ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

การเปรียบเทียบครอสเอ็นโทรพีฟังก์ชั่นและเออเลอร์กราเดียร์ฟังก์ชั่น ในการปรับค่าถ่วงน้ำหนักของโครงข่ายประสาทเทียม

COMPARISON OF CROSS ENTROPY FUNCTION AND ERROR GRADIENT FUNCTION ALGORITHM FOR LEARNING OF MULTI-LAYER NEURAL NETWORK

H004884

โดย

รุ่งโรจน์ แสงธรรมชัย

RUNGROJ SAENGTUMCHAI

วน.
ร 636 ก
2550

เลขหมู่.....
เลขทะเบียน.....04884
วัน,เดือน,ปี.....9 ก.ค. 2551

b.19785691
i.....

อาจารย์ที่ปรึกษา

รศ.ดร. อาริต ธรรมโน

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่ 2 ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPARISON OF CROSS ENTROPY FUNCTION AND ERROR GRADIENT FUNCTON ALGORITHM FOR LEARNING OF MULTI-LAYER NEURAL NETWORK

RUNGROJ SAENGTUMCHAI

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/ 2007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2008

FACULTY ON INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	การเปรียบเทียบครอสเอ็นโทรพีฟังก์ชั่นและเออเลอร์กราเดียร์ฟังก์ชั่น ในการปรับค่าถ่วงน้ำหนักของโครงข่ายประสาทเทียม
นักศึกษา	รุ่งโรจน์ แสงธรรมชัย
รหัสนักศึกษา	46066720
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2550
อาจารย์ที่ปรึกษา	รศ.ดร. อาริต ธรรมโน

บทคัดย่อ

โครงการพัฒนาระบบงานนี้ได้พัฒนาขึ้นเนื่องจากปัจจุบันข้อมูลเป็นสิ่งสำคัญต่อการตัดสินใจทางธุรกิจ ในการพัฒนาระบบนี้มีจุดประสงค์เพื่อการนำข้อมูลที่มีเหล่านั้นมาใช้ประโยชน์โดยใช้เทคนิคดาต้าไมนิ่ง (Data Mining) โดยศึกษาแบบจำลองโครงข่ายประสาทเทียม (Neural Network) ซึ่งลักษณะเป็นโมเดลทางคณิตศาสตร์สำหรับการประมวลผลสารสนเทศด้วยการประมวลผลแบบคอนเน็คชันนิสต์ (Connectionist) ที่มีการทำงานจำลองแบบการเรียนรู้ของโครงข่ายประสาทของมนุษย์ โดยการศึกษาครอสเอ็นโทรพีฟังก์ชัน (Cross entropy function) เพื่อใช้ในการฝึกฝนโครงข่ายประสาทเทียมสำหรับการจัดหมวดหมู่โดยใช้ในรูปแบบโครงสร้างแบบการประมวลผลโครงข่ายแบบทิศทางเดียว ที่มีการเรียนรู้แบบแบคพรอพพาเกชันเน็ทเวิร์ค (Back propagation network) โดยเปรียบเทียบกับการเรียนรู้ที่ใช้อัลกอริทึม Error gradient function เพื่อให้เห็นประสิทธิภาพการทำงานเพื่อนำไปใช้งานต่อไป จากผลการศึกษาฉบับนี้จะได้นำไปประยุกต์ในทางธุรกิจต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Comparison of Cross Entropy Function and Error Gradient Function Algorithm for learning of multi-layer Neural Network
Student	Mr.Rungroj Saengtumchai
Student ID.	46066720
Degree	Master of Science
Program	Information Science
Academic Year	2007
Advisor	Assoc.Prof. Dr.Arit Thammano

ABSTRACT

This system development project is that data impact to business of decision. In this project will receive business data to Data mining techniques. One of many techniques is Neural Network ,a mathematic model for information proceeding by a Connectionist pattern ,simulating to human brain which is compose of nerve cell "neurons" by studying Cross entropy function for training classification feed forward Neural Network ,learning Back propagation network pattern by comparing Error gradient function for Learning algorithms in order to find better performance .Finally, this studying project will apply to business .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานนี้สำเร็จได้ ด้วยความกรุณาของ รศ.ดร. อาริต ธรรมโน อาจารย์ที่ปรึกษาโครงการ ซึ่งได้ให้คำปรึกษา ข้อชี้แนะ และความช่วยเหลือจนกระทั่งลุล่วงไปด้วยดี ขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ขอกราบพระคุณคณาจารย์คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกๆท่านที่ให้ความรู้ คำแนะนำให้แก่ข้าพเจ้าตลอดการศึกษาที่ผ่านมา และขอขอบคุณเจ้าหน้าที่ของคณะเทคโนโลยีสารสนเทศทุกท่านที่ช่วยเหลืออำนวยความสะดวก และให้คำแนะนำต่างๆ

ขอขอบคุณ เอื้ออนงค์ ที่เอื้อเฟื้อข้อมูลที่นำมาใช้ในการพัฒนาระบบงานในครั้งนี้ จนสำเร็จด้วยดี

ขอขอบคุณเพื่อนๆ IS16.1 ทุกคนที่คอยช่วยเหลือให้คำแนะนำ ให้ความรู้ตอบข้อซักถามต่างๆ และคอยให้กำลังใจตลอดมา

สุดท้ายนี้ขอกราบขอบพระคุณพ่อคุณแม่ ที่คอยเป็นกำลังใจและให้การสนับสนุนในทุกเรื่องๆ ทำให้สามารถทำโครงการพัฒนาระบบงานนี้สำเร็จลุล่วงลงได้ในที่สุด

รุ่งโรจน์ แสงธรรมชัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... I

บทคัดย่อภาษาอังกฤษ..... II

กิตติกรรมประกาศ..... III

สารบัญ..... IV

สารบัญตาราง..... VI

สารบัญรูป..... VII

บทที่ 1 บทนำ..... 1

1.1 ความเป็นมาและความสำคัญของปัญหา..... 1

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา..... 1

1.3 ขอบเขตของการศึกษา..... 2

1.4 ขั้นตอนการศึกษา..... 2

1.5 ผลที่คาดว่าจะได้รับ..... 3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง..... 4

2.1 โครงข่ายประสาทเทียม..... 4

2.2 โครงสร้างโครงข่ายประสาทเทียม..... 5

2.3 โครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ..... 6

2.4 Back-propagation : Error gradient..... 7

2.5 Back-propagation : Cross entropy function..... 11

2.6 ตัวอย่างการคำนวณ..... 16

บทที่ 3 วิเคราะห์และออกแบบระบบ..... 22

3.1 Usecase Diagram..... 22

3.2 Activity Diagram..... 26

บทที่ 4 การออกแบบส่วนต่อประสานกับผู้ใช้..... 30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

4.1 เครื่องมือและภาษาที่ใช้ในการพัฒนาระบบ.....30
4.2 รายละเอียดการทำงานของโปรแกรม.....30
4.3 ค่าพารามิเตอร์.....33
4.4 ผลการทดลองและสรุปผลการทดลอง.....34
4.5 การทำงานของโปรแกรม.....46

บทที่ 5 สรุปผลการศึกษาและข้อเสนอแนะ.....51
5.1 สรุปการพัฒนาระบบงาน.....51
5.2 ประโยชน์ที่ได้รับจากการพัฒนาระบบ.....52
5.3 ปัญหาและอุปสรรค.....52

บรรณานุกรม.....53

ประวัติผู้เขียน.....54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

3.1 แสดงรายละเอียดของ Use Case ที่ 1 23

3.2 แสดงรายละเอียดของ Use Case ที่ 2 23

3.3 แสดงรายละเอียดของ Use Case ที่ 3 24

3.4 แสดงรายละเอียดของ Use Case ที่ 4 25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 โครงสร้างของเซลล์ประสาท..4

2.2 การทำงานเปรียบเทียบเซลล์ประสาทเทียม.....5

2.3 ลักษณะโครงสร้างการทำงาน Back-propagation.....7

2.4 รูปโครงสร้างโครงข่ายประสาทเทียม.....17

3.1 Use Case Diagram ของระบบ Neural Network 26

3.2 Activity Diagram ของ Normalized Data.....27

3.3 Activity Diagram ของ Process Cross entropy function.....28

3.4 Activity Diagram ของ Process Error gradient function.....29

4.1 ตัวอย่างข้อมูลที่นำเข้าระบบ..... 31

4.2 ตัวอย่างไฟล์ที่ผ่านการ Normalized Data 33

4.3 หน้าจอเพื่อผู้ใช้ป้อนพารามิเตอร์..... 34

4.4 ผลการทดสอบ Class 0..... 36

4.5 ผลการทดสอบ Class 1..... 36

4.6 ผลการทดสอบ Class 2..... 36

4.7 ผลการทดสอบ Class 0..... 37

4.8 ผลการทดสอบ Class 1..... 37

4.9 ผลการทดสอบ Class 2..... 37

4.10 แสดงประสิทธิภาพการทำงานของระบบ..... 38

4.11 กราฟแสดงค่าความผิดพลาด..... 39

4.12 ผลการทดสอบ Class 0 โดยใช้ Cross Entropy Function..... 41

4.13 ผลการทดสอบ Class 1 โดยใช้ Cross Entropy Function..... 42

4.14 ผลการทดสอบ Class 0 โดยใช้ Error Gradient Function Test..... 43

4.15 ผลการทดสอบ Class 1 โดยใช้ Error Gradient Function Test..... 44

4.16 แสดงการเปรียบเทียบการทำงานของสองอัลกอริทึม..... 44

4.17 แสดงในส่วนของ RMSE ของสองอัลกอริทึม..... 45

4.18 แสดงโปรแกรม..... 46

4.19 การเปิดไฟล์..... 46

4.20 โปรแกรมหน้าหลัก..... 47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

4.21 หน้าจอแสดงการทำงานNormalized..... 48
4.22 หน้าจอแสดงการทำงาน..... 49
4.23 หน้าจอแสดงส่วนของการทดสอบ..... 49
4.24 หน้าจอแสดงส่วนการทำงาน Gradient..... 49
4.25 หน้าจอหลักแสดงการเปรียบเทียบ..... 50
4.26 หน้าจอหลักแสดงกราฟเส้น..... 50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันการดำเนินธุรกิจไม่ว่าจะขนาดธุรกิจมีขนาดเท่าใด ความสำคัญที่ไม่สามารถจะละเลยได้เลยนั้นได้แก่ ข้อมูล ซึ่งจะเห็นได้ว่าการทำงานทางธุรกิจนั้น ข้อมูลเป็นสิ่งจำเป็นในการดำเนินงานต่างๆและใช้ในการประกอบการตัดสินใจในเชิงธุรกิจ เมื่อมีข้อมูลที่ใช้ในการตัดสินใจมากจึงจำเป็นต้องมีวิธีการที่ช่วยทำให้เกิดประสิทธิภาพ ทันเวลาและเป็นข้อมูลที่เหมาะสมต่อการตัดสินใจได้อย่างรวดเร็วจึงส่งผลให้เป็นผู้ที่ได้เปรียบในเชิงการค้า เมื่อเล็งเห็นประโยชน์และความสำคัญต่อข้อมูลที่มีจำนวนมากจึงมีความจำเป็นที่ต้องใช้เทคโนโลยีมาช่วยในการตัดสินใจในทางธุรกิจและการทำงาน เพื่อให้เกิดความถูกต้องและมีความรวดเร็วและน่าเชื่อถือรวมทั้งเป็นข้อมูลในเชิงสารสนเทศที่มีประสิทธิภาพ ดังนั้นสารสนเทศแขนงหนึ่งที่มีประโยชน์ในด้านการช่วยในการตัดสินใจที่เรียกว่า Data Mining ซึ่งมีหลายเทคนิคและประโยชน์ประกาศหนึ่งที่นำมาศึกษาและสามารถนำมาใช้ปรับเพื่อแก้ไขปัญหาได้แก่ ในส่วนของ Classification ซึ่งมีบทบาทและความสำคัญอย่างมากในเชิงธุรกิจ เช่น การจัดกลุ่มหรือแยกประเภทของกลุ่มลูกค้า , การจำแนกประเภทของวัตถุ หรือ อาจจะประยุกต์ใช้ในการจำแนกใบหน้า เป็นต้น ซึ่งการทำงานได้เลือกการทำงานในส่วนของ การทำงานของโครงข่ายประสาทเทียมหรือที่เรียกกันว่า Neural Network ซึ่งมีลักษณะการทำงานคล้ายการทำงานหรือเรียกได้ว่า เลียนแบบการทำงานของสมองเป็นลักษณะของการทำงานโดยการใช้หลักคณิตศาสตร์ในการคำนวณ โดยลักษณะการทำงานไม่ยุ่งยากมากนะ โดยการทำงานนั้นมีหลายกอลิทึมที่ใช้ในการเรียนรู้ของระบบเพื่อให้การฝึกฝนของระบบทำงานได้อย่างถูกต้องและมีประสิทธิภาพ จึงได้เลือกการเรียนรู้ที่มีการทำงานแบบแพร่กระจายย้อนกลับ ที่เรียกว่า Back Propagation โดยอัลกอริทึมที่เลือกใช้ได้แก่ Cross entropy function และได้นำมาเปรียบกับอีกหนึ่งอัลกอริทึมซึ่งได้แก่ Gradient Error Function เพื่อให้เห็นความแตกต่างของประสิทธิภาพในการทำงานของทั้งสองอัลกอริทึมเพื่อนำไปประยุกต์ใช้งานต่อไป

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

การศึกษาโครงงานนี้มีวัตถุประสงค์ดังนี้

1. เพื่อศึกษาการทำงานของเทคนิคต่างๆที่ใช้ในการจัดกลุ่ม (Classification Technique)
2. เพื่อศึกษาการทำงานของโครงข่ายประสาทเทียมแบบแพร่กระจายย้อนกลับ โดยใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัลกอริทึมฟังก์ชั่นครอสเอ็นโทรพี (Cross Entropy Function)

3. เพื่อศึกษาการทำงานของโครงข่ายประสาทเทียมแบบแพร่กระจายแบบย้อยกลับโดยใช้อัลกอริทึม Error Gradient Function
4. พัฒนาโปรแกรมเพื่อศึกษาเปรียบเทียบอัลกอริทึมที่ใช้ในการปรับค่าถ่วงน้ำหนักของโครงข่ายประสาทเทียมโดยนำไปใช้ให้เกิดประโยชน์สูงสุด

1.3 ขอบเขตการศึกษา

การศึกษาโครงงานนี้ขอบเขตในการศึกษาพัฒนาโครงข่ายประสาทเทียมโดยใช้การเรียนรู้แบบแพร่กระจายย้อนกลับโดยใช้อัลกอริทึมในการเรียนรู้สองชนิดเพื่อเปรียบเทียบในเรื่องของประสิทธิภาพและเวลาในการทำงาน โดยมีหัวข้อหลักดังนี้

1. โครงข่ายประสาทเทียม (Artificial Neural Network)
2. โครงข่ายประสาทเทียมแบบ feed forward technique
3. ลักษณะการเรียนรู้แบบแพร่กระจายแบบย้อนกลับโดยใช้อัลกอริทึม Cross Entropy Function
4. ลักษณะการเรียนรู้แบบแพร่กระจายแบบย้อนกลับโดยใช้อัลกอริทึม Error Gradient Function
5. เปรียบเทียบการทำงานของอัลกอริทึมที่เลือกใช้ในการพัฒนาโครงงาน

1.4 ขั้นตอนของการศึกษา

การศึกษาโครงการพัฒนาระบบงานนี้มีขั้นตอนและวิธีการดำเนินงาน โดยโครงงานฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ ขอบเขตของการวิจัย และขั้นตอนการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการพัฒนาโครงงาน และพื้นฐานของการทำงานของโครงข่ายประสาทเทียมที่มีการทำงานแบบ feed forward technique และการเรียนรู้แบบแพร่กระจายย้อนกลับโดยเลือกอัลกอริทึมมาใช้ในการทำงานได้แก่ Cross Entropy Function และ Error Gradient Function แล้วนำมาเปรียบเทียบให้เห็นถึงประสิทธิภาพและเวลาที่ใช้ในการทำงานของแต่ละอัลกอริทึม

บทที่ 3 กล่าวถึงการออกแบบระบบ เพื่อนำมาใช้ในการพัฒนาโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 กล่าวถึงการพัฒนาให้ระบบสามารถใช้งานได้โดยนำทฤษฎีที่เกี่ยวข้องมาประยุกต์ เพื่อให้เกิดระบบที่สามารถใช้งานได้จริง และเปรียบเทียบให้เห็นถึงประสิทธิภาพของอัลกอริทึมที่นำมาประยุกต์ใช้งาน

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะ

1.5 ผลที่คาดว่าจะได้รับ

การศึกษาโครงการพัฒนาระบบงานนี้มีผลที่คาดว่าจะได้รับคือ

1. เพื่อเข้าใจหลักการและเทคนิคของโครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ
2. เพื่อเข้าใจหลักการและเทคนิคการเรียนรู้แบบแพร่ย้อนกลับโดยใช้อัลกอริทึม Error Gradient Function
3. เพื่อเข้าใจหลักการและเทคนิคการเรียนรู้แบบแพร่ย้อนกลับโดยใช้อัลกอริทึม Cross Entropy Function
4. เพื่อให้เข้าใจหลักการและเทคนิคของโปรแกรมภาษา Java ที่ใช้ในการพัฒนาโปรแกรมโครงข่ายประสาทเทียม
5. เพื่อนำระบบที่พัฒนาไปใช้ให้เกิดประโยชน์สูงสุด

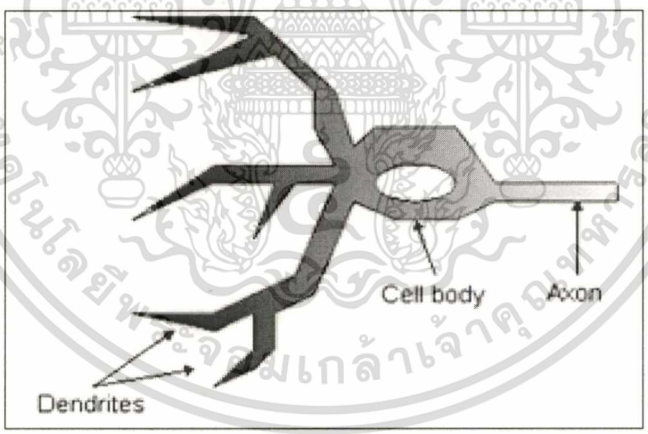
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

ในหัวข้อนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องที่ใช้ในการทำโครงงานพัฒนาระบบ โดยจะกล่าวถึงรายละเอียดเกี่ยวกับ ทฤษฎีโครงข่ายประสาทเทียม ทฤษฎีโครงข่ายประสาทเทียมแบบแพร่กระจายย้อนกลับ (Backpropagation) อัลกอริทึมที่ศึกษาเพื่อใช้ในการปรับค่าถ่วงน้ำหนักของระบบ Error gradient function และ Cross entropy function

2.1 โครงข่ายประสาทเทียม (Neural network) (Setiono.,2003)

โครงข่ายประสาทเทียม มีหลักการโดยอ้างอิงถึงการทำงานของสมองมนุษย์ ในสมอง ซึ่งประกอบด้วย เซลล์ประสาท หรือ "นิวรอน" (Neurons) และ จุดประสานประสาท (Synapses) แต่ละเซลล์ประสาทประกอบด้วยปลายในการรับกระแสประสาท เรียกว่า "เดนไดรท์" (Dendrite) ซึ่งเป็น input และปลายในการส่งกระแสประสาทเรียกว่า "แอคซอน" (Axon) ซึ่งเป็นเหมือน Output ของเซลล์ ดังแสดงดังรูปที่ 2.1



รูปที่ 2.1 โครงสร้างของเซลล์ประสาท

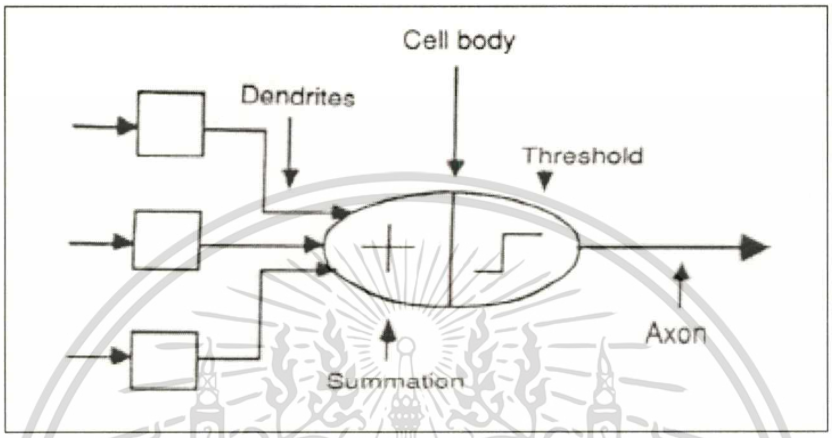
ซึ่งเทียบกับโครงข่ายการทำงานประสาทเทียมได้โดยการแทนด้วยโมเดลทางคณิตศาสตร์ โดยเปรียบเทียบกับเซลล์ประสาทของมนุษย์ที่แสดงให้เห็นดังรูปที่ 2.1 โดยโครงข่ายประสาทเทียมจะประกอบด้วย 3 ส่วนหลักได้แก่

- ชั้นรับข้อมูล (Input Layer) ทำหน้าที่รับข้อมูลดิบเพื่อเข้าสู่โครงข่ายประสาทเทียม และกระจายข้อมูลออกให้แก่โหนดทุกโหนดที่เชื่อมต่อในชั้นซ่อนต่อไป
- ชั้นซ่อน (Hidden Layer) เป็นชั้นที่ใช้ในการคำนวณโดยทำหน้าที่รับข้อมูลดิบจากชั้นรับข้อมูลทุกโหนดและค่าถ่วงน้ำหนักส่วนที่เชื่อมจากชั้นรับข้อมูลเข้ามายังชั้นซ่อนเพื่อส่งให้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชั้นข้อมูลออกต่อไป เนื่องจากชั้นนี้จะไม่เห็นในส่วนของข้อมูลเข้าและข้อมูลออกในชั้นนี้จึงถูกเรียกว่า ชั้นซ่อน ในส่วนการทำงานของชั้นซ่อนนั้นสามารถมีมากกว่าหนึ่งชั้นได้ส่วนใหญ่มีเพียง 3 ชั้นเท่านั้น เพราะการเพิ่มจำนวนชั้นของชั้นซ่อนที่มากขึ้นหมายถึงการคำนวณที่มากขึ้น

- ชั้นข้อมูลออก (Output Layer) ใช้แสดงผลลัพธ์ที่เกิดขึ้นในโครงข่ายประสาทเทียม



รูปที่ 2.2. การทำงานเปรียบเทียบเซลล์ประสาทมนุษย์

2.2 โครงสร้างของโครงข่ายประสาทเทียม (Neural Network)

ในโครงข่ายประสาทเทียมจะแบ่งการทำงานได้เป็นการเชื่อมต่อของโหนดภายในโครงข่ายซึ่งหมายถึงการส่งข้อมูลไปในโครงข่ายประสาทเทียมและลักษณะของการเรียนรู้ของระบบซึ่งการทำงานของระบบจะกล่าวถึงระบบที่มีการเรียนรู้ด้วยตัวระบบเองและระบบที่มีการให้ข้อมูลไว้เพื่อสอนแก่ระบบ ซึ่งลักษณะของการส่งข้อมูลภายในโครงข่ายประสาทเทียมและลักษณะของการเรียนรู้จะสัมพันธ์กัน โดยลักษณะของการส่งข้อมูลภายในของโครงข่ายประสาทเทียมนั้นมีด้วยกัน 2 รูปแบบ คือ

2.2.1 Feed-forward Network

ลักษณะการไหลของข้อมูลเป็นการไหลของข้อมูลไปในทิศทางเดียวกัน โดยการเชื่อมโยงโหนดเป็นการเชื่อมโยงโหนดระหว่างชั้นเท่านั้นในระดับชั้นเดียวกันไม่มีการเชื่อมโยงกัน โดยการส่งข้อมูลไปข้างหน้า (Feed-forward Network) โดยข้อมูลจากชั้นรับข้อมูลทุกโหนดไปยังโหนดในชั้นซ่อนและส่งต่อไปยังชั้นข้อมูลออกต่อไปดังนั้น Input Layer จะเชื่อมต่อกับ Hidden Layer และ Hidden Layer เชื่อมต่อกับ Output Layer เสมอ เพื่อแสดงผลลัพธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 Feedback Network

แสดงลักษณะของการไหลของข้อมูลสามารถไหลย้อนกลับได้ อาจมีลักษณะของการไหลของข้อมูลแบบ Recursive คือโครงสร้างแบบมีการป้อนกลับหรือเรียกว่ารีเคอร์เร็นท์นิวรอลเน็ตเวิร์กนั้น มีการนำค่าเอาต์พุตที่ได้จากนิวรอนใดๆ นั้นมาย้อนกลับเป็นค่าอินพุตให้กับเครือข่ายได้

ในส่วนรูปแบบของการเรียนรู้ในโครงข่ายประสาทเทียมมีลักษณะการเรียนรู้ 2 รูปแบบ

2.2.3 Supervised Learning

เป็นการเรียนแบบที่มีลักษณะของการตรวจคำตอบเพื่อให้โครงข่ายประสาทเทียมปรับตัวชุดข้อมูลที่ใช้สอนโครงข่ายประสาทเทียมจะมีคำตอบไว้คอยตรวจดูว่าโครงข่ายประสาทเทียมให้คำตอบที่ถูกหรือไม่ ถ้าตอบไม่ถูกโครงข่ายประสาทเทียมก็จะปรับตัวเองเพื่อให้ได้คำตอบที่ดีขึ้น

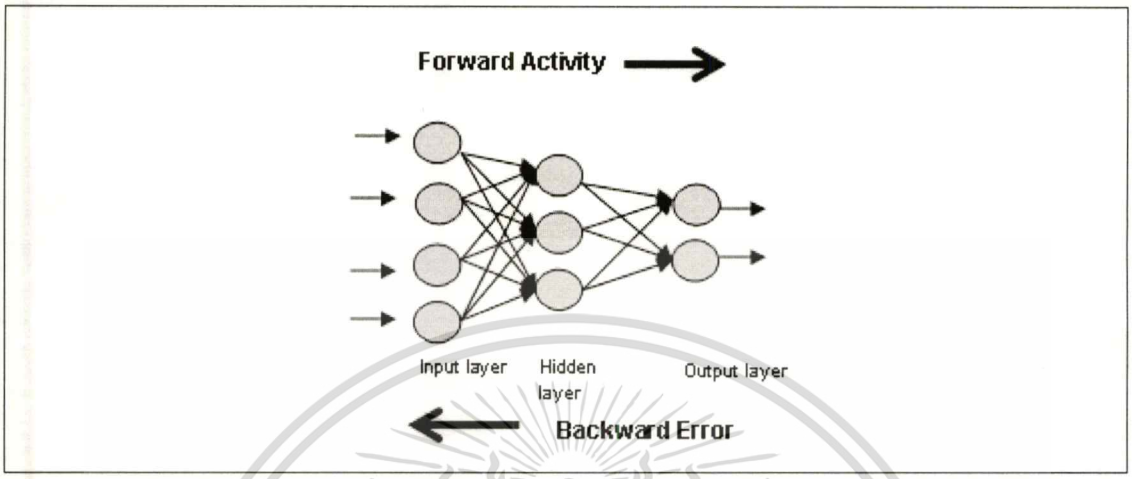
2.2.4 Unsupervised Learning

เป็นการเรียนแบบไม่มีผู้แนะนำไม่มีการตรวจคำตอบว่าถูกหรือผิด โครงข่ายประสาทเทียมจะจัดเรียงโครงสร้างด้วยตัวเองโดยหาความสัมพันธ์กันของเซตของข้อมูล และแยกข้อมูลตามความสัมพันธ์

2.3 โครงข่ายประสาทเทียมแบบแพร่กระจายย้อนกลับ (Back-propagation Neural Network)

โครงข่ายประสาทเทียมแบบแพร่ย้อนกลับเป็นลักษณะการทำงานของโครงข่ายหลายชั้น ซึ่งอย่างน้อยประกอบด้วย 3 ชั้นคือ ชั้นรับข้อมูล (Input Layer) ชั้นซ่อน (Hidden Layer) และชั้นข้อมูลออก (Output Layer) ซึ่งชั้นซ่อนจะมีมากกว่า 1 ชั้นได้โดยการใช้การเรียนรู้แบบชี้นำในการทำงาน ดังแสดงดังรูปที่ 2.3 ได้กล่าวถึงการเรียนรู้โครงข่ายประสาทเทียมแบบแพร่ย้อนกลับใน 2 รูปแบบได้แก่โครงข่ายประสาทเทียมแพร่กระจายย้อนกลับโดยใช้ Error Gradient Function และโครงข่ายประสาทเทียมแพร่ย้อนกลับโดยใช้เป็น Cross Entropy Function โดยลักษณะที่แตกต่างกันอยู่ในส่วนของ Error Function ที่ใช้แตกต่างกัน ในโครงงานฉบับนี้จะกล่าวทั้ง 2 แบบเพื่อให้สามารถเข้าใจได้มากยิ่งขึ้น โดยการทำงานของระบบโครงข่ายประสาทเทียมนั้นมีหลักการทำงาน 2 ส่วนได้แก่ Forward Activity และ Backward Error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ลักษณะโครงสร้างการทำงาน Back-propagation

2.4 Back-propagation: Error gradient

โดยการเรียนรู้นั้นเกิดขึ้นโดยให้ข้อมูลเพื่อฝึกฝนแก่ระบบโดยค่าสุ่มค่าถ่วงน้ำหนักให้แก่ระบบ ระบบทำการคำนวณเทียบกับค่าที่เป้าหมายที่ต้องการ หากผิดพลาดจะปรับค่าถ่วงน้ำหนักเพื่อลดค่าผิดพลาดลง โดยลักษณะของการเรียนรู้ของ Back-propagation ใดๆ ก็ตามจะประกอบไปด้วย Activation Function , Learning Rate การเรียนรู้แบบแพร่กระจายย้อนกลับนั้นจะใช้การเชื่อมต่อของโหนดจะเชื่อมถึงกันหมด (fully connected) การส่งข้อมูลจะส่งไปข้างหน้าไปยังชั้นข้อมูลออกซึ่งจะได้ผลลัพธ์ออกมา

Forward Activity

การทำงานเริ่มต้นด้วยชั้นอินพุททำหน้าที่รับค่าอินพุทส่งผ่านไปยังชั้นซ่อน โดยชั้นซ่อนทำหน้าที่ประมวลผลด้วยการคำนวณจากผลรวมของค่าอินพุทคูณค่าถ่วงน้ำหนักจากทุกโหนดในชั้นอินพุท และนำผลรวมผ่าน Activation Function เพื่อให้มีค่าอยู่ระหว่างที่กำหนดจะได้เป็นค่าเอาท์พุทของชั้นซ่อน ดังแสดงสมการที่ 2.1

$$Y_j = \sigma(\sum_{i=1}^{n} x_i w_{ij}) - \theta_j \qquad (2.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยกำหนดให้

i คือชั้นของข้อมูลเข้า

j คือชั้นซ่อน

Y_j คือค่าเอาท์พุทในชั้นซ่อน

n คือจำนวนของอินพุท

x_i คืออินพุทโหนดที่ i

Wij คือค่าถ่วงน้ำหนักของชั้นรับข้อมูลกับชั้นซ่อน

θ_j คือค่าไบแอสของชั้นซ่อน

σ คือ sigmoid activation function

โดย Activation Function ของชั้นซ่อนใช้ Sigmoid Activation Function โดยจะให้ค่าอยู่ระหว่าง [0, 1] ดังสมการที่ 2.2

$$\sigma(X) = 1/1 + e^{-X} \tag{2.2}$$

เมื่อได้ค่าเอาท์พุทในชั้นซ่อนส่งมายังชั้นข้อมูลออกก็จะนำค่านั้นคูณกับค่าถ่วงน้ำหนักระหว่างชั้นซ่อนและชั้นข้อมูลออกแล้วนำค่านั้นผ่าน Activation function ซึ่งใช้ Sigmoid Activation Function ซึ่งให้ค่าอยู่ระหว่าง [0, 1] ดังสมการที่ 2.3

$$Y_k = \sigma(\sum_{j=1}^{m} y_j w_{jk}) - \theta_k \tag{2.3}$$

โดยกำหนดให้

m คือจำนวนโหนดชั้นซ่อน

Y_k คือค่าเอาท์พุทของชั้นข้อมูลออก

w_{jk} คือค่าถ่วงน้ำหนักระหว่างชั้นซ่อน และชั้นข้อมูลออก

Y_k คือค่าผลลัพธ์ของชั้นข้อมูลออก

θ_k คือค่าไบแอสของชั้นข้อมูลออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Backward Error: Gradient function

เมื่อได้ค่าผลลัพธ์ทางชั้นข้อมูลออกจะนำมาหาค่าผิดพลาดเพื่อปรับปรุงค่าถ่วงน้ำหนัก หากเกิดค่าผิดพลาดเกิดขึ้น ซึ่งเรียกการทำงานส่วนนี้ว่า Calculate Error Gradient โดยการปรับค่าถ่วงน้ำหนักจะแสดงดังสมการที่ 3 การคำนวณหาค่า Error Gradient ในชั้นข้อมูลออก ดังสมการที่ 2.4

$$\delta_k = y_k \times [1 - y_k] \times e_k \tag{2.4}$$

โดยกำหนดให้

y_k คือ ค่าผลลัพธ์ในชั้นข้อมูลออก

e_k คือ ค่าผิดพลาดที่เกิดขึ้นในชั้นข้อมูลออก

การคำนวณค่าผิดพลาดคำนวณได้จากสมการที่ 2.5

$$e_k = y_{d,k} - y_k \tag{2.5}$$

โดยกำหนดให้

e_k คือค่าผิดพลาดของชั้นข้อมูลออก

$y_{d,k}$ คือค่าผลลัพธ์เป้าหมายที่กำหนดไว้

y_k คือค่าผลลัพธ์ที่คำนวณจริงในชั้นข้อมูลออก

การคำนวณหาค่าถ่วงน้ำหนักระหว่างชั้นข้อมูลออกและชั้นซ่อนเพื่อใช้เป็นค่าถ่วงน้ำหนักในรอบถัดไป ได้จากสมการที่ 2.6

$$\Delta w_{jk} = \alpha \times y_j \times \delta_k \tag{2.6}$$

โดยกำหนดให้

Δw_{jk} คือค่าปรับน้ำหนักระหว่างชั้นซ่อนและชั้นข้อมูลออก

α คืออัตราการเรียนรู้

y_j คือข้อมูลออกในชั้นซ่อน

δ_k ค่า error gradient ในชั้นข้อมูลออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การปรับปรุงหาค่าถ่วงน้ำหนักใหม่ จากสมการที่ 2.7

$$w_{jk}^{new} = w_{jk}^{old} + \Delta w_{jk} \tag{2.7}$$

โดยกำหนดให้

Δw_{jk} คือค่าปรับน้ำหนักระหว่างชั้นซ่อนและชั้นข้อมูลออก

w_{jk}^{new} คือค่าถ่วงน้ำหนักใหม่ระหว่างชั้นซ่อนและชั้นข้อมูลออก

w_{jk}^{old} คือค่าถ่วงน้ำหนักเดิม

การปรับค่า Error Gradient ในชั้นซ่อน (Hidden Layer) แสดงดังสมการที่ 2.8

$$\delta_j = y_j \times [1 - y_j] \times \sum_{k=1}^{l} \delta_k \times w_{jk} \tag{2.8}$$

โดยกำหนดให้

y_j คือค่าเอาท์พุทของโหนดในชั้นซ่อน

$\sum_{k=1}^{l} \delta_k$ คือผลรวมของ Error Gradient ในชั้นข้อมูลออกโดยรวมทุก Pattern

w_{jk} คือค่าถ่วงน้ำหนักระหว่างชั้นซ่อนและชั้นข้อมูลออก

δ_j คือค่า Error Gradient ในชั้นซ่อน

การปรับหาค่าถ่วงน้ำหนักใหม่ แสดงดังสมการที่ 2.9

$$\Delta w_{ij} = \alpha \times x_i \times \delta_j \tag{2.9}$$

โดยกำหนดให้

Δw_{ij} คือค่าปรับน้ำหนักระหว่างชั้นรับข้อมูลและชั้นซ่อน

α คืออัตราการเรียนรู้

x_i คือค่าข้อมูลเข้า

δ_j คือค่า Error gradient ในชั้นซ่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การปรับปรุงค่าถ่วงน้ำหนักใหม่ แสดงสมการที่ 2.10

$$w_{ij}^{new} = w_{ij}^{old} + \Delta w_{ij} \tag{2.10}$$

โดยกำหนดให้

w_{ij}^{new} คือค่าถ่วงน้ำหนักใหม่ในชั้นรับข้อมูลและชั้นซ่อน

w_{ij}^{old} คือค่าถ่วงน้ำหนักเดิมระหว่างชั้นรับข้อมูล และชั้นซ่อน

Δw_{ij} คือค่าปรับน้ำหนักระหว่างชั้นรับข้อมูลและชั้นซ่อน

ประสิทธิภาพของระบบสามารถวัดได้จากการคำนวณหาค่าของ Root mean squared error **[5]** โดยผลของ RMSE เป็นค่าข้อมูลผิดพลาดเฉลี่ยที่เกิดขึ้น ซึ่งต้องการให้มีค่าน้อยที่สุดซึ่งเป็นค่าที่ยอมรับให้เกิดค่าผิดพลาดให้ได้น้อยที่สุดแสดงดังสมการที่ 211

$$\text{RMSE} = \sqrt{\frac{\sum_n \sum_k (y_{d,k} - y_k)^2}{nk}} \tag{2.11}$$

โดยกำหนดให้

n คือ จำนวนของชุดข้อมูลที่ใช้ฝึกฝนระบบ

k คือ จำนวนของโหนดในชั้นข้อมูลออก

$y_{d,k}$ คือ ค่าผลลัพธ์ที่ต้องการ

y_k คือ ค่าผลลัพธ์ที่คำนวณได้

โดยลักษณะของการฝึกฝนข้อมูลที่ป้อนให้แก่ระบบที่ใช้ในการฝึกฝนนั้นจะเรียกว่า Epoch โดยจากสมการที่ 11 ค่าที่ยอมรับได้ว่าระบบทำงานได้อย่างถูกต้องปกติจะกำหนดให้ค่า RMSE มีค่าต่ำมากๆ ซึ่งเมื่อทำการฝึกฝนจนได้ค่าที่ยอมรับได้ก็จะนำระบบไปใช้งานจริง

2.5 Forward Activity: Cross entropy function (Rudy Setiono,2000)

การทำงานของระบบเป็นลักษณะโครงสร้างของการส่งข้อมูลแบบ Feed-forward และการเรียนรู้ของระบบ Back-propagation โดย Error Function ที่เลือกใช้คือ Cross Entropy Function ซึ่งจะใช้ในรูปแบบของ Classification โดยลักษณะของการเรียนรู้เป็นแบบ Supervised Learning โดยการทำงานของระบบโครงข่ายประสาทเทียมแบ่งหลักการทำงานเป็น 2 ส่วนได้แก่ Forward Activity และ Backward Error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 Forward Activity

การเริ่มต้นทำงานต้องนำข้อมูลเข้ามาทำการ Normalized ข้อมูลที่นำเข้า ให้มีค่าอยู่ระหว่าง 0.0 ถึง 1.0 เพื่อให้เป็นค่าที่สอดคล้องกับฟังก์ชั่นอัลกอริทึมที่ได้เลือกใช้ในขั้นตอนการทำงานประกอบด้วย

1. การกำหนดค่าถ่วงน้ำหนักน้ำหนักเพื่อให้แก่ส่วนที่เชื่อมระหว่างชั้น(Random weight) โดยการสุ่มค่าน้ำหนักมีค่าระหว่าง [-1.0, 1.0] โดยกำหนดให้

m คือชั้นซ่อนเร้น (1, 2, …., h)

l คือชั้นรับข้อมูล (1, 2, …., n)

p คือชั้นข้อมูลออก (1, 2, …., C)

2. ชั้นรับข้อมูลทำหน้าที่รับค่าอินพุทและส่งไปยังชั้นซ่อน การคำนวณหาอินพุทของโหนดในชั้นซ่อนได้จากผลรวมของค่าถ่วงน้ำหนักคูณอินพุทของแต่ละโหนดในชั้นรับข้อมูล ดังสมการที่ 2.12

$$\alpha^m = \delta(\sum_{l=1}^{n}(x_l^i w_l^m) - \theta^m) \qquad (2.12)$$

โดยกำหนดให้

α^m คือ ค่าเอาท์พุทของชั้นซ่อน (m) โดยผ่าน $\delta(x)$

x_l^i คือ ค่าอินพุทของชั้นรับข้อมูล(l) ตัวที่ i

w_l^m คือ ค่าถ่วงน้ำหนักระหว่างชั้นรับข้อมูล (l) และข้อมูลเข้าชั้นซ่อน (m)

θ^m คือ ค่าไบแอสในชั้นซ่อน

เมื่อได้ค่าเอาท์พุทของชั้นซ่อนตัวที่ m จะนำผลรวมนั้นมาผ่าน Activation Function ซึ่งกอลิทึมตัวหนึ่งที่ใช้ ซึ่งมีใช้กันหลายตัวซึ่งขึ้นอยู่กับผลที่ต้องการ เช่น Sigmoid Activation Function จะให้ผลลัพธ์ของฟังก์ชั่นนี้มีค่าอยู่ระหว่าง [0, 1] โดย Activation Function ที่ใช้ในชั้นซ่อนคือ Sigmoid Function ซึ่งให้ค่าของแต่ละโหนดในชั้นข้อมูลออกมีค่าระหว่าง [0,1] ดังสมการที่ 2.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\sigma(x) = 1/(1 + e^{-X}) \qquad (2.13)$$

โดยนำค่า x ที่ได้มาผ่านฟังก์ชั่นโดยข้อมูลออกของโหนดในชั้นซ่อนนี้จะอยู่ระหว่าง [0, 1]

3. นำข้อมูลออกในชั้นซ่อนส่งค่าไปยังชั้นข้อมูลออก โดย
การนำค่าเอาท์พุทในชั้นซ่อนคูณค่าถ่วงน้ำหนักได้ผลลัพธ์นำทุกโหนดมารวมกันและผ่าน Activation Function แสดงดังสมการที่ 2.14

$$S_p^i = \sigma\left(\sum_{m=1}^{h} \alpha^m v_p^m\right) \qquad (2.14)$$

โดยกำหนดให้

s_p^i คือ ค่าเอาท์พุทของชั้นข้อมูลออก

σ (x) คือ activation function ในชั้นข้อมูลออก

v_p^m คือ ค่าถ่วงน้ำหนักระหว่างชั้นซ่อนไปยังชั้นข้อมูลออก

α^m คือ ค่าเอาท์พุทในชั้นข้อมูลซ่อน

โดย Activation Function ที่ใช้ในชั้นข้อมูลออกคือ Sigmoid Function ซึ่งให้ค่าของแต่ละโหนดในชั้นข้อมูลออกมีค่าระหว่าง [0,1] ดังสมการที่ 2.15

$$\sigma(x) = 1/(1 + e^{-X}) \qquad (2.15)$$

โดยผลลัพธ์ที่ได้ของแต่ละโหนดในชั้นข้อมูลจะได้เป็นค่าที่เป็นผลลัพธ์จากการคำนวณโดยในจำนวนในการจัดกลุ่มให้เท่ากับจำนวนโหนดในชั้นข้อมูลออก ก็อาจจะกำหนด เป็น 3 โหนดโดยการกำหนดดังนี้ เช่น ให้ t_p^i=0 โดย t_1^i=1 ถ้า $x^i \in$ A, $t_2^i = 1$ ถ้า $x^i \in B$,$t_3^i = 1$ ถ้า $x^i \in C$ การวัดผลลัพธ์ในการจัดกลุ่มได้ถูกต้องหรือไม่นั้น ดังสมการที่ 2.16

$$\max_p \left|e_p^i\right| = \max_p \left|s_p^i - t_p^i\right| \le \eta_1 \qquad (2.16)$$

โดยกำหนดให้

$\max_p$ คือ หาค่าผิดพลาดมากที่สุดในชั้นข้อมูลออก

s_p^i คือ ค่าผลลัพธ์คำนวณจริงของโหนดที่ p ตัวที่ i

t_p^i คือ ค่าผลลัพธ์ที่ต้องการ

η_1 คือ ค่าคงที่กำหนดให้ไม่เกิน 0.5 เป็นค่าที่ยอมรับได้ในการจัดกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความผิดพลาดของการพยากรณ์

ความถูกต้องของการพยากรณ์เป็นสิ่งที่บอกถูกความถูกต้องในการทำงานของโปรแกรมในทางที่ถูกต้อง โดยสามารถจะสังเกตเห็นได้ ค่าผิดพลาดมีแนวโน้มที่จะลดลงในแต่ละรอบ แต่ก็ไม่ได้แสดงถึงความถูกต้องในการจัดกลุ่มที่จะออกมาอย่างถูกต้อง เนื่องจากจะเกี่ยวเนื่องกับหลายตัวแปรเช่น Node Hidden , Learning Rate เป็นต้น โดยได้ใช้สูตรในการวัดความถูกต้องของการพยากรณ์ ได้แก่ RMSE (Root Mean Square Error)

2.5.2 Backward Activity: Cross entropy function

การแพร่ย้อนกลับหากผลลัพธ์ที่ได้มีความผิดพลาดมากกว่าค่าที่กำหนดจึงกระจายค่าผิดพลาดกลับสู่โครงข่ายประสาทเทียม โดย Error Function สำหรับการกระจายค่าผิดพลาดแบบจัดกลุ่มนั้น อัลกอริทึมที่เลือกใช้คือ Cross Entropy Function ใช้สำหรับการจัดกลุ่ม ดังสมการที่ 2.17

$$F(w,v) = -\sum_{i=1}^{k}\sum_{p=1}^{o}(t_p^i \log S_p^i + (1-t_p^i)\log(1-S_p^i)) \qquad (2.17)$$

โดยขั้นตอนในการปรับปรุงค่าถ่วงน้ำหนัก Gradient function นี้ การหาค่าถ่วงน้ำหนัก w_{ml} ซึ่งเป็นการกระจายค่าถ่วงน้ำหนักลักษณะของ back-propagation โดยจากสมการเป็นการหาค่าถ่วงน้ำหนักระหว่างชั้นซ่อนเร้นไปยังชั้นรับข้อมูล ได้แก่ดังสมการที่ 2.18

$$\frac{\partial F(w,v)}{\partial w_{ml}} = \frac{\partial F(w,v)}{\partial s_{pi}} \times \frac{\partial S_{pi}}{\partial w_{ml}} \qquad (2.18)$$

$$\Delta w_{ml} = \sum_{i=1}^{k}\sum_{p=1}^{c}[e_{pi} \times v_{pm} \times (1-\delta(x_i w_m)^2) \times x_{il}]$$

โดยกำหนดให้

e_{pi} คือ ค่า Error ของชั้นข้อมูลออกในโหนดที่ p , Patten ที่ i

v_{pm} คือ ค่าถ่วงน้ำหนักที่ได้จากการแพร่ย้อนกลับ ระหว่างชั้นข้อมูลออกไปยังชั้นซ่อน

P คือ โหนดในชั้นข้อมูลออก P=1, 2,..C

K คือ จำนวนของรูปแบบ i=1, 2, K

δ คือ ฟังก์ชั่น Sigmoid Function

Δw_{ml} คือค่าปรับน้ำหนักระหว่างชั้นรับข้อมูลและชั้นซ่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชั่นในการคำนวณที่ใช้ในสมการที่ 18 นั้น เป็นฟังก์ชั่น Sigmoid Function ซึ่งให้ค่าอยู่ระหว่าง [0,1] แสดงดังสมการที่ 2.19

$$\sigma(x) = 1/(1 + e^{-x}) \tag{2.19}$$

การค่าผิดพลาดในแต่ละโหนดของชั้นข้อมูลออกนั้นหาได้จากสมการที่ 2.20

$$e_{pi} = S_{pi} - t_{pi} \tag{2.20}$$

โดยกำหนดให้

e_{pi} คือค่าผิดพลาดที่เกิดขึ้นของชั้นข้อมูลออก

S_{pi} คือค่าผลลัพธ์ที่คำนวณจริงชั้นข้อมูลออก

t_{pi} คือค่าผลลัพธ์ที่ต้องการของชั้นข้อมูลออก

i คือแต่ละโหนดในชั้นข้อมูลออก (1, 2, ., k)

การปรับปรุงค่าถ่วงน้ำหนักระหว่างชั้นรับข้อมูลและชั้นซ่อน

$$w_{ml}^{new} = w_{ml}^{old} + \eta \Delta w_{ml} \tag{2.21}$$

โดยกำหนดให้

η คืออัตราในการเรียนรู้

Δw_{ml} คือ ค่าปรับน้ำหนักระหว่างชั้นรับข้อมูลและชั้นซ่อน

w_{ml}^{old} คือค่าถ่วงน้ำหนักเดิมระหว่างชั้นรับข้อมูลและชั้นซ่อน

w_{ml}^{new} คือค่าถ่วงน้ำหนักใหม่ระหว่างชั้นรับข้อมูลและชั้นซ่อน

ในส่วนขั้นตอนนี้เป็นการหาค่าถ่วงน้ำหนัก w_{pm} เป็นการหาค่าถ่วงน้ำหนักในชั้นซ่อนเร้นไปยังชั้นข้อมูลออกแสดงดังสมการที่ 2.22

$$\frac{\partial F(w,v)}{\partial v_{pm}} = \frac{\partial F(w,v)}{\partial S_{pi}} \times \frac{\partial S_{pi}}{\partial v_{pm}} \tag{2.22}$$

$$\Delta v_{pm} = \sum_{i=1}^{k} [e_{pi} \times \delta(x_i w_m)]$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยกำหนดให้

$\sum_{i=1}^{k}$ คือค่าผลรวมของ Patten

e_{pi} คือค่าผิดพลาดในชั้นข้อมูลออก

x_i คือค่าอินพุทโหนดที่ i

w_m คือค่าถ่วงน้ำหนักในชั้นรับข้อมูลส่งไปยังโหนดที่ m

η คืออัตราในการเรียนรู้

Δv_{pm} คือค่าปรับน้ำหนักระหว่างชั้นซ่อนและชั้นข้อมูลออก

การปรับค่าถ่วงน้ำหนักในชั้นซ่อนและชั้นรับข้อมูลดังสมการที่ 2.23

$$v_{pm}^{new} = v_{pm}^{old} + \eta \Delta v_{pm} \qquad (2.23)$$

โดยกำหนดให้

v_{pm}^{old} คือค่าถ่วงน้ำหนักเดิมระหว่างชั้นซ่อนและชั้นข้อมูลออก

v_{pm}^{new} คือค่าถ่วงน้ำหนักใหม่ระหว่างชั้นซ่อนและชั้นข้อมูลออก

η คืออัตราในการเรียนรู้

Δv_{pm} คือค่าปรับน้ำหนักระหว่างชั้นซ่อนและชั้นข้อมูลออก

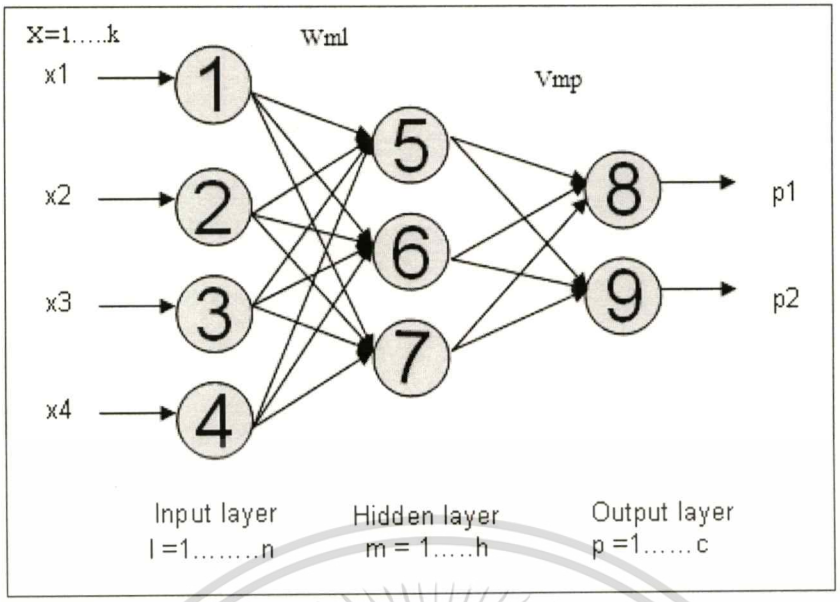
จากสมการเมื่อมีการปรับค่าถ่วงน้ำหนักแล้วจะได้ค่าถ่วงน้ำหนักใหม่เพื่อใช้ในรอบการคำนวณถัดไปซึ่งจะทำการวนรูปทำงานจนกว่าจะได้ผลลัพธ์ในการจำแนกกลุ่มได้อย่างถูกต้องโดยสมการที่ 16 เป็นสมการที่ชี้วัดประสิทธิภาพการทำงานของระบบในการจำแนกกลุ่ม

2.6 ตัวอย่างการคำนวณ

แสดงตัวอย่างการคำนวณโดยใช้ Cross entropy function โดยกำหนดให้ มีชั้นรับข้อมูลมี 4 โหนดให้ชั้นซ่อนมี 3 โหนด และชั้นข้อมูลออกเป็น 2 โหนดเพื่อให้จำแนกกลุ่มโดยใช้ฟังก์ชั่นครอสเอ็นโทพีดังแสดงดังรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง



รูปที่ 2.4 รูปโครงสร้างโครงข่ายประสาทเทียม

โดยกำหนดให้

m คือชั้นซ่อน (1, 2, …., h)

l คือชั้นรับข้อมูล (1, 2, …., n)

p คือชั้นข้อมูลออก (1, 2, ..., C)

x คืออินพุท

p1 คือผลลัพธ์ของโหนดที่ 1 กำหนดให้เป็น “1”

p2 คือผลลัพธ์ของโหนดที่ 2 กำหนดให้เป็น “0”

โดยกำหนดให้ค่าอินพุทเข้าเป็นดังนี้ x1=0.5, x2=0.5714, x3=0.4444, x4=0.2 โดยค่าอินพุทได้ผ่านการ Normalized แล้วกำหนดค่าถ่วงน้ำหนักโดยการสุ่มให้มีค่าอยู่ระหว่าง [-1,1]

W15= -0.7, w16=0.3, w17=-0.9, v58=0.2, v78=-0.4

W25=-0.5, w26=-0.8, w27=0.8, v59=0.9, v79=-0.2

W35=0.9, w36=0.1, w37=-0.5, v68=-0.3

W45=0.2, w46=-0.2, w47=0.4, v69=-0.1

เริ่มต้นการทำงานโดยป้อนข้อมูลเข้าชั้นรับข้อมูลเพื่อส่งไปยังโหนดในชั้นซ่อนโดยหาค่าของเอาท์พุทของโหนดที่ 5 ในชั้นซ่อนโดยใช้สมการที่ 2.24

$$\alpha^m = \delta(\sum_{l=1}^{n}(x_l^i w_l^m) - \theta^m) \qquad (2.24)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

04884

โดยแจกแจงจากอัลกอริทึมได้ดังนี้

$= \delta[(0.5(-0.7)+0.5714(-0.5)+0.4444(0.9)+0.2(0.2))]$

$= \delta(-0.19574)$

$\alpha^6 = \delta[(x_1 w_{16}) + (x_2 w_{26}) + (x_3 w_{36}) + (x_4 w_{46})] - 0$

$= \delta(-0.30268)$

$\alpha^7 = \delta[(x_1 w_{17}) + (x_2 w_{27}) + (x_3 w_{37}) + (x_4 w_{47})] - 0$

$= \delta(-0.13508)$

นำมาผ่าน activation function โดยใช้ sigmoid activation function จะให้ค่าอยู่ระหว่าง [0,1] ดังสมการที่ 2.25

$$\sigma(x) = 1/(1 + e^{-x}) \tag{2.25}$$

คำนวณค่าเอาท์พุทของชั้นซ่อนในโหนดที่ 5 ,6,7

นำข้อมูลที่ได้ออกจากชั้นข้อมูลในชั้นซ่อนส่งไปยังชั้นข้อมูลออกโดยคำนวณผ่านอัลกอริทึมเพื่อส่งแสดงผลลัพธ์ของระบบ ดังแสดงสมการที่ 2.26

$$S_p^i = \sigma(\sum_{m=1}^{h} \alpha^m v_p^m) \tag{2.26}$$

โดยแจกแจงจากอัลกอริทึมได้ดังนี้

$S_8 = \sigma[(\alpha^5 v_{58}) + (\alpha^6 v_{68}) + (\alpha^7 v_{78})]$

$= \sigma[(-0.193292(0.2)+-0.293766(-0.3)+(-0.134264)(-0.4))]$

$= \sigma(0.103177)$

$S_9 = \sigma[(\alpha^5 v_{59}) + (\alpha^6 v_{69}) + (\alpha^7 v_{79})]$

$= \sigma[(-0.193292(0.9) +-0.293766(-0.1) +(-0.134264)(0.2))]$

$= \sigma(-0.1177334)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำมาผ่าน activation function โดยใช้ sigmoid activation function จะให้ค่าอยู่ระหว่าง [0,1] ดังสมการที่ 2.27

$$\sigma(x) = 1/(1 + e^{-x}) \tag{2.27}$$

ค่าผลลัพธ์ในชั้นข้อมูลออกโหนดที่ 8

$S_8 = 1/(1+e^{-0.103177})$

$S_8 = 0.474228$

ค่าผลลัพธ์ในชั้นข้อมูลออกโหนดที่ 9

$S_9 = 1/(1+e^{0.1177334})$

$S_9 = 0.470600$

ในสมการนี้จะหาค่าผิดพลาดของระบบโดยการกำหนดให้โหนดที่ 8 หรือ ep1 ให้มีค่าผลลัพธ์มีค่าเป็น 1 และโหนดที่ 9 ให้มีค่าผลลัพธ์เป็น 0 จึงสามารถคำนวณหาค่าผิดพลาดโดยใช้สมการที่ 2.28

$$e_{pi} = S_{pi} - t_{pi} \tag{2.28}$$

แสดงค่าผิดพลาดของ e_{p1} (กำหนดให้ผลลัพธ์เป้าหมาย=1)

$e_{p1} = 0.474228 - 1$

$e_{p1} = -0.525772$

แสดงค่าผิดพลาดของ e_{p2} (กำหนดให้ผลลัพธ์เป้าหมาย=0)

$e_{p2} = 0.470600 - 0$

$e_{p2} = 0.47060$

Error Function : Cross entropy function

เนื่องจากผลลัพธ์ที่ได้มีความผิดพลาดมากกว่าค่าที่กำหนดจึงกระจายค่าผิดพลาดกลับสู่โครงข่ายประสาทเทียม โดยการใช้ Error Function สำหรับการกระจายค่าผิดพลาดแบบจัดกลุ่มนั้น อัลกอริทึมที่เลือกใช้คือ Cross Entropy Function ดังสมการที่ 2.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$F(w,v)=-\sum_{i=1}^{k}\sum_{p=1}^{o}(t_p^i \log S_p^i+(1-t_p^i)\log(1-S_p^i)) \quad (2.29)$$

การปรับค่าถ่วงน้ำหนัก Cross Entropy Function ซึ่งเป็นการกระจายค่าถ่วงน้ำหนักลักษณะของ back-propagation โดยจากสมการที่ 2.30 เป็นการหาค่าถ่วงน้ำหนักระหว่างชั้นซ่อนไปยังชั้นรับข้อมูล

$$\Delta v_{pm}=\sum_{i=1}^{k}[e_{pi}\times\delta(x_i w_m)] \quad (2.30)$$

จากตัวอย่างเป็นการคำนวณหาค่าถ่วงน้ำหนักระหว่างชั้นข้อมูลออกไปยังชั้นซ่อนโดยเป็นการคำนวณเพียง Pattern เดียว เมื่อคำนวณได้แล้วเราต้องเอาค่าทุก Patten มารวมกันทั้งหมดก่อน ค่าจากนั้น เราจึงนำไปปรับปรุงค่าถ่วงน้ำหนัก
ปรับปรุงค่าถ่วงน้ำหนัก v_{pm} ดังสมการที่ 2.31

$$v_{pm}^{new}=v_{pm}^{old}+\eta\Delta v_{pm} \quad (2.31)$$

ปรับปรุงค่าถ่วงน้ำหนัก w_{ml} ดังสมการที่ 2.32 เป็นการหาค่าถ่วงน้ำหนักในชั้นซ่อนและชั้นข้อมูลเข้า

$$\Delta w_{ml}=\sum_{i=1}^{k}\sum_{p=1}^{c}[e_{pi}\times v_{pm}\times(1-\delta(x_i w_m)^2)\times x_{il}] \quad (2.32)$$

การคำนวณส่วนนี้เป็นการหาค่าถ่วงน้ำหนัก เพื่อนำไปเป็นค่าถ่วงน้ำหนักในรอบถัดไป โดยการคำนวณในแต่ละPatten ดังนั้นจึงต้องรวมค่าทุก Patten ก่อนจึงจะนำไปปรับปรุงค่าถ่วงน้ำหนักใน Cycle ถัดเพื่อใช้เป็นค่าถ่วงน้ำหนักใหม่ในรอบ Cycle ถัดไป

ปรับปรุงค่าถ่วงน้ำหนัก w_{ml} ดังสมการที่ 2.33

$$w_{ml}^{new}=w_{ml}^{old}+\eta w_{ml} \quad (2.33)$$

ปรับปรุงค่าถ่วงน้ำหนักระหว่างชั้นรับข้อมูลและชั้นซ่อนกำหนดให้ อัตราการเรียนรู้ $\eta=0.1$ เป็นการเรียนรู้ของระบบโดยการปรับค่าถ่วงน้ำหนักจนกว่าจะจัดกลุ่มได้อย่างถูกต้องโดยสามารถวัดได้จากสมการที่ 2.34

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\max_p \left|e_p^i\right| = \max_p \left|s_p^i - t_p^i\right| \le \eta_1 \qquad (2.34)$$

ผลลัพธ์ที่ได้ออกมาทางชั้นข้อมูลออกนั้นจะเปรียบเทียบกับค่าผลลัพธ์ที่ต้องการในแต่ละโหนด จะได้ผิดพลาดโดยเลือกค่าผิดพลาดที่มีค่าผิดพลาดที่มากที่สุดในโหนดของชั้นข้อมูลออกมา หากว่ามีค่ามากกว่า η_1 (ค่าผิดพลาดที่ยอมรับได้) แสดงว่ายังมีค่าผิดพลาดเกินกว่าจะยอมรับได้ ดังนั้นระบบจะทำการฝึกฝนเพื่อให้ได้ค่าถ่วงน้ำหนักที่เหมาะสมต่อไป จนกว่าจะได้ผลลัพธ์ที่มีค่าผิดพลาดยอมรับได้ แต่หากค่าผิดพลาดที่มากที่สุดในชั้นข้อมูลออกน้อยกว่าค่า η_1 ก็แสดงให้เห็นว่าการจัดกลุ่มถูกต้องและจะหยุดการเรียนรู้เพื่อนำค่าถ่วงน้ำหนักนั้นไปใช้งานในระบบจริง

เพื่อให้เห็นการทำงานของระบบที่ชัดเจนจะทำป้อนข้อมูลอินพุทอีกชุดหนึ่งเพื่อแสดงให้เห็นว่าระบบมีการเรียนรู้โดยการสังเกตได้จาก ค่าผิดพลาดจะลดลงในแต่ละรอบของการเรียนรู้ และแสดงถึงการปรับค่าถ่วงน้ำหนักของระบบ

จะเห็นได้ว่าระบบมีการเรียนรู้โดยปรับค่าถ่วงน้ำหนักโดยจะสังเกตได้ว่า ค่าผิดพลาดเมื่อเปรียบเทียบกับค่าผลลัพธ์เป้าหมายมีค่าผิดพลาดที่ลดลงเพื่อที่ระบบจะปรับค่าถ่วงน้ำหนักให้สามารถได้ค่าผลลัพธ์ที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

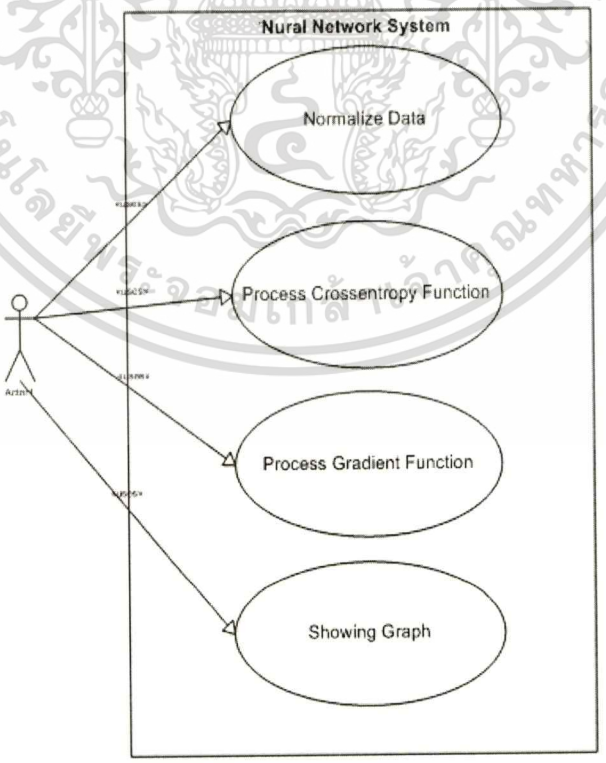
บทที่ 3
การวิเคราะห์และออกแบบระบบ

ในบทนี้จะกล่าวถึงการวิเคราะห์และออกแบบระบบงานใหม่ โดยได้นำรายละเอียดข้อมูลความต้องการของระบบที่ได้จากบทที่แล้วมาใช้ในการออกแบบ ซึ่งใช้ UML (Unified Modeling Language) เป็นโมเดลมาตรฐานที่ใช้หลักการออกแบบ OOP (Object oriented programming) มาใช้ในการอธิบายภาพรวมส่วนประกอบ และลักษณะการทำงานของระบบ

3.1 ยูสเคสไดอะแกรม

ภาพรวมของฟังก์ชันการทำงานของระบบสารสนเทศนักศึกษาสามารถแสดงได้ โดยใช้ยูสเคสไดอะแกรม (Use Case Diagram) เพื่อแสดงให้เห็นว่าในระบบแบ่งส่วนประกอบของการทำงานออกเป็นส่วนย่อยๆ เป็นส่วนใดบ้าง สำหรับยูสเคสไดอะแกรมของระบบ Neural Network สามารถแสดงได้ดังในรูปที่ 3.1 โดยประกอบด้วยแอกเตอร์หลักที่สำคัญดังต่อไปนี้

ผู้ใช้ (USER) ผู้ที่เข้ามาใช้งานโปรแกรมและฟังก์ชั่นการทำงานหลักๆที่ทำงานอยู่ในระบบเพื่อให้เห็นภาพรวมของการทำงานของระบบ



รูปที่ 3.1 use case diagram ของระบบ Neural Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1 แสดงยูสเคสไดอะแกรมของภาพรวมการทำงานของระบบ ในส่วนรายละเอียดของยูสเคสสามารถอธิบายรายละเอียดขั้นตอนการทำงาน รายละเอียดของยูสเคสสามารถแสดงเป็นแอกทิวิตี้ไดอะแกรมได้ดังรูปที่ 3.2 ถึง 3.5 ดังต่อไปนี้

UC1: Normalize Data

Description: นำข้อมูลดิบนำเข้าสู่ระบบเพื่อทำการนอร์มอลไรซ์ให้มีค่าอยู่ระหว่าง 0 ถึง 1

Primary actor : user

Stakeholder : -

Precondition: ผู้ใช้ต้องทำการระบุข้อมูลเบื้องต้นของโครงข่าย เช่น Learning Rate, จำนวน Hidden Layer และ จำนวนรอบ

Basic Flow

1 ผู้ใช้เลือกไฟล์ข้อมูลที่ต้องการทำ(เลือกไฟล์ .data หรือ .xls)

2. ระบบทำการหาจำนวน Input Layer และ Output Layer ของไฟล์ที่ผู้ใช้ระบุ

3. ระบบทำการเก็บข้อมูลเข้าเป็นแต่ละคลาส

4. ระบบทำการหาค่า Max และ Min แต่ละคลาส

5. ระบบการ Normalize ข้อมูลแล้วเก็บข้อมูลเข้าสู่ไฟล์ "NoramalizedData.xls"

6. ระบบนำข้อมูลที่ทำการ normalized แล้วมาแบ่งเป็นข้อมูลที่ใช้ในการ Training and Testing Process

Alternate Flow

1a) ถ้าผู้ใช้เลือกไฟล์ข้อมูลที่ไม่ใช่ .xlsและ.data ระบบจะทำการแสดงข้อความเตือน

Post Condition :

-ได้ไฟล์ที่ผ่านการ Normalize ได้ค่าระหว่าง 0 ถึง 1(ข้อมูลอยู่ในไฟล์ NormalizedData.xls"

-ได้ไฟล์ "Train.xls" และ "Test.xls"

UC2: Process Cross entropy Function

Description : นำข้อมูลที่ผ่านการ Normalize แล้วผ่านการฝึกฝนโครงข่ายประสาทเทียมโดยใช้การเรียนรู้แบคพรอคพาเกชั่น Cross Entropy function เพื่อให้ได้ค่า Error ที่ได้ต่ำกว่าค่าที่กำหนด

Precondition: ข้อมูลผ่านการ Normalized

Primary Actor System(ระบบ)

Stakeholder: -

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Basic Flow

1.ระบบทำการเปิดไฟล์ “Train.xls”

2. ระบบทำการสร้างโครงข่ายประสาทเทียม โดยสอดคล้องกับชุดข้อมูลของ input และ output

3. ระบบใช้การเรียนรู้แบบแพร่กระจายย้อนกลับ โดยใช้ Cross Entropy Function

4. ระบบเก็บข้อมูลของ ค่า Error และ ค่า Weight Adjustment โดยไม่ได้บวกค่า Old Weight และไม่มีการคูณด้วยค่า Learning Rate

5. ระบบทำการตรวจสอบว่า ค่า Error ที่ได้จากการคำนวณของแต่ละรอบ

6. ระบบทำการเปิดไฟล์ “Test.xls” โดย Weight ที่นำมาใช้เป็น Weight ที่มาจากการ Training

7. ระบบทำการตรวจสอบถึงประสิทธิภาพการ Classification

Alternative :

2a) ถ้าเป็นการสร้างโครงข่ายประสาทเทียมในครั้งแรก ระบบจะสุ่มค่า Weight -0.5 ถึง 0.5

2b) ถ้าไม่ใช่การสร้างโครงข่ายประสาทเทียมรอบแรก ระบบจะนำค่า Weight ที่ปรับปรุงจากรอบที่แล้วมาใช้เป็นค่า Weight ในการสร้างโครงข่ายประสาทเทียม

5a) ถ้าค่า Error ต่ำกว่าค่า Tolerance Error ของแต่ละรอบ ระบบจะหยุดการทำงาน

5b) ถ้าค่า Error มากกว่าค่า Tolerance Error ของแต่ละรอบ ระบบจะทำงานต่อและทำการ Update ค่า Weight

Post Condition

-ได้ไฟล์ “WeightCross.xls” และ “ErrorCross.xls”

- ได้ไฟล์ “BestWeightCross.xls” และ “TimeCross.xls”

- ได้ไฟล์ “TestCross.xls”

UC:3 Process Error Gradient Function

Description การนำข้อมูลที่ผ่านการปรับค่าแล้วเข้าสู่ระบบเพื่อทำการจัดกลุ่มโดยใช้การเรียนรู้แบบย้อนกลับโดยอัลกอริทึมที่ใช้ Error Gradient Function

Precondition: ข้อมูลผ่านการ Normalized

Primary Actor System(ระบบ)

Stakeholder : -

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Basic Flow

1.ระบบทำการเปิดไฟล์ “Train.xls”

2. ระบบทำการสร้างโครงข่ายประสาทเทียม โดยสอดคล้องกับชุดข้อมูลของ Input และ Output

3. ระบบใช้การเรียนรู้แบบแพร่กระจายย้อนกลับ โดยใช้ Gradient Function

4. ระบบเก็บข้อมูลของ ค่า Error และ ค่า Weight Adjustment

5. ระบบทำการตรวจสอบว่า ค่า Error ที่ได้จากการคำนวณของแต่ละรอบ

6. ระบบทำการเปิดไฟล์ “Test.xls” โดย Weight ที่นำมาใช้เป็น Weight ที่มาจากการ Training

7. ระบบทำการตรวจสอบถึงประสิทธิภาพการ Classification

Alternative :

2a) ถ้าเป็นการสร้างโครงข่ายประสาทเทียมในครั้งแรก ระบบจะสุ่มค่า Weight -0.5 ถึง 0.5

2b) ถ้าไม่ใช่การสร้างโครงข่ายประสาทเทียมรอบแรก ระบบจะนำค่า Weight ที่ปรับปรุงจากรอบที่แล้วมาใช้เป็นค่า weight ในการสร้างโครงข่ายประสาทเทียม

5a) ถ้าค่า Error ต่ำกว่าค่า Tolerance Error ของแต่ละรอบ ระบบจะหยุดการทำงาน

5b) ถ้าค่า Error มากกว่าค่า Tolerance Error ของแต่ละรอบ ระบบจะทำงานต่อ

Post condition

-ได้ไฟล์ “WeightGrad.xls” และ “ErrorGrad.xls”

- ได้ไฟล์ “BestWeightGrad.xls” และ “TimeGrad.xls”

- ได้ไฟล์ “TestGrad.xls”

UC:4 Comparing Algorithm

Description เป็นการแสดงกราฟเชิงเส้น เปรียบเทียบข้อมูลของ Classification และทำการแสดงเวลา

Precondition: ข้อมูล Cross Entropy and Gradient Function Process

Primary Actor System(ระบบ)

Stakeholder: -

Basic Flow

1.ระบบทำการเปิดไฟล์ “TestCross.xls”, “TestGrad.xls” และ “NormalizedData.xls”

2. ระบบทำเก็บค่า Output ของแต่ละรอบที่ใช้ในการ Test ของทั้งสองอัลกอริทึมเปรียบเทียบกับข้อมูล Output ที่ได้จริง

3. ระบบทำการแสดงข้อมูลกราฟเชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ระบบทำการเปิดไฟล์ “TimeCross.xls” และ “TimeGrad.xls”

5. ระบบทำการเฉลี่ยค่าเวลาของแต่ละอัลกอริทึม

6. ระบบแสดงค่าเฉลี่ยเป็นตัวเลข

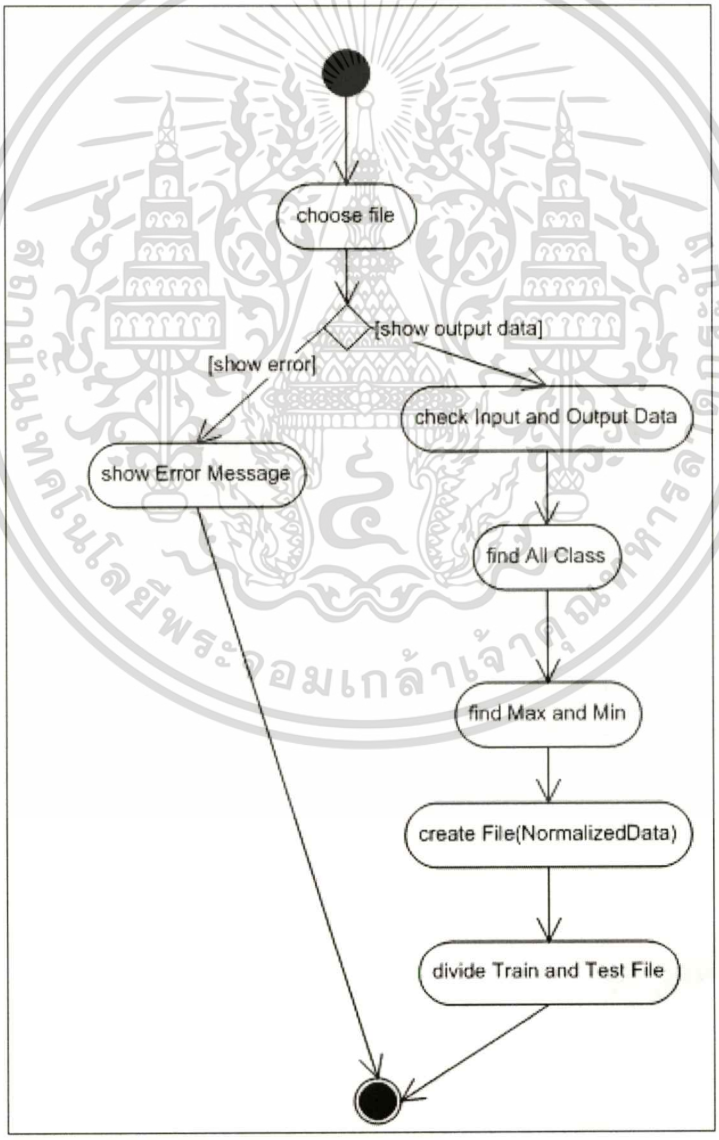
Alternative :

Post condition

- ได้กราฟเชิงเส้นและเวลา ทีใช้ในการคำนวณ

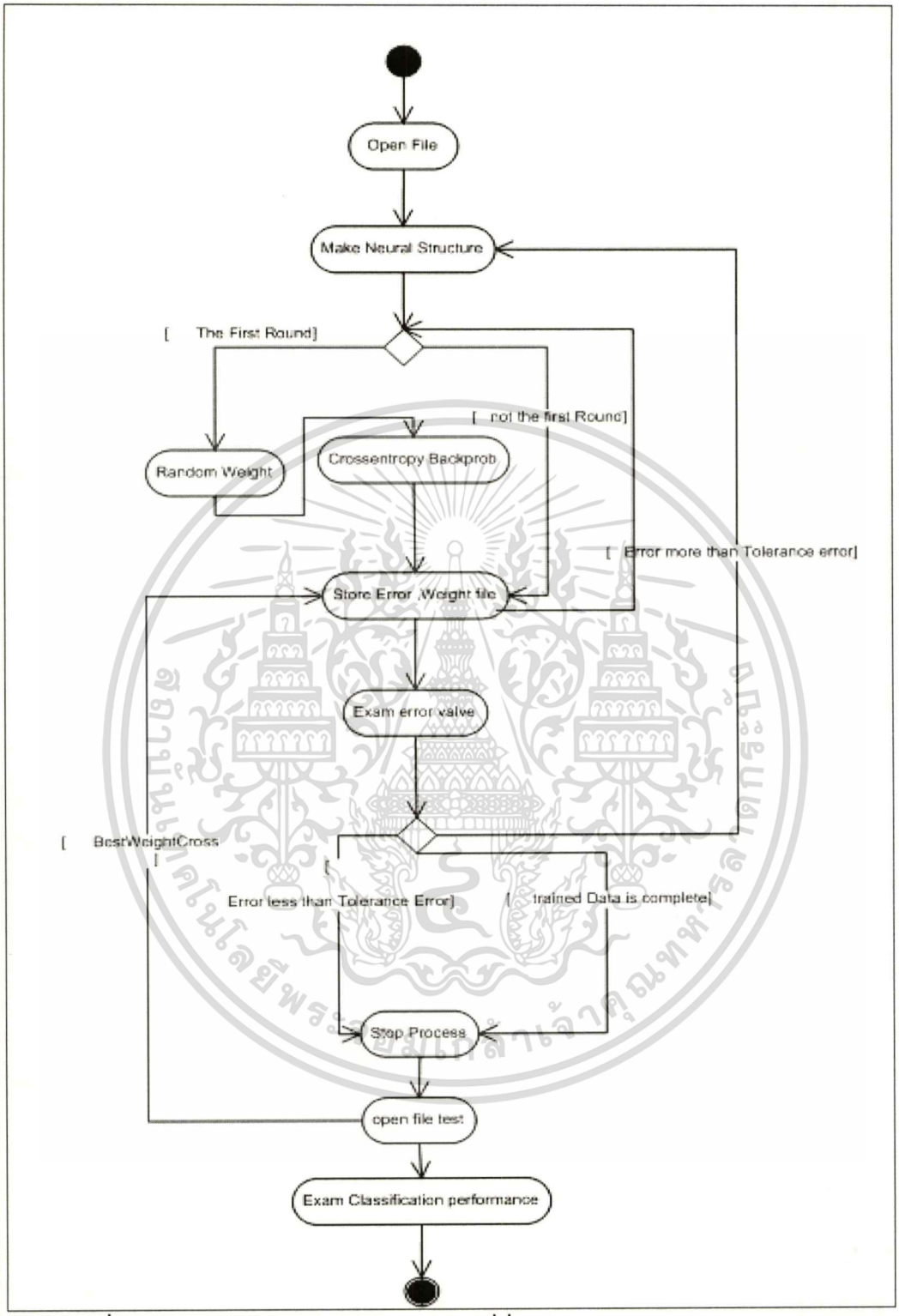
3.2 Activity Diagram

ในส่วนถัดไปแสดงในส่วนของ Activity Diagram



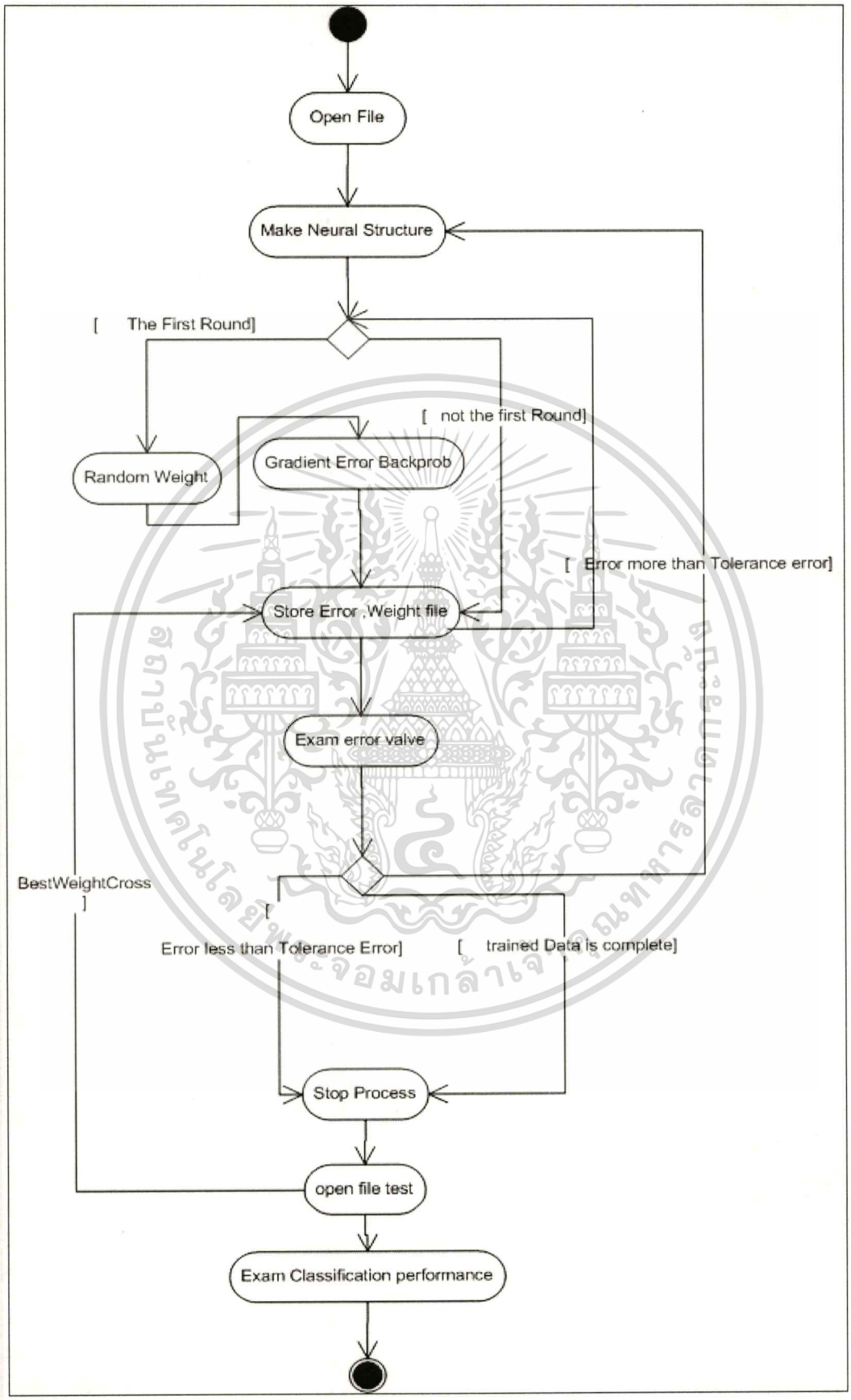
รูปที่ 3.2 Activity Diagram ของ Use Case ที่ชื่อ Normalized Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



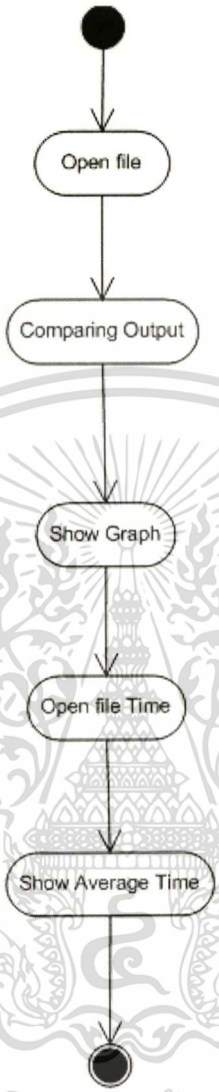
รูปที่ 3.3 Activity Diagram ของ Use Case ที่ชื่อ Process Cross entropy function

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 Activity Diagram ของ Use Case ที่ชื่อ Process Error gradient function

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 Activity Diagram ของ Use Case ที่ชื่อ Comparing algorithm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบส่วนต่อประสานกับผู้ใช้

ในกระบวนการพัฒนาระบบงาน หลังจากได้มีการออกแบบระบบเรียบร้อยแล้วจะต้องพัฒนาส่วนต่อประสานกับผู้ใช้ ซึ่งจะต้องศึกษาข้อมูลต่างๆดังต่อไปนี้ คือ เครื่องมือที่ใช้ในการพัฒนาระบบงานเป็นต้น โดยสามารถแสดง

4.1 เครื่องมือและภาษาที่ใช้ในการพัฒนาระบบ

4.11 ฮาร์ดแวร์

เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนาและทดสอบระบบ โดยมีคุณสมบัติดังนี้

- Intel Pentium 4 1.70Ghz
- RAM 512 MB
- Hard Disk 80 GB

4.12 ซอฟต์แวร์

ซอฟต์แวร์ที่นำมาใช้เป็นเครื่องมือในการพัฒนา และทดสอบระบบได้แก่

1) ระบบปฏิบัติการที่ใช้ คือ Microsoft Windows XP Professional
2) โปรแกรมที่ใช้ในการพัฒนาระบบ โดยใช้ Tool ของ NetBeans IDE 6.0.1
3) โปรแกรม JAVA

4.2 รายละเอียดการทำงานของระบบ

การทำงานของระบบแบ่งเป็นสองส่วนด้วยกันได้แก่ การฝึกฝนแก่ระบบ และการทดสอบระบบ โดยการทำงานนั้นจะทำการเปรียบเทียบอัลกอริทึมในส่วนของการเรียนรู้แบบ Back propagation ในส่วนของการเรียนรู้แบบแพร่กระจายย้อนกลับ ในส่วนของ Feed-forward นั้นจะเป็นการทำงานที่เหมือนกันของทั้งสองอัลกอริทึมจะแตกต่างกันตรงที่การเรียนรู้ในส่วนการแพร่กระจายย้อนกลับ เพื่อให้เห็นการเปลี่ยนแปลงของแต่ละอัลกอริทึมที่เกิดขึ้นต่อการฝึกฝนต่อระบบเพื่อนำไปใช้หรือเป็นแนวทางพัฒนาต่อไป

โดยเริ่มจากการทำงานเกิดจากเริ่มนำข้อมูลเข้าสู่ระบบเพื่อใช้ในส่วนของการฝึกฝนระบบและการทดสอบการทำงานของระบบ โดยข้อมูลที่นำเข้ามาฝึกฝนและทดสอบระบบตัวอย่างมีดังรูปที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

sepal length	sepal width	petal length	petal width	class
5.1	3.5	1.4	0.2	Iris-setosa
4.9	3	1.4	0.2	Iris-setosa
4.7	3.2	1.3	0.2	Iris-setosa
4.6	3.1	1.5	0.2	Iris-setosa
5	3.6	1.4	0.2	Iris-setosa
5.4	3.9	1.7	0.4	Iris-setosa
4.6	3.4	1.4	0.3	Iris-setosa
5	3.4	1.5	0.2	Iris-setosa
4.4	2.9	1.4	0.2	Iris-setosa
4.9	3.1	1.5	0.1	Iris-setosa
5.4	3.7	1.5	0.2	Iris-setosa
4.8	3.4	1.6	0.2	Iris-setosa
4.8	3	1.4	0.1	Iris-setosa
4.3	3	1.1	0.1	Iris-setosa
5.8	4	1.2	0.2	Iris-setosa
5.7	4.4	1.5	0.4	Iris-setosa

รูปที่ 4.1 ตัวอย่างข้อมูลที่นำเข้าระบบ

โดยตัวอย่างข้อมูลทั้งหมด 150 แถว แบ่งเป็น 3 กลุ่ม โดยในการฝึกฝนระบบจะนำมาใช้ในการฝึกฝน 80% และแบ่งข้อมูลมาใช้ในการทดสอบระบบ 20% โดยจากตัวอย่างข้อมูลที่ได้มาจึงกำหนดให้ระบบมีข้อมูลเข้า 4 Input และกำหนดให้ Output node มี 3 class เพื่อการฝึกฝนในการจัดกลุ่ม โดยข้อมูลกลุ่มแรกคือ Iris-setosa กลุ่มที่สองมีชื่อว่า Iris-versicolor และกลุ่มที่สามมีชื่อว่า Iris-virginica

ต่อมาจึงทำการ Normalized Data เพื่อเป็นการคลีนข้อมูลก่อนเพื่อให้เกิดความถูกต้องและทำให้ระบบทำงานได้อย่างถูกต้อง โดยเมื่อผ่านการ Normalized Data แล้วค่าของข้อมูลจะถูกแปลงให้มีค่าอยู่ในช่วงระหว่าง 0-1 ซึ่งจะสังเกตได้จากการทดลอง ตัวอย่างที่ยกมาบางช่วงดังรูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0.222222	0.625	0.067797	0.041667
0.166667	0.416667	0.067797	0.041667
0.111111	0.5	0.050847	0.041667
0.083333	0.458333	0.084746	0.041667
0.194444	0.666667	0.067797	0.041667
0.305556	0.791667	0.118644	0.125
0.083333	0.583333	0.067797	0.083333
0.194444	0.583333	0.084746	0.041667
0.027778	0.375	0.067797	0.041667
0.166667	0.458333	0.084746	0
0.305556	0.708333	0.084746	0.041667
0.138889	0.583333	0.101695	0.041667
0.138889	0.416667	0.067797	0
0	0.416667	0.016949	0
0.416667	0.833333	0.033898	0.041667
0.388889	1	0.084746	0.125
0.305556	0.791667	0.050847	0.125
0.222222	0.625	0.067797	0.083333
0.388889	0.75	0.118644	0.083333
0.222222	0.75	0.084746	0.083333
0.305556	0.583333	0.118644	0.041667
0.222222	0.708333	0.084746	0.125
0.083333	0.666667	0	0.041667
0.222222	0.541667	0.118644	0.166667
0.138889	0.583333	0.152542	0.041667
0.194444	0.416667	0.101695	0.041667
0.194444	0.583333	0.101695	0.125
0.25	0.625	0.084746	0.041667
0.25	0.583333	0.067797	0.041667
0.111111	0.5	0.101695	0.041667
0.138889	0.458333	0.101695	0.041667

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0.305556	0.583333	0.084746	0.125
0.25	0.875	0.084746	0
0.333333	0.916667	0.067797	0.041667
0.166667	0.458333	0.084746	0
0.194444	0.5	0.033898	0.041667

รูปที่ 4.2 ตัวอย่างไฟล์ที่ผ่านการ Normalized Data

โดยการทำงานของระบบจะทำการแบ่ง ไฟล์ไปใช้ในการฝึกฝนระบบและการทดสอบระบบ โดยจะแบ่งไฟล์ไปใช้ในการทดสอบระบบ 80 เปอร์เซ็นต์ และเก็บไว้ใช้เพื่อใช้ในการทดสอบระบบ 20 เปอร์เซ็นต์ เพื่อจะได้ดูว่าระบบสามารถทำงานได้อย่างถูกต้องตามที่ได้ฝึกฝนแก่ระบบแล้วหรือไม่ โดยลักษณะของการแบ่งนำมาฝึกฝนและการทดสอบนั้นจำเป็นจะต้องนำ 80 เปอร์เซ็นต์ของแต่ละประเภทนำมาฝึกฝน ดังนั้นจะเห็นได้ว่าไฟล์ที่นำมาใช้ในการทดสอบมีจำนวนสามประเภท จึงดึงมาอย่างละ 80 เปอร์เซ็นต์ของแต่ละประเภทเพื่อใช้ในการฝึกฝนและอีก 20 เปอร์เซ็นต์ของแต่ละประเภทก็เก็บไว้เพื่อใช้ในการทดสอบระบบในภายหลังจากการฝึกฝนระบบจนต่ำกว่าค่า Tolerance rate หรือครบ Cycle ที่กำหนดไว้แล้ว ซึ่งลักษณะของการที่จะกำหนดว่าระบบจะสามารถนำไปใช้ในการทดสอบได้นั้นเกิดจากการที่ error ต่ำกว่าค่าerror ที่กำหนดหรือเรียกอีกอย่างว่า Tolerance error หรือเรียกว่า ค่า error ที่ยอมรับได้ปกติกำหนดไว้ 0.001 (จากการทดสอบยังเกิดการทำนายที่ผิดพลาด ในการเรียนรู้ของระบบจึงกำหนดเป็น 0.0001) แต่ลักษณะของระบบจะมีการกำหนดพารามิเตอร์อีกหนึ่งอย่างได้แก่ Cycle หรือว่ารอบของการทดสอบ ซึ่งเกิดจากการทำงานโดยการฝึกฝนระบบของข้อมูลทั้งหมด เรียกว่า 1 Cycle ซึ่งจะสามารถกำหนดให้ระบบ หยุดโดยกำหนดเงื่อนไขว่า จะให้ระบบหยุดการเรียนรู้เมื่อครบกี่รอบหรือต่ำกว่าerror ที่เรากำหนด เมื่อระบบพบกับเงื่อนไขใดเงื่อนไขหนึ่งระบบก็จะหยุดทำงาน

4.3 ค่าพารามิเตอร์

ค่าพารามิเตอร์ต่างๆที่ต้องกำหนดให้แก่ระบบเพื่อให้ระบบสามารถทำงานได้อย่างถูกต้องมากที่สุดโดยมีพารามิเตอร์ดังนี้

Node Input คือพารามิเตอร์ที่ผู้ใช้ต้องป้อนเพื่อให้สอดคล้องกับข้อมูลที่เข้าสู่ระบบ

Node Hidden คือพารามิเตอร์ที่ทำหน้าที่หลักคือการคำนวณที่อยู่ในชั้นกลางระหว่าง Input Node และOutput Nodeซึ่งมีผลต่อการทำนายของระบบในการกำหนดจำนวน Hidden node

Node Output คือพารามิเตอร์ที่ทำหน้าที่จัดแบ่งประเภทแก่ระบบเพื่อให้สอดคล้องกับประเภทของข้อมูลที่ฝึกฝนให้แก่ระบบเพื่อแบ่งประเภทได้อย่างถูกต้อง

Learning Rate คือพารามิเตอร์ที่ให้ผู้ใช้ป้อนเข้าสู่ระบบ พารามิเตอร์นี้มีผลต่อการดำเนินการของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

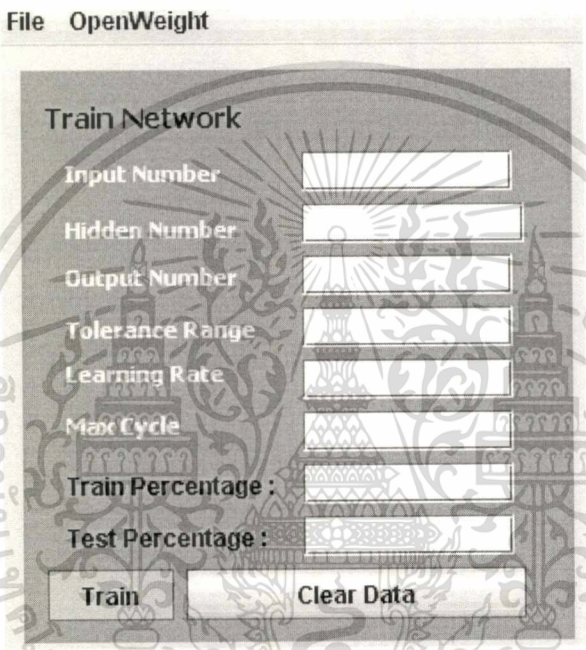
ระบบโดยจะให้ค่าอยู่ระหว่าง 0.0 ถึง 0.1 โดยมีผลต่อการเรียนรู้ของระบบ

Tolerant rate คือพารามิเตอร์ที่ผู้ใช้กำหนดเพื่อให้ระบบหยุดเมื่อ error มีค่าต่ำกว่าที่กำหนด ปกติจะกำหนดค่า 0.0001 เป็นพารามิเตอร์ที่สำคัญ

Cycle คือพารามิเตอร์ที่กำหนดให้ระบบทำงานกี่รอบตามที่ผู้ใช้กำหนดเมื่อครบระบบจะหยุดการฝึกฝน

Train Percentage กำหนดค่าโดยผู้ใช้เพื่อระบุจำนวนสัดส่วนของข้อมูลเพื่อใช้ในการเรียนรู้

Test Percentage กำหนดค่าโดยผู้ใช้เพื่อระบุจำนวนสัดส่วนของข้อมูลที่ใช้ในการทดสอบระบบ



รูปที่ 4.3 หน้าจอเพื่อผู้ใช้ป้อนพารามิเตอร์

4.4 ผลการทดลองและสรุปผลการทดลอง

การทดลองที่ 1

การนำข้อมูลเข้าโดยข้อมูลที่แสดงข้างต้นดังรูปที่ 4.2 นำมาใช้ในการทดลองจึงกำหนดพารามิเตอร์ดังนี้

Input Node 4 node

Hidden node 17 node

Output node 3 node

Learning rate 0.1

Tolerant rate 0.0001

Cycle 2500 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Train Percentage 80 %

Test Percentage 20 %

เมื่อได้กำหนดค่าให้แก่ระบบเป็นที่เรียบร้อยแล้ว ระบบจะสุ่มค่าถ่วงน้ำหนักให้แก่ระบบ โดยระบบจะสุ่มค่าอยู่ระหว่าง -0.5 ถึง 0.5 ดังตัวอย่างต่อไปนี้

Input-Hidden0------->-0.4194517384296125

Input-Hidden0------->0.07347840575168962

Input-Hidden0------->0.2786914103104746

Input-Hidden0------->0.10918567967752735

Input-Hidden0------->0.03364720169482971

Input-Hidden0------->-0.07862751488353836

Input-Hidden0------->-0.03305201799040036

Input-Hidden0------->-0.20060658540959564

Input-Hidden0------->-0.4699780440737755

Input-Hidden0------->0.03767898957186733

โดยเมื่อสุ่มค่าถ่วงน้ำหนักเป็นที่เรียบร้อยแล้วต่อมาโหลดข้อมูลแถวแรกเข้าสู่ระบบ โดยการทำงานในส่วนของ Forward นั้นระบบนี้เป็นการเปรียบเทียบของสองอัลกอริทึมในการเรียนรู้ในส่วนของ Backpropagation ดังนั้นในส่วนของ forward จึงทำงานเหมือนกัน โดยในการทำงานนี้ได้เลือก Activate ฟังก์ชั่นเป็น Sigmoid activation function ซึ่งใช้ทั้งในส่วนของ Hidden layer และ Output Layer เป็น Sigmoid activation function ดังนั้นข้อมูลที่ใช้ในการทดสอบจะต้องแปลงค่าให้อยู่ระหว่าง 0 ถึง 1 และจากการทำงานเมื่อได้ค่า Output ออกมาของแต่ละโหนดแล้วนั้นก็จะนำค่า Output ที่ได้ไปหาค่า Error เพื่อนำไปใช้ในการเรียนรู้แก่ระบบในแบบ Backpropagation โดยจะแยกเป็นการเรียนรู้ของสองอัลกอริทึม

การทดลองที่ 1

เป็นการนำข้อมูลเข้า โดยมีไฟล์ชื่อว่า Iris.xls ซึ่งมีอินพุท 4 ตัว และเอาท์พุทจำนวน 3 คลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 ผลการทดสอบ โดยใช้ Cross Entropy Function

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
6.677365	0.998742	1	0.001258	-9.20259	0.000101	0	-0.0001	-75.0458	2.56E-33	0	-2.6E-33	0
1.053359	0.741419	1	0.258581	21.11846	1	0	-1	-73.3449	1.4E-32	0	-1.4E-32	1
7.488619	0.999441	1	0.000559	-7.47373	0.000567	0	-0.00057	-80.8019	8.09E-36	0	-8.1E-36	0
5.899583	0.997267	1	0.002733	-8.45039	0.000214	0	-0.00021	-68.7555	1.38E-30	0	-1.4E-30	0
6.281664	0.998133	1	0.001867	-9.45262	7.85E-05	0	-7.8E-05	-72.5538	3.09E-32	0	-3.1E-32	0
6.265146	0.998102	1	0.001898	-4.5398	0.010563	0	-0.01056	-72.5873	2.99E-32	0	-3E-32	0
6.711888	0.998785	1	0.001215	-10.0689	4.24E-05	0	-4.2E-05	-75.3278	1.93E-33	0	-1.9E-33	0
7.084324	0.999163	1	0.000837	-7.3285	0.000656	0	-0.00066	-78.1946	1.1E-34	0	-1.1E-34	0
6.570135	0.9986	1	0.0014	-9.65449	6.41E-05	0	-6.4E-05	-74.6023	3.99E-33	0	-4E-33	0
6.656321	0.998716	1	0.001284	-8.03186	0.000325	0	-0.00032	-75.2697	2.05E-33	0	-2E-33	0

รูปที่ 4.4 ผลการทดสอบ Class 0

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
-24.4501	2.41E-11	0	-2.4E-11	-0.08606	0.478498	1	0.521502	-0.55414	0.364904	0	-0.3649	1
-1.8855	0.131758	0	-0.13176	0.033179	0.508294	1	0.491706	-1.07612	0.25424	0	-0.25424	1
-22.5923	1.54E-10	0	-1.5E-10	-0.02489	0.493777	1	0.506223	-0.78975	0.312222	0	-0.31222	1
-33.8658	1.96E-15	0	-2E-15	3.679786	0.975392	1	0.024608	-5.57765	0.003767	0	-0.00377	1
-22.1501	2.4E-10	0	-2.4E-10	-0.28872	0.428318	1	0.571682	-0.54219	0.367679	0	-0.36768	1
-0.30532	0.424257	0	-0.42426	0.067139	0.516778	1	0.483222	-3.08057	0.043916	0	-0.04392	1
-4.1602	0.015365	0	-0.01536	0.10011	0.525007	1	0.474993	-1.85987	0.134718	0	-0.13472	1
-2.3021	0.09095	0	-0.09095	0.109779	0.527417	1	0.472583	-1.31377	0.211856	0	-0.21186	1
-30.806	4.18E-14	0	-4.2E-14	1.599156	0.8319	1	0.1681	-4.9907	0.006755	0	-0.00675	1
-11.3245	1.21E-05	0	-1.2E-05	-0.02744	0.49314	1	0.50686	-1.32126	0.210609	0	-0.21061	1

รูปที่ 4.5 ผลการทดสอบ Class 1

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
-34.9588	6.57E-16	0	-6.6E-16	-1.5141	0.180332	0	-0.18033	1.901291	0.870038	1	0.129962	2
-34.3126	1.25E-15	0	-1.3E-15	-1.48208	0.185114	0	-0.18511	1.832548	0.862065	1	0.137935	2
-35.2047	5.14E-16	0	-5.1E-16	-1.29103	0.215679	0	-0.21568	1.610766	0.833518	1	0.166482	2
-31.7214	1.67E-14	0	-1.7E-14	-1.38095	0.200856	0	-0.20086	1.719432	0.848056	1	0.151944	2
-33.7639	2.17E-15	0	-2.2E-15	-1.47725	0.185843	0	-0.18584	1.837708	0.862677	1	0.137323	2
-35.0689	5.89E-16	0	-5.9E-16	-1.5	0.182426	0	-0.18243	1.875146	0.867053	1	0.132947	2
-35.3017	4.66E-16	0	-4.7E-16	-1.2225	0.227497	0	-0.2275	1.617137	0.8344	1	0.1656	2
-31.9941	1.27E-14	0	-1.3E-14	-1.34458	0.206757	0	-0.20676	1.612056	0.833697	1	0.166303	2
-23.4888	6.29E-11	0	-6.3E-11	-1.02966	0.26315	0	-0.26315	1.121685	0.754301	1	0.245699	2
-26.4105	3.39E-12	0	-3.4E-12	-1.05289	0.258671	0	-0.25867	1.06171	0.743017	1	0.256983	2

รูปที่ 4.6 ผลการทดสอบ Class 2

จากการทดสอบในส่วนของ Cross Entropy Function นั้น สามารถทำงานได้ดีโดยสามารถจัดแบ่งประเภทของข้อมูลได้ดีมาก โดยสังเกตได้จาก Class ที่เป็น Target หมายถึงการให้ระบบเรียนรู้เพื่อบอกให้ระบบทราบว่าต้องการให้ระบบเรียนรู้คลาสดังกล่าว และจากคลาสที่ระบบจัดประเภทมานั้น ก็ทำนายออกมาเป็นประเภทที่ได้เรียนรู้มาโดยคิดเป็นเปอร์เซ็นต์ความถูกต้องแล้วเกือบ 100 % ซึ่งมีประสิทธิภาพสูงมากในข้อมูลนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 ผลการทดสอบ โดยใช้ Error Gradient Function

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
2.893641	0.947531	1	0.052469	-1.51883	0.179634	0	-0.17963	-8.5012	0.000203	0	-0.0002	0
1.843271	0.863335	1	0.136665	0.910465	0.713095	0	-0.7131	-7.89332	0.000373	0	-0.00037	0
3.130556	0.958136	1	0.041864	-0.96701	0.275476	0	-0.27548	-8.69567	0.000167	0	-0.00017	0
1.95738	0.876249	1	0.123751	-1.72636	0.151054	0	-0.15105	-7.18153	0.00076	0	-0.00076	0
2.451794	0.920693	1	0.079307	-2.23608	0.096557	0	-0.09656	-7.64138	0.00048	0	-0.00048	0
2.372578	0.914712	1	0.085288	-0.51872	0.373152	0	-0.37315	-8.13406	0.000293	0	-0.00029	0
3.153101	0.959031	1	0.040969	-2.09025	0.110048	0	-0.11005	-8.63038	0.000179	0	-0.00018	0
2.937061	0.949648	1	0.050352	-0.92859	0.283212	0	-0.28321	-8.57088	0.00019	0	-0.00019	0
3.007885	0.952929	1	0.047071	-1.82168	0.139233	0	-0.13923	-8.68139	0.00017	0	-0.00017	0
2.820916	0.943796	1	0.056204	-1.04614	0.259967	0	-0.25997	-8.60336	0.000183	0	-0.00018	0

รูปที่ 4.7 ผลการทดสอบ Class 0

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
-2.94505	0.049971	0	-0.04997	-0.29401	0.427023	1	0.572977	-1.56161	0.173415	0	-0.17342	1
-3.44732	0.030849	0	-0.03085	-1.12501	0.245083	1	0.754917	-0.88849	0.291422	0	-0.29142	2
-2.79694	0.05749	0	-0.05749	-0.19114	0.45236	1	0.54764	-1.99817	0.119395	0	-0.1194	1
-1.67157	0.158215	0	-0.15822	0.429802	0.605826	1	0.394174	-3.29998	0.035572	0	-0.03557	1
-2.96122	0.049209	0	-0.04921	-0.52535	0.371601	1	0.628399	-1.51275	0.180531	0	-0.18053	1
-2.4516	0.079322	0	-0.07932	-1.07016	0.255374	1	0.744626	-1.99757	0.119459	0	-0.11946	1
-2.79467	0.057613	0	-0.05761	-0.91844	0.285277	1	0.714723	-1.61826	0.165444	0	-0.16544	1
-3.15018	0.041084	0	-0.04108	-0.80902	0.308099	1	0.691901	-1.48636	0.184469	0	-0.18447	1
-1.49884	0.182598	0	-0.1826	-0.0038	0.499051	1	0.500949	-3.41604	0.031798	0	-0.0318	1
-2.82839	0.055809	0	-0.05581	-0.69998	0.331817	1	0.668183	-1.67457	0.157816	0	-0.15782	1

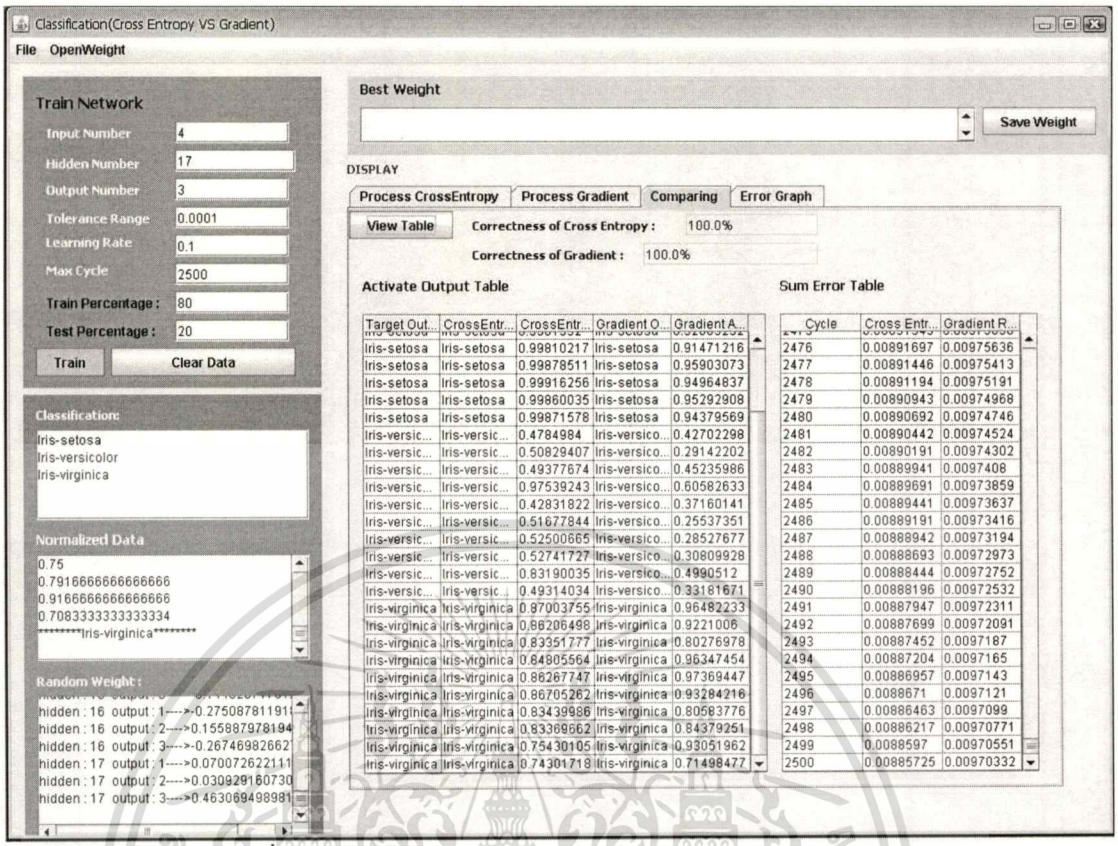
รูปที่ 4.8 ผลการทดสอบ Class 1

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
-6.66855	0.001269	0	-0.00127	-1.81809	0.139663	0	-0.13966	3.311532	0.964822	1	0.035178	2
-6.17458	0.002077	0	-0.00208	-1.68437	0.156517	0	-0.15652	2.471236	0.922101	1	0.077899	2
-5.13549	0.00585	0	-0.00585	-0.89452	0.290177	0	-0.29018	1.403696	0.80277	1	0.19723	2
-6.617	0.001336	0	-0.00134	-1.96891	0.122506	0	-0.12251	3.272536	0.963475	1	0.036525	2
-6.74038	0.001181	0	-0.00118	-2.28386	0.092468	0	-0.09247	3.611318	0.973694	1	0.026306	2
-6.25661	0.001914	0	-0.00191	-1.53685	0.176994	0	-0.17699	2.63119	0.932842	1	0.067158	2
-5.55212	0.003864	0	-0.00386	-0.35916	0.411164	0	-0.41116	1.423188	0.805838	1	0.194162	2
-5.4819	0.004144	0	-0.00414	-1.41812	0.194957	0	-0.19496	1.686721	0.843793	1	0.156207	2
-5.69488	0.003352	0	-0.00335	-2.46864	0.078086	0	-0.07809	2.594699	0.93052	1	0.06948	2
-4.62715	0.009688	0	-0.00969	-1.43363	0.192534	0	-0.19253	0.919719	0.714985	1	0.285015	2

รูปที่ 4.9 ผลการทดสอบ Class 2

ในส่วนของการเรียนรู้ Error Gradient Function ในส่วนของข้อมูลนี้ก็ทำได้ดีมากเช่นกัน โดยผลของการจัดแบ่งประเภทของข้อมูลก็ทำได้ถูกต้องเกือบ 100 % เช่นเดียวกันในส่วนของข้อมูลนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 แสดงประสิทธิภาพการทำงานของระบบ

ต่อไปเป็นการสังเกตค่าความผิดพลาดของแต่ละฟังก์ชั่น

RMSE: Cross Entropy Function

sumError :1

0.9999853780280684

sumError :220

0.032620645120388286

sumError :2500

0.008857245694994043

RMSE: Error Gradient Function

sumError :1

0.8801820968918068

sumError :220

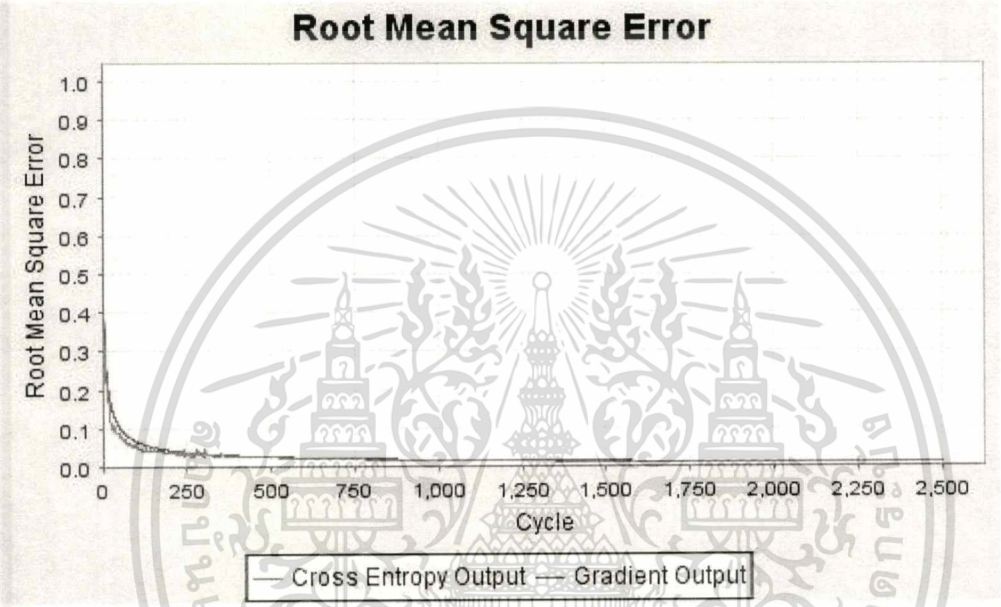
0.040595158277040605

sumError :2500

0.009703318548152528

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก Error ที่เกิดขึ้นจะสังเกตได้ว่าในช่วงแรกค่าผิดพลาดจะลดต่ำลงอย่างรวดเร็วในส่วนของ Cross Entropy Function ซึ่งแสดงให้เห็นถึงการทำงานของฟังก์ชั่นนี้ โดยอาจจะสังเกตเห็น Sum Error ที่รอบ 220 มีค่าต่ำกว่า Gradient Function ในขณะที่เริ่มต้นมีความผิดพลาดที่สูงกว่า และจะสังเกตได้จากกราฟ



รูปที่ 4.11 กราฟแสดงค่าความผิดพลาด

สรุปผลการทดลอง

การเรียนรู้ของ Cross Entropy Function นั้น มีการทำงานที่รวดเร็วกว่าในส่วนของการทำงานของอัลกอริทึมโดยการสังเกตว่าค่าผิดพลาดลดลงได้รวดเร็วมากกว่า จึงส่งผลให้ Activate วิ่งเข้าหา Target ได้เร็วกว่า แต่ลักษณะของผลทำงานออกมานั้นใกล้เคียงกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองที่ 2

การทดสอบในส่วนนี้ใช้ไฟล์ข้อมูลที่มีชื่อว่า Haberman.xls ซึ่งมีอินพุท 3 ตัว และมีเอาท์พุท 2 คลาส โดยมีรายละเอียดดังนี้

ตัวอย่างข้อมูลที่นำเข้าสู่ระบบ

39,66,0,2

39,63,0,1

39,67,0,1

39,58,0,1

39,59,2,1

39,63,4,1

40,58,2,1

40,58,0,1

40,65,0,1

41,60,23,27

โดยลักษณะของข้อมูลจึงกำหนดพารามิเตอร์ดังนี้

Input Node 3 node

Hidden node 37 node

Output node 2 node

Learning rate 0.1

Tolerant rate 0.0001

Cycle 2700 รอบ

Train Percentage 70 %

Test Percentage 30 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 การทดสอบระบบโดยใช้ Cross Entropy Function

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
0.655688	0.658291	1	0.341709	-0.64913	0.343185	0	-0.34318	0
2.713359	0.93781	1	0.06219	-2.73349	0.061026	0	-0.06103	0
1.83245	0.862053	1	0.137947	-1.8392	0.137146	0	-0.13715	0
2.34318	0.912391	1	0.087609	-2.36238	0.086087	0	-0.08609	0
1.221721	0.772366	1	0.227634	-1.225	0.227057	0	-0.22706	0
0.627683	0.651964	1	0.348036	-0.62309	0.34908	0	-0.34908	0
1.508116	0.818782	1	0.181218	-1.51582	0.180077	0	-0.18008	0
0.374887	0.592639	1	0.407361	-0.3693	0.408711	0	-0.40871	0
0.274665	0.568238	1	0.431762	-0.27195	0.43243	0	-0.43243	0
2.883037	0.947002	1	0.052998	-2.906	0.051858	0	-0.05186	0
0.280677	0.569712	1	0.430288	-0.27737	0.431099	0	-0.4311	0
-0.07029	0.482435	1	0.517565	0.069819	0.517448	0	-0.51745	1
1.638287	0.837302	1	0.162698	-1.65015	0.161088	0	-0.16109	0
1.658668	0.840059	1	0.159941	-1.66496	0.159098	0	-0.1591	0
1.658668	0.840059	1	0.159941	-1.66496	0.159098	0	-0.1591	0
1.77597	0.855199	1	0.144801	-1.79219	0.142805	0	-0.1428	0
0.251078	0.562442	1	0.437558	-0.24767	0.438397	0	-0.4384	0
0.258676	0.564311	1	0.435689	-0.25596	0.436358	0	-0.43636	0
1.658668	0.840059	1	0.159941	-1.66496	0.159098	0	-0.1591	0
1.558864	0.82619	1	0.17381	-1.57361	0.171703	0	-0.1717	0
0.605601	0.646937	1	0.353063	-0.60503	0.353193	0	-0.35319	0
0.605601	0.646937	1	0.353063	-0.60503	0.353193	0	-0.35319	0
3.002632	0.952693	1	0.047307	-2.9838	0.048163	0	-0.04816	0
2.854297	0.94554	1	0.05446	-2.87828	0.053238	0	-0.05324	0
3.200714	0.960861	1	0.039139	-3.2349	0.037873	0	-0.03787	0
2.815233	0.943493	1	0.056507	-2.85045	0.054658	0	-0.05466	0
2.815233	0.943493	1	0.056507	-2.85045	0.054658	0	-0.05466	0
0.660894	0.659461	1	0.340539	-0.66222	0.34024	0	-0.34024	0
1.040713	0.738988	1	0.261012	-1.05794	0.257703	0	-0.2577	0
-0.21204	0.447187	1	0.552813	0.211174	0.552598	0	-0.5526	1
2.988551	0.952054	1	0.047946	-2.97215	0.0487	0	-0.0487	0
0.136065	0.533964	1	0.466036	-0.13702	0.4658	0	-0.4658	0
0.727939	0.674353	1	0.325647	-0.73147	0.324872	0	-0.32487	0
2.781893	0.941689	1	0.058311	-2.8075	0.05692	0	-0.05692	0
0.24848	0.561802	1	0.438198	-0.24631	0.438732	0	-0.43873	0
2.983802	0.951837	1	0.048163	-2.96884	0.048854	0	-0.04885	0
2.339206	0.912072	1	0.087928	-2.37308	0.085249	0	-0.08525	0
0.422084	0.603982	1	0.396018	-0.42186	0.396071	0	-0.39607	0
-0.29696	0.4263	1	0.5737	0.279296	0.569374	0	-0.56937	1
2.339206	0.912072	1	0.087928	-2.37308	0.085249	0	-0.08525	0
0.339483	0.584065	1	0.415935	-0.33818	0.416253	0	-0.41625	0
2.979904	0.951658	1	0.048342	-2.96661	0.048957	0	-0.04896	0
2.690167	0.936444	1	0.063556	-2.68661	0.063768	0	-0.06377	0
0.260915	0.564861	1	0.435139	-0.25957	0.43547	0	-0.43547	0
0.998274	0.730719	1	0.269281	-1.01018	0.266944	0	-0.26694	0
0.998274	0.730719	1	0.269281	-1.01018	0.266944	0	-0.26694	0
2.511912	0.924973	1	0.075027	-2.54222	0.072951	0	-0.07295	0
1.781212	0.855846	1	0.144154	-1.80871	0.140795	0	-0.14079	0
0.56471	0.637542	1	0.362458	-0.56921	0.36142	0	-0.36142	0
0.293383	0.572824	1	0.427176	-0.29337	0.427178	0	-0.42718	0
0.538388	0.631437	1	0.368563	-0.56529	0.362324	0	-0.36232	0
2.0831	0.88925	1	0.11075	-2.11962	0.107204	0	-0.1072	0
1.242507	0.776	1	0.224	-1.26169	0.220684	0	-0.22068	0
0.094404	0.523583	1	0.476417	-0.09365	0.476605	0	-0.47661	0
0.713252	0.671119	1	0.328881	-0.72228	0.326891	0	-0.32689	0
0.350807	0.586813	1	0.413187	-0.35278	0.412709	0	-0.41271	0
-2.162	0.103215	1	0.896785	2.108669	0.891743	0	-0.89174	1
3.268728	0.96334	1	0.03666	-3.32212	0.03482	0	-0.03482	0
0.234151	0.558272	1	0.441728	-0.23455	0.441631	0	-0.44163	0
2.973133	0.951346	1	0.048654	-2.97232	0.048692	0	-0.04869	0
3.073656	0.955793	1	0.044207	-3.13147	0.041828	0	-0.04183	0
0.38765	0.595717	1	0.404283	-0.39111	0.403451	0	-0.40345	0
3.138142	0.958439	1	0.041561	-3.18775	0.039629	0	-0.03963	0
0.49432	0.621123	1	0.378877	-0.50073	0.377369	0	-0.37737	0
3.354857	0.966264	1	0.033736	-3.41415	0.031856	0	-0.03186	0
3.028983	0.953866	1	0.046134	-3.08465	0.043745	0	-0.04374	0
1.432353	0.807268	1	0.192732	-1.46212	0.188143	0	-0.18814	0
2.295976	0.908543	1	0.091457	-2.34614	0.087373	0	-0.08737	0

รูปที่ 4.12 ผลการทดสอบ Class 0โดยใช้ Cross Entropy Function

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
0.173531	0.543274	0	-0.54327	-0.1766	0.455964	1	0.544036	0
-3.96315	0.018649	0	-0.01865	3.960139	0.981296	1	0.018704	1
0.93005	0.717085	0	-0.71709	-0.93121	0.28268	1	0.71732	0
1.75185	0.852186	0	-0.85219	-1.76761	0.14584	1	0.85416	0
1.018791	0.734737	0	-0.73474	-1.02286	0.264471	1	0.735529	0
0.240445	0.559823	0	-0.55982	-0.23812	0.44075	1	0.55925	0
-3.08408	0.043769	0	-0.04377	3.058516	0.955149	1	0.044851	1
3.002632	0.952693	0	-0.95269	-2.9838	0.048163	1	0.951837	0
0.167805	0.541853	0	-0.54185	-0.16863	0.457942	1	0.542058	0
0.894333	0.709784	0	-0.70978	-0.91329	0.286328	1	0.713672	0
2.983802	0.951837	0	-0.95184	-2.96884	0.048854	1	0.951146	0
2.556391	0.928002	0	-0.928	-2.59278	0.069604	1	0.930396	0
0.063396	0.515844	0	-0.51584	-0.06216	0.484464	1	0.515536	0
0.133759	0.53339	0	-0.53339	-0.13373	0.466618	1	0.533382	0
2.979904	0.951658	0	-0.95166	-2.96661	0.048957	1	0.951043	0
0.120096	0.529988	0	-0.52999	-0.12813	0.468011	1	0.531989	0
0.474493	0.616447	0	-0.61645	-0.47567	0.383275	1	0.616725	0
2.954538	0.950478	0	-0.95048	-3.00067	0.047396	1	0.952604	0
0.125349	0.531296	0	-0.5313	-0.12415	0.469004	1	0.530996	0
2.972972	0.951338	0	-0.95134	-2.9677	0.048907	1	0.951093	0
0.596859	0.644937	0	-0.64494	-0.61721	0.350417	1	0.649583	0
3.322549	0.965194	0	-0.96519	-3.37913	0.032954	1	0.967046	0
1.84	0.862949	0	-0.86295	-1.87704	0.132729	1	0.867271	0
2.924658	0.949052	0	-0.94905	-2.98914	0.047919	1	0.952081	0
2.747179	0.939754	0	-0.93975	-2.793	0.057703	1	0.942297	0

รูปที่ 4.13 ผลการทดสอบ Class 1 โดยใช้ Cross Entropy Function

ผลการทดสอบของ Cross Entropy Function โดยใช้ Data ที่มีชื่อว่า Haberman.xls เป็นที่น่าพอใจระดับหนึ่ง โดยผลการทดสอบนั้นทำได้ถูกต้องถึง 73 เปอร์เซ็นต์ และมี Error ลดต่ำลงอย่างรวดเร็วอย่างเห็นได้ชัดจากกราฟที่แสดงในส่วนถัดในรูปที่ 4.17 แสดงให้เห็นถึงประสิทธิภาพการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การทดสอบระบบโดยใช้ Error Gradient Function

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
-1.67316	0.158003	1	0.841997	1.672669	0.841931	0	-0.84193	1
-1.46842	0.187183	1	0.812817	1.46692	0.812589	0	-0.81259	1
-1.50109	0.182262	1	0.817738	1.499569	0.81751	0	-0.81751	1
-1.39016	0.199383	1	0.800617	1.388237	0.800311	0	-0.80031	1
-1.44972	0.190044	1	0.809956	1.448057	0.809699	0	-0.8097	1
-1.75274	0.147702	1	0.852298	1.752047	0.852211	0	-0.85221	1
-1.35027	0.205826	1	0.794174	1.348291	0.79385	0	-0.79385	1
-2.05483	0.113565	1	0.886435	2.055134	0.886465	0	-0.88647	1
-1.52228	0.179126	1	0.820874	1.521259	0.820724	0	-0.82072	1
-1.7614	0.146615	1	0.853385	1.760517	0.853274	0	-0.85327	1
-1.61352	0.1661	1	0.8339	1.612706	0.833787	0	-0.83379	1
-4.1426	0.015633	1	0.984367	4.150186	0.984483	0	-0.98448	1
-1.54636	0.175613	1	0.824387	1.54487	0.824172	0	-0.82417	1
-1.9262	0.127171	1	0.872829	1.925816	0.872786	0	-0.87279	1
-1.9262	0.127171	1	0.872829	1.925816	0.872786	0	-0.87279	1
-1.74233	0.149017	1	0.850983	1.741338	0.850857	0	-0.85086	1
-2.50266	0.075672	1	0.924328	2.504204	0.924436	0	-0.92444	1
-1.56343	0.173156	1	0.826844	1.56267	0.826736	0	-0.82674	1
-1.9262	0.127171	1	0.872829	1.925816	0.872786	0	-0.87279	1
-2.61706	0.068048	1	0.931952	2.618683	0.932054	0	-0.93205	1
-1.85176	0.135667	1	0.864333	1.851296	0.864279	0	-0.86428	1
-1.85176	0.135667	1	0.864333	1.851296	0.864279	0	-0.86428	1
-2.1528	0.10407	1	0.89593	2.153103	0.895958	0	-0.89596	1
-2.24956	0.095387	1	0.904613	2.249763	0.90463	0	-0.90463	1
-2.2128	0.098607	1	0.901393	2.212882	0.9014	0	-0.9014	1
-2.17409	0.102101	1	0.897899	2.174104	0.8979	0	-0.8979	1
-2.17409	0.102101	1	0.897899	2.174104	0.8979	0	-0.8979	1
-2.04721	0.114334	1	0.885666	2.047255	0.88567	0	-0.88567	1
-3.14985	0.041097	1	0.958903	3.153154	0.959033	0	-0.95903	1
-5.01203	0.006613	1	0.993387	5.02212	0.993453	0	-0.99345	1
-2.54292	0.072904	1	0.927096	2.544143	0.927179	0	-0.92718	1
-4.46937	0.011325	1	0.988675	4.477117	0.988762	0	-0.98876	1
-2.24256	0.095993	1	0.904007	2.243106	0.904054	0	-0.90405	1
-2.44513	0.079796	1	0.920204	2.445797	0.920254	0	-0.92025	1
-2.14821	0.104499	1	0.895501	2.149016	0.895577	0	-0.89558	1
-2.72746	0.061372	1	0.938628	2.729116	0.938723	0	-0.93872	1
-2.51476	0.07483	1	0.92517	2.515682	0.925234	0	-0.92523	1
-2.38105	0.08463	1	0.91537	2.382176	0.915458	0	-0.91546	1
-2.89524	0.05239	1	0.94761	2.897371	0.947716	0	-0.94772	1
-2.51476	0.07483	1	0.92517	2.515682	0.925234	0	-0.92523	1
-2.48022	0.077257	1	0.922743	2.481631	0.922844	0	-0.92284	1
-2.92207	0.051073	1	0.948927	2.924181	0.949029	0	-0.94903	1
-3.02097	0.046488	1	0.953512	3.023433	0.953622	0	-0.95362	1
-2.52737	0.073962	1	0.926038	2.52918	0.926162	0	-0.92616	1
-2.81766	0.056377	1	0.943623	2.819668	0.943729	0	-0.94373	1
-2.81766	0.056377	1	0.943623	2.819668	0.943729	0	-0.94373	1
-3.0206	0.046504	1	0.953496	3.022632	0.953586	0	-0.95359	1
-2.86253	0.054037	1	0.945963	2.86445	0.946061	0	-0.94606	1
-2.96547	0.04901	1	0.95099	2.968111	0.951113	0	-0.95111	1
-2.91637	0.05135	1	0.94865	2.919206	0.948788	0	-0.94879	1
-3.44409	0.030946	1	0.969054	3.447176	0.969147	0	-0.96915	1
-3.2524	0.037241	1	0.962759	3.255284	0.962863	0	-0.96286	1
-3.20729	0.038892	1	0.961108	3.210275	0.961219	0	-0.96122	1
-4.79364	0.008214	1	0.991786	4.80146	0.991849	0	-0.99185	1
-3.35448	0.033749	1	0.966251	3.358118	0.96637	0	-0.96637	1
-3.30492	0.035403	1	0.964597	3.30877	0.964728	0	-0.96473	1
-4.46704	0.011351	1	0.988649	4.473607	0.988723	0	-0.98872	1
-3.53035	0.028461	1	0.971539	3.533645	0.97163	0	-0.97163	1
-3.69689	0.0242	1	0.9758	3.701788	0.975915	0	-0.97592	1
-4.07532	0.016703	1	0.983297	4.080094	0.983375	0	-0.98338	1
-3.8689	0.020454	1	0.979546	3.873118	0.97963	0	-0.97963	1
-4.03479	0.017382	1	0.982618	4.040239	0.982711	0	-0.98271	1
-4.14149	0.01565	1	0.98435	4.146155	0.984421	0	-0.98442	1
-3.8766	0.0203	1	0.9797	3.881936	0.979805	0	-0.97981	1
-4.2974	0.013421	1	0.986579	4.302502	0.986646	0	-0.98665	1
-4.62815	0.009678	1	0.990322	4.634033	0.990378	0	-0.99038	1
-4.50796	0.010901	1	0.989099	4.514507	0.98917	0	-0.98917	1
-5.0972	0.006077	1	0.993923	5.104726	0.993969	0	-0.99397	1

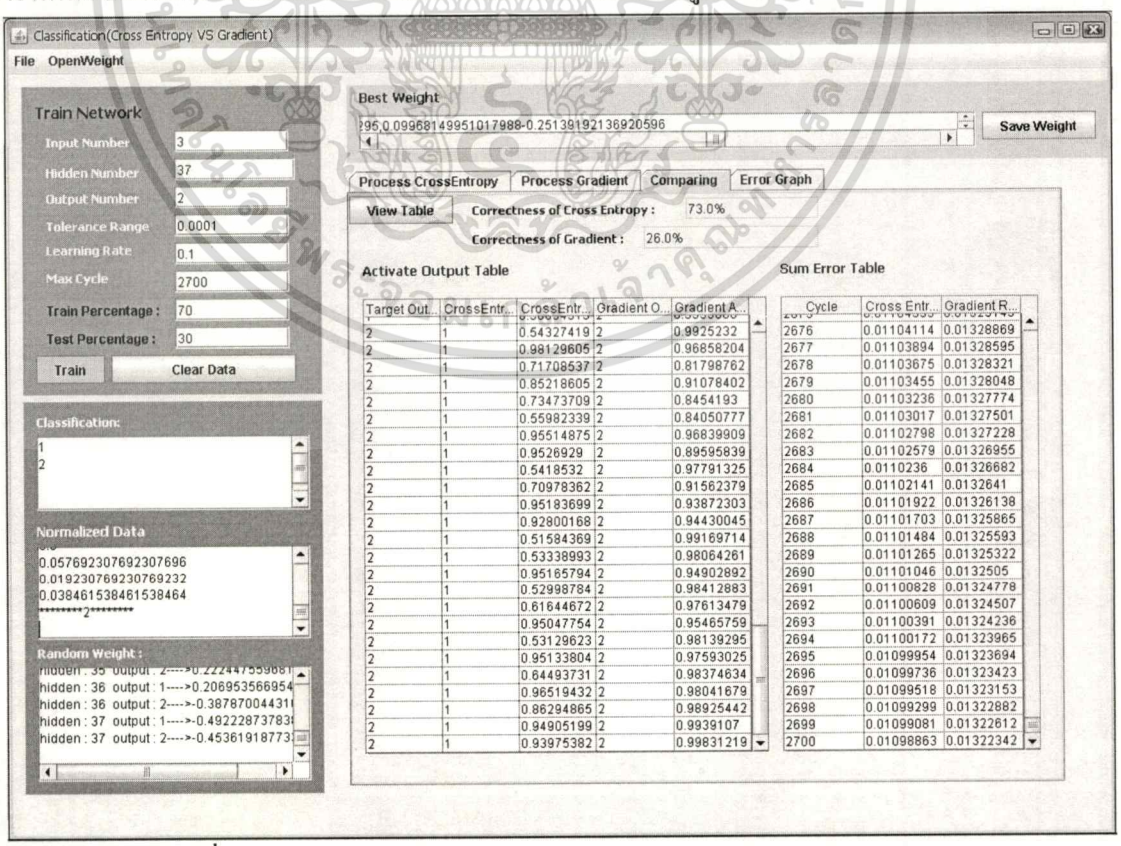
รูปที่ 4.14 ผลการทดสอบ Class 0โดยใช้ Error Gradient Function Test

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sum	Activate	Target	Error	Sum	Activate	Target	Error	Class
0.173531	0.543274	0	-0.54327	-0.1766	0.455964	1	0.544036	0
-3.96315	0.018649	0	-0.01865	3.960139	0.981296	1	0.018704	1
0.93005	0.717085	0	-0.71709	-0.93121	0.28268	1	0.71732	0
1.75185	0.852186	0	-0.85219	-1.76761	0.14584	1	0.85416	0
1.018791	0.734737	0	-0.73474	-1.02286	0.264471	1	0.735529	0
0.240445	0.559823	0	-0.55982	-0.23812	0.44075	1	0.55925	0
-3.08408	0.043769	0	-0.04377	3.058516	0.955149	1	0.044851	1
3.002632	0.952693	0	-0.95269	-2.9838	0.048163	1	0.951837	0
0.167805	0.541853	0	-0.54185	-0.16863	0.457942	1	0.542058	0
0.894333	0.709784	0	-0.70978	-0.91329	0.286328	1	0.713672	0
2.983802	0.951837	0	-0.95184	-2.96884	0.048854	1	0.951146	0
2.556391	0.928002	0	-0.928	-2.59278	0.069604	1	0.930396	0
0.063396	0.515844	0	-0.51584	-0.06216	0.484464	1	0.515536	0
0.133759	0.53339	0	-0.53339	-0.13373	0.466618	1	0.533382	0
2.979904	0.951658	0	-0.95166	-2.96661	0.048957	1	0.951043	0
0.120096	0.529988	0	-0.52999	-0.12813	0.468011	1	0.531989	0
0.474493	0.616447	0	-0.61645	-0.47567	0.383275	1	0.616725	0
2.954538	0.950478	0	-0.95048	-3.00067	0.047396	1	0.952604	0
0.125349	0.531296	0	-0.5313	-0.12415	0.469004	1	0.530996	0
2.972972	0.951338	0	-0.95134	-2.9677	0.048907	1	0.951093	0
0.596859	0.644937	0	-0.64494	-0.61721	0.350417	1	0.649583	0
3.322549	0.965194	0	-0.96519	-3.37913	0.032954	1	0.967046	0
1.84	0.862949	0	-0.86295	-1.87704	0.132729	1	0.867271	0
2.924658	0.949052	0	-0.94905	-2.98914	0.047919	1	0.952081	0
2.747179	0.939754	0	-0.93975	-2.793	0.057703	1	0.942297	0

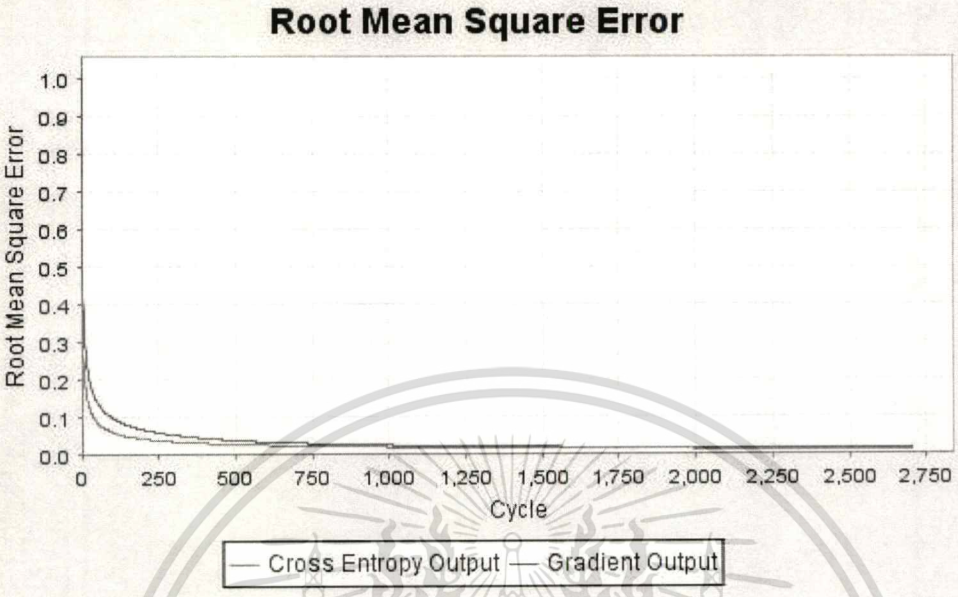
รูปที่ 4.15 ผลการทดสอบ Class 1 โดยใช้ Error Gradient Function Test

จากผลการทดสอบการทำงานของทั้งสองอัลกอริทึมเปรียบเทียบได้ดังรูปที่ 4.15



รูปที่ 4.16 แสดงการเปรียบเทียบการทำงานของสองอัลกอริทึม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



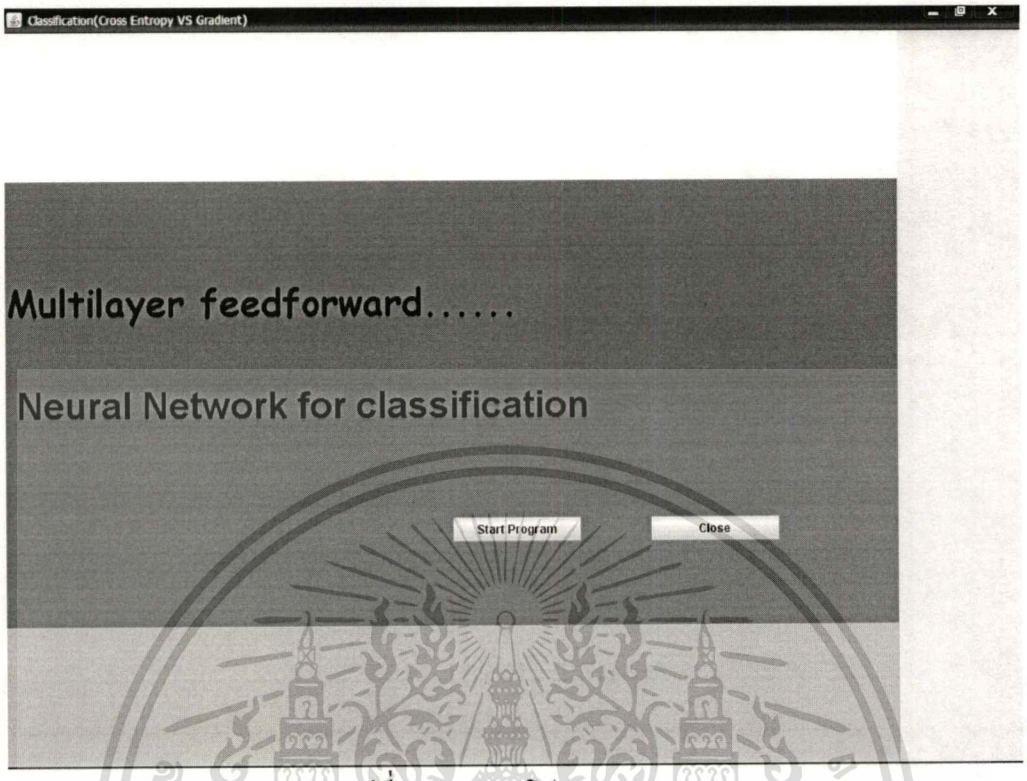
รูปที่ 4.17 แสดงในส่วนของ RMSE ของสองอัลกอริทึม

สรุปผลการทดลอง

จากการทำงานที่เกิดขึ้นจะสังเกตได้ว่าการทำงานของ Cross Entropy Function ทำงานได้ถูกต้องกว่าในจำนวนรอบที่เท่ากัน โดยได้ถึง73 % ในขณะที่ Error Gradient Function ทำงานได้ถูกต้องเพียง 26 % จึงจะเห็นได้ว่า Cross Entropy Function ทำงานได้อย่างเป็นที่น่าพอใจ จากกราฟที่แสดงทำให้เห็นว่า Error ของ Cross Entropy Function มีค่า error ลดลงอย่างรวดเร็ว เป็นผลทำงานเข้าใกล้สู่เป้าหมายซึ่งส่งผลให้ระบบเข้าใกล้ค่าเป้าหมายได้รวดเร็วกว่า Error Gradient Function และส่งผลให้ทำงานได้รวดเร็วกว่าในจำนวนรอบที่เท่ากัน

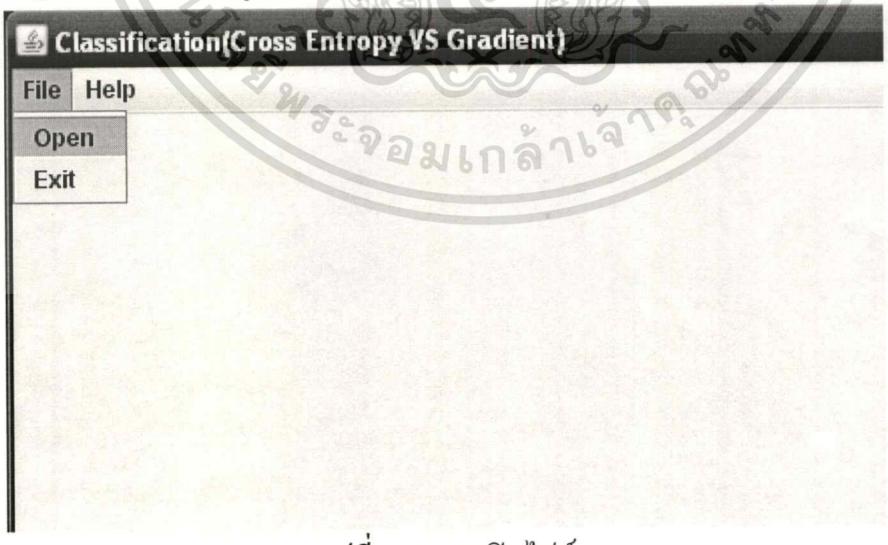
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 การทำงานของโปรแกรม



รูปที่ 4.18 แสดงโปรแกรม

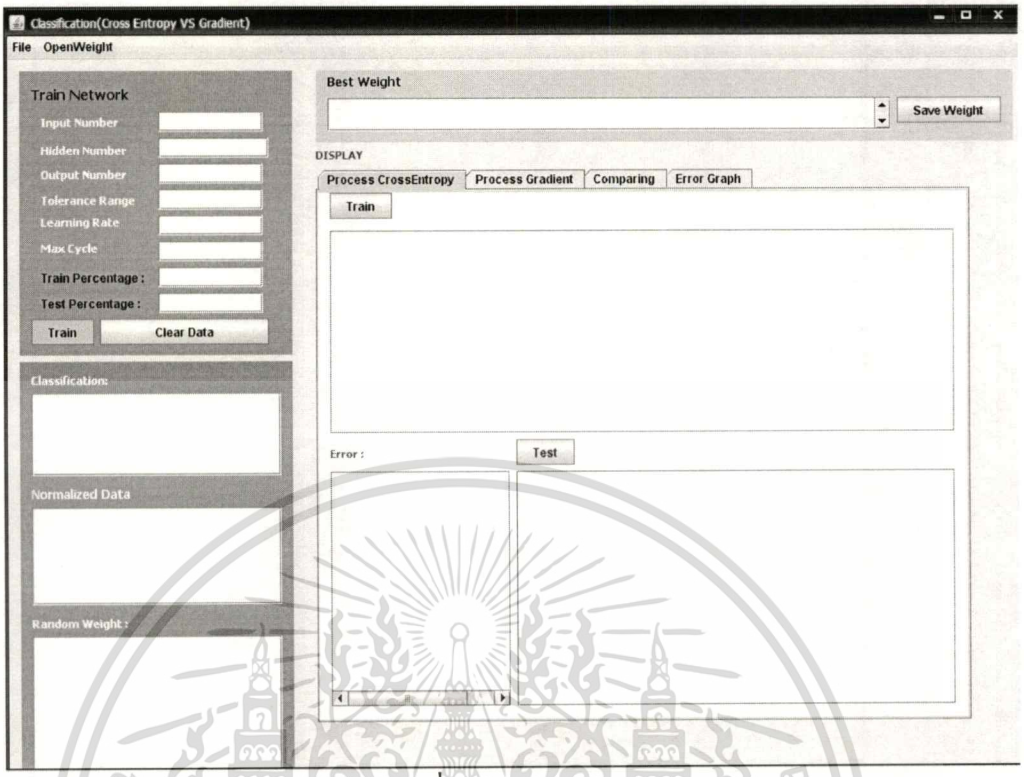
เมื่อเปิดโปรแกรมจะแสดงในส่วนของหน้าตาของโปรแกรม ซึ่งส่วนนี้เป็นเสมือนส่วนต้อนรับของโปรแกรมเพื่อให้สามารถเลือกที่จะ Start Program เพื่อเข้าสู่การทำงานของระบบ หรือต้องการออกจากโปรแกรมเมื่อต้องการออกจากระบบ เลือก Close เมื่อผู้ใช้เลือก Start Program แล้วก็จะเข้าสู่ระบบดังรูปที่ 4.18



รูปที่ 4.19 การเปิดไฟล์

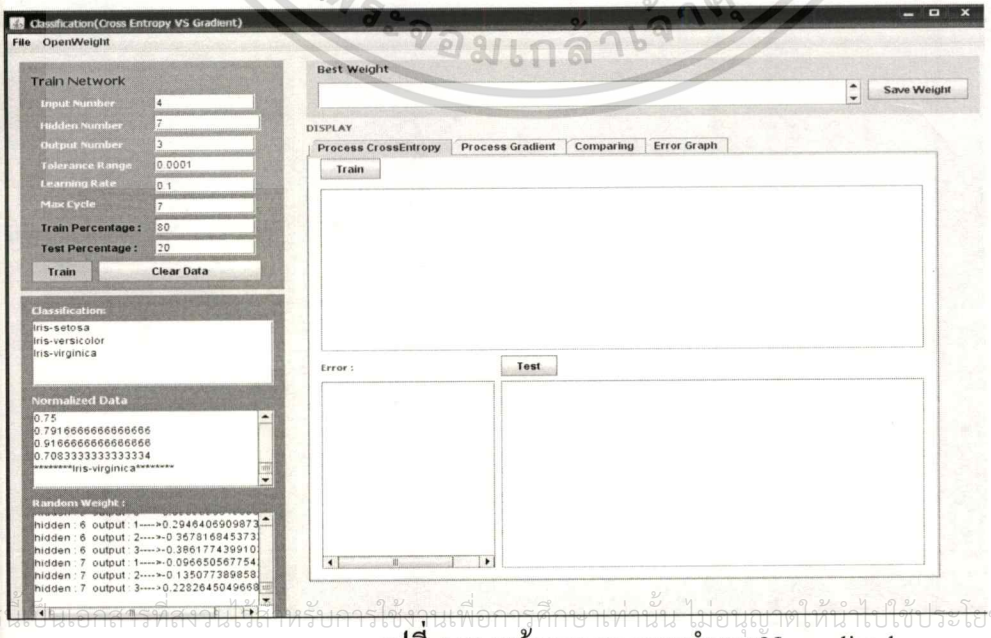
ผู้ใช้เลือกเปิดไฟล์ OPEN ที่ต้องการนำข้อมูลเข้าสู่ระบบ และ HELP คำแนะนำในการป้อนพารามิเตอร์ต่างรวมถึงการทำงานโดยรวม ดังรูปที่ 4.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 โปรแกรมหน้าหลัก

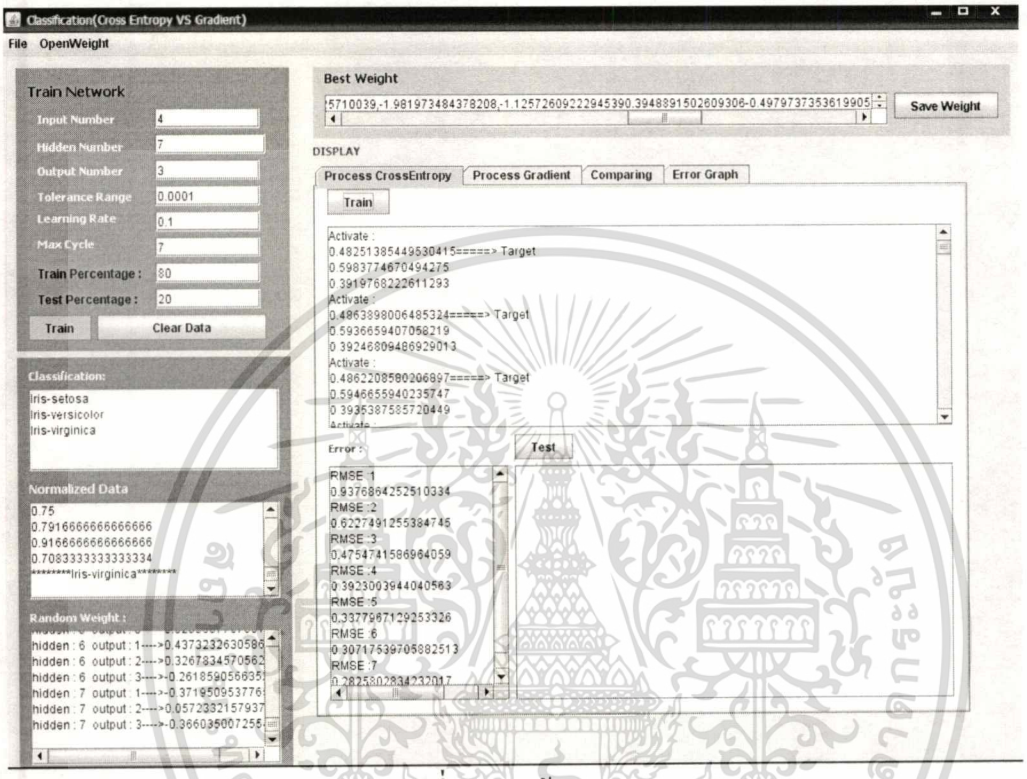
เมื่อเลือกไฟล์ที่ต้องการนำข้อมูลเข้าเรียบร้อยแล้ว ผู้ใช้ป้อนค่าพารามิเตอร์ต่างๆที่กำหนดไว้เพื่อให้ระบบทำงาน โดยการทำงานเริ่มจากที่ผู้ใช้กดปุ่มเพื่อทำการฝึกฝนแก่ระบบ โดยระบบจะเก็บชื่อของคลาสเพื่อแสดงให้ผู้ใช้เห็น ระบบจะนำไฟล์ที่นำเข้ามาทำการ Normalized Data และระบบจะสุ่มค่าถ่วงน้ำหนักให้แก่ระบบโดยอัตโนมัติ ระบบจะทำการฝึกฝนแก่ระบบเพื่อให้ผู้ใช้สามารถติดตามดังรูปที่ 4.20



รูปที่ 4.21 หน้าจอแสดงการทำงานNormalized

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

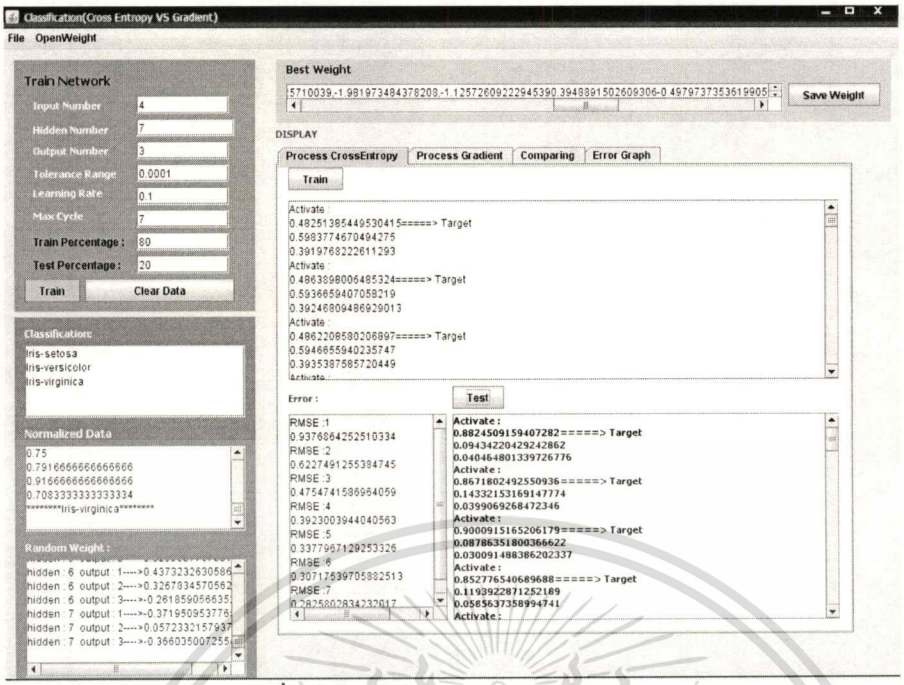
ความเคลื่อนไหวในการฝึกฝนได้โดยในส่วนของการฝึกฝนระบบจะแสดงค่า Activate Output Node ซึ่งหมายถึงค่าระบบให้ออกมาแต่ละโหนดจะมีค่าแนวโน้มที่จะเพิ่มขึ้นเพื่อเข้าใกล้ค่า Target และส่วนโหนดอื่นๆค่ามีแนวโน้มที่จะออกห่างเรื่อยๆ และค่า Mean Square Error แสดงในเพื่อให้เห็นถึงค่าแนวโน้มที่ลดลงเรื่อยและจำนวนรอบที่ระบบทำงานดังรูปที่ 4.22



รูปที่ 4.22 หน้าจอแสดงการทำงาน

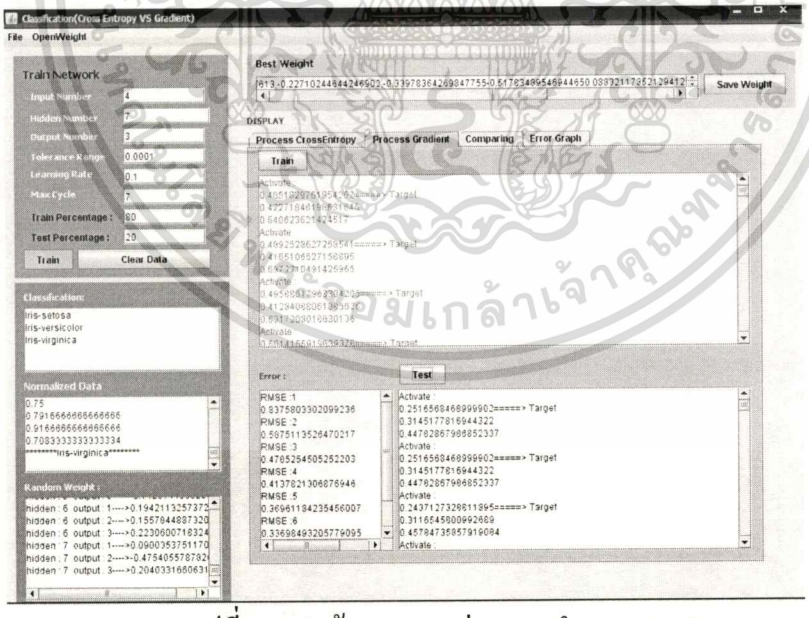
เมื่อการฝึกฝนระบบเป็นที่เรียบร้อยแล้วนั้นผู้ใช้สามารถเก็บค่าถ่วงน้ำหนักที่ดีที่สุดในการฝึกฝนระบบมาเก็บไว้เป็นไฟล์เพื่อนำไปทดสอบแก่ระบบได้ โดยผู้ใช้สามารถจะกดปุ่ม TEST เพื่อให้ระบบทำการทดสอบข้อมูลที่เก็บไว้เพื่อทดสอบได้ทันที ดังรูปที่ 4.23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 หน้าจอแสดงส่วนของการทดสอบ

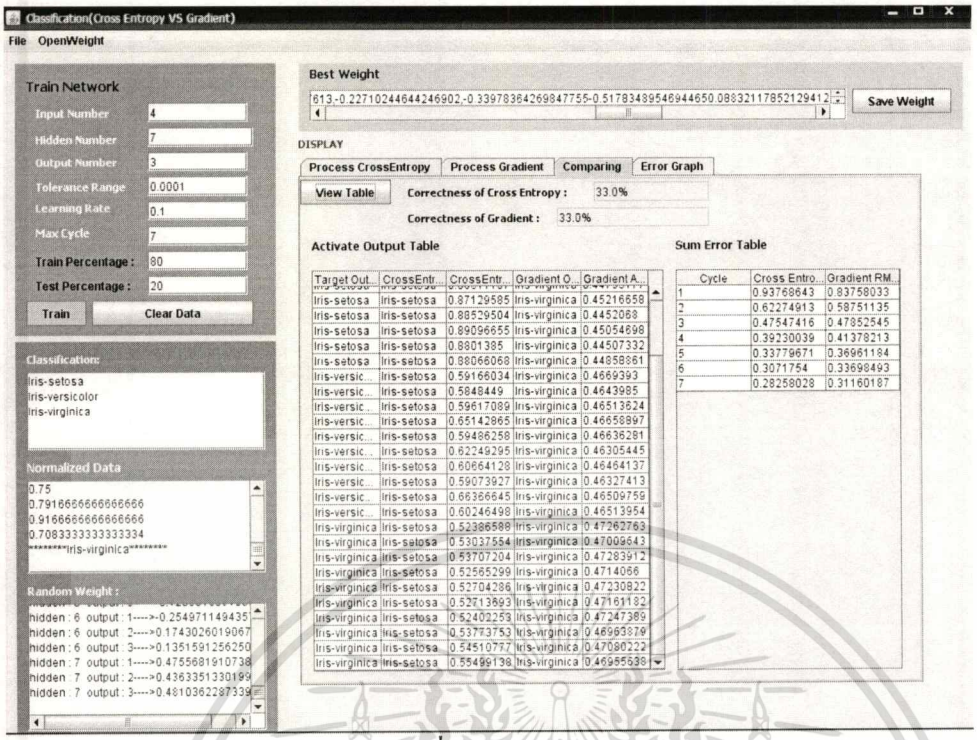
ในส่วนของ Process Gradient ก็ทำงานลักษณะเดียวกันเพียงต่างกันที่ การเรียนรู้ใช้อัลกอริทึม Error Gradient function ในการปรับเพื่อให้เครือข่ายเกิดการเรียนรู้ดังรูป 4.24



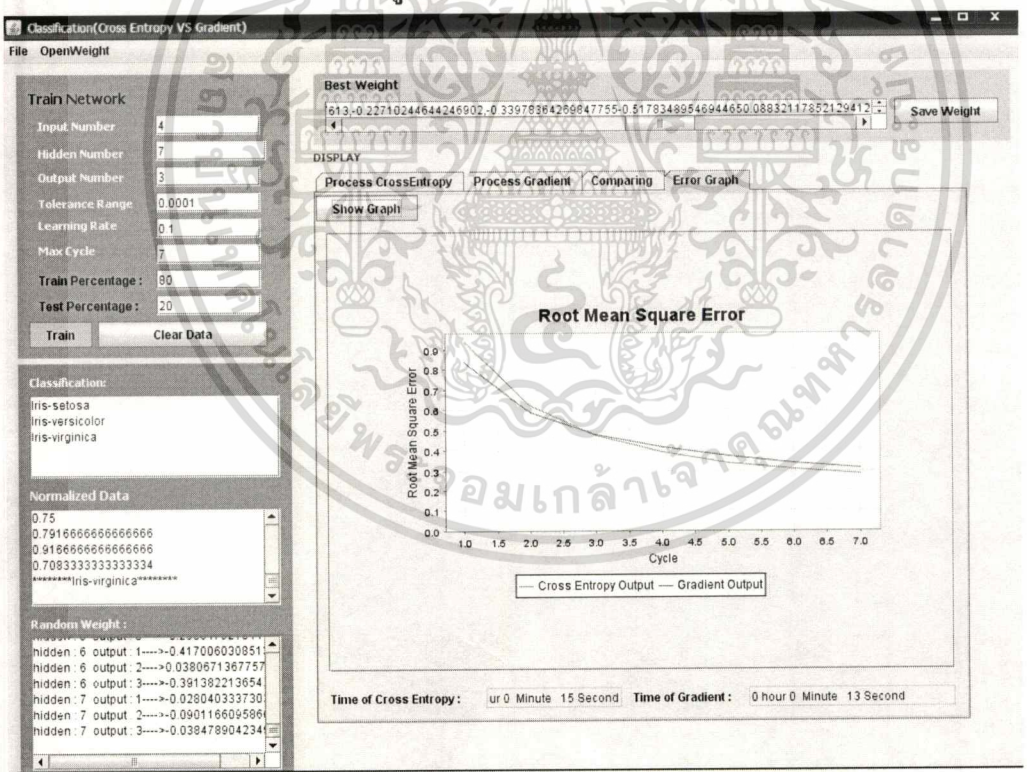
รูปที่ 4.24 หน้าจอแสดงส่วนการทำงาน Gradient

ในส่วนของ Comparing นั้น ระบบจะนำผลของการ TEST มาใช้ในการเปรียบเทียบว่าอัลกอริทึมตัวไหนทำงานได้ดีกว่ากันในแง่ของประสิทธิภาพในการจัดแบ่งประเภท และเรื่องของเวลาการทำงานของแต่ละอัลกอริทึมดังรูปที่ 4.25 ในส่วนสุดท้ายนี้เป็นส่วนแสดงผลในรูปแบบของกราฟเส้นเพื่อให้เห็นและเข้าใจได้ง่ายขึ้นดังรูปที่ 4.26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ทำซ้ำ ดัดแปลง หรือเผยแพร่ต่อ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.25 หน้าจอหลักแสดงการเปรียบเทียบ



รูปที่ 4.26 หน้าจอหลักแสดงกราฟเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการศึกษาและข้อเสนอแนะ

5.1 สรุปการพัฒนาระบบงาน

จากการทดลองที่ได้มานั้น ค่าพารามิเตอร์มีความสำคัญอย่างมาก ซึ่งผู้ใช้งานระบบที่ได้พัฒนาขึ้นนี้ต้องมีความเข้าใจในระดับหนึ่งจึงจะสามารถใช้งานได้ดีและใช้งานได้อย่างถูกต้อง โดยพารามิเตอร์ที่มีได้แก่

Input Node มีความสำคัญเพราะผู้ใช้ต้องป้อนจำนวนเท่ากับจำนวนของอินพุทที่ป้อนเข้าสู่ระบบ

Hidden Node มีผลต่อการจัดประเภทของข้อมูลเนื่องจากหากกำหนดให้จำนวนของ Hidden Node มากไปก็จะไม่ถูกต้องและทำให้การคำนวณเพิ่มขึ้นตาม ซึ่งจะถูกเรียกว่า Over fitting และการประมวลผลจึงจำเป็นต้องใช้ทรัพยากรต่างๆเพิ่มขึ้นอีก และหากผู้ใช้กำหนดจำนวน Hidden Node น้อยไปผลก็ไม่ถูกต้องเช่นกันเนื่องจากการที่ Hidden Node น้อยไปจะไม่เกิดการเรียนรู้ การที่ผลถูกต้องต้องลองกำหนดดูโดยจากการทดลองประมาณสองเท่าของอินพุทน่าจะมีแนวโน้มที่ถูกต้องแต่ก็ไม่แน่เสมอไปทั้งนี้ยังขึ้นกับอีกหลายๆปัจจัยที่ถูกกำหนดในพารามิเตอร์อื่นๆ

Output Node เป็นการที่ผู้ใช้ป้อนเพื่อให้สอดคล้องกับประเภทที่ต้องการจะให้ระบบจะดูข้อมูลเพื่อให้สอดคล้องกับข้อมูลที่ฝึกฝนให้แก่ระบบด้วย

Tolerance Error คือค่า Error ที่ยอมรับได้เป็นค่าหนึ่งที่บอกว่าระบบควรจะหยุดเมื่อใด ปกติเมื่อค่า Error ของระบบต่ำกว่า ระบบจะหยุดทำงาน หากตั้งค่านี้สูงก็จะทำให้ระบบทำนายผิดพลาดเนื่องจากยังค่า Error ยังมีค่าไม่ต่ำเพียงพอที่จะยอมรับได้

Learning Rate คือการที่ไบแอสค่าให้แก่ระบบเพื่อให้ระบบสามารถเรียนรู้ได้รวดเร็วขึ้น ปกติจะกำหนดไม่เกิน 0.1

Cycle ค่าแต่ละรอบของการทำงานของข้อมูลชุดหนึ่ง ซึ่งเป็นตัวแปรหนึ่งที่ให้ระบบหยุดการฝึกฝนเมื่อครบตามรอบที่ได้กำหนดไว้

จากการเปรียบเทียบของทั้งสองอัลกอริทึมสรุปได้ว่า การจัดแบ่งประเภทเมื่อมีการฝึกฝนแล้ว ให้ผลได้ไม่แตกต่างกันมากนักในบางข้อมูล แต่บางข้อมูลที่นำเข้าก็ส่งผลแตกต่างกัน ข้อแตกต่างนั้นอยู่ที่ Cross Entropy Function มีการทำงานที่เร็วกว่าโดยสังเกตได้ว่าแต่ละรอบของการฝึกฝนที่ข้อมูลเข้านั้น ค่า Summation และค่า Error แต่ละรอบจะลดลงได้เร็วกว่า Error Gradient Function

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ประโยชน์ที่ได้รับจากการพัฒนาระบบ

สามารถนำโปรแกรมไปปรับใช้หรือประยุกต์ใช้ในเชิงธุรกิจทำให้ช่วยลดเวลาการทำงานและลดภาระงานได้มากยิ่งขึ้นและมีความถูกต้อง

5.3 ปัญหาและอุปสรรค

เกิดจากความซับซ้อนในส่วนของอัลกอริทึมซึ่งมีความยุ่งยากอยู่พอสมควร และบางครั้งโครงข่ายประสาทเทียมเป็นการทำงานลักษณะการทำนายผล บางครั้งอาจจะไม่สามารถหาสาเหตุของความผิดพลาดที่อาจจะเกิดขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ธนิศา เครือไวศยวรรณ. 2547. **Computer Program Development 2 (JAVA).** กรุงเทพฯ: สถาบันเทคโนโลยีเจ้าคุณทหารลาดกระบัง

Bishop, Christopher M, **Neural Networks for Pattern Recognition.** Oxford university press,2000.Rudy Setiono. **A penalty-function approach for pruning feedforward neural network.** [Online]. Available : http: //www.mitpressjournals.org/doi/pdf/10.1162/neco.1997.9.1.185

Deite,H.M. l, **JAVA How to Program.** PEARSON Education, Inc.,2005

Hongjun Lu, Setiono, Rudy. et al. **NeuroRule: A Connectionist Appoach to Data mining.** [Online]. Available :http :// www.vldb.org/conf/1995/p478.pdf

Negnevitsky, Michael, **Artificial Intelligence.** Addison Wesley,2002.

Richard J. Roiger,Michael W.Geatz. **Data Mining A Tutorial-Based Prime.** Addison Wesley,2003.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ-นามสกุล นาย รุ่งโรจน์ แสงธรรมชัย

วันเดือนปีเกิด 5 กุมภาพันธ์ 2520

สถานที่เกิด กรุงเทพมหานคร

ประวัติการศึกษา

ปริญญาตรี	บริหารธุรกิจบัณฑิต
สถานที่สำเร็จการศึกษา	มหาวิทยาลัยศรีปทุม
ปีที่สำเร็จการศึกษา	2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้