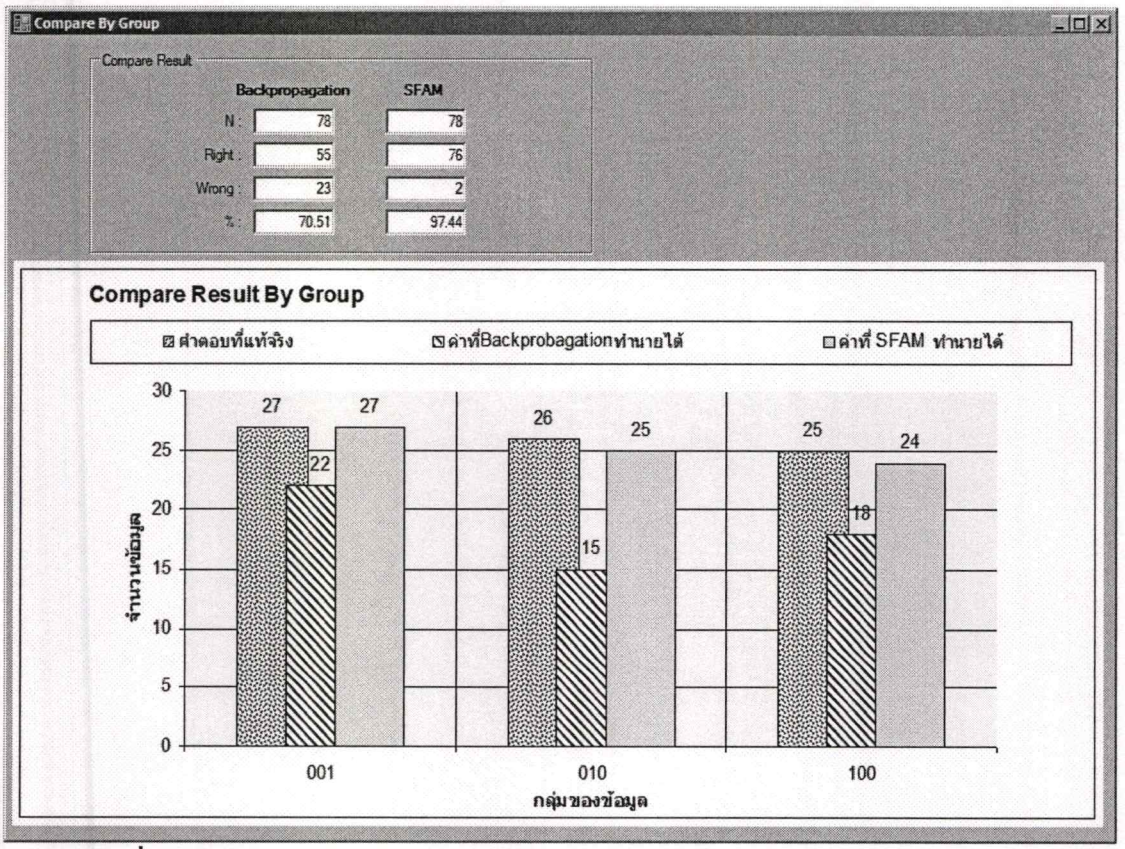
รูปที่ 4.69 ผลการทดลองของ Wine Data จำนวน 78 ตัวอย่าง (1รอบ)



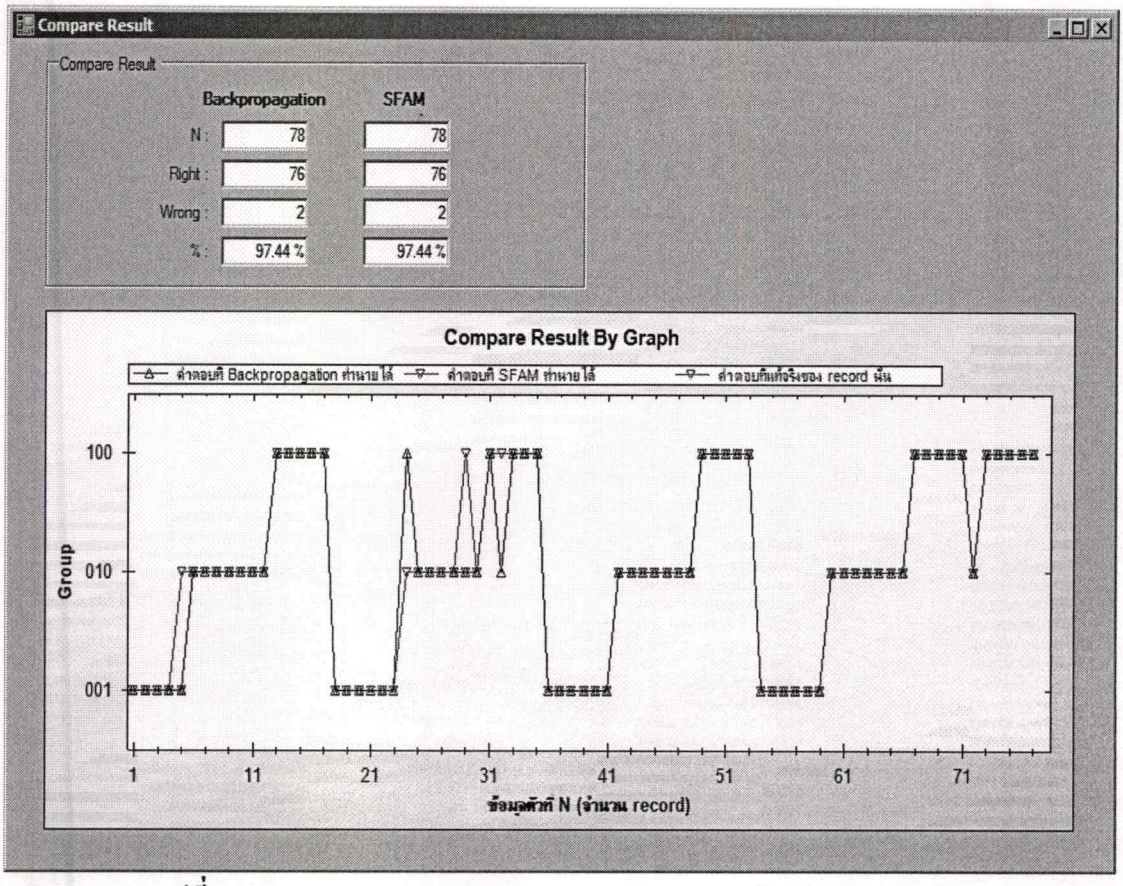
รูปที่ 4.70 ผลการทดลองของ Wine Data 78 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล



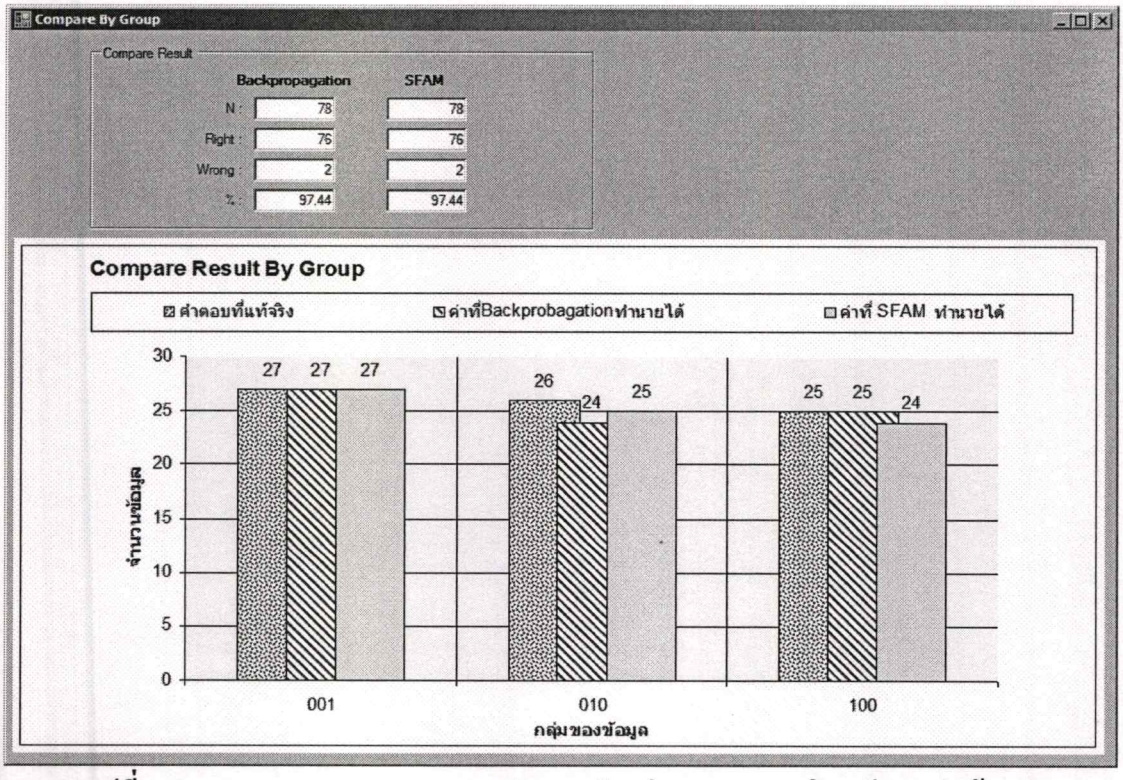
รูปที่ 4.71 ผลการทดลองของ Wine Data จำนวน 78 ตัวอย่าง (10รอบ)



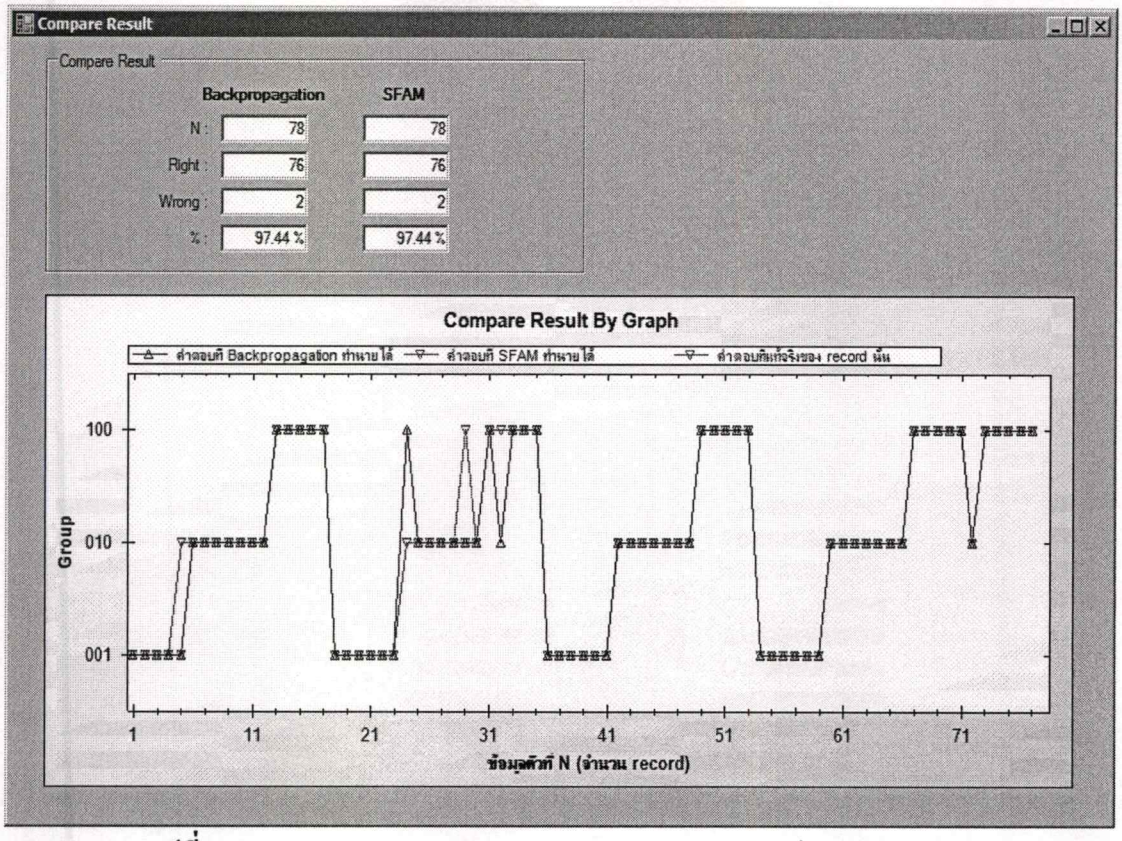
รูปที่ 4.72 ผลการทดลองของ Wine Data 78 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล



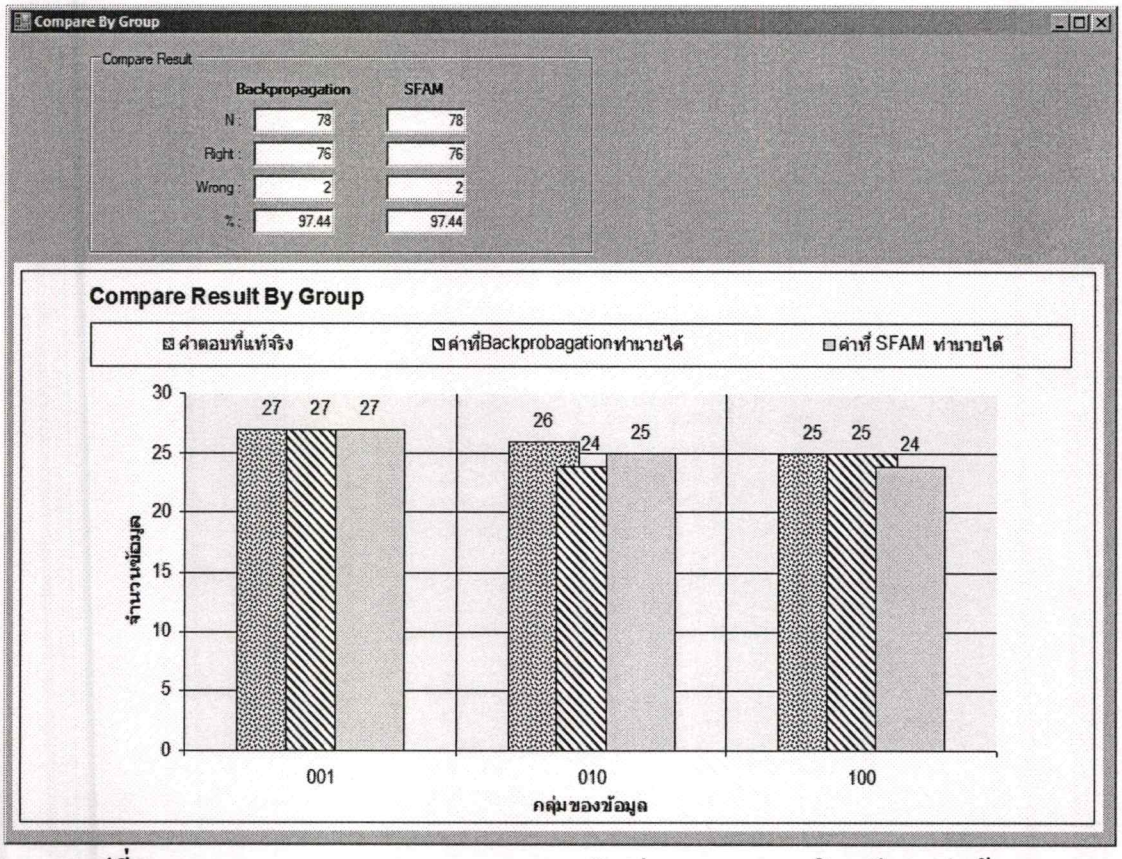
รูปที่ 4.73 ผลการทดลองของ Wine Data จำนวน 78 ตัวอย่าง (100รอบ)



รูปที่ 4.74 ผลการทดลองของ Wine Data 78 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล



รูปที่ 4.75 ผลการทดลองของ Wine Data จำนวน 78 ตัวอย่าง (1000รอบ)



รูปที่ 4.76 ผลการทดลองของ Wine Data 78 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล

4.3.4 Breast Data

ทำการเปรียบเทียบความถูกต้องข้อมูล Wine Data จำนวน 420 ตัวอย่าง ระหว่าง Backpropagation และ SFAM

โดยกำหนด

-จำนวนตัวอย่างฝึกสอนโครงข่าย 420 ตัวอย่าง (Data Train)

-จำนวนตัวอย่างที่ใช้ทดสอบโครงข่าย 420 ตัวอย่าง (Data Train)

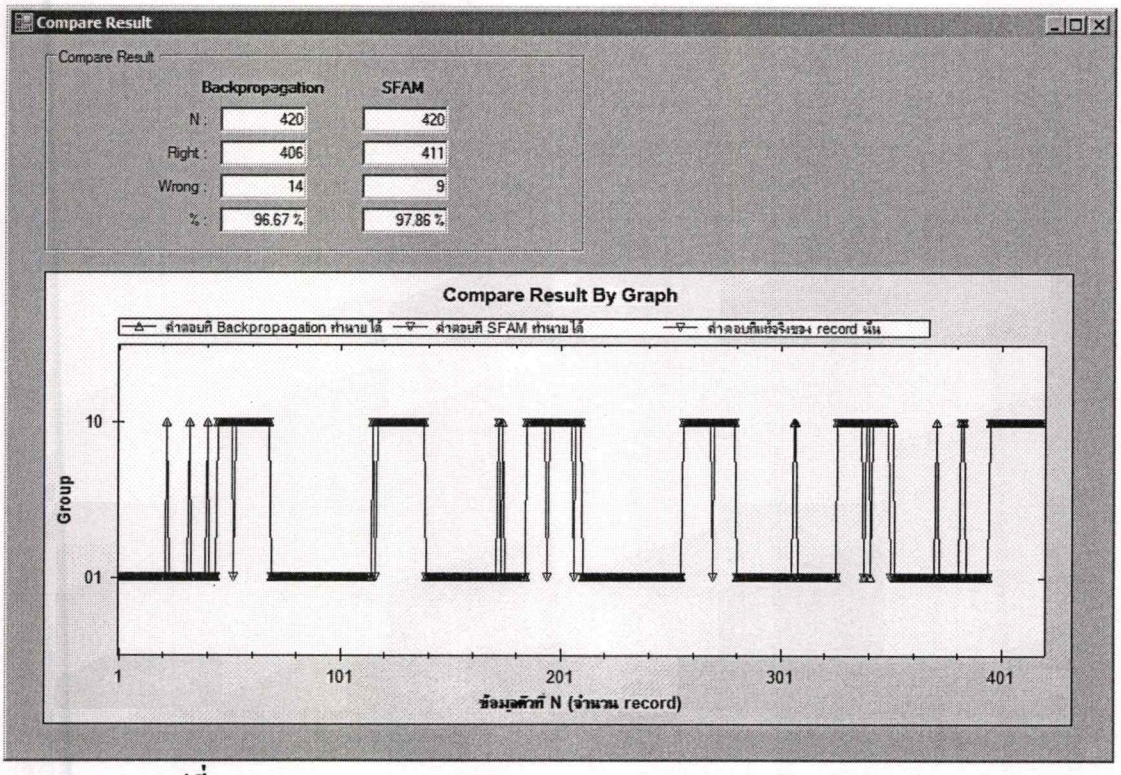
-Backpropagation ทำการวนที่ 1รอบ,10รอบ,100 รอบ,1000รอบ

-SFAM กำหนดค่า Vigilance (ρ) = 0.4

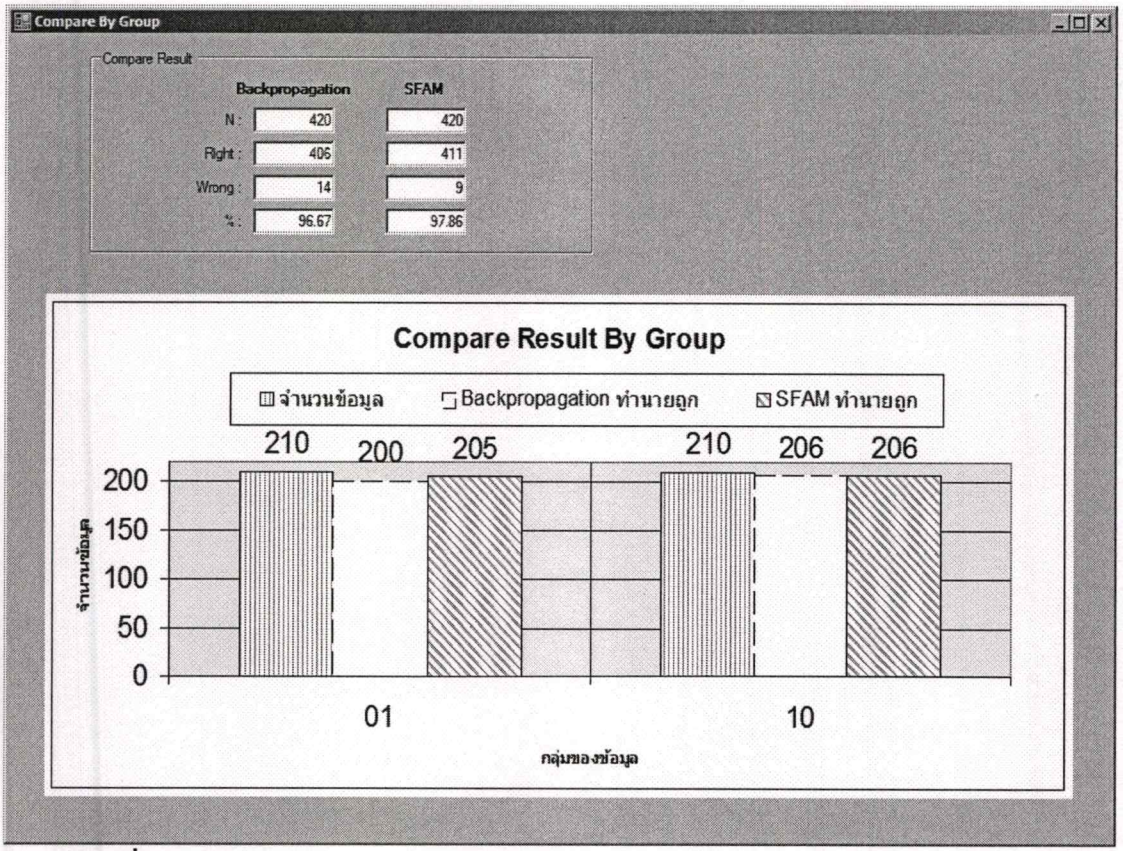
ได้ผลการทดลองดังนี้

ตารางที่ 4.7 แสดงผลลัพธ์การทดลอง Breast Data ด้วยตัวอย่าง 420 ตัวอย่าง

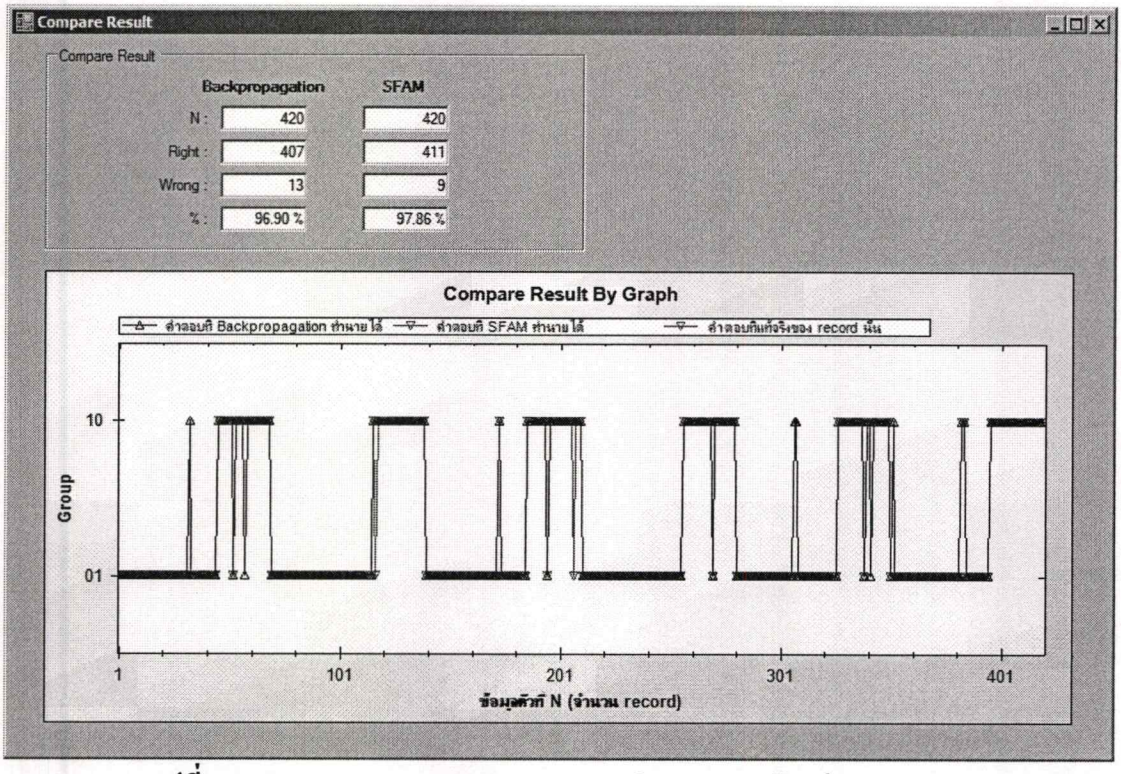
อัลกอริทึม	Backpropagation (1รอบ)	Backpropagation (10รอบ)	Backpropagation (100รอบ)	Backpropagation (1000รอบ)	SFAM
จำนวนตัวอย่าง	420	420	420	420	420
จำนวนถูก	406	407	408	408	411
จำแนกผิด	14	13	12	12	9
เปอร์เซ็นความถูกต้อง	96.67%	96.90%	97.14%	97.14%	97.86%



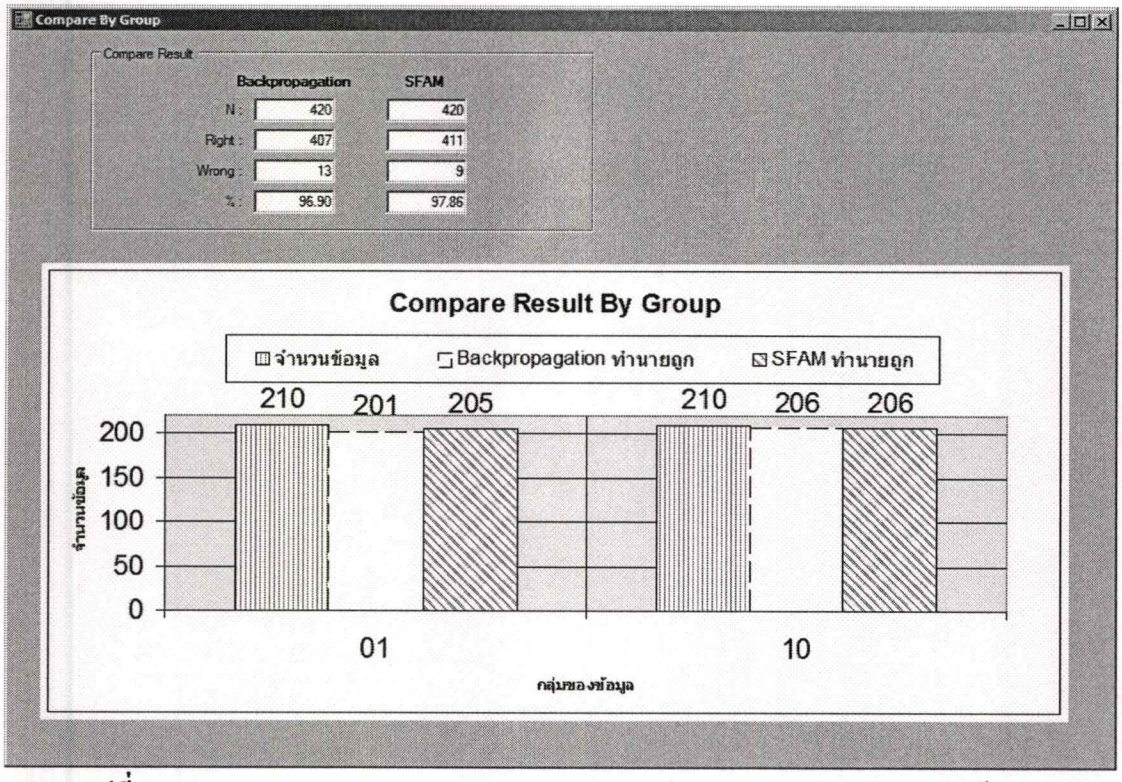
รูปที่ 4.77 ผลการทดลองของ Breast Data จำนวน 420 ตัวอย่าง (1รอบ)



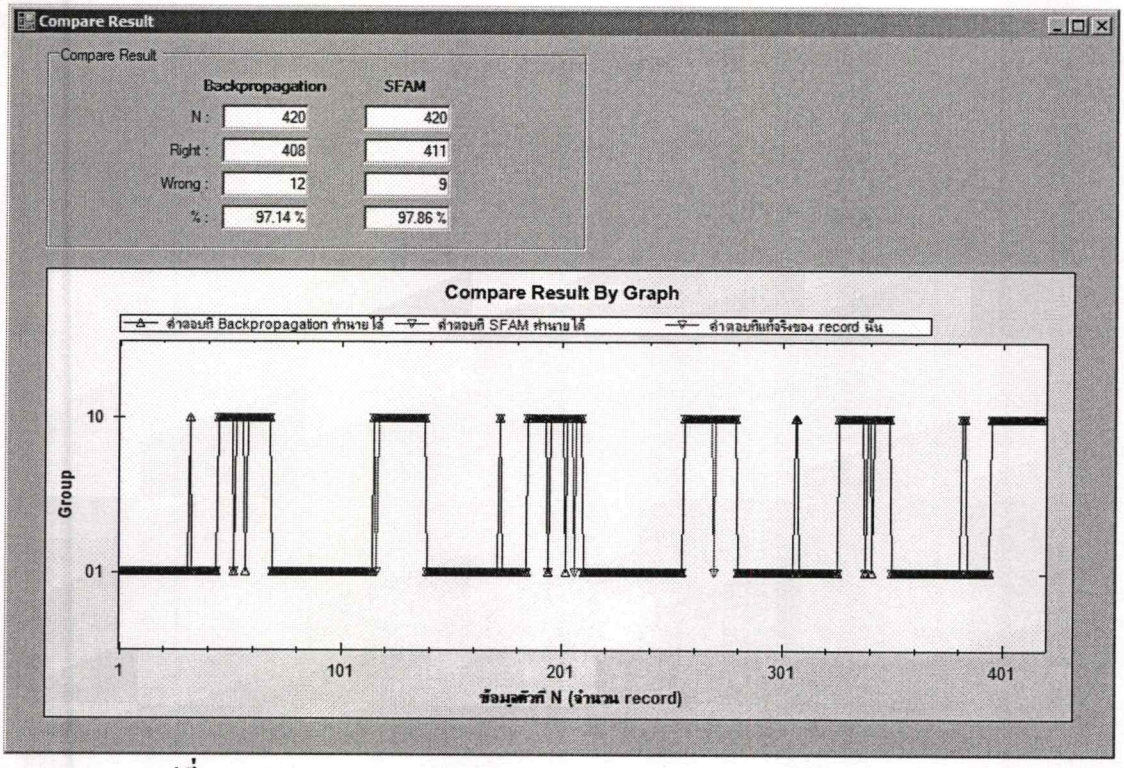
รูปที่ 4.78 ผลการทดลองของ Breast Data 420 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล



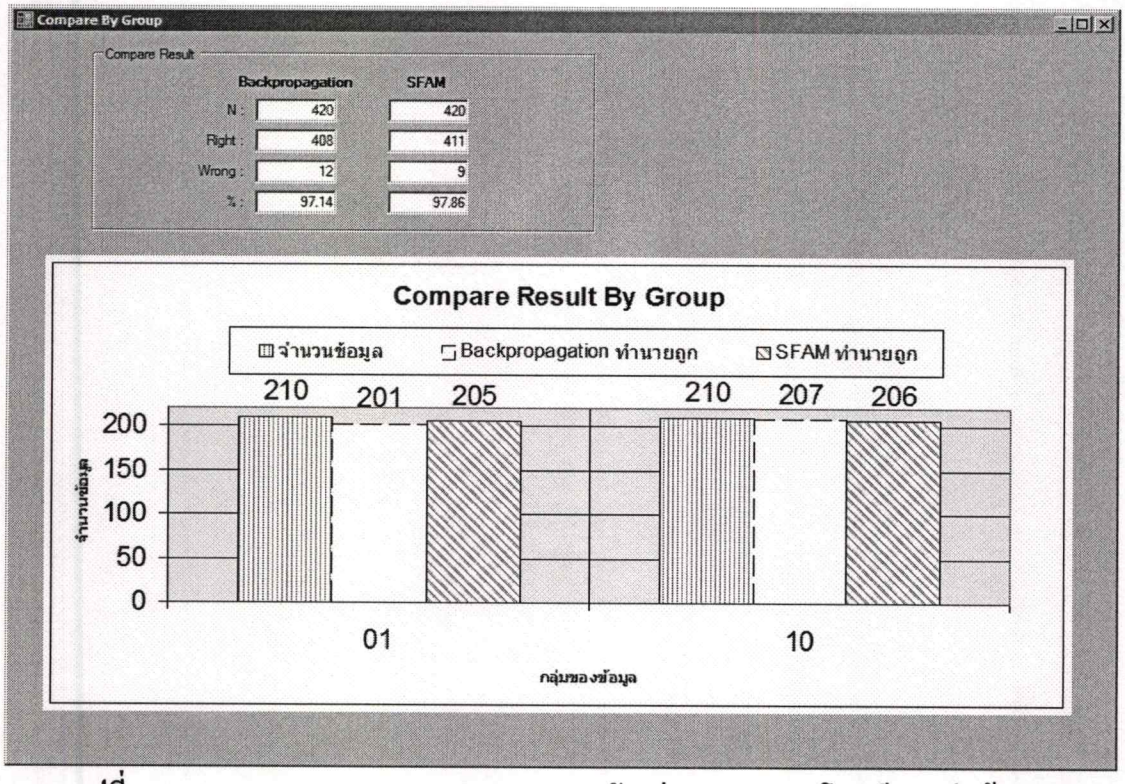
รูปที่ 4.79 ผลการทดลองของ Breast Data จำนวน 420 ตัวอย่าง (10รอบ)



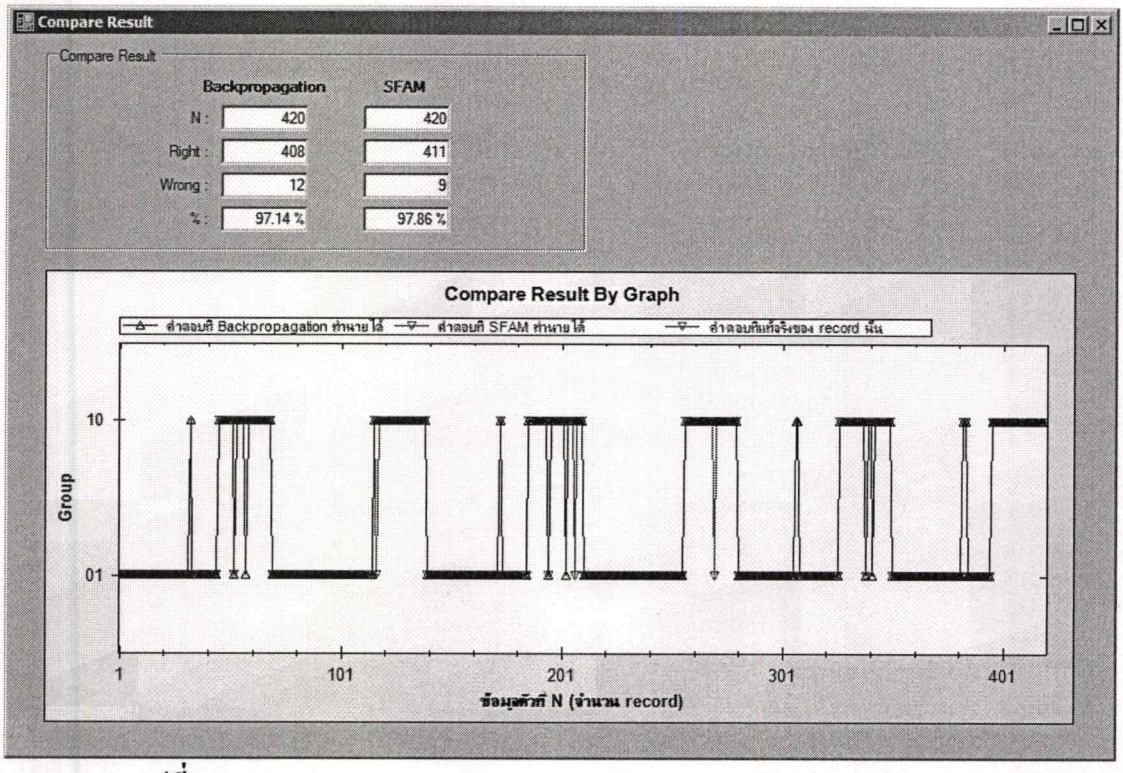
รูปที่ 4.80 ผลการทดลองของ Breast Data 420 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล



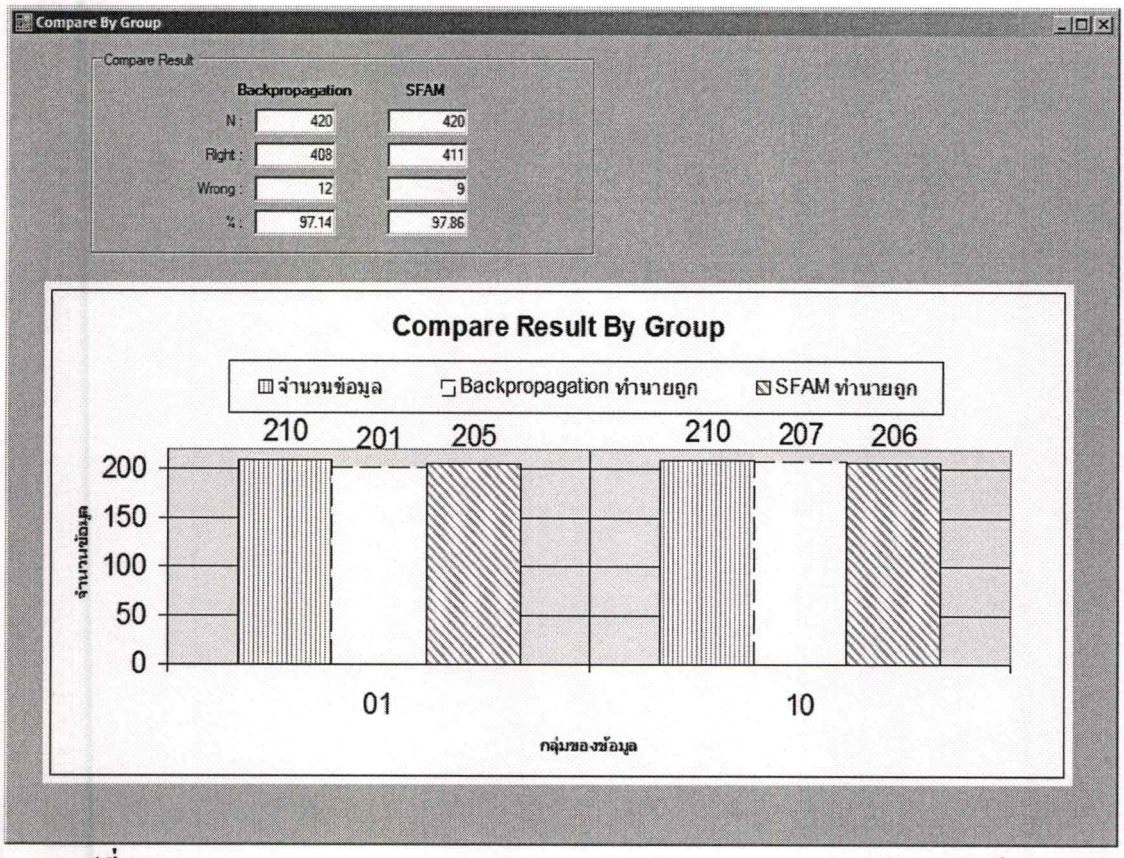
รูปที่ 4.81ผลการทดลองของ Breast Data จำนวน 420 ตัวอย่าง (100รอบ)



รูปที่ 4.82 ผลการทดลองของ Breast Data 420 ตัวอย่าง (100รอบ) โดยเทียบกลุ่มข้อมูล



รูปที่ 4.83 ผลการทดลองของ Breast Data จำนวน 420 ตัวอย่าง (1000รอบ)



รูปที่ 4.84 ผลการทดลองของ Breast Data 420 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล

ต่อมานำข้อมูล Breast Data มาทดสอบโครงข่าย (Test Data)จำนวน 279 ตัวอย่างมาทดสอบ โดยใช้จำนวนข้อมูลในการฝึกสอนโครงข่าย 420 ตัวอย่าง เมื่อนำมาฝึกสอนโครงข่ายแล้วนำมาทดสอบ

โดยกำหนด

-จำนวนตัวอย่างฝึกสอนโครงข่าย 420 ตัวอย่าง (Data Train)

-จำนวนตัวอย่างที่ใช้ทดสอบโครงข่าย 279 ตัวอย่าง (Data Test)

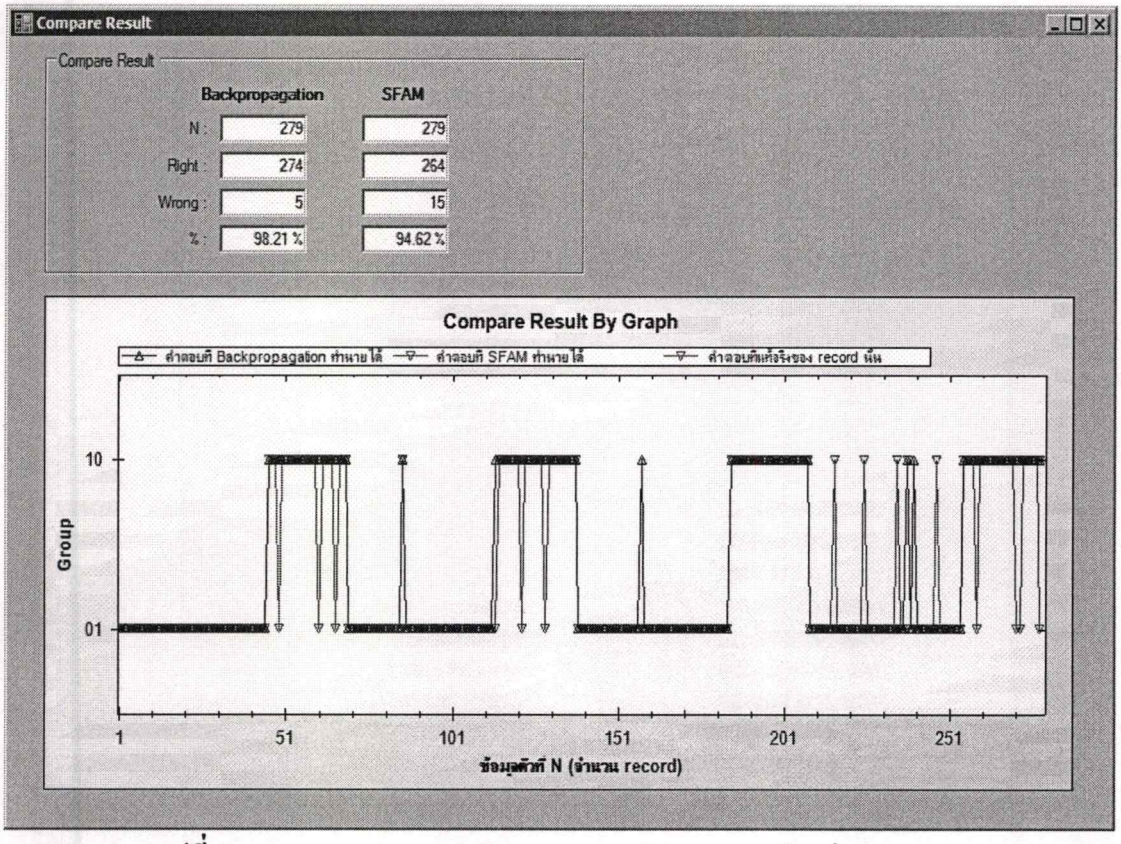
-Backpropagation ทำการวนที่ 1 รอบ, 10 รอบ, 100 รอบ, 1000 รอบ

-SFAM กำหนดค่า Vigilance (ρ) = 0.4

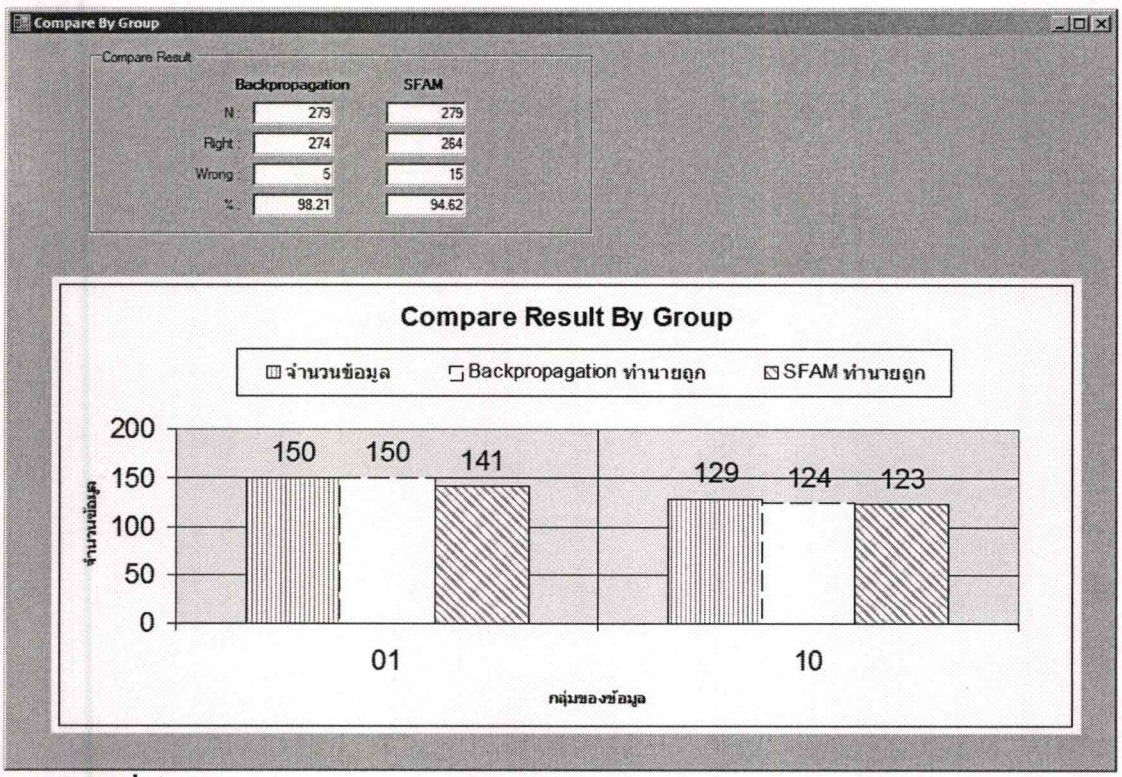
ได้ผลการทดลองดังนี้

ตารางที่ 4.8 แสดงผลลัพธ์การทดลอง Breast Data ด้วยตัวอย่าง (Data Test) 279 ตัวอย่าง

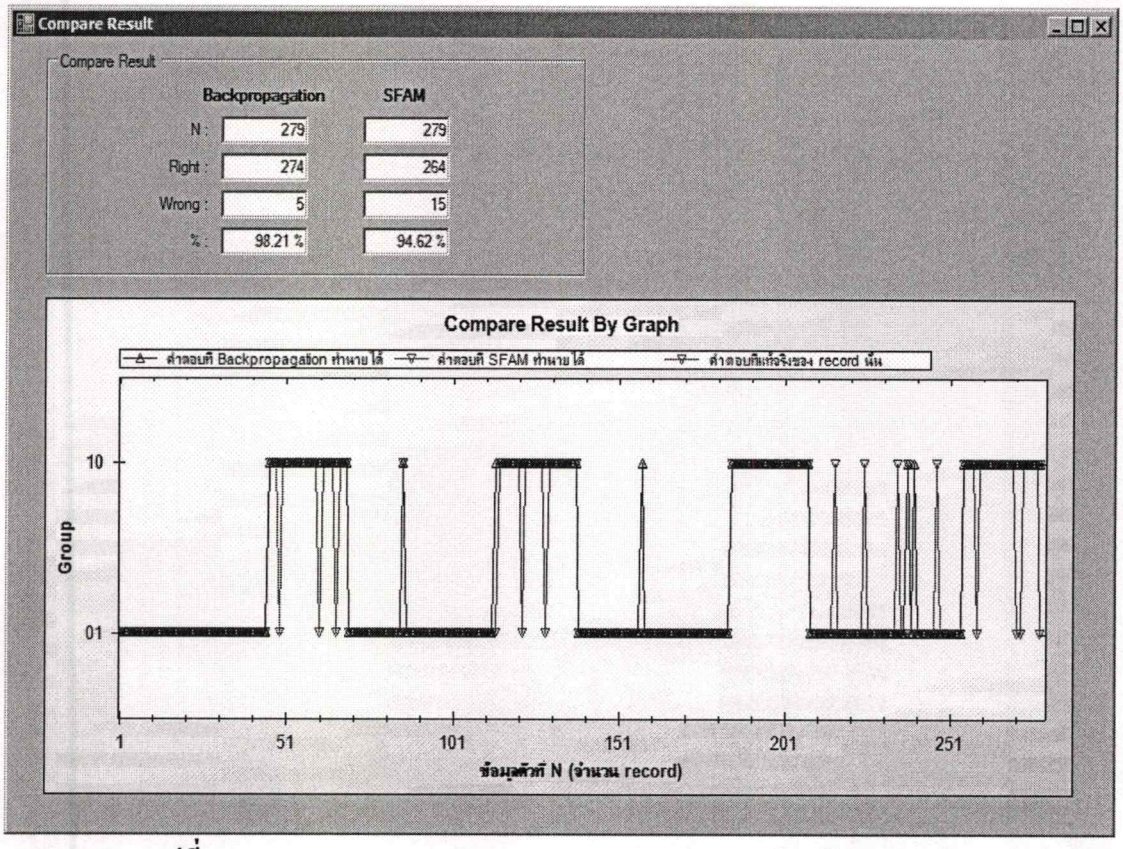
อัลกอริทึม	Backpropagation (1รอบ)	Backpropagation (10รอบ)	Backpropagation (100รอบ)	Backpropagation (1000รอบ)	SFAM
จำนวนตัวอย่าง	279	279	279	279	279
จำนวนถูก	274	274	274	274	264
จำแนกผิด	5	5	5	5	15
เปอร์เซ็นความถูกต้อง	98.21%	98.21%	98.21%	98.21%	94.62%



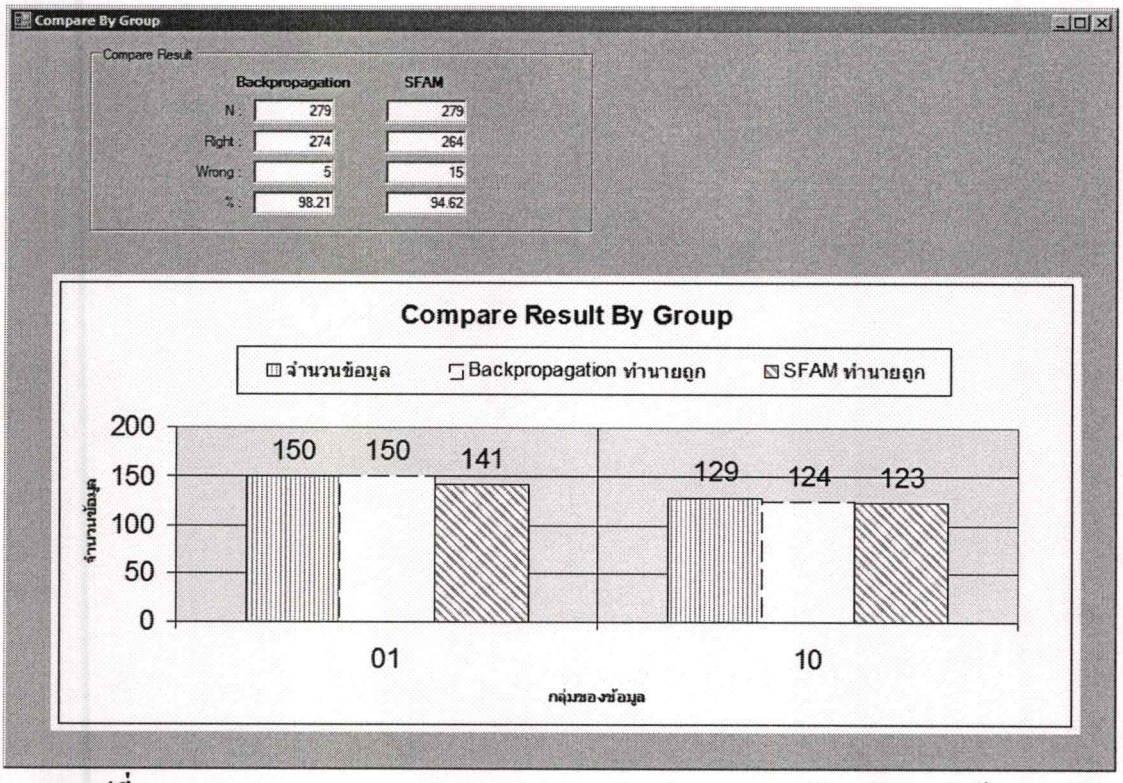
รูปที่ 4.85 ผลการทดลองของ Breast Data จำนวน 279 ตัวอย่าง (1รอบ)



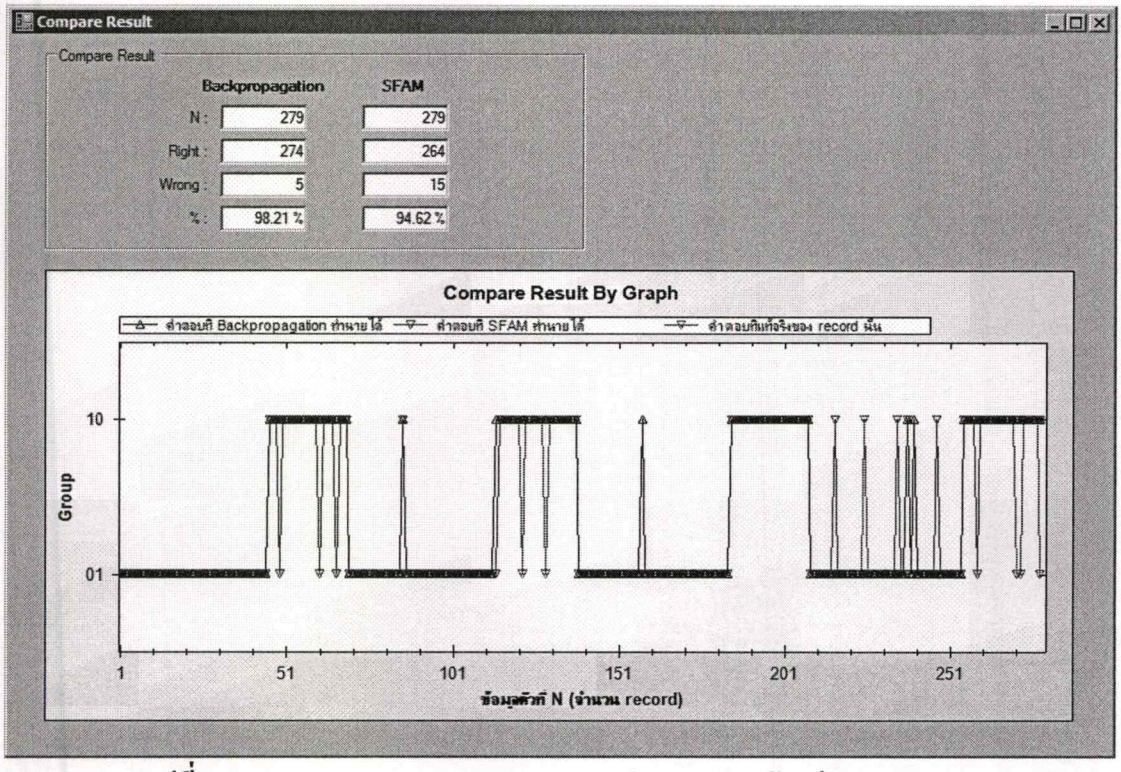
รูปที่ 4.86 ผลการทดลองของ Breast Data 279 ตัวอย่าง (1รอบ) โดยเทียบกลุ่มข้อมูล



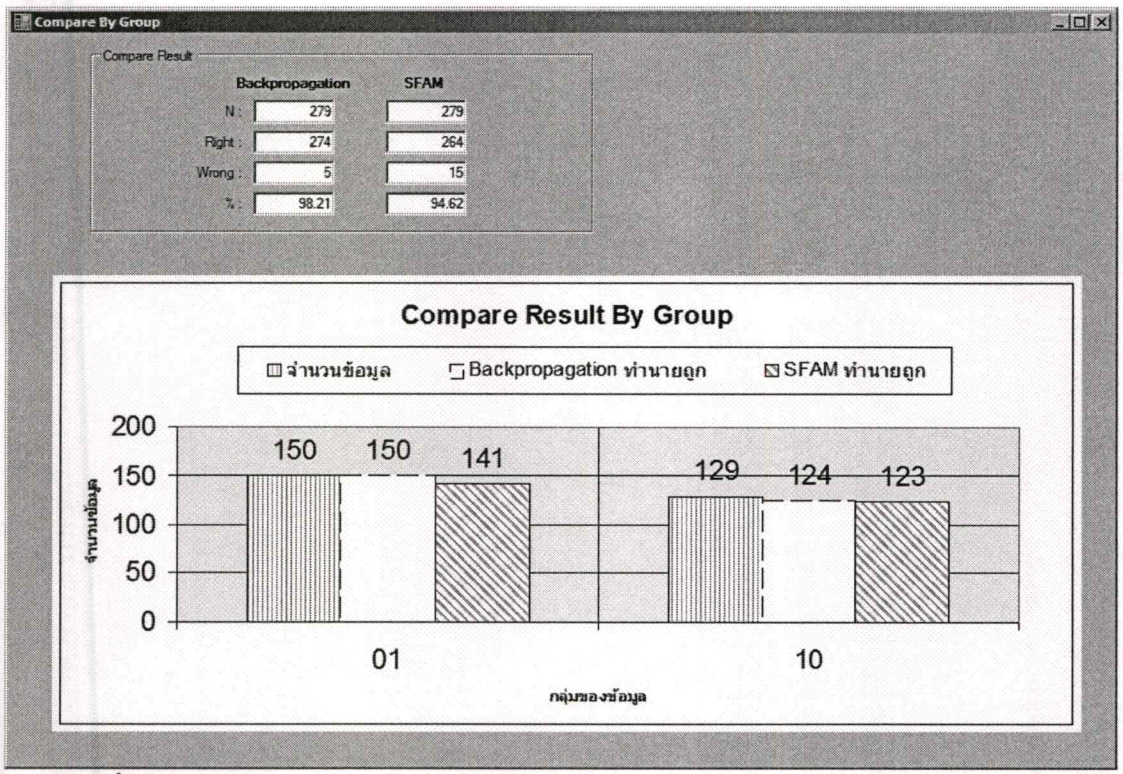
รูปที่ 4.87 ผลการทดลองของ Breast Data จำนวน 279 ตัวอย่าง (10รอบ)



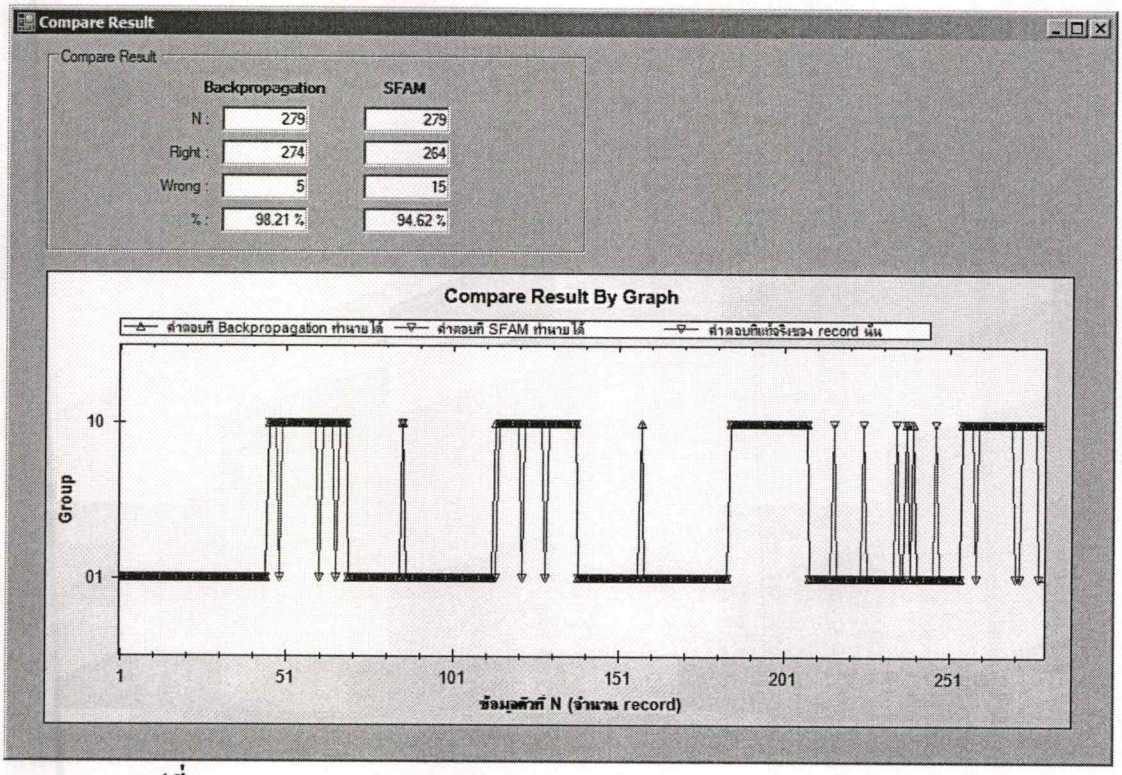
รูปที่ 4.88 ผลการทดลองของ Breast Data 279 ตัวอย่าง (10รอบ) โดยเทียบกลุ่มข้อมูล



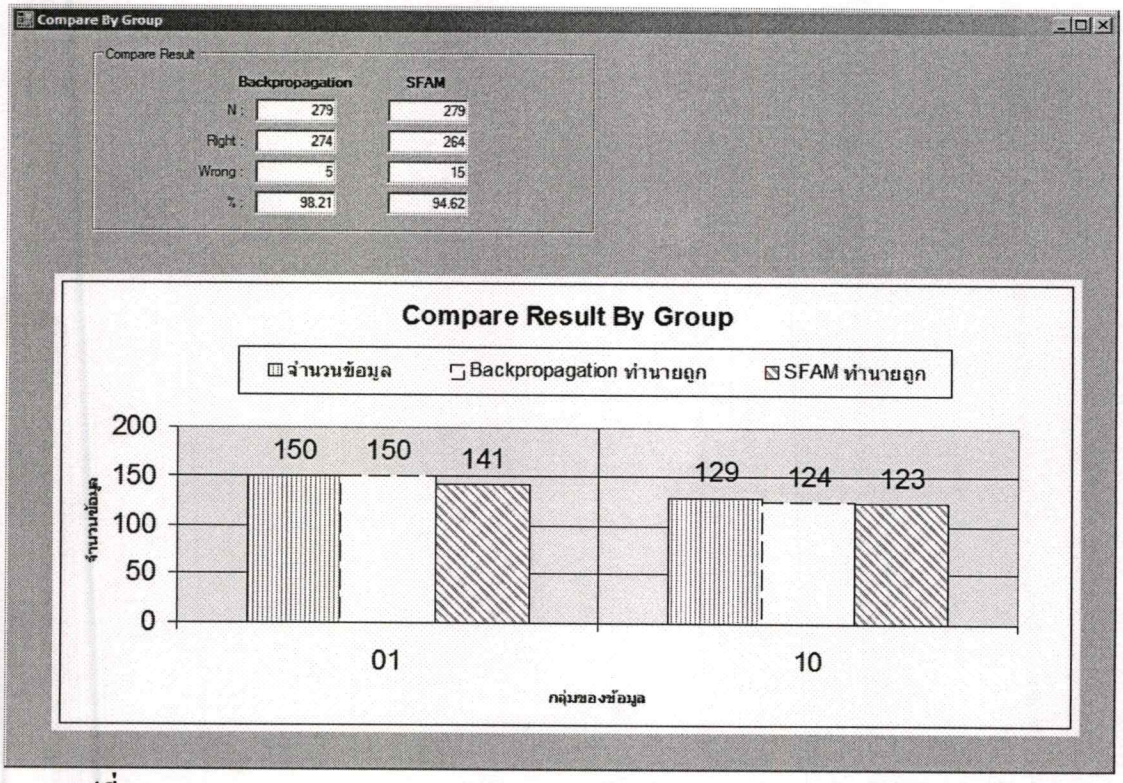
รูปที่ 4.89 ผลการทดลองของ Breast Data จำนวน 279 ตัวอย่าง (100รอบ)



รูปที่ 4.90 ผลการทดลองของ Breast Data 279 ตัวอย่าง (100 รอบ) โดยเทียบกลุ่มข้อมูล



รูปที่ 4.91 ผลการทดลองของ Breast Data จำนวน 279 ตัวอย่าง (1000รอบ)



รูปที่ 4.92 ผลการทดลองของ Breast Data 279 ตัวอย่าง (1000รอบ) โดยเทียบกลุ่มข้อมูล

4.3.5 วิเคราะห์ผลการทดลอง

จากผลการทดลองการจำแนกหมวดหมู่ จากตัวอย่างข้อมูลทั้ง 4 ชุดคือ Iris Data , Lung Cancer, Wine ,Breast ระหว่างอัลกอริทึมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และ ซิมพลิไฟล์ฟัชซี่อาร์ทแมป (Simplified Fuzzy ARTMAP:SFAM) จะเห็นได้ว่า อัลกอริทึมซิมพลิไฟล์ฟัชซี่อาร์ทแมป (Simplified Fuzzy ARTMAP:SFAM) มีการเรียนรู้ได้เร็วกว่า อัลกอริทึมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) แต่อัลกอริมทึมอัลกอริทึมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) มีความสามารถไม่ด้อยกว่าอัลกอริทึมซิมพลิไฟล์ฟัชซี่อาร์ทแมป (Simplified Fuzzy ARTMAP:SFAM) ในการจำแนกข้อมูล แต่ข้อเสียเปรียบของอัลกอริทึมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) คือ ต้องใช้เวลาในการฝึกสอน (Train) โครงข่ายหลายรอบกว่าอัลกอริทึมซิมพลิไฟล์ฟัชซี่อาร์ทแมป (Simplified Fuzzy ARTMAP:SFAM)ถึงจะทำให้มีความสามารถในการจำแนกหมวดหมู่ได้มีประสิทธิภาพ

ข้อสังเกตุจากการทดลองจะเห็นได้ว่าจำนวนรอบในการฝึกสอน(Train) โครงข่ายจะมีผลต่ออัลกอริทึมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) เป็นอย่างมากยิ่งรอบยิ่งมากขึ้นจะทำให้โครงข่ายมีประสิทธภาพในการจำแนกข้อมูลได้แม่นยำมากขึ้นเมื่อดูจากผลการทดลอง

อย่างไรก็ตามจำนวนรอบในการฝึกสอนอาจไม่ใช่ปัจจัยเดียวที่มีผลต่อประสิทธิภาพในการจำแนกข้อมูล การจำแนกข้อมูลนั้นอาจขึ้นอยู่กับหลายๆปัจจัย เช่น รูปแบบของข้อมูล พารามิเตอร์ที่ใช้ จำนวนข้อมูลที่ใช้ฝึกสอน(Train) โครงข่าย เป็นต้น ซึ่งการพิจารณาถึงความสัมพันธ์เหล่านี้มีความยุ่งยาก ใช้เวลาในการค้นหาความสัมพันธ์ค่อนข้างนาน

บทที่ 5
สรุปผลการศึกษาและข้อเสนอแนะ

จากการศึกษาการการจำแนกหมวดหมู่ข้อมูลโดยใช้ อัลกอริทึมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) และซิมพลิไฟล์ฟัชซี่อาร์ทแมป (Simplified FuzzyARTMAP:SFAM) ในโครงการนี้สามารถสรุปผลการดำเนินงานและสรุปผลการทดลองรวมถึงข้อเสนอแนะดังต่อไปนี้

5.1 สรุปผลการศึกษา

จากการศึกษาในโครงการนี้สามารถสรุปผลดำเนินงาน ได้ดังนี้

- ในการทำ Data Mining ซึ่งเป็นกระบวนการที่ใช้วิเคราะห์และค้นหาข้อมูลที่ซ่อนเร้นอยู่ภายในฐานข้อมูล ซึ้งข้อมูลที่ได้จากการทำ Data Mining นั้นมีประโยชน์อย่างมากในการนำไปใช้ ซึ่งในการเตรียมข้อมูลในการทำการจำแนกข้อมูลนั้น ข้อมูลที่นำไปฝึกสอนโครงข่ายนั้นมีความสำคัญมากเพราะว่าถ้าข้อมูลที่นำไปฝึกสอนโครงข่ายมีลักษณะครบถ้วนในทุกลักษณะของข้อมูลจะทำให้ความสามารถในการจำแนกกลุ่มข้อมูลของโครงข่ายมีประสิทธิภาพดีขึ้น
- ในการนำข้อมูลมาฝึกสอนและทดสอบโครงข่ายนั้น ข้อมูลจะต้องทำการนอร์มอลไลเซชันเพื่อให้ข้อมูลเหล่านั้นสามารถทำงานกับโปรแกรมประยุกต์ได้
- ในการเรียนรู้แต่ละครั้งของอัลกอริทึมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) ผลจากการทดลองจะให้ค่าที่ต่างกัน เนื่องจากผลลัพธ์จากการจำแนก จะขึ้นอยู่กับการกำหนดค่าโครงสร้างของโครงข่าย ต่าง ๆ ที่ใช้ในการ Train เช่น ค่าอัตราการเรียนรู้ และค่าจำนวนรอบในการ Train เป็นต้น ดังนั้นการเรียนรู้ต้องอาศัยการเรียนรู้หลาย ๆ ครั้ง เพื่อสังเกตว่าการกำหนดค่าต่าง ๆ นั้น ควรจะกำหนดค่าเหล่านั้นอย่างไร เพื่อให้ได้ประสิทธิภาพในการจำแนกข้อมูลได้ถูกต้องมากที่สุด
- จากผลทดลองอัลกอริทึมซิมพลิไฟล์ฟัชซี่อาร์ทแมป(SimplifiedFuzzyARTMAP:SFAM) นั้นจะมีความสามารถในการเรียนรู้ของโครงข่ายที่รวดเร็วกว่าอัลกอริทึมแบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) ทำให้เหมาะสมกับงานการจำแนกกลุ่มข้อมูลที่มีเวลาค่อนข้างน้อยในการจำแนกข้อมูล

5.2 ประโยชน์ที่ได้รับจากการศึกษาและพัฒนาระบบ

1. ทำให้เข้าใจทฤษฎี และหลักการทำงานของอัลกอริทึมต่างๆในการทำ Classification ข้อมูลมากขึ้น รวมถึงทำให้ทราบวิธีแก้ปัญหาเมื่อเจอปัญหาที่พบบ่อยในการทำ Classification
2. ทำให้ได้โปรแกรมในการเปรียบเทียบประสิทธิภาพในการทำ Classification ที่ทำให้ทราบว่าข้อมูลชุดนั้นเหมาะสมกับอัลกอริทึมแบบใด ทำให้เราสามารถเลือกใช้อัลกอริทึมได้ตามวัตถุประสงค์

5.3 ข้อเสนอแนะ

เพื่อปรับปรุงโปรแกรมประยุกต์นี้ในอนาคต ผู้ศึกษามีความคิดเห็นว่า เพื่อเพิ่มความเร็วให้กับการเรียนรู้แบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back-propagation) ควรค้นหาวิธีที่เหมาะสมเพื่อเพิ่มประสิทธิภาพในการสุ่มค่าน้ำหนัก หรือการปรับค่าน้ำหนักให้มีประสิทธิภาพมากขึ้น เนื่องจากค่าน้ำหนักเริ่มต้นมีผลต่อการทำงานของโปรแกรม และในส่วนของข้อมูลที่จะนำมาฝึกสอนโครงข่ายนั้นข้อมูลควรจะต้องมีความสมบูรณ์ในทุกกลุ่มข้อมูล

บรรณานุกรม

กนกวรรณ เฮงวัฒนะ. 2549. **เทคนิคการจำแนกข้อมูลโดยใช้ฟัซซีออร์แกนไนซิงแมปนิวรอลเน็ตเวิร์ก.** โครงการศึกษา วิทยาศาสตรมหาบัณฑิต. กรุงเทพฯ: บัณฑิตมหาวิทยาลัยสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประพนธ์ เอื้ออานันท์. 2549. **การศึกษาเปรียบเทียบการจัดกลุ่มข้อมูลโดยใช้อัลกอริทึมเค-ฟลานและ ฟัซซี่อาร์ท.** โครงการศึกษา วิทยาศาสตรมหาบัณฑิต. กรุงเทพฯ: บัณฑิตมหาวิทยาลัยสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

อทิตตา ไพบูลย์. 2549. **การศึกษาเปรียบเทียบการจำแนกหมวดหมู่โดยใช้ฟัซซีอาร์ทแมพและอาร์บีเอฟฟัซซี่อาร์ทแมพ.** โครงการศึกษา วิทยาศาสตรมหาบัณฑิต. กรุงเทพฯ: บัณฑิตมหาวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Kate A. Smith and Jatinder N. D. Gupta. 1970. **Neural Networks in Business : Techniques and Applications**. United States of America.

L.X. Wang and Jerry M. Mendel. 1992. "Fuzzy basic function , Universal approximation & Orthogonal least – Squares learning." **IEEE Transaction Neural Network.** 3(5): 807-814.

Michael Negnevitasky , 2002. **Artificial intelligence : a guide to intelligent sysyems .** Addison - Wesley.

Min Liu and Yean Ying Ling. 2003. **Using fuzzy neural network approach to estimate contrators' markup.** [Online]. Available : www.sciencedirect.com.

Martin Hellmann. 2001. **Fuzzy Logic Introduction**. [Online]. Available : http://www.seattlerobotics.org/encoder/mar98/fuz /fl_part1.html.

Steven D. Kaehler. **Fuzzy Logic – An Introduction.** [Online]. Available : http://www.seattlerobotics.org

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายพินิจ เล้ารุ่งเรืองเดช
วัน เดือน ปีเกิด	24 มิถุนายน 2526
วุฒิการศึกษา	วิทยาศาสตรบัณฑิต สาขาวิชาสถิติประยุกต์
สถานที่สำเร็จการศึกษา	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีที่สำเร็จการศึกษา	2549