ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การทำนายราคาหลักทรัพย์โดยใช้โครงข่ายประสาทเทียม แบบแบคพรอพพาเกชั้นและดาต้าไมนิ่ง

Prediction of Stock Price By Backpropagation Neural Network and Data Mining

โดย

นางสาวกัลยาณี ทองเจริญสกุล

รหัส 42067054

H001762

อาจารย์ที่ปรึกษา

ดร. วรพจน์ กรีสุระเดช

วัน เดือน ปี 09 ส.ค. 2550
เลขทะเบียน 01762
เลขเรียกหนังสือ วพ ก412ก 2543
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2543
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การทำนายราคาหลักทรัพย์โดยใช้โครงข่ายประสาทเทียมแบบแบพรอพพาเกชั้น (Backpropagation Neural Network) และดาต้าไมนิ่ง
นักศึกษา	นางสาวกัลยาณี ทองเจริญสกุล
อาจารย์ที่ปรึกษา	ดร.วรพจน์ กรีสุระเดช
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2543

บทคัดย่อ

ปัจจุบันการลงทุนในตลาดหลักทรัพย์เป็นทางเลือกหนึ่งในการลงทุนที่ได้รับความสนใจจากผู้มีเงินออม เนื่องจากมีโอกาสที่จะได้รับผลตอบแทนที่สูงกว่าการออมเงินประเภทอื่น ดังนั้นโครงการนี้จะเป็นการเก็บรวบรวมราคาหลักทรัพย์ที่ผ่านมาในอดีต และศึกษาทฤษฎีอนุกรมเวลาและโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชั่น (Backpropagation Neural Network) พร้อมทั้งทฤษฎีดาต้าไมนิ่ง เพื่อนำทฤษฎีดังกล่าวมาประยุกต์ใช้เพื่อช่วยในการพยากรณ์ราคาหลักทรัพย์ แล้วเปรียบเทียบราคาหลักทรัพย์ที่ได้จากการใช้โครงข่ายประสาทเทียมแบบแบคพรอพพาเกชั่น (Backpropagation Neural Network) กับราคาหลักทรัพย์จริงว่ามีความคาดเคลื่อนเท่าใด เพื่อเป็นแนวทางให้นักลงทุนต่างๆสามารถพยากรณ์ราคาหลักทรัพย์ในอนาคตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Prediction of Stock Price By Backpropagation Neural Network and Data Mining
Student	Miss Kalayanee Thongcharoensakul
Advisor	Dr. Worapoj Kreesuradej
Level of Student	Master of Science in Information Technology
Major	Information Science
Academic Year	2000

ABSTRACT

The present investment in the stock market is alternative for investment that interest for saving Because it has opportunity high reward more than other investment. So that Investor have analysis from basic factor for efficiency of decision. This project studies the theory of Backpropagation Neural Network ,Time Series and Data Mining is used to forecase the Stock price. By comparing the result received from Backpropagation Neural Network with result data in the past. It is forecast technique use Stock price in the past for forecast the present Stock price.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณท่านอาจารย์ ดร.วรพจน์ กรีสุระเดช ที่กรุณาให้คำปรึกษา และช่วยให้แนวทางในการทำโครงงานนี้ เป็นผลให้โครงงานนี้สำเร็จลุล่วงไปด้วยดี

และขอบคุณเพื่อนๆพี่ที่ให้คำปรึกษา และเป็นกำลังใจในการทำโครงงานนี้ตลอดมา

กัลยาณี ทองเจริญสกุล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญภาพ	VII
บทที่	
1. บทนำ	1
1.1 ความเป็นมา	1
1.2 วัตถุประสงค์	3
1.3 ขอบเขตของการศึกษา	3
1.4 ขั้นตอนของการดำเนินการศึกษา	4
1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงงาน	4
2. แนวคิดและทฤษฎีที่เกี่ยวข้อง	5
2.1 ทฤษฎีดาต้าไมนิ่ง	5
2.1.1 บทนำ	5
2.1.2 ความหมายของดาต้าไมนิ่ง	6
2.1.3 กระบวนการทำงานของดาต้าไมนิ่ง	6
2.1.4 การปฏิบัติการของดาต้าไมนิ่ง : โมเดลการพยากรณ์	10
2.2 การวิเคราะห์อนุกรมเวลา	11
2.2.1 คุณสมบัติของอนุกรมเวลา	12
2.3 ทฤษฎีนิวรอนเน็ตเวิร์ค	13
2.3.1 ทฤษฎีเซลล์ประสาททางชีววิทยาเบื้องต้น	13
2.3.2 ตัวแบบของเซลล์ประสาทเทียม	14
2.3.3 การฝึกสอนโครงข่ายประสาทเทียม	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4 โครงข่ายนิวรอนแบบแบพรอพพาเกชั่น 17
2.4 การใช้ทฤษฎีนิวรอนเน็ตเวิร์คมาพยากรณ์ราคาหลักทรัพย์ 19
3. วิธีดำเนินการศึกษา 22
3.1 ส่วนของโครงสร้างแบบจำลอง 22
3.2 ส่วนของอัลกอริทึม 22
3.3 ส่วนของข้อมูล 28
3.4 ส่วนของโปรแกรม 28
4. ผลการดำเนินการศึกษา 36
4.1 การเปรียบเทียบผลที่ได้จากการฝึกสอน 36
4.2 การเปรียบเทียบผลที่ได้จากการทดสอบ 37
5. สรุปผลการศึกษาและข้อเสนอแนะ 38
5.1 สรุปผลที่ได้จากการดำเนินการศึกษา 38
5.2 ข้อเสนอแนะ 38
5.2.1 การเพิ่มความแม่นยำในการทำนาย 38
5.2.3 การนำไปใช้ 38
บรรณานุกรม 40
ภาคผนวก 41
ประวัติผู้เขียน 105

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่

2.1 แสดง Transformation of Time Series to Strandard Form 12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ภาพที่		หน้า
2.1	แสดงกระบวนการทำ Data Mining	6
2.2	แสดงเปอร์เซ็นต์การทำงานแต่ละขั้นตอนใน Data Mining	7
2.3	แสดงข้อมูลอายุ และรายได้ที่ได้ ซึ่งจะมีข้อมูลที่แตกต่างจากข้อมูลอื่น	9
2.4	แสดงถึง Predictive modeling approach	10
2.5	แสดงTime Series with a Window of Size 5	11
2.6	แสดง Iterative Data Preparation Model	12
2.7	แสดงเซลล์ประสาทมนุษย์ทางชีววิทยา	14
2.8	แสดง Multi-layer feedforward neural network	14
2.9	แสดง A unit of an artificial neural network	15
2.10	แสดงลักษณะของสมการ Threshold	16
2.11	แสดงกราฟของฟังก์ชั่นไบนารี่ซิกมอยด์	16
2.12	แสดงกราฟของฟังก์ชั่นโพล่าซิกมอยด์	17
2.13	แสดง Multilayer Feedforward Networks	18
2.14	แสดง Neural net architecture for predicting stock price	20
2.15	แสดง Neural net architecture for predicting daily gold prices based on 5 day pattern	21
3.1	แสดงหน้าจอเมนูหลัก	29
3.2	แสดงหน้าจอ Load Data	30
3.3	แสดงหน้าจอ Load DataBase เข้ามาใน Neuron Network	31
3.4	แสดงหน้าจอ Create Neuron Network Model	32
3.5	แสดง Make prediction	33
3.6	แสดงกราฟเปรียบเทียบระหว่างค่าจริงกับค่าที่พยากรณ์ได้	34
4.1	แสดงกราฟเปรียบเทียบระหว่างค่าจริงกับค่าที่พยากรณ์ได้ (ช่วงการฝึกสอน)	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 แสดงกราฟเปรียบเทียบระหว่างค่าจริงกับค่าที่พยากรณ์ได้ (ช่วงการทดสอบ) 37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมา

ตลาดทุน (Capital Market) เป็นแหล่งกลางระดมเงินและจัดสรรเงินทุนระยะยาวที่ มีบทบาทสำคัญต่อการขยายการลงทุนของธุรกิจต่างๆในระบบเศรษฐกิจ การขยายหรือการเพิ่มการลงทุนในกิจการใดๆ นอกจากการกู้ยืมเงินจากสถาบันการเงิน ซึ่งจะต้องมีภาระการใช้เงินคืนทั้งเงินต้นและดอกเบี้ยแล้ว อีกวิธีหนึ่งก็คือการขยายทุนเพิ่มหุ้นออกขายให้ประชาชน กระจายความเป็นเจ้าของกิจการออกไป การซื้อขายหุ้นในครั้งแรกเรียกว่า ตลาดแรกหรือตลาดหลักทรัพย์ออกใหม่ (Primary Market or Mew Issued Market) หรือเรียกโดยทั่วๆไปว่า ตลาดซื้อขาย ซึ่งเป็นแหล่งระดมเงินทุนเพื่อนำไปใช้ในธุรกิจอย่างแท้จริง นักลงทุนที่ซื้อหลักทรัพย์ในตลาดแรก ย่อมมีความต้องการให้หลักทรัพย์ที่ซื้อมาได้สามารถแปรเปลี่ยนเป็นเงินสดเมื่อใดก็ได้ตามต้องการ จึงมีตลาดรองหรือตลาดซื้อขายหลักทรัพย์ (Secondary Market or Trading Market) เกิดขึ้น เพื่อสนับสนุนให้ตลาดแรกมีสภาพคล่อง

การซื้อขายหลักทรัพย์ในตลาดหลักทรัพย์มีการขยายตัวค่อนข้างรวดเร็วในระยะเวลาที่ผ่านมา จำนวนบริษัทจดทะเบียนได้เพิ่มขึ้นอย่างต่อเนื่อง ระดับราคาและปริมาณการซื้อขายโดยเฉลี่ยได้ปรับตัวสูงขึ้นมาก ทำให้มีผู้สนใจเข้ามาลงทุนในตลาดหลักทรัพย์เพิ่มจำนวนมากขึ้นทุกขณะ

การลงทุนในหลักทรัพย์อาจเป็นการลงทุนที่ให้ผลตอบแทนสูงแต่มีความเสี่ยงสูงเช่นกัน ดังนั้นในการลงทุนในหลักทรัพย์อย่างมีหลักเกณฑ์ ผู้ลงทุนจำเป็นต้องพิจารณาปัจจัยต่างๆที่มีผลในการกำหนดมูลค่าและราคาหลักทรัพย์เพื่อวิเคราะห์ว่าควรเลือกลงทุนในหลักทรัพย์ใด จะอาศัยการวิเคราะห์ปัจจัยพื้นฐาน (Fundamental Analysis) ซึ่งเป็นการวิเคราะห์ภาวะเศรษฐกิจ การวิเคราะห์ภาวะอุตสาหกรรม การวิเคราะห์ภาวะการดำเนินงานของบริษัท และการวิเคราะห์ภาวะการเมือง เพื่อช่วยตัดสินใจว่าควรจะเลือกซื้อหลักทรัพย์ใดจึงจะคุ้มกับการลงทุนและลดความเสี่ยง

การวิเคราะห์หลักทรัพย์เป็นการวิเคราะห์ที่ประกอบด้วยขั้นตอนการเก็บรวบรวมข้อมูลการศึกษาและการวิเคราะห์ข้อมูลและการตีความผลที่ได้จากการศึกษาเพื่อนำมากำหนดหลักทรัพย์หรือกลุ่มหลักทรัพย์ที่จะลงทุน

สำหรับเหตุผลทางทฤษฎีที่ใช้สนับสนุนการวิเคราะห์ปัจจัยพื้นฐานนั้นมาจากรากฐานแนวความคิดที่ว่ามูลค่าที่แท้จริง (Intrinsic) และราคาตลาดของหลักทรัพย์ (Market Value) ควรจะอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใกล้เคียงกัน ในทางปฏิบัตินักวิเคราะห์ทางปัจจัยพื้นฐานจะวิเคราะห์หลักทรัพย์เพื่อหามูลค่าที่แท้จริงของหลักทรัพย์ โดยคิดในรูปแบบของมูลค่าปัจจุบัน (Present Value) ซึ่งเป็นการประเมินมูลค่าของหลักทรัพย์ที่จะลงทุน ซึ่งคาดว่าจะได้รับผลตอบแทนในอนาคตทั้งในรูปกำไรจากการขายหลักทรัพย์ (Capital Gain) และผลตอบแทนจากการลงทุนในรูปของดอกเบี้ยหรือเงินปันผล (Yield) หลังจากนั้นจึงนำมาเปรียบเทียบกับราคาของหลักทรัพย์ในตลาด หากมูลค่าที่แท้จริงสูงกว่าราคาตลาดของหลักทรัพย์ก็จะตัดสินใจซื้อ ในทำนองกลับกันหากมูลค่าที่แท้จริงต่ำกว่าราคาตลาดก็ตัดสินใจขาย

การวิเคราะห์หลักทรัพย์โดยใช้ปัจจัยพื้นฐาน เป็นการวิเคราะห์ที่ลำดับการพิจารณาจากสภาพเศรษฐกิจมายังสภาพอุตสาหกรรมตลอดถึงผลการดำเนินงานของบริษัท โดยรวบรวมข้อมูลทางเศรษฐกิจ อุตสาหกรรมและของบริษัทมาวิเคราะห์แต่ละส่วนเพื่อนำไปสู่การคาดการณ์ผลการดำเนินงานในอนาคตของบริษัท

การวิเคราะห์หลักทรัพย์ตามวิธีปัจจัยพื้นฐาน มีขั้นตอนที่สำคัญดังนี้

- การวิเคราะห์เศรษฐกิจโดยทั่วไป (Economic Analysis)

การวิเคราะห์เศรษฐกิจเน้นการวิเคราะห์ภาวะเศรษฐกิจที่เป็นมาตลอดจนแนวโน้มของภาวะเศรษฐกิจในอนาคต ซึ่งอาจเกี่ยวข้องสัมพันธ์กับเศรษฐกิจโลกและแนวนโยบายเศรษฐกิจของรัฐบาลว่าจะมีผลกระทบต่อธุรกิจที่ออกหลักทรัพย์หรือไม่ เพียงใด นอกจากนั้น การวิเคราะห์เศรษฐกิจนี้ยังรวมถึงการวิเคราะห์วัฏจักรธุรกิจและนโยบายทางเศรษฐกิจที่สำคัญของรัฐบาลด้วย เช่น นโยบายการเงินและการคลัง เป็นต้น

- การวิเคราะห์อุตสาหกรรม (Industry Analysis)

การวิเคราห์อุตสาหกรรมเน้นการวิเคราะห์วงจรอุตสาหกรรม (Industry Life Cycle) สภาพการตลาดและการแข่งขันตลอดจนอนาคตของอุตสาหกรรมว่าจะมีแนวโน้มอัตราการเจริญเติบโตอย่างไร ซึ่งขึ้นอยู่กับปัจจัยหลายอย่างด้วยกัน เช่น นโยบายของรัฐบาลที่จะให้การสนับสนุนหรือเป็นอุปสรรค โดยโครงสร้างการเปลี่ยนแปลงของระบบภาษีของรัฐบาล โครงสร้างของอุตสาหกรรมแต่ละประเภท เป็นต้น

- การวิเคราะห์บริษัท (Company Analysis)

การวิเคราะห์บริษัทเป็นขั้นสุดท้ายของการวิเคราะห์ โดยเน้นการวิเคราะห์ประเภทของบริษัท และประเภทของหลักทรัพย์ โดยจะวิเคราะห์ทั้งเชิงคุณภาพ (Qualitative Analysis) อันได้แก่ การวิเคราะห์ประสิทธิภาพการบริหารของผู้บริหาร ส่วนแบ่งหารตลาดของบริษัท โครงการขยายโรงงานของบริษัท เป็นต้น และการวิเคราะห์เชิงปริมาณ (Quantitative Analysis) อันได้แก่ การ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิเคราะห์จากงบแสดงฐานะการเงินของธุรกิจในอดีตและปัจจุบัน เพื่อนำมาประมาณการกำไรต่อหุ้นและราคาหุ้นในอนาคตได้

อีกทางหนึ่งเป็นวิธีการวิเคราะห์ทางเทคนิค (Technical Analysis) ซึ่งเป็นการศึกษาข้อมูลในอดีตและพฤติกรรมการเคลื่อนไหวของราคาหลักทรัพย์ เพื่อนำไปคาดการณ์ราคาหลักทรัพย์นั้นๆในอดีต

แนวคิดเกี่ยวกับพฤติกรรมการเคลื่อนไหวของราคาหลักทรัพย์ในอดีตสามารถอธิบายการเคลื่อนไหวของราคาหลักทรัพย์ในอนาคต โดยการวิเคราะห์อนุกรมเวลา (Time Series Analysis) ซึ่งเป็นเครื่องมือทางคณิตศาสตร์และสถิติชนิดหนึ่ง ซึ่งจะต้องสร้างตัวแบบ (Model) ที่สามารถแสดงความเคลื่อนไหวของราคาหลักทรัพย์ ตัวแบบที่ดีและเหมาะสมจะสามารถนำมาใช้อธิบายการเคลื่อนไหวของราคาหลักทรัพย์และนำมาใช้ในการทำนายราคาหลักทรัพย์ในอนาคตได้

1.2 วัตถุประสงค์ของการศึกษา

การศึกษาโครงงานนี้มีวัตถุประสงค์คือ

1. ศึกษาทฤษฎีคาต้าไมนิ่ง

2..ศึกษาวิธีการนำทฤษฎีโครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backprogation Neuron Network) มาประยุกต์ใช้ในการทำนายข้อมูลอนุกรมเวลา

3.ทำนายราคาหลักทรัพย์โดยใช้ทฤษฎีโครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backprogation Neuron Network)

4.เพื่อศึกษาเปรียบราคาหลักทรัพย์จริงกับราคาหลักทรัพย์ที่ได้จากการนำทฤษฎีโครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backprogation Neuron Network) มาใช้ว่ามีความคลาดเคลื่อนเท่าใด

5. สามารถป้อนข้อมูลเข้าไประบบเพื่อให้ระบบทำนายราคาหลักทรัพย์

1.3 ขอบเขตของการศึกษา

การศึกษาโครงงานนี้กำหนดขอบเขตในการศึกษาเป็นการพัฒนาทฤษฎีโครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backpropagation Neuron Network) ในการทำนายราคาหลักทรัพย์ในอนาคต โดยใช้ราคาปิดหลักทรัพย์ของ TFB ย้อนหลังตั้งแต่วันที่ 5 มิถุยายน 2536 ถึงวันที่ 5 มิถุนายน 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนของการดำเนินการศึกษา

การศึกษาโครงงานนี้จะดำเนินการพัฒนาทฤษฎีโครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backpropagation Neuron Network) เพื่อทำนายราคาหลักทรัพย์ โดยใช้โปรแกรม Microsoft Visual basic 6.0 เป็นเครื่องมือในการพัฒนา โดยมีขั้นตอนในการดำเนินการดังนี้

1. จัดเตรียมและปรับปรุงข้อมูลราคาปิดหลักทรัพย์ของ TFB ย้อนหลังตั้งแต่วันที่ 5 มิถุนายน 2536 ถึงวันที่ 5 มิถุนายน 2541ให้มีค่าอยู่ในช่วง 0 ถึง 1 เพื่อกำหนดเป็น Input-Output pairs

2. ข้อมูลที่ได้จากข้างต้นจะถูกนำไปสร้างโครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backpropagation Neuron Network) ซึ่งมีขั้นตอนดังนี้

2.1 ให้ค่าเริ่มต้นกับ weight เป็นค่า random น้อยๆ

2.2 เริ่มทำการเรียนรู้โดยการทดสอบก่อนว่าจะหยุดเรียนรู้แล้วหรือยัง ถ้ามีข้อผิดพลาดในการเรียนรู้น้อยก็หยุดการเรียนรู้ แต่ถ้าข้อผิดพลาดในการเรียนรู้ยังมากอยู่ก็ต้องเรียนรู้ต่อไป ในที่กำหนด Input และ Output ขึ้นมา 1 คู่ แล้วนำเข้าไปใน Network เพื่อเรียนรู้และปรับค่าน้ำหนักให้เหมาะสม

2.3 การปรับค่าความผิดพลาดโดยการย้อนเอาค่าความผิดพลาดกลับเข้ามาใน Neural Network โดยย้อนกลับทีละ Layer เพื่อปรับค่าน้ำหนัก แล้วเอาข้อมูลคู่ใหม่เข้ามาทดสอบต่อไป แล้วก็ปรับน้ำหนักไปเรื่อยๆจนความผิดพลาดน้อยก็หยุดการเรียนรู้

1.5 ประโยชน์ที่คาดว่าจะได้รับจากโครงงาน

1. เข้าใจถึงหลักการของดาต้าไมนิ่ง
2. เข้าใจถึงหลักการของโครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backpropagation Neuron Network)
3. สามารถประยุกต์ใช้โครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backpropagation Neuron Network) ช่วยในการพยากรณ์ราคาหลักทรัพย์ได้
4. สามารถช่วยให้นักลงทุนต่าง ๆ สามารถวิเคราะห์ได้ว่าจะเลือกลงทุนในหลักทรัพย์ใด
5. สามารถเป็นต้นแบบในการนำไปประยุกต์ใช้กับการทำนายอื่นๆที่มีลักษณะเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
แนวคิดและทฤษฎีที่เกี่ยวข้อง

แนวคิดในการนำทฤษฎีใช้โครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backpropagation Neuron Network) มาช่วยในการพยากรณ์ราคาหลักทรัพย์ซึ่งเป็นข้อมูลอนุกรมเวลารูปแบบหนึ่ง เริ่มจากลักษณะของราคาหลักทรัพย์จะมีการเคลื่อนไหวที่มีลักษณะไม่เป็นเชิงเส้น จึงทำให้การวิเคราะห์และการทำนายด้วยวิธีการวิเคราะห์อนุกรมเวลาแบบต่างๆที่เป็นเชิงเส้น อาจทำให้เกิดความคลาดเคลื่อนได้มากและอาจจะนำไปสู่การตัดสินใจในการลงทุนมีโอกาสผิดพลาดสูง จึงควรที่จะสร้างระบบที่สามารถนำมาใช้ในการวิเคราะห์และทำนายข้อมูลที่มีลักษณะไม่เป็นเชิงเส้นโดยเฉพาะมาทำนายราคาหลักทรัพย์ ระบบใช้โครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backpropagation Neuron Network) เป็นระบบหนึ่งที่สามารถตอบสนองความต้องการนี้ได้

การทำนายราคาหลักทรัพย์โดยใช้ใช้โครงข่ายนิวรอนแบบแบคพรอพเกชั่น (Backpropagation Neuron Network) มีแนวคิดพื้นฐานที่ว่า พฤติกรรมของราคาหลักทรัพย์ในอดีตจะต้องมีลักษณะเกี่ยวเนื่องหรือมีอิทธิพลต่อพฤติกรรมของราคาหลักทรัพย์ในปัจจุบันและสามารถนำไปใช้ในการทำนายราคาหลักทรัพย์ในอนาคตได้

2.1 ทฤษฎี Data Mining

2.1.1 บทนำ

เมื่อข้อมูลข่าวสารมีความสำคัญต่อการดำเนินธุรกิจซึ่งปัจจุบันมีการแข่งขันกันสูง จึงมีความจำเป็นที่จะต้องเก็บข้อมูลและหาสารสนเทศมาสนับสนุนในการตัดสินใจต่างๆและให้เขาถึงพฤติกรรมของข้อมูลเพราะการมีสารสนเทศที่ดีและมีวิธีการวิเคราะสารสนเทศที่ได้เหมาะสมแล้วก็เท่ากับเป็นผู้นำทางธุรกิจ ส่งผลให้มีการเพิ่มจำนวนฐานข้อมูลในการเก็บข้อมูลมากขึ้นจนทำให้เกิดเป็นคลังข้อมูล หรือ Data Warehouse แต่ปัญหาที่ตามมาคือการวิเคราะห์ข้อมูลจำนวนมากไม่ใช่เรื่องง่าย จึงได้มีการคิดค้นเทคโนโลยีที่เรียกว่า Data Mining มาช่วยในการวิเคราะห์เพื่อให้ทราบถึงความสัมพันธ์ในรูปแบบต่างๆที่ซ่อนอยู่ในคลังข้อมูล

Data Mining ยังเป็นเครื่องมือในการทำนายแนวโน้ม และพฤติกรรมของข้อมูลในอนาคต ตลอดจนถึงการเก็บความรู้เพื่อใช้ในการตัดสินใจ ทำให้ Data Mining เป็นเครื่องมือที่ใช้ในระบบช่วยตัดสินใจ (decision support system)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

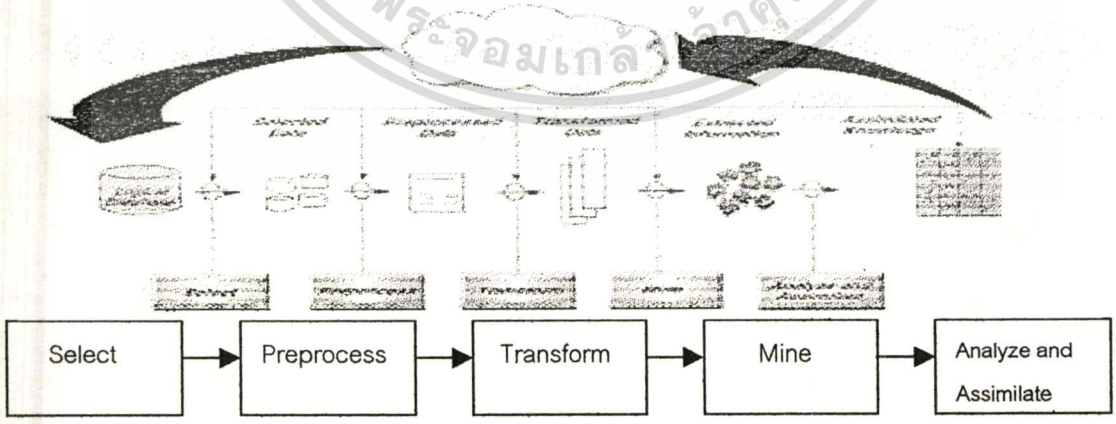
2.1.2 ความหมายของดาต้าไมนิ่ง (Data Mining)

Data Mining เป็นกระบวนการที่สำคัญที่จะดึงส่วนที่เป็นนัยของข้อมูลที่เราไม่ทราบ และทำให้เกิดศักยภาพในการใช้ข้อมูลในฐานข้อมูล

การนำเอาแนวโน้มของข้อมูลและ Information ต่างๆที่ซ่อนอยู่ในฐานข้อมูลขนาดใหญ่ออกมานั้นเป็นสิ่งที่มีความสำคัญมาก เนื่องจากว่าแม้เราจะมีข้อมูลจำนวนมากแต่ไม่สามารถนำมาใช้ประโยชน์ได้ก็เท่ากับว่าสูญเปล่า จึงเกิดการพัฒนาเครื่องมือที่ช่วยค้นหาความรู้ในฐานข้อมูลและให้เข้าใจถึงความสัมพันธ์ของข้อมูลต่างๆ เพื่อนำไปใช้ประโยชน์ ความแตกต่างของ Data Mining กับเครื่องมือในการวิเคราะห์อื่นๆก็คือเทคนิคในการสำรวจข้อมูลและหาความสัมพันธ์ที่เกิดขึ้นในฐานข้อมูล เครื่องมือจำนวนมากจะสนับสนุนสิ่งที่ใกล้เคียงกับความเป็นจริง โดยใช้สมมติฐานเกี่ยวกับความสัมพันธ์ระหว่างข้อมูลและใช้เครื่องมือนั้นพิสูจน์ว่าเป็นจริงหรือไม่ ซึ่งผลการวิเคราะห์หรือลักษณะการตอบคำถามจะแสดงในรูปของแฟกเตอร์ตัวเลขที่ประกอบด้วยการวิเคราห์ความเป็นไปได้ต่างๆซึ่งมีโอกาสที่จะพบความยุ่งยากได้ในกรณีที่มีข้อมูลจำนวนมาก แต่ Data Mining สามารถมองความสัมพันธ์ในหลายมิติในข้อมูลขนาดใหญ่ได้และจะเน้นเฉพาะจุดเด่นหรือความพิเศษของข้อมูลนั้น

2.1.3 กระบวนการทำงานของ Data Mining

ถ้ากล่าวถึง Data Mining ส่วนใหญ่จะให้ความสำคัญกับการ Mining หรือการค้นหาลักษณะทิศทางของข้อมูล แต่ที่จริงแล้วการ Mining ข้อมูลเป็นเพียงกระบวนการหนึ่งของการทำ Data Mining ดังรูปที่ 2.1 แสดงกระบวนการทำ Data Mining

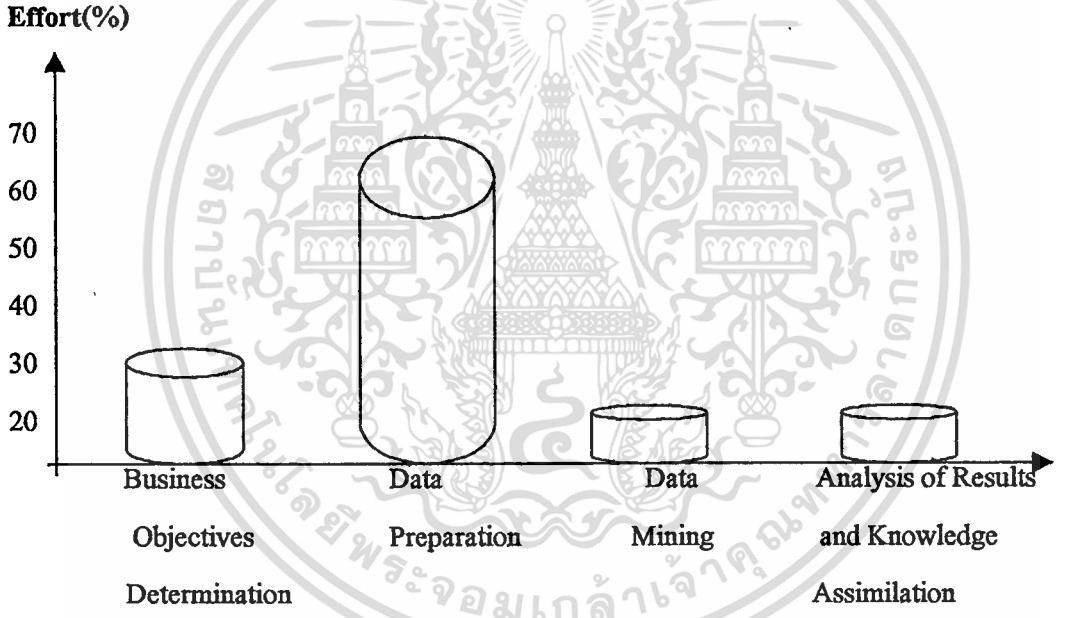


รูปที่ 2.1 กระบวนการทำ Data Mining

[Jacek M. Zurada, Artificial Neural systems, West Publishing Company, 1992]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของวัตถุประสงค์ทางธุรกิจเป็นพื้นฐานหลักในกระบวนการทำ Data Mining ด้วย ซึ่งส่วนนี้เป็นส่วนที่บอกถึงวัตถุประสงค์และสิ่งที่ต้องการจากการทำ Mining ในกระบวนการทำ Data Mining นั้นประกอบด้วยหลายขั้นตอนและในแต่ละขั้นตอนก็จะมีการทำซ้ำๆในขั้นตอนนั้น หรือต้องมีการวนกลับมาทำซ้ำใหม่ ถึงแม้ปัจจุบันเทคโนโลยีมีความก้าวหน้าไปมาก แต่ในการทำ Data Mining ก็ยังคงต้องใช้ความพยายามในการทำมากคือยังคงทำงานแบบ Labor-intensive แต่การทำงานในลักษณะนี้ก็ไม่ได้เป็นในทุกขั้นตอน ดังรูปที่ 2.2 จะแสดงเปอร์เซ็นต์การทำงานแต่ละขั้นตอน จะเห็นว่าการเตรียมข้อมูลสำหรับการทำ Data Mining (Data Preparation) นั้นมีถึง 60 เปอร์เซ็นต์ ฉะนั้นจุดที่ต้องให้ความสำคัญคือการ Clean ข้อมูลและประเด็นของข้อมูล ส่วนการทำ Mining จริงๆมีแค่เพียง 10 เปอร์เซ็นต์



รูปที่ 2.2 เปอร์เซ็นต์การทำงานแต่ละขั้นตอนใน Data Mining

รายละเอียดของการทำงานในแต่ละขั้นตอนใน Data Mining มีดังนี้

ขั้นตอนที่ 1 Business Objectives Determination

การกำหนดวัตถุประสงค์ทางธุรกิจคือจะต้องเข้าใจถึงปัญหาและความต้องการทางธุรกิจ การกำหนดวัตถุประสงค์ทางธุรกิจนั้นจะเป็นส่วนที่กำหนดว่าเมื่อไหร่ที่จะใช้ Data Mining ในการแก้ปัญหา ซึ่งในส่วนนี้จะประกอบด้วยการวิเคราะห์ทางธุรกิจและการวิเคราะห์ข้อมูลเบื้องต้นว่าเรามีข้อมูลใดอยู่บ้างและต้องการอะไรจากข้อมูลนั้น ซึ่งขั้นตอนนี้จะสามารถมองถึงอัลกอริทึมและฐานข้อมูลที่จะใช้ในเบื้องต้นที่สัมพันธ์กับวัตถุประสงค์ทางธุรกิจได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 Data Preparation

ในกระบวนการทำ Data Mining จะใช้กระบวนการนี้ถึง 60 เปอร์เซ็นต์ซึ่งจะประกอบด้วย 3 ขั้นตอน

Data Selection

จุดประสงค์หลักคือการระบุลักษณะและเลือกข้อมูลที่ต้องการ และนำข้อมูลที่ไม่ต้องการออกไป ซึ่งเป็นการเริ่มต้นของการเตรียมการ Mining การเลือกข้อมูลจะเปลี่ยนแปลงตามวัตถุประสงค์ทางธุรกิจ การเลือกไม่ว่าจะเป็นตัวแปร ความสัมพันธ์ จำเป็นจะต้องเข้าใจความหมาย ประเภทข้อมูล ค่าที่สามารถจะเป็นไปได้ ตัวแปรข้อมูลมี 2 ลักษณะ

1. Categorical

- nominal ตัวแปรนี้ลำดับข้อมูลไม่มีผลกับค่า เช่น สถานะการแต่งงาน (single,married,divorced)
- Ordinal ตัวแปรนี้ลำดับข้อมูลมีผลกับค่า เช่น อัตราการใช้บัตรเครดิตของลูกค้า (good,regular,poor)

2. Quantitative

- continuous ค่าที่เก็บจะเป็นเลขจำนวนจริง (real number) เช่น ข้อมูลรายได้
- discrete ค่าที่เก็บจะเป็นเลขจำนวนเต็ม (integer) เช่น ข้อมูลจำนวนลูกค้า

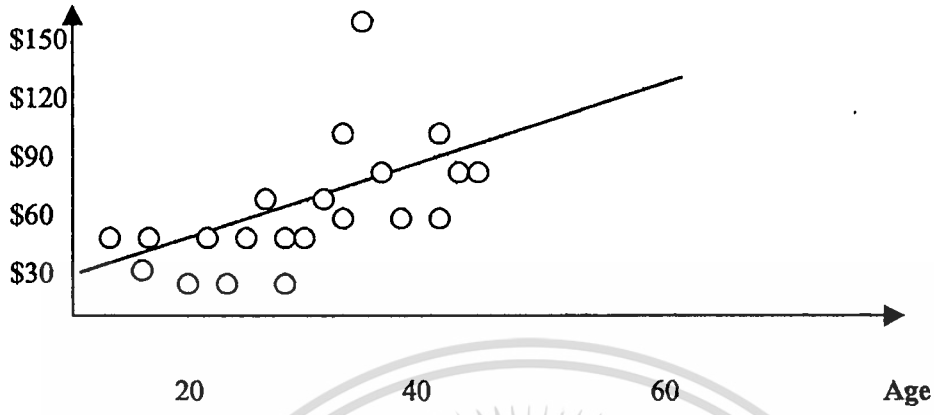
ตัวแปรสำหรับการเลือกใน Data Mining เรียกว่า "active variable"
ในการเลือกข้อมูลต้องคำนึงถึงอายุของข้อมูลด้วยเช่น ข้อมูลอาชีพของลูกค้า ซึ่งจะมีการเปลี่ยนแปลงบ่อยเมื่อเวลาผ่านไป

Data Proprocessing

ในกระบวนการนี้จะมีปริมาณข้อมูลจำนวนหนึ่งที่ถูกเลือกเข้ามาจากกระบวนการ Data Selection ซึ่งข้อมูลเหล่านี้จะต้องเป็นข้อมูลที่ถูกต้องที่พร้อมสำหรับการทำ Mining บางครั้งข้อมูลที่ได้จากการเลือกข้อมูล (Data Selection) อาจจะมีข้อมูลที่ไม่ถูกต้อง ดังนั้นในขั้นตอนนี้มีประเด็นที่จะต้องพิจารณาเพิ่มเติม 2 ประเด็นคือ

- Noisy Data เป็นข้อมูลที่มีลักษณะแตกต่างจากข้อมูลที่คาดการณ์ไว้ เรียกข้อมูลนี้ว่า outliers ดังรูป 2.3 แสดงข้อมูลอายุ และรายได้ที่ได้ ซึ่งจะมีข้อมูลที่แตกต่างจากข้อมูลอื่น
- Missing Values ข้อมูลที่ไม่ได้ถูกเลือกมาจากขั้นตอนการเลือกข้อมูล (Data Selection) ซึ่งถ้าข้อมูลที่มีขาดน้อยก็อาจจะตัดทิ้งได้ แต่ถ้าข้อมูลที่ขาดมีมากอาจจะแทนด้วยค่าเฉลี่ย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.3 แสดงข้อมูลอายุ และรายได้ที่ได้ ซึ่งจะมีข้อมูลที่แตกต่างจากข้อมูลอื่น

Data Transformation

เป็นการแปลงข้อมูลไปสู่โมเดลการวิเคราะห์ซึ่งจะแบ่งตามชนิดของปัญหาและจะแปลงให้อยู่ในลักษณะ format ข้อมูลที่ตรงกับอัลกอริทึมของ Data Mining ที่ใช้

ขั้นตอนที่ 3 Data Mining

เป็นการประมวลผลข้อมูลตามอัลกอริทึมที่ได้กำหนดไว้ในขั้นตอนนี้จะมีความสัมพันธ์กับการวิเคราะห์ข้อมูลและขั้นตอนที่ผ่านมา โดยเมื่อทำในส่วนของ Data Mining แล้วอาจจะต้องย้อนกลับไปทำในขั้นตอนที่ 2 ใหม่ ในการพัฒนาในส่วนของ Data Mining จะเกี่ยวข้องกับการใช้อัลกอริทึมหลายๆแบบ

ขั้นตอนที่ 4 Analysis of Results

เป็นการวิเคราะห์ผลของการประมวลผลซึ่งจำทำการแปลความหมายและประเมินผลที่ได้จากการ Mining การทำงานส่วนนี้จำเป็นต้องใช้ทักษะในการวิเคราะห์ข้อมูลและการวิเคราะห์ทางธุรกิจ ซึ่งจะต้องสอดคล้องกับการทำงานของ Data Mining

ขั้นตอนที่ 5 Assimilation of Knowledge

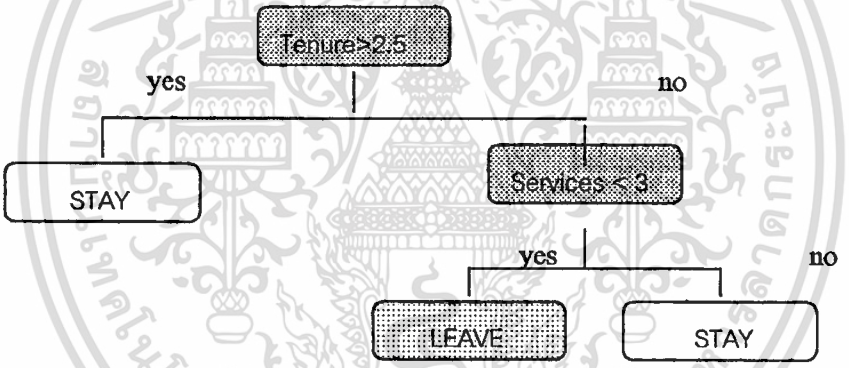
เป็นการรวบรวมความเข้าใจทางธุรกิจที่เป็นผลมาจาก Analysis of Results เข้าในส่วนความรู้เพื่อนำไปใช้ในโอกาสต่อไป ในขั้นตอนนี้จะมีหลักอยู่ 2 ประการคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แสดงแนวคิดทางธุรกิจที่ค้นพบใหม่
- กฎเกณฑ์ที่จะใช้ Information ใหม่ให้ได้ประโยชน์สูงสุด

2.1.4 Data Mining Operations : Predictive Modeling

เป็นโมเดลหนึ่งใน 4 โมเดล (Predictive Modeling, Database Segmentation, Link Analysis, Deviation Detection) ซึ่งลักษณะที่คล้ายกับการเรียนรู้ของมนุษย์ คือจะต้องเข้าใจลักษณะของสิ่งที่จะศึกษาอย่างแท้จริง เราจะใช้โมเดลนี้ในการวิเคราะห์ฐานข้อมูลที่มีอยู่เพื่อตัดสินใจเลือกลักษณะข้อมูลที่ต้องการ ฉะนั้นข้อมูลที่มีอยู่ต้องเป็นข้อมูลที่สมบูรณ์ จึงจะทำให้โมเดลให้คำทำนายได้อย่างเหมาะสม โดยการเริ่มต้นจะต้องให้คำตอบที่ถูกต้องกับโมเดลเพื่อโมเดลจะได้เห็นถึงข้อสังเกตใหม่ๆ เมื่ออัลกอริทึมทำงานจะเรียกว่า "supervised learning" ซึ่งลักษณะการทำงานจะมีลักษณะคล้ายกับ IF THEN ดังรูปที่ 2.4 จะแสดงถึง Predictive modeling approach



รูปที่ 2.4 จะแสดงถึง Predictive modeling approach

จากรูปที่ 2.4 เป็นตัวอย่างของบริษัทประกันภัย ซึ่งแสดงด้วย decision tree ที่ใช้วิเคราะห์ทางเลือกที่จะเกิดขึ้น เช่น ถ้าลูกค้าใช้บริการประกันของบริษัทนี้มามากกว่า 2.5 ปี และใช้บริการ 2 บริการ ลูกค้าก็จะยกเลิกการทำประกัน เป็นต้น

โมเดลนี้จะถูกพัฒนาใน 2 phase

- Training phase หมายถึงการสร้างโมเดลขึ้นมาใหม่โดยใช้ข้อมูลในอดีต
- Testing phase หมายถึงการตรวจสอบว่าโมเดลนี้ดีหรือไม่ โดยใช้ข้อมูลที่ถูกแบ่งเอาไว้สำหรับ Testing ซึ่งต้องเป็นข้อมูลคนละส่วนกับข้อมูล Training

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Predictive Modeling ประกอบด้วย 2 ลักษณะ

- Classification

จะแบ่งว่าข้อมูลแต่ละ record ในฐานข้อมูลนี้ควรจะอยู่ในกลุ่มใดบ้าง โดยแต่ละกลุ่มต้องระบุค่าที่เป็นไปได้อย่างชัดเจน

- Value Prediction

จะใช้ประเมินค่าความต่อเนื่องของตัวเลขที่มีความเกี่ยวข้องกับฐานข้อมูล เช่นต้องการทำนายช่วงเวลาที่จะได้ลูกค้าใหม่ในการขายรถ โดยจะต้องทำการ Mining ข้อมูลเก่าที่เก็บข้อมูลลูกค้าเก่าๆรวมถึงข้อมูลอื่นๆที่สามารถวัดได้เช่น ความสามารถทางด้านการเงิน ซึ่งตัวแปรที่จะใช้ในการสร้างโมเดลเช่น อายุของลูกค้า รายได้ ประวัติการนำรถมาเข้าซ่อม จำนวนคนในครอบครัว ระดับการศึกษา เป็นต้น

2.2 การวิเคราะห์อนุกรมเวลา (Time Series Analysis)

ปัญหาอนุกรมเวลา (Time Series Problem) จะเป็นการคาดหวังค่าที่จะได้มาจากความสัมพันธ์ของค่าก่อนหน้า

ข้อมูลตัวสุดท้ายจะมีความสำคัญมากในลำดับสมมติเป็น t(n)

ค่าที่เราต้องการในการพยากรณ์ t(n+1)

รูปแบบของการแปลงข้อมูลใน Time Series {t(n+1),t(n)}

ค่าที่เราต้องการจะนิยามอยู่ในรูปแบบเดียวกับกับรูปแบบของข้อมูลก่อนหน้า

สมมติมีข้อมูลอยู่ในลำดับ t(1).....t(I).....t(n) จำนวนข้อมูลในช่วง (Window) จะพิจารณาจากความสัมพันธ์ของเวลาก่อนหน้านี้ จากรูป 2.5 จำนวนข้อมูลในช่วงหรือ time lag จะเท่ากับ 5 ลำดับ (Series) นี้จะมีหมายความว่าจะมี 5 กรณีที่เราจะต้องพิจารณา ดังรูปที่ 2.5 ซึ่ง column สุดท้ายจะเป็นผลลัพธ์ที่เราต้องการ

t0 t1 t2 t3 t4 t5 t6 t7 t8 t9

lag = 5

รูปที่ 2.5 Time Series with a Window of Size 5

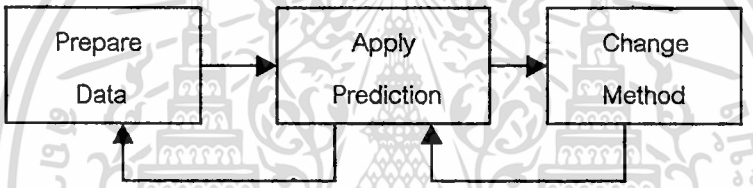
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Case	F1	F2	F3	F4	F5	Goal
1	t0	t1	t2	t3	t4	t5
2	t1	t2	t3	t4	t5	t6
3	t2	t3	t4	t5	t6	t7
4	t3	t4	t5	t6	t7	t8
5	t4	t5	t6	t7	t8	t9

ตารางที่ 2.1 Transformation of Time Series to Strandard Form

กระบวนการเตรียมข้อมูล (Data Preparation) จะดำเนินไปในรูปแบบ

- การหาช่วงของข้อมูล time lag จะตัดสินใจได้จากการทดสอบข้อมูล ดังรูปที่ 2.6



รูปที่ 2.6 Iterative Data Preparation Model

- ขณะที่หาผลลัพธ์ที่จะพยากรณ์ในครั้งต่อไป เช่น มีข้อมูลของทุกๆเดือนในเวลา 1 ปี แล้วต้องการพยากรณ์ค่าว่าในปีต่อไปจะมีค่าเป็นเท่าใด สมมติ {t(1)....t(n-12),t(n)} ซึ่ง t(n-12) คือค่าในเวลาปัจจุบัน และ t(n) คือค่าที่เราจะพยากรณ์ในปีถัดไป ถ้าเราต้องการพยากรณ์ค่าที่ไกลออกไปก็จะยาก และไม่น่าเชื่อถือ

2.2.1 คุณสมบัติจาก Time Series

2.2.1.1 ค่าปัจจุบัน (Current Value)

ค่าในปัจจุบันจะถูกใช้ในการพยากรณ์ที่ดีจะต้องพยากรณ์ในเวลาถัดไปที่ติดกัน เช่น ค่าปัจจุบัน t(I) ค่าที่จะพยากรณ์ต้องเป็น t(i+1) และ t(i+1) / t(I) จะบอกถึงเปอร์เซ็นต์การเปลี่ยนแปลงที่ได้รับ ค่าความแตกต่างของ t(I+1) – t(I) และ t(I) – t(I – 1) ควรจะเป็นไปในรูปแบบเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.2 ค่าเฉลี่ยการเปลี่ยนแปลง (Moving Averages)

$$ma\ (i,m) = \frac{1}{m} \sum_{j=i-m+1}^{i} t(j) \qquad (2.1)$$

จากสมการ 2.1 ความกว้างของช่วงเท่ากับ m จะหาค่าเฉลี่ยการเปลี่ยนแปลงของข้อมูลได้ และค่าที่ได้จากการคาดหวังเมื่อมีการเปลี่ยงแปลงอย่างอิสระแสดงดังสมการ 2.2

$$t(i) = mean(i) + error \qquad (2.2)$$

2.2.1.3 แนวโน้ม (Trends)

ผลรวมของค่าเฉลี่ยการเปลี่ยนแปลงของข้อมูลในอดีตจะแสดงแนวโน้มของการพยากรณ์ได้ คุณลักษณะของแนวโน้มจะวัดด้วยการเปรียบเทียบระหว่างค่าปัจจุบันกับค่าเฉลี่ยการเปลี่ยนแปลง

$$t(i) - ma(i,m) \qquad (2.3)$$

หรืออาจจะเปรียบเทียบระหว่างค่าเฉลี่ยการเปลี่ยนแปลง ดังนั้นข้อมูลแบบ Time Series จะมีความสำคัญมากในการพยากรณ์

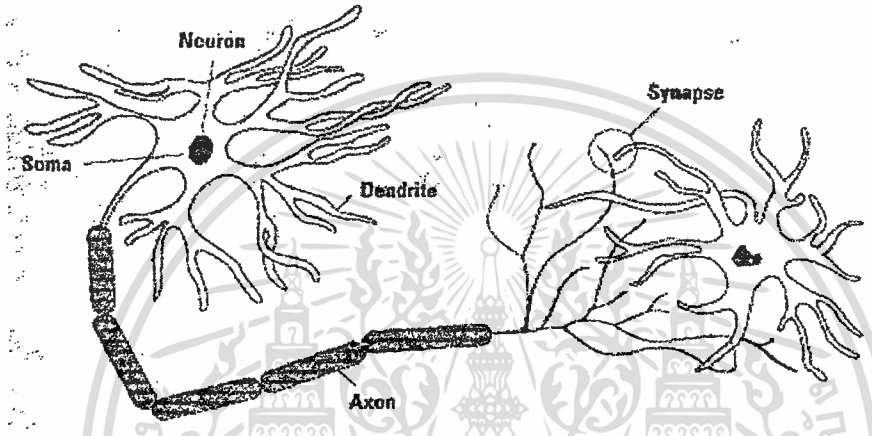
2.3 ทฤษฎีนิวรอนเน็คเวิร์ค (Neuron Network)

2.3.1 ทฤษฎีเซลล์ประสาททางชีววิทยาเบื้องต้น (Biological Neurons)

องค์ประกอบของสมองมนุษย์นั้นประกอบไปด้วยเซลล์ประสาท หรือนิวรอนประมาณ 10^{11} ตัว โดยเซลล์ประสาทหนึ่งๆมีเส้นเชื่อมโยงต่อไปยังอีกเซลล์ประสาทอื่นๆประมาณ 10^{4} ตัว และเวลาในการสับเปลี่ยนข้อมูลระหว่างนิวรอนประมาณ 10^{-3} วินาที ซึ่งนับว่าช้ามากเมื่อเทียบกับเวลาในการสลับการทำงานของคอมพิวเตอร์ที่ใช้เวลาเพียง 10^{-10} วินาที ลักษณะของเซลล์ประสาททางชีวภาพประกอบไปด้วย 3 ส่วนหลักคือ เดนไดรท์ (Dendrites) โซมา (Soma) หรือตัวเซลล์ (Cell body) หรือแอคซอน (Axon) ส่วนที่เชื่อมระหว่างตัวเซลล์กับแอคซอนของเซลล์อื่นๆที่อยู่รอบๆข้างเรียกว่า ไซแนปส์ (Synapse) โดยตัวเดนไดรท์จะรับสัญญาณจากเซลล์ประสาทที่อยู่รอบๆข้างผ่านทางไซแนปส์ด้วยปฏิกิริยาทางเคมี ซึ่งสัญญาณที่เข้ามานี้คือ ประจุอิเล็กตรอน โดยปฏิกิริยาที่เกิดขึ้นจะทำการปรับเปลี่ยนความถี่ของสัญญาณที่เข้ามา ตัวเซลล์จะทำหน้าที่รวมสัญญาณที่เข้ามาแล้วทำการส่งสัญญาณออกไปให้กับแอคซอน เพื่อผ่านต่อให้กับเซลล์อื่นๆ ซึ่งสัญญาณที่ออกจากตัวเซลล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีลักษณะเป็นแบบสัญญาณกระตุ้นให้กับเซลล์อื่น กล่าวคือ ถ้าค่าสัญญาณที่ส่งออกจากตัวเซลล์มีจำนวน 100 ครั้งต่อวินาที ถือว่าค่าสัญญาณที่ส่งออกเป็นสถานะการกระตุ้น (Fire) โดยทั่วไปราสนใจสัญญาณที่ออกจากเซลล์ประสาทในรูปของสถานะการกระตุ้น และไม่กระตุ้น (Not Fire) ให้กับเซลล์ที่อยู่รอบข้างมากกว่าสนใจเงื่อนไขที่เกี่ยวข้องที่ทำให้เซลล์ประสาทเกิดสถานะนี้ขึ้นมาดังรูปที่ 2.7

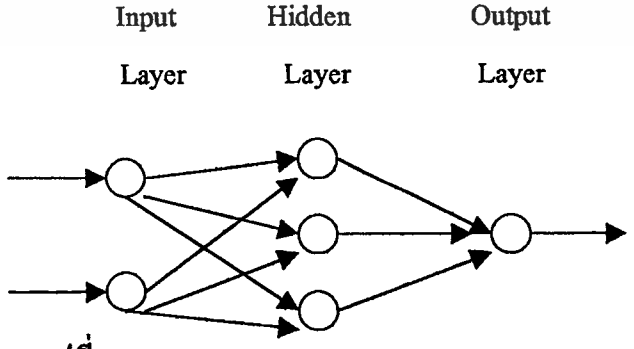


รูปที่ 2.7 แสดงเซลล์ประสาทมนุษย์ทางชีววิทยา

[Jacek M. Zurada, Artificial Neural systems, West Publishing Company, 1992]

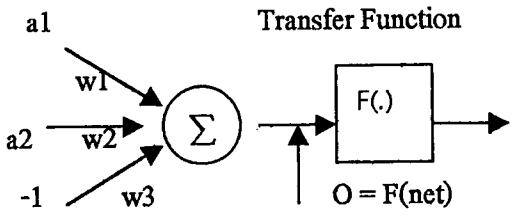
2.3.2 ตัวแบบของเซลล์ประสาทเทียม (Artificial Neural Model)

เป็นกระบวนการเรียนแบบการทำงานของเซลล์สมองของสิ่งมีชีวิต ซึ่งแสดงโดย node และ **น้ำหนัก** (weight) ที่อยู่บน Link ที่เชื่อมแต่ละ node ดังรูปที่ 2.8 จะแสดงถึง neural network ที่มี Input Layer ,Hidden Layer และ Output Layer ที่มี Input เป็น a1,a2..... และมี weight เป็น w1,w2.....หรืออาจจะแสดง 1 หน่วยของ Neural Network ได้ด้วยรูปที่ 2.9



รูปที่ 2.8 Multi-layer feedforward neural network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



$$net = \sum_{I=1}^{3} w_I a_I$$

Transfer function คือ

$$F(net) = \frac{2}{1+\exp(-\lambda \cdot net)} - 1, \lambda > 0$$

$$F(net) = \frac{1}{1+\exp(-\lambda \cdot net)}, \lambda > 0$$

$$F(net) = sgn(net) = \begin{cases} +1, & net > 0 \\ -1, & net < 0 \end{cases}$$

$$F(net) = sgn(net) = \begin{cases} +1, & net > 0 \\ 0, & net < 0 \end{cases}$$

รูปที่ 2.9 A unit of an artificial neural network

ส่วน Output ที่ออกมาจะถูกกระทำต่อไปโดยส่วนที่เรียกว่า ฟังก์ชั่นกระตุ้น (Activation Function) ค่าที่ได้จากฟังก์ชั่นกระตุ้นนี้จะทำให้เซลล์ประสาทเทียมมีลักษณะเหมือนเซลล์ประสาททางชีววิทยา กล่าวคือ ค่าผลลัพธ์ที่ได้จากฟังก์ชั่นกระตุ้นเป็นค่าที่อยู่ในช่วง 0 ถึง 1 ถ้าค่าสัญญาณที่ออกจากเซลล์ทางชีววิทยามีลักษณะเป็นสถานะของการกระตุ้น จะสามารถแทนความหมายของสถานะการกระตุ้นของเซลล์ประสาทด้วยค่า 1 และการไม่กระตุ้นด้วยค่า 0 รูปแบบของฟังก์ชั่นมีหลายรูปแบบดังนี้

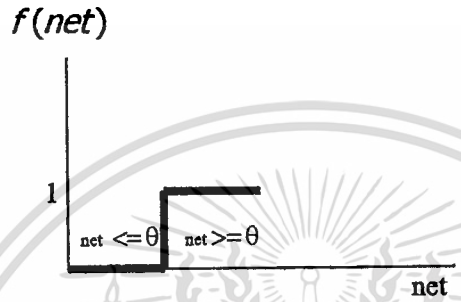
1.Step Function เป็นฟังก์ชั่นที่ใช้ในโครงข่ายแบบชั้นเดียว (Single Layer) ซึ่งใช้ในการแปลงค่า net input ที่อยู่ในรูปของค่าต่อเนื่องให้อยู่ในรูปของไบนารี (Binary) คือ 1 และ 0 หรือไบโพลล่า (Bipolar) คือ 1 และ –1 ฟังก์ชั่นนี้จะใช้ค่าเทรโชด์ (Threshold) ในการกำหนดการแปลงค่า ซึ่งเรียกฟังก์ชั่นนี้ว่า Threshold Function หรือ Heavised Function ดังสมการ 2.4 และรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(net) = 1 \quad \text{If net} >= \theta \qquad (2.4)$$
$$0 \quad \text{If net} <= \theta$$

โดยที่ net คือค่า net input ของเซลล์ประสาทเทียม

θ คือค่าเทรโชลด์ (Threshold)



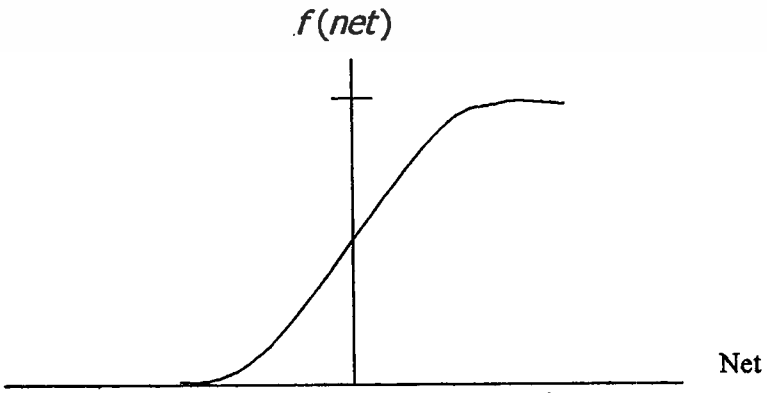
รูปที่ 2.10 ลักษณะของสมการ Threshold

2.Sigmoid Function เป็นฟังก์ชั่นที่ใช้ในการแปลงค่า net input ให้อยู่ในช่วง 0 ถึง ซึ่งเรียกว่า Binary Sigmoid function หรืออยู่ในช่วง –1 ถึง 1 ซึ่งเรียกว่า Bipolar Sigmoid ดังแสดงในสมการ 2.5,2.6 และ และรูปที่ 2.11,2.12

$$f(net) = 1/(1 + e^{-net}) \qquad (2.5)$$

โดยที่ net คือ ค่า net input ของเซลล์ประสาทเทียม

e คือ ฟังก์ชั่นเอกส์โปร์แนนเชียล



รูปที่ 2.11 กราฟของฟังก์ชั่นไบนารี่ซิกมอยด์

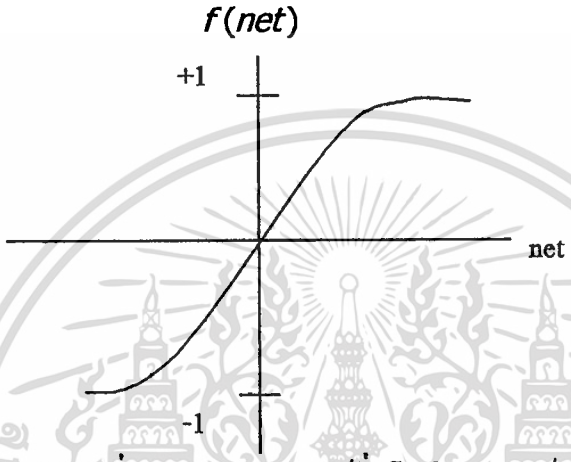
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

$$f(net) = (e^{net} - e^{-net}) / (e^{net} + e^{-net}) \qquad (2.6)$$

โดยที่ net คือ ค่า net input ของเซลล์ประสาทเทียม

e คือ ฟังก์ชั่นเอกส์โปร์แนนเชียล



รูปที่ 2.12 กราฟของฟังก์ชั่นโพล่าซิกมอยด์

2.3.3 การฝึกสอนโครงข่ายประสาทเทียม (Neural Network)

การทำงานใน Neural Network มี 2 วิธีคือ

- การเรียนรู้ (Training) แบ่งเป็น
  - Supervised Learning การสอนที่บอกว่ามี Input แบบนี้จะมี Output ในลักษณะใด
  - Unsupervised Learning การสอนจะไม่บอกว่า Output ควรเป็นอย่างไร โดยใส่ Input เข้าไปแล้ว Neural Network มันจะเรียนรู้ของมันเองว่า Output ควรจะเป็นอย่างไร
- Query (Interrogation) หลังจากเรียนรู้เสร็จแล้ว แล้วนำเอาไปใช้งาน

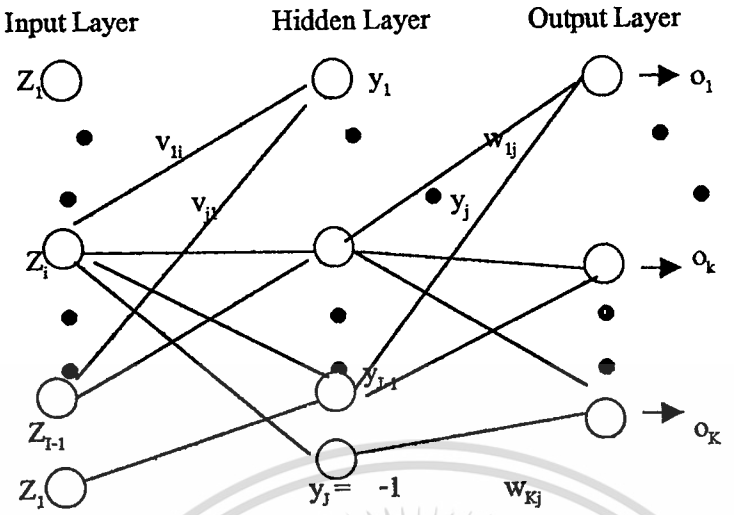
ในที่นี้จะใช้ Backpropagation Algorithm ในการสอนให้ Neural Network เรียนรู้ (ดูรูป 13 ประกอบ)

2.3.4 โครงข่ายนิวรอนแบบแบพรอพพาเกชั่น

(Backpropagation Neural Network,BPN)

โครงข่ายแบพรอพพาเกชั่นเป็นโครงข่ายที่มีชั้น Input Layer,Hidden Layer และ Output Layer ดังรูปที่ 2.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 Multilayer Feedforward Networks

ซึ่งมีขั้นตอนในการทำงานดังนี้

- ให้ค่าเริ่มต้นกับ weight เป็นค่า random น้อยๆ
- เริ่มทำการเรียนรู้โดยการทดสอบก่อนว่าจะหยุดเรียนรู้แล้วหรือยัง ถ้ามีข้อผิดพลาดในการเรียนรู้น้อยก็หยุดการเรียนรู้ แต่ถ้าข้อผิดพลาดในการเรียนรู้ยังมากอยู่ก็ต้องเรียนรู้ต่อไป ในที่กำหนด Input และ Output ขึ้นมา 1 คู่ $(\vec{Z}_p, \vec{d}_p)$ แล้วค่า Input จะถูกป้อนให้กับชั้น Input ของโครงข่าย และที่ชั้น Input นี้จะถูกกระจายไปยังนิวรอนในชั้นซ่อน ที่ชั้นซ่อนนี้จะเองจะทำการคำนวณค่า net และ f(net)
- ปรับค่าความผิดพลาดโดยการย้อนเอาค่าความผิดพลาดกลับเข้ามาใน Neural Network โดยย้อนกลับที่ละ Layer

● กำหนดแต่ละ Network มี Output unit k ดังนั้นค่าความผิดพลาด (error signal term) ดังสมการ 2.7

$$\delta_{ok} = 1/2(d_k - o_k)(1 - o_k^2), k = 1,2,3,\ldots,k \quad (2.7)$$

d_k คือ ค่า Output จริง

O_k คือ ค่า Output ที่ได้จาก Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

● กำหนดแต่ละ hidden unit j ดังนั้นค่าความผิดพลาด (error signal term) ดังสมการ 2.8

$$\delta_{yj} = 1/2(1 - y_j^2)\sum_{k=1}^{k} \delta_{ok} w_{kj}, j = 1,2,\ldots, j \qquad (2.8)$$

Y คือ ค่า Output ของแต่ละ Hidden

W คือ ค่า Weight ของ Hidden

- ปรับค่า weight ให้สอดคล้องกับค่าความผิด

● สำหรับ Output Layer weight ดังสมการ 2.9

$$w_{kj} \longleftarrow w_{kj} + \eta\delta_{ok} y_j \quad , \; k = 1,2,3,..,K \quad (2.9)$$

W คือ ค่า Weight เดิมของชั้น Output

η คือ ค่า Learning Rate

● สำหรับแต่ละ hidden weight ดังสมการ 2.10

$$v_{ji} \longleftarrow v_{ji} + \eta\delta_{yj} z_i, \; j = 1,2,3,..,J \qquad (2.10)$$

$$, i = 1,2,3,..,I$$

V คือ ค่า Weight เดิมของชั้น Hidden

η คือ ค่า Learning Rate

เมื่อทำจนเสร็จตามขั้นตอนดังกล่าวก็เลือกคู่ Input และ Output อื่นต่อไปมาทำจนครบทั้งหมดก็จะได้ Neural Network ที่มีค่า weight ต่างๆถูกต้อง แล้วสามารถเอาข้อมูลต่างๆมาผ่าน Neural Network นี้ได้เพื่อหาคำตอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีและข้อจำกัดใน Backpropagation Algorithm

ข้อดีของ Backpropagation Algorithm

- สามารถเรียนรู้ความสัมพันธ์ของรูปแบบได้มากมาย โดยการเรียนรู้ต้องการรูปแบบตัวอย่างที่จะเรียนรู้
- มีความยืดหยุ่นในการเรียนรู้ เช่นสามารถเลือกจำนวน Layer ได้

ข้อเสียของ Backpropagation Algorithm

- ต้องใช้เวลามากในการเรียนรู้ของ Network
- อาจเกิดความล้มเหลวในการเรียนรู้ของ Network ถึงแม้จะเพิ่มข้อมูลเข้าไปก็ไม่สามารถปรับ Network ได้ แต่อาจจะแก้ได้โดยเพิ่มจำนวน Layer หรือเปลี่ยนชุดของข้อมูลที่ใช้ Train

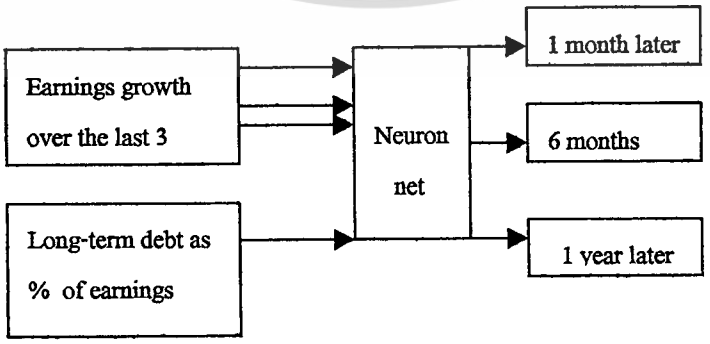
2.4 การใช้ทฤษฎี Neuron Network มาพยากรณ์ราคาหลักทรัพย์

มี 2 approach มาตรฐานในการพยากรณ์ทางการเงินคือ trend prediction และ price prediction

trend prediction จะใช้ในการตัดสินใจเมื่อ trend มีการเปลี่ยนแปลงทิศทาง ซึ่งการวิเคราะห์จะใช้เวลานาน (เป็นสัปดาห์หรือเดือน)

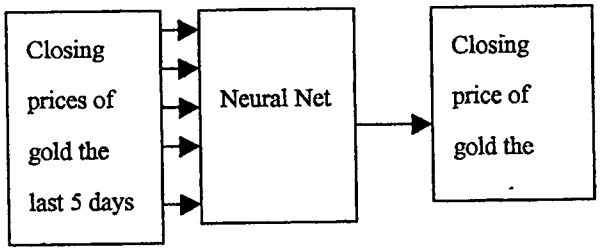
price prediction จะใช้ในการตัดสินใจซึ่งราคาจะอยู่บน time frame ซึ่งการวิเคราะห์จะใช้เวลาสั้น (ชั่วโมงหรือวัน) Neuron Net จะสามารถเรียนรู้ถึงตัวแปรที่ดีสำหรับการตัดสินใจซื้อหลักทรัพย์ ข้อมูลในอดีตสามารถจะนำมาใส่ใน neuron net ซึ่งข้อมูลเหล่านี้เรารู้ผลลัพธ์อยู่แล้ว และ neuron net สามารถจะเรียนรู้ถึงตัวแปรที่ชี้ถึงการขึ้นลงของราคาหลักทรัพย์ ซึ่งแสดงดังรูป 2.11

Neuron net สามารถที่จะหาข้อมูลที่เราเห็นว่าไม่สำคัญที่จะทำให้เกิดการผิดพลาด และจะชี้ถึงการทำซ้ำของ Pattern และทำนายค่าในอนาคต คุณลักษณะของ Neuron Net จะถูกสร้างจากข้อมูลในอดีต ซึ่งจะแสดง block diagram ที่สร้าง Neuron Net ที่ประสบความสำเร็จในการพยากรณ์ ดังรูปที่ 2.12



รูปที่ 2.11 Neural net architecture for predicting stock price

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 Neural net architecture for predicting daily gold prices based on 5 day pattern

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินการศึกษา

ในการศึกษาโครงงานนี้ จะเป็นการนำทฤษฎีโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชั่น (Backpropagation Neural Network) มาพัฒนาเป็นโปรแกรมคอมพิวเตอร์เพื่อมาประยุกต์ใช้ในการพยากรณ์ราคาหลักทรัพย์ โดยแบ่งการดำเนินการศึกษาออกเป็น 4 ส่วนคือ

3.1 ส่วนของโครงสร้างแบบจำลอง

3.2 ส่วนของอัลกอริธึม

3.3 ส่วนของข้อมูล

3.4 ส่วนของโปรแกรม

3.1 ส่วนของโครงสร้างแบบจำลอง

โครงสร้างแบบจำลองของโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชั่น (Backpropagation Neural Network) ที่ใช้ในโครงงานนี้ประกอบด้วย 3 ชั้น คือ

1. ชั้นอินพุต (Input Layer) สามารถระบุจำนวนอินพุตได้
2. ชั้นฮิดเดน (Hidden Layer) สามารถระบุจำนวนชั้นของฮิดเดนและจำนวน Node ในฮิดเดนได้
3. ชั้นเอาต์พุต (Output Layer)

3.2 ส่วนของอัลกอริธึม

อัลกอริธึมของโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชั่น (Backpropagation Neural Network) ที่ใช้ในการพัฒนาโปรแกรมสำหรับโครงงานนี้มีขั้นตอนดังนี้

ขั้นตอนที่ 1 ทำการแปลงค่าที่อ่านจาก Text File ให้อยู่ในรูป 0 ถึง 1 หรือกรณีที่เป็น Database ก็จะแปลงมาเป็น Text File ก่อนแล้วจึงแปลงให้อยู่ในรูป 0 ถึง 1

ขั้นตอนที่ 2 กำหนดค่าน้ำหนัก (weight) โดยการสุ่มค่าตัวเลขระหว่าง –1 ถึง 1

ขั้นตอนที่ 3 ตรวจสอบว่าถ้า Error น้อยกว่า 0.05 ก็จะหยุดการฝึกสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 แต่ละนิวรอนบนชั้นอินพุท (Input Layer) จะถูกกำหนดด้วยค่าใน Text File ที่ได้แปลงให้มีค่าอยู่ระหว่าง 0 ถึง 1 ซึ่งจะมีกี่อินพุทนั้นสามารถกำหนดได้ด้วยการระบุจากจำนวน Lag ที่หน้าจอ

ขั้นตอนที่ 5 คำนวณหาค่าเน็ท (nethi1) ของแต่ละนิวรอนบนชั้นฮิตเดนชั้นที่ 1 (Hidden Layer 1) เป็นค่าที่ได้จากการหาผลรวมของการคูณกันของตัวแปรเวต (wh1) และอินพุตที่เข้ามาของแต่ละคู่ ซึ่งแสดงด้วยสมการ (3.1)

$$nethi1[h] = \sum_{I=1}^{\text{จำนวน Lag ที่ระบุเข้ามา}} (input[I] * wh1[I][h]) \qquad (3.1)$$

wh1 คือ ค่าน้ำหนักของชั้น Hidden layer 1

h คือ จำนวนนิวรอนของชั้น Hidden layer 1

i คือ จำนวนนิวรอนของชั้น Input layer

Input คือ ข้อมูลอินพุทจากชั้น Input layer

ขั้นตอนที่ 6 คำนวณหาค่าเอาต์พุท (outputhi1) ของแต่ละนิวรอนบนฮิตเดนชั้นที่ 1 (Hidden Layer 1) โดยนำผลลัพธ์จาก nethi1[h] เข้าสู่ฟังก์ชั่นแอคติเวชั่น (Activation Function) ซึ่งแสดงด้วยสมการ (3.2)

$$outputhi1[h] = 1.0/(1.0 + \exp(-nethi1[h])) \qquad (3.2)$$

h คือ จำนวนนิวรอนของชั้น Hidden layer 1

nethi1 คือ ค่าเน็ทของชั้น Hidden layer 1

exp() คือ ฟังก์ชั่นเอกซ์โปร์แนนเชียล

ขั้นตอนที่ 7 (กรณีมี 2Hidden Layer) คำนวณหาค่าเน็ท (nethi2) ของแต่ละนิวรอนบนชั้นฮิตเดนชั้นที่ 2 (Hidden Layer 2) เป็นค่าที่ได้จากการหาผลรวมของการคูณกันของตัวแปรเวต (wh2) และอินพุตที่เข้ามา (outputhi1(h) ของแต่ละคู่ ซึ่งแสดงด้วยสมการ (3.3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวน Node ของ Hidden 1 ที่ระบุเข้ามา

$$nethi2[j] = \sum_{i=1} (outputhi1[i] * wh2[i][j]) \qquad (3.3)$$

wh2	คือ	ค่าน้ำหนักของชั้น Hidden layer 2
j	คือ	จำนวนนิวรอนของชั้น Hidden layer 2
i	คือ	จำนวนนิวรอนของชั้น Hidden layer 1
outputhi1	คือ	เอาต์พุทจากชั้น Hidden layer 1

ขั้นตอนที่ 8 *(กรณีมี 2Hidden Layer)* คำนวณหาค่าเอาต์พุท (outputhi2) ของแต่ละนิวรอนบนฮิดเดนชั้นที่ 2 (Hidden Layer 2) โดยนำผลลัพธ์จาก nethi2[j] เข้าสู่ฟังก์ชั่นแอคติเวชั่น (Activation Function) ซึ่งแสดงด้วยสมการ (3.4)

$$outputhi2[j] = 1.0/(1.0 + \exp(-nethi2[j])) \qquad (3.4)$$

j	คือ	จำนวนนิวรอนของชั้น Hidden layer 2
nethi2	คือ	ค่าเน็ทของชั้น Hidden layer 2
exp()	คือ	ฟังก์ชั่นเอกซ์โปร์แนนเชียล

ขั้นตอนที่ 9 คำนวณหาค่าเน็ท (netout) ของแต่ละนิวรอนบนชั้นเอาพุท (Output Layer) ค่าที่ได้จากการหาผลรวมของการคูณกันของตัวแปรเวต (wo) และอินพุตที่เข้ามาของแต่ละคู่ ดังสมการ (3.5)

จำนวน neuron ใน hidden layer ที่ป้อนเข้ามา

$$netout = \sum_{i=1} (outputhi2[i] * wo[i]) \qquad (3.5)$$

i	คือ	จำนวนนิวรอนของชั้น Input layer 1 หรือ 2 แล้วแต่ว่ามีกี่ Hidden Layer
outputhi2	คือ	ค่าเอาพุทของชั้น Hidden layer 2
wo	คือ	ค่าน้ำหนักของชั้น Output layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 10 คำนวณหาค่าเอาพุต (output) ของแต่ละนิวรอนบนชั้นเอาต์พุต (Output Layer) โดยนำผลลัพธ์จาก nethout เข้าสู่ฟังก์ชันแอคติเวชั่น (Activation Function) ดังสมการ (3.6)

$$output = 1.0 / (1.0 + exp(-netout) \quad (3.6)$$

netout	คือ	ค่าเน็ตของชั้น Output layer
exp()	คือ	ฟังก์ชั่นเอกซ์โปร์แนนเชียล

ขั้นตอนที่ 11 คำนวณหาความผิดพลาดจากสมการ (3.7)

$$errorp = (1/2) * (target - output)^2 \quad (3.7)$$

target	คือ	ค่าเป้าหมาย
output	คือ	ค่าเอาต์พุทของชั้น Output Layer

ขั้นตอนที่ 12 คำนวณหาค่าเดลต้า (deltaout) ของชั้น Output Layer ดังสมการ (3.8)

$$deltaout = (target - output) * ((1 - output) * output) \quad (3.8)$$

target	คือ	ค่าเป้าหมาย
output	คือ	ค่าเอาต์พุทของชั้น Output layer

ขั้นตอนที่ 13 คำนวณหาค่าเดลต้าเวต (delta_wo) ที่ต้องใช้เป็นค่าสำหรับปรับปรุงเวตที่ชั้นเอาต์พุท (wo) ดังสมการ (3.9)

$$delta_wo[j] = deltaout * outputhi[j] \quad (3.9)$$

j	คือ	จำนวนนิวรอนของชั้น Hidden layer
deltaout	คือ	ค่าเดลต้าของชั้น Output layer
outputhi	คือ	ค่าเอาต์พุทของชั้น Hidden layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 14 คำนวณหาค่า deltahi2 ของฮิดเดนชั้นที่ 2 ดังสมการ (3.10)

$$deltahi2[j] = (delout * wh[h][j]) * ((1-outputhi2[j]) * outputhi2[j])) \quad (3.10)$$

j	คือ	จำนวนนิวรอนของชั้น Hidden layer 2
deltaout	คือ	ค่าเดลต้าของชั้น Output layer
outputhi2	คือ	ค่าเอาต์พุทของชั้น Hidden layer 2
wh	คือ	ค่าน้ำหนักของชั้น Output layer

ขั้นตอนที่ 15 คำนวณหาค่าเดลต้าเวต (delta_wh2) ที่ต้องใช้เป็นค่าสำหรับปรับปรุงเวตฮิดเดน 2 (wh2) ดังสมการ (3.11)

$$delta_wh2[j][h] = deltahi2[h] * outputhi1[h] \quad (3.11)$$

j	คือ	จำนวนนิวรอนของชั้น Hidden layer 2
h	คือ	จำนวนนิวรอนของชั้น Hidden layer 1
deltahi2	คือ	ค่าเดลต้าของชั้น Hidden layer 2
outputhi1	คือ	ข้อมูลเอาต์พุทจากชั้น Hidden layer 1

ขั้นตอนที่ 16 คำนวณหาค่า deltahi1 ของแต่ละฮิดเดนชั้นที่ 1 ซึ่งมีสมการ3.12

$$deltahi1[h] = \left(\sum_{j=1}^{n} (deltahi2[j] * wh2[h][j]) \right) * ((1 - outputhi1[h]) * outputhi1[h])) \quad (3.12)$$

h	คือ	จำนวนนิวรอนของชั้น Hidden layer 1
j	คือ	จำนวนนิวรอนของชั้น Hidden layer 2
deltahi2	คือ	ค่าเดลต้าของชั้น Hidden layer 2
outputhi1	คือ	ค่าเอาต์พุตของชั้น Hidden layer 1
wh2	คือ	ค่าน้ำหนักของชั้น Hidden layer 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 17 คำนวณหาค่าเดลต้า (delta_wh1) ที่ต้องใช้เป็นค่าสำหรับปรับปรุงเวตฮิตเดน 1 (wh1) ดังสมการ (3.13)

$$delta_wh1[i][h] = deltahi1[h] * input[i] \qquad (3.13)$$

h	คือ	จำนวนนิวรอนของชั้น Hidden layer 1
i	คือ	จำนวนนิวรอนของชั้น Input layer
deltahi1	คือ	ค่าเดลต้าของชั้น Hidden layer 1
input	คือ	ค่าอินพุตของชั้น Input layer

ขั้นตอนที่ 18 ปรับค่าเวตที่ชั้นเอาต์พุท (wo) ดังสมการ (3.14)

$$wo[j]_{new} = wo[j]_{old} + \text{Learning Rate}\ (delta_wo[j]) \qquad (3.14)$$

wo_{new}	คือ	ค่าน้ำหนักหลังปรับค่า
wo_{old}	คือ	ค่าน้ำหนักก่อนปรับค่า
delta_wo	คือ	ค่าเดลต้าเวตของชั้น Output layer
j	คือ	จำนวนนิวรอนของชั้น Hidden layer

ขั้นตอนที่ 19 ปรับค่าเวตที่ชั้นฮิตเดน2 (wh2) ดังสมการ (3.15)

$$wh2[h][j]_{new} = wh2[h][j]_{old} + \text{Learning Rate}\ (delta_wh2[h][j]) \qquad (3.15)$$

$wh2_{new}$	คือ	ค่าน้ำหนักของ Hidden 2 หลังปรับค่า
$wh2_{old}$	คือ	ค่าน้ำหนักของ Hidden 2 ก่อนปรับค่า
delta_wh2	คือ	ค่าเดลต้าเวตของชั้น Hidden layer 2
j	คือ	จำนวนนิวรอนของชั้น Hidden layer 2
h	คือ	จำนวนนิวรอนของชั้น Hidden layer 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 20 ปรับค่าเวตที่ชั้นฮิตเดน1 (wh1) ดังสมการ (3.16)

$$wh1[h][j]_{new} = wh1[h][j]_{old} + \text{Learning Rate}(delta_wh1[h][j]) \qquad (3.16)$$

$wh1_{new}$ คือ ค่าน้ำหนักของ Hidden 1 หลังปรับค่า

$wh1_{old}$ คือ ค่าน้ำหนักของ Hidden 1 ก่อนปรับค่า

delta_wh1 คือ ค่าเดลต้าเวตของชั้น Hidden layer 1

j คือ จำนวนนิวรอนของชั้น Hidden layer 2

h คือ จำนวนนิวรอนของชั้น Hidden layer 1

ขั้นตอนที่ 21 ทดสอบเงื่อนไขตามที่กำหนด คือ ถ้าค่า Error น้อยกว่า 0.05 ให้หยุดการฝึกสอน

3.3 ส่วนของข้อมูล

ในโปรแกรมนี้ข้อมูลที่รับเข้ามาจะเป็นข้อมูลราคาหลักทรัพย์อาจจะเป็นเราเปิดหรือราคาปิด รูปแบบของข้อมูลที่รับเข้ามาแบ่งเป็น 2 แบบ

1. Text File
   จะแปลงให้ค่าอยู่ระหว่าง 0 ถึง 1 โดยหาค่าที่มากที่สุดแล้วนำมาหาร
2. Database
   จะแปลงเป็น Text File แล้วแปลงให้ค่าอยู่ระหว่าง 0 ถึง 1 อีกที

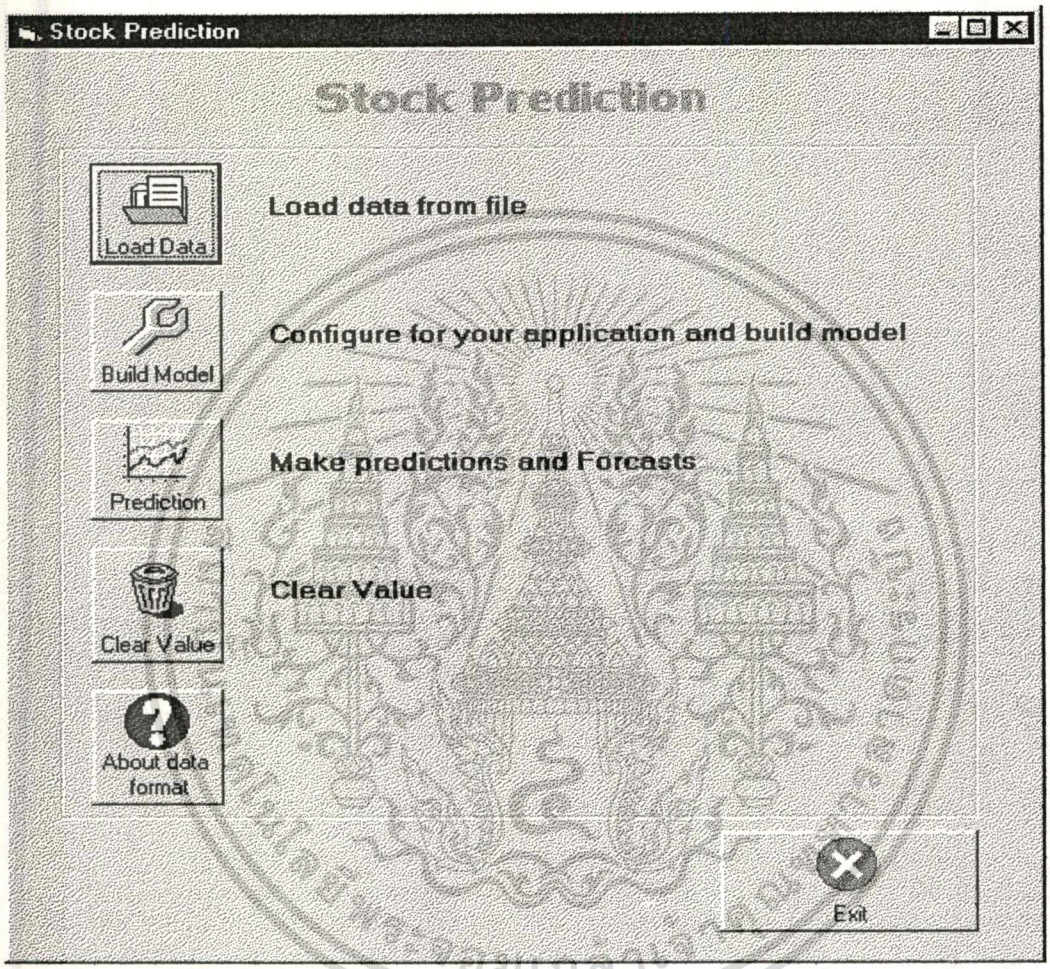
3.4 ส่วนของโปรแกรม

การศึกษาโครงงานนี้ได้ใช้โปรแกรมวิชวลเบสิค เวอร์ชั่น 6.0 (Visual basic v 6.0) ในการพัฒนาโปรแกรม เพื่อนำทฤษฎีดาต้าไมนิ่ง ทฤษฎีโครงข่ายประสาทเทียมแบบแบคพรอพพาเกชั่น และทฤษฎีอนุกรมเวลามาประยุกต์เพื่อใช้ในการพยากรณ์ราคาหลักทรัพย์ ซึ่งแสดงดังหน้าจอต่างๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.1 หน้าจอเมนูหลัก

เมื่อเริ่มเรียกใช้งานโปรแกรมสำหรับโครงงานนี้จะแสดงหน้าจอเมนูหลักสำหรับที่ผู้ใช้สามารถจะเลือกทำตามขั้นตอนต่างๆของโปรแกรมได้ ดังรูปที่ 3.1



รูปที่ 3.1 เมนูหลัก

จากรูปจะเห็นว่ามีปุ่มดังนี้

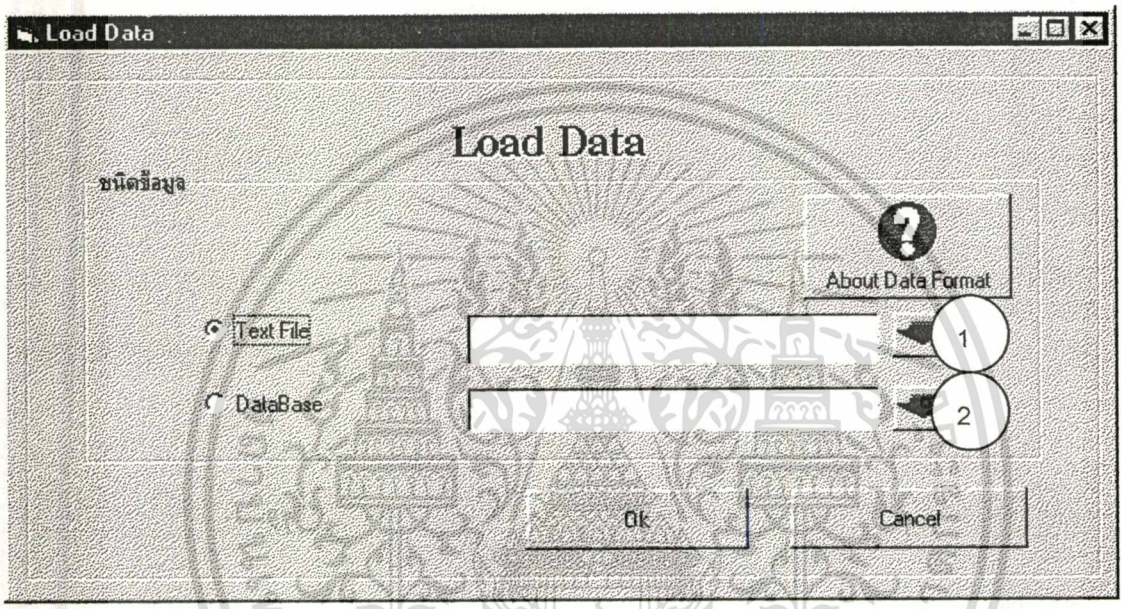
Load Data	ใช้สำหรับนำข้อมูลเข้าในโปรแกรมเพื่อสร้าง Neuron Network
Build Model	ใช้สำหรับสร้าง Neuron Network Model
Prediction	ใช้สำหรับนำข้อมูลที่จะพยากรณ์เข้ามาใน Network แล้วทำการพยากรณ์ค่าออกมาพร้อมกับค่า Error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Clear Value ล้างค่าต่างๆที่เก็บไว้ใน File เช่น ค่าในการเปลี่ยนแปลง Weight,ค่า Output

About data format บอกถึงลักษณะข้อมูลที่สามารถจะนำเข้า Network ได้

3.4.2 หน้าจอ Load Data

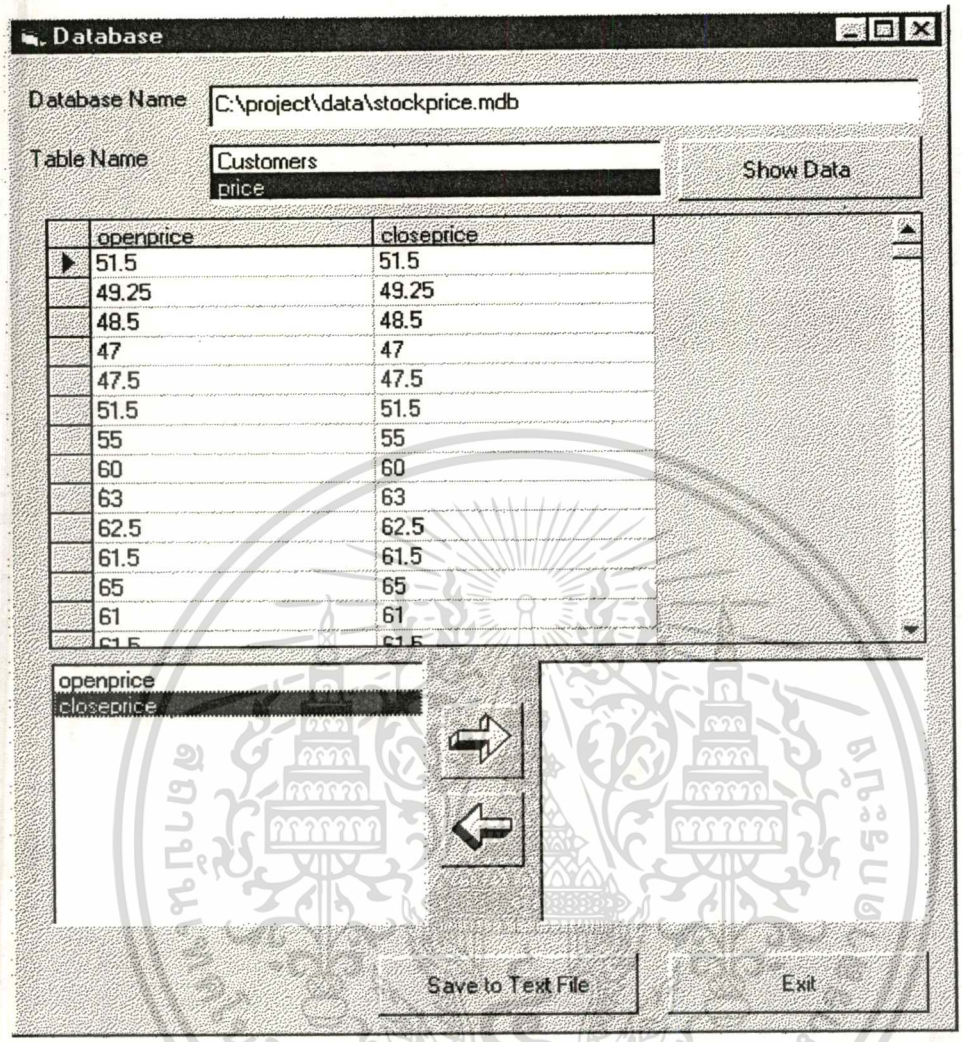


รูปที่ 3.2 แสดงหน้าจอ Load Data

จากรูปที่ 3.2 จะแสดงถึงการนำข้อมูลเข้าสู่การสร้าง Neuron Network ซึ่งจะแบ่ง

- Text File ซึ่งสามารถเลือกว่าจะเอา Text File ไหนเข้ามาโดยกดที่หมายเลข 1 จะสามารถเลือกได้ว่าจะเอา Text File จากที่ไหน แล้วจากนั้นกดปุ่ม OK โปรแกรมก็จะทำการแปลงให้มีค่าอยู่ระหว่าง 0 ถึง 1 เพื่อนำไปใช้สร้าง Neuron Network ต่อไป
- DataBase ซึ่งจะสามารถเลือกว่าจะเอา DataBase ไหนเข้ามาโดยกดที่หมายเลข 2 จะปรากฏดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

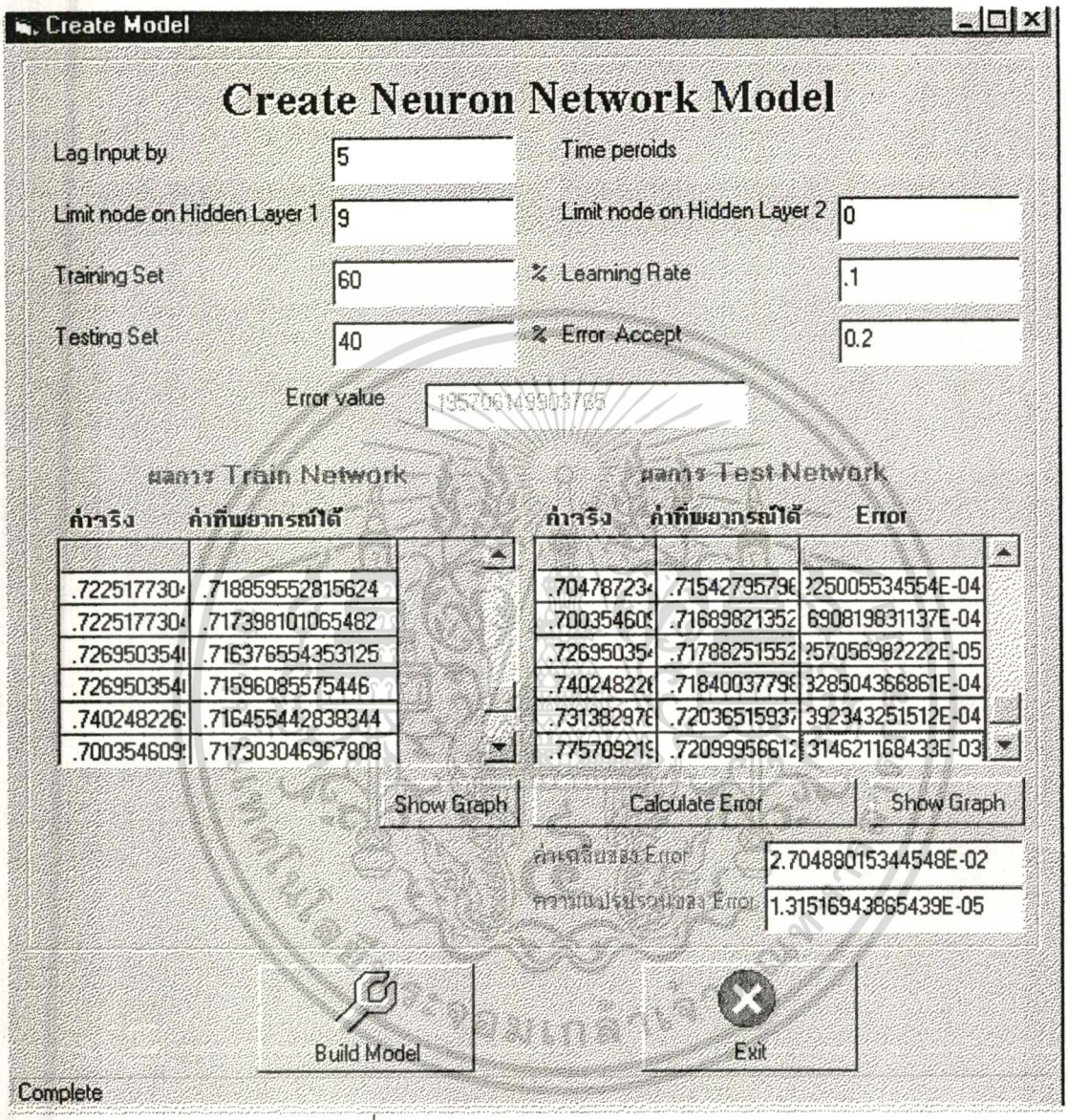


รูปที่ 3.3 Load DataBase เข้ามาใน Neuron Network

จากรูปจะเห็นว่าจะแสดงชื่อ DataBase ที่เราเลือก แล้วจะขึ้นชื่อ Table ทั้งหมดนั้นที่อยู่ใน DataBase ซึ่งเราจะสามารถเลือกได้ แล้วกดปุ่ม Show Data จะแสดงข้อมูลทั้งหมดออกมาและชื่อ Field ทั้งหมดออกมาด้วย แล้วจึงเลือกว่าเราจะเอา Field ไหนเป็นข้อมูลเข้าไปสร้าง Neuron Network แล้วกดปุ่ม Save to Text File เพื่อแปลงไปเป็น Text File อีกที แล้วจะแปลงให้มีค่าอยู่ระหว่าง 0 ถึง 1 จึงจะสามารถนำไปใช้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3 หน้าจอ Build Model



รูปที่ 3.4 Create Neuron Network Model

จากรูปจะต้องระบุ

- จำนวน Input ที่จะใช้ในการสร้าง Neuron Network (Lag Input by)
- จำนวน Node ของ Hidden Layer ชั้นที่ 1 (Limit Node on Neuron Network 1)
- จำนวน Node ของ Hidden Layer ชั้นที่ 2 (Limit Node on Neuron Network 2) กรณีมี Hidden 2 Layer
- ขนาดของ Training Set (เป็น %)

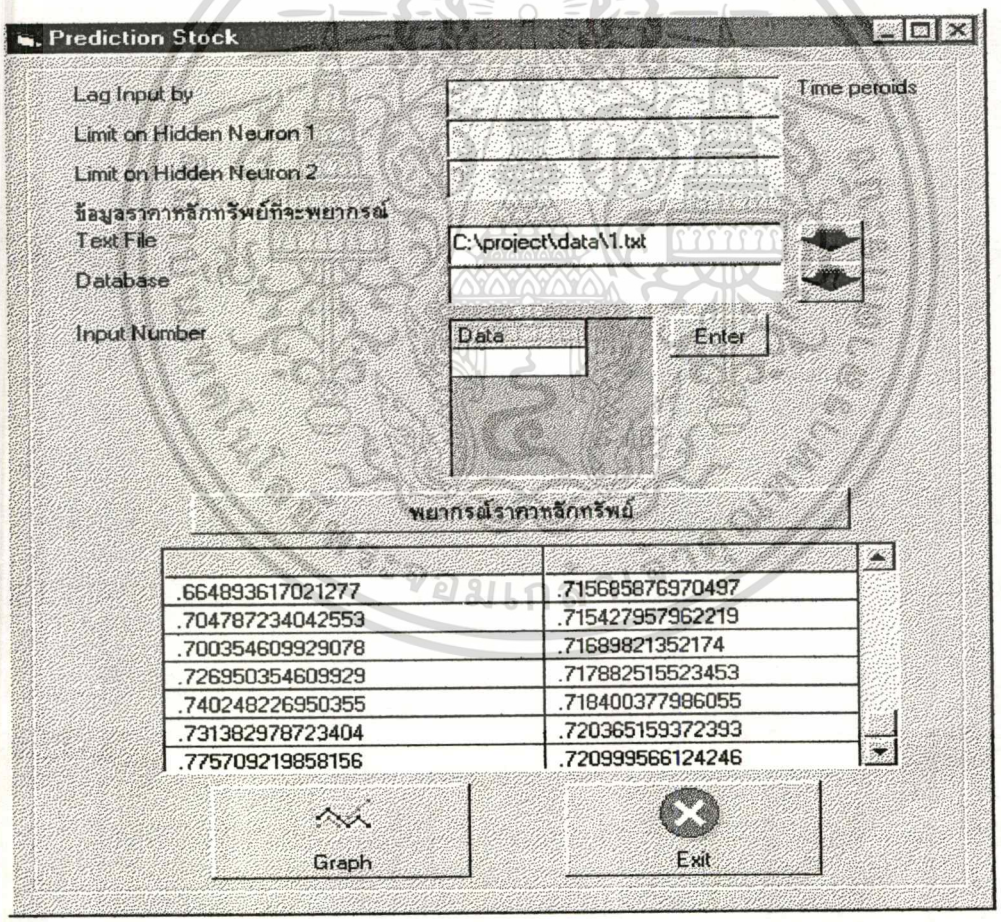
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขนาดของ Testing Set (เป็น %)
- ค่า Learning Rate
- ค่า Error Accept

แล้วกดปุ่ม Build Model เพื่อสร้าง Neuron Network ตามเงื่อนไขที่ระบุไว้ข้างต้น โปรแกรมจะทำการ Train Network ไปจนค่า Error น้อยกว่า 0.05 ก็จะหยุดการสร้าง Network

แล้วสังเกตว่าจะปรากฏค่าข้อมูลจริงที่เราใส่เข้าไปกับค่าที่เราพยากรณ์ได้ทั้งตอน Train และ Test ในตาราง แล้วถ้าต้องการหาค่า Error ,ค่าเฉลี่ยของ Error และค่าความแปรปรวนของ Error ให้กดปุ่ม Calculate Error ก็จะปรากฏต่างๆให้ดู

3.4.4 หน้าจอ Prediction



รูปที่ 3.5 Make Prediction

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปสิ่งที่ต้องระบุเพื่อพยากรณ์ค่า

- ข้อมูลอาจจะเป็น

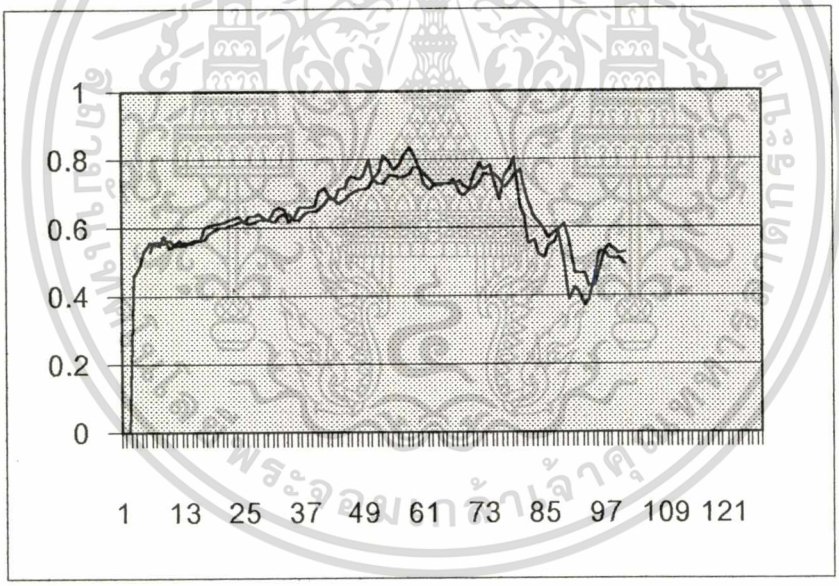
  Text File

  DataBase

  ป้อนเข้าโดยตรงจากหน้าจอ

- จำนวนข้อมูลเข้า (Lag input by) ปกติจะ defalut เป็นค่าที่เราใช้ตอนสร้าง Model
- จำนวน Node ใน Hidden Layer 1 ปกติจะ defalut เป็นค่าที่เราใช้ตอนสร้าง Model
- จำนวน Node ใน Hidden Layer 2 กรณีมี 2 Layer ปกติจะ defalut เป็นค่าที่เราใช้ตอนสร้าง Model

แล้วกดปุ่มพยากรณ์ราคาหลักทรัพย์ ก็จะได้ผลหลักทรัพย์ที่พยากรณ์ได้ และค่า Error ที่หาได้ หรือสามารถแสดงเป็นกราฟด้วยการกดปุ่ม Graph ดังรูปที่ 3.6



รูปที่ 3.6 กราฟเปรียบเทียบระหว่างค่าจริงกับค่าที่พยากรณ์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.5 หน้าจอ Clear Value

เมื่อสร้าง Neuron Network ขึ้นมาเรียบร้อยแล้ว แล้วถ้าต้องการเปลี่ยนค่าข้อมูลเข้า แล้วสร้าง Network ใหม่ จะต้องทำการล้างค่าต่างๆออกก่อน เพราะในการสร้าง Network นั้นจะเก็บค่าต่างๆไว้เป็น Text File

3.4.5 หน้าจอ About Data Format

จะบอกรายละเอียดของข้อมูลที่จะนำเข้าสู่การสร้าง Neuron Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

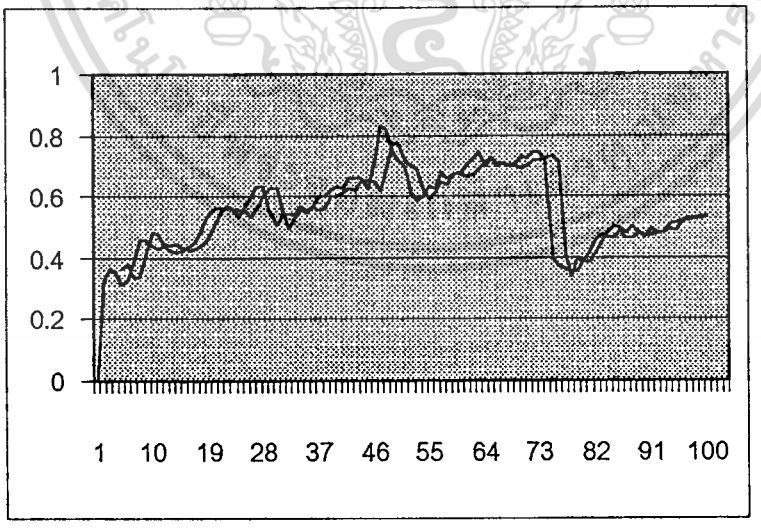
บทที่ 4

ผลการดำเนินการศึกษา

ผลที่ได้จากการพัฒนาโปรแกรมการพยากรณ์ราคาหลักทรัพย์โดยใช้โครงข่ายประสาทเทียมแบบแบคพรอพพาเกชั่น โดยใช้ข้อมูลราคาปิดหลักทรัพย์ของ TFB ย้อนหลังตั้งแต่วันที่ 5 มิถุยายน 2536 ถึงวันที่ 5 มิถุนายน 2541 เป็นข้อมูลที่ใช้ในการฝึกสอน ผลที่ได้จากการดำเนินการศึกษาสามารถสรุปได้ดังนี้

4.1 การเปรียบเทียบผลที่ได้จากการ Training

จากข้อมูลดังตารางภาคผนวกที่ 1 (ราคาหลักทรัพย์จริง) นำมาสร้าง Network โดยกำหนดค่า Lag Input by = 3 , Limit node on Hidden Layer1 = 6 , Learning Rate = 0.5 , Error Accept = 0.1, Training set = 60% ของข้อมูลทั้งหมด จะได้ราคาหลักทรัพย์ที่พยากรณ์ได้ดังตารางภาคผนวกที่ 1 (ราคาหลักทรัพย์ที่พยากรณ์ได้) และกราฟเปรียบเทียบระหว่างราคาหลักทรัพย์จริงกับราคาหลักทรัพย์ที่ได้จากการพยากรณ์ดังรูปที่ 4.1

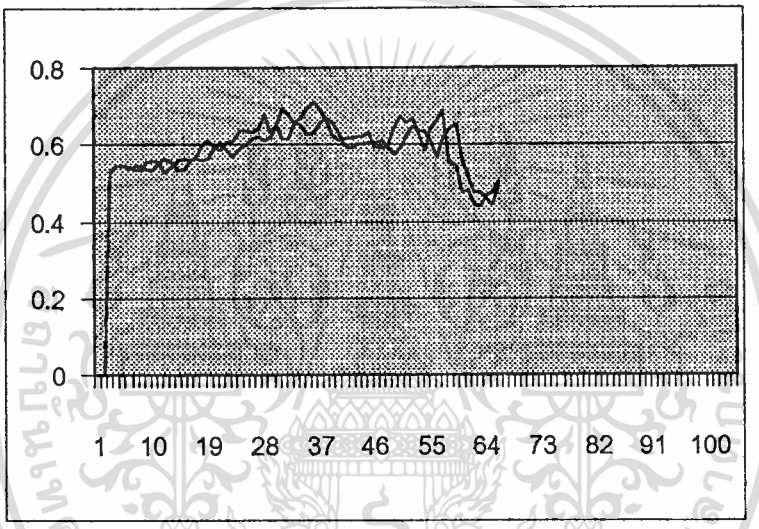


รูปที่ 4.1 Training : กราฟเปรียบเทียบระหว่างราคาหลักทรัพย์จริงกับราคาหลักทรัพย์ที่ได้จากการพยากรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การเปรียบเทียบผลที่ได้จากการ Testing

จากข้อมูลดังตารางที่ 2 (ราคาหลักทรัพย์จริง) ในภาคผนวก เอาเข้าไปใน Network ที่ได้สร้างเรียบร้อยแล้วโดยกำหนดค่า Lag Input by = 3 , Limit node on Hidden Layer1 = 6 , Learning Rate = 0.5 , Error Accept = 0.1, Testing set = 40% ของข้อมูลทั้งหมด จะได้กราฟเปรียบเทียบระหว่างราคาหลักทรัพย์จริงกับราคาหลักทรัพย์ที่ได้จากการพยากรณ์ดังรูปที่ 4.2 (ราคาหลักทรัพย์ที่พยากรณ์ได้) ในภาคผนวก และได้ค่าเฉลี่ยของ Error ประมาณ 1.95979053445629E-03 และค่าความแปรแรวนของ Error ประมาณ 6.01501722908322E-08



รูปที่ 4.2 Testing : กราฟเปรียบเทียบระหว่างราคาหลักทรัพย์จริงกับราคาหลักทรัพย์ที่ได้จากการพยากรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการดำเนินการศึกษาและข้อเสนอแนะ

5.1 สรุปผลที่ได้จากการดำเนินการศึกษา

การพยากรณ์ราคาหลักทรัพย์โดยใช้โครงข่ายประสาทเทียมแบบแบคพรอพพาเกชั่น (Backpropagation Neural Network) การดำเนินการศึกษานี้ได้ดำเนินการโดยใช้ข้อมูลราคาปิดหลักทรัพย์ของ TFB ย้อนหลังตั้งแต่วันที่ 5 มิถุยายน 2536 ถึงวันที่ 5 มิถุนายน 2541 จะเห็นว่าผลที่ได้บางช่วงมีค่าไม่ใกล้เคียงกัน ทั้งนี้เพราะเนื่องจากราคาหลักทรัพย์ไม่ได้ขึ้นอยู่กับราคาหลักทรัพย์ในอดีตเพียงอย่างเดียว แต่ขึ้นอยู่กับปัจจัยอย่างอื่นด้วย เช่นปัจจัยทางเศรษฐกิจ อื่นๆ และขึ้นกับการ Train Network ว่าเพียงพอหรือไม่ การแบ่งข้อมูลที่ไว้ Train และ Test เป็นต้น

5.2 ข้อเสนอแนะ

5.2.1 การเพิ่มความแม่นยำในการทำนาย

สิ่งที่สำคัญที่สุดของการทำนายอนุกรมเวลาคือ ความแม่นยำของการทำนายหรืออาจจะกล่าวได้ว่า ทำอย่างไรจึงจะทำนายข้อมูลให้มีความถูกต้องมากที่สุด ประเด็นต่างๆที่มีผลต่อความแม่นยำในการทำนายข้อมูล

1. จำนวนข้อมูลย้อนหลังที่นำไปใช้ในการทำนาย
2. วิธีการวิเคราะห์และปรับปรุงข้อมูลก่อนที่จะมาใช้ในการทำนาย

5.2.2 การนำไปใช้

การนำผลการศึกษาไปใช้งานจริงอาจจะมีความคลาดเคลื่อน เพราะราคาหลักทรัพย์จะขึ้นอยู่กับปัจจัยอื่นๆด้วยเช่น

- การวิเคราะห์เศรษฐกิจโดยทั่วไป (Economic Analysis)

การวิเคราะห์เศรษฐกิจเน้นการวิเคราะห์ภาวะเศรษฐกิจที่เป็นมาตลอดจนแนวโน้มของภาวะเศรษฐกิจในอนาคต ซึ่งอาจเกี่ยวข้องสัมพันธ์กับเศรษฐกิจโลกและแนวนโยบายเศรษฐกิจของรัฐบาลว่าจะมีผลกระทบต่อธุรกิจที่ออกหลักทรัพย์หรือไม่ เพียงใด นอกจากนั้น การวิเคราะห์เศรษฐกิจนี้ยังรวมถึงการวิเคราะห์วัฏจักรธุรกิจและนโยบายทางเศรษฐกิจที่สำคัญของรัฐบาลด้วย เช่น นโยบายการเงินและการคลัง เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การวิเคราะห์อุตสาหกรรม (Industry Analysis)

การวิเคราห์อุตสาหกรรมเน้นการวิเคราะห์วงจรอุตสาหกรรม (Industry Life Cycle) สภาพการตลาดและการแข่งขันตลอดจนอนาคตของอุตสาหกรรมว่าจะมีแนวโน้มอัตราการเจริญเติบโตอย่างไร ซึ่งขึ้นอยู่กับปัจจัยหลายอย่างด้วยกัน เช่น นโยบายของรัฐบาลที่จะให้การสนับสนุนหรือเป็นอุปสรรค โดยโครงสร้างการเปลี่ยนแปลงของระบบภาษีของรัฐบาล โครงสร้างของอุตสาหกรรมแต่ละประเภท เป็นต้น

-การวิเคราะห์บริษัท (Company Analysis)

การวิเคราะห์บริษัทเป็นขั้นสุดท้ายของการวิเคราะห์ โดยเน้นการวิเคราะห์ประเภทของบริษัท และประเภทของหลักทรัพย์ โดยจะวิเคราะห์ทั้งเชิงคุณภาพ (Qualitative Analysis) อันได้แก่ การวิเคราะห์ประสิทธิภาพการบริหารของผู้บริหาร ส่วนแบ่งทารตลาดของบริษัท โครงการขยายโรงงานของบริษัท เป็นต้น และการวิเคราะห์เชิงปริมาณ (Quantitative Analysis) อันได้แก่ การวิเคราะห์จากงบแสดงฐานะการเงินของธุรกิจในอดีตและปัจจุบัน เพื่อนำมาประมาณการกำไรต่อหุ้นและราคาหุ้นในอนาคตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ตลาดหลักทรัพย์แห่งประเทศไทย. 1998. **การวิเคราะห์หลักทรัพย์โดยใช้ปัจจัยพื้นฐาน.**: ISBN

Cabena and Hadjinian and Stadler and Verhees and Zanasi. 1998. **DISCOVERING DATA MINING From Concept to Implementation.**: Prentice Hall ,New Jersey.

David M. Skapura. 1995. **Building neural networks.**: ACM Press Newyork,New York.

Joseph S. Zirilli. 1997. **Financial Prediction using Neural Networks.**: INTERNATIONAL THOMSON COMPUTER PRESS.

Michael J.A.Berry and Gordon Linoff. **Data Mining Techniques for Marketing,Sales,and Customer Support.**: WILEY COMPUTER PUBLISHING

Sholom M. Weiss and Nitin Indurkhya. 1998. **Predictive Data Mining.**: Morgan Kaufmann,San Francisco.

Tom M. Mitchell. 1997. **MACHING LEARNING.**: McGRAW-HILL INTERNATIONAL EDITIONS.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1 Trainingเปรียบเทียบราคาหลักทรัพย์จริงกับราคาหลัก
หลักทรัพย์ที่พยากรณ์ได้

ราคาหลักทรัพย์จริง	ราคาหลักทรัพย์ที่พยากรณ์ได้
0.329545455	
0.363636364	
0.352272727	
0.314393939	0.364935337
0.329545455	0.379748053
0.390151515	0.331833015
0.454545455	0.340687837
0.454545455	0.410184744
0.435606061	0.481054081
0.431818182	0.478688072
0.439393939	0.444411001
0.424242424	0.43631707
0.420454545	0.446841713
0.424242424	0.433223998
0.435606061	0.424259301
0.458333333	0.430257315
0.503787879	0.442670095
0.541666667	0.466088636
0.564393939	0.511845039
0.560606061	0.550538409
0.560606061	0.568457154
0.560606061	0.561352824
0.537878788	0.556353528
0.568181818	0.557476011
0.602272727	0.536392732
0.636363636	0.561140831
0.636363636	0.601249398
0.556818182	0.62992294
0.511363636	0.629954279
0.541666667	0.554718675

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ราคาหลักทรัพย์จริง	ราคาหลักทรัพย์ที่พยากรณ์ได้
0.545454545	0.496836694
0.545454545	0.536012808
0.568181818	0.554066953
0.553030303	0.548158835
0.568181818	0.56827836
0.602272727	0.558128366
0.598484848	0.564847644
0.621212121	0.600812517
0.632575758	0.600396883
0.625	0.613246125
0.662878788	0.627490061
0.662878788	0.619613508
0.659090909	0.647893816
0.628787879	0.655634827
0.71969697	0.647501455
0.833333333	0.623399317
0.818181818	0.690140858
0.746212121	0.77163866
0.71969697	0.769421098
0.700757576	0.718431002
0.613636364	0.696151506
0.590909091	0.691083939
0.598484848	0.625405928
0.636363636	0.593490396
0.625	0.611392718
0.681818182	0.646684952
0.662878788	0.641182938
0.670454545	0.676438807
0.678030303	0.672263677
0.704545455	0.666350121
0.723484848	0.676123737
0.742424242	0.695157863
0.700757576	0.710885387

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ราคาหลักทรัพย์จริง	ราคาหลักทรัพย์ที่พยากรณ์ได้
0.723484848	0.722871989
0.704545455	0.69687663
0.708333333	0.705049804
0.696969697	0.701818968
0.704545455	0.699276
0.734848485	0.69507786
0.727272727	0.698531177
0.742424242	0.721376429
0.746212121	0.720176124
0.71969697	0.725740077
0.390151515	0.731816946
0.373106061	0.714774966
0.367424242	0.414639515
0.356060606	0.337199918
0.359848485	0.399264825
0.390151515	0.387900017
0.416666667	0.388489465
0.454545455	0.420194709
0.477272727	0.449934878
0.473484848	0.484306941
0.465909091	0.506195172
0.492424242	0.496668255
0.462121212	0.48177167
0.465909091	0.506094401
0.481060606	0.48123281
0.46969697	0.472257719
0.477272727	0.493734189
0.484848485	0.483356966
0.481060606	0.484504535
0.511363636	0.495085608
0.518939394	0.490271935
0.522727273	0.516963553
0.522727273	0.529866071

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ราคาหลักทรัพย์จริง	ราคาหลักทรัพย์ที่พยากรณ์ได้
0.526515152	0.527896499
0.534090909	0.526591362
0.537878788	0.536789807

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2 Testing: เปรียบเทียบราคาหลักทรัพย์จริงกับราคาหลัก-หลักทรัพย์ที่พยากรณ์ได้

ราคาหลักทรัพย์จริง	ราคาหลักทรัพย์ที่พยากรณ์ได้
0.53030303	
0.541666667	
0.541666667	
0.545454545	0.540359135
0.537878788	0.544506943
0.534090909	0.545535988
0.556818182	0.539724217
0.560606061	0.534420759
0.556818182	0.554560785
0.53030303	0.562234431
0.541666667	0.555423601
0.560606061	0.531453906
0.564393939	0.537409576
0.560606061	0.560420175
0.568181818	0.564573351
0.602272727	0.558501829
0.613636364	0.563442348
0.59469697	0.592785556
0.583333333	0.605504824
0.606060606	0.58651843
0.613636364	0.572260991
0.636363636	0.592068319
0.640151515	0.603457626
0.632575758	0.61781628
0.643939394	0.622670495
0.681818182	0.613782225
0.632575758	0.620456811
0.651515152	0.650098397
0.693181818	0.618639576
0.678030303	0.619006426

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ราคาหลักทรัพย์จริง	ราคาหลักทรัพย์ที่พยากรณ์ได้
0.659090909	0.658730667
0.666666667	0.651227812
0.693181818	0.629199645
0.712121212	0.634435715
0.696969697	0.65671747
0.662878788	0.671346125
0.625	0.660013227
0.609848485	0.631470106
0.617424242	0.600869621
0.617424242	0.589316946
0.621212121	0.599654024
0.621212121	0.603473086
0.632575758	0.605047418
0.598484848	0.605622828
0.590909091	0.613551918
0.606060606	0.589074737
0.647727273	0.575509185
0.674242424	0.59257956
0.659090909	0.627538113
0.666666667	0.649557474
0.628787879	0.636113912
0.583333333	0.635052674
0.643939394	0.61097574
0.662878788	0.567854895
0.689393939	0.615106533
0.560606061	0.644787616
0.541666667	0.65599458
0.473484848	0.559691771
0.481060606	0.516226305
0.446969697	0.478082976
0.439393939	0.476417568
0.465909091	0.460720265
0.477272727	0.445739166

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ราคาหลักทรัพย์จริง	ราคาหลักทรัพย์ที่พยากรณ์ได้
0.503787879	0.492583237

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์สโค้ดของโปรแกรม

```
Public type_mem As Integer
Option Explicit
Dim sSql As String
Dim db As Database
Dim rs As Recordset

Function randweight()
Dim numrnd
 Randomize
 numrnd = (2 * Rnd) - 1
 randweight = numrnd
 'MsgBox (randweight)
End Function

Function nethidden()
 nethidden = (wh * hinput)
End Function

Function acthidden(netih)
 acthidden = 1 / (1 + Exp(-netih))
End Function

Function netout(wo, acthi)
 netout = (wo * acthi)
End Function

Function actout(neto)
 actout = 1 / (1 + Exp(-neto))
End Function

Function deltaout(deltar, act_out)
 deltaout = (deltar - act_out) * (act_out * (1 - act_out))
End Function

Function deltaweightout(delo, fhidden)
 deltaweightout = (delo * fhidden)
End Function
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Function deltaweight(delh, inhi)
 deltaweighthi = (delh * inhi)
End Function

Function newweight(delw, wold)
 newweight = wold + delw
End Function

Function writewhi1(vin, vhi1)  'weight for hidden 1,vin = num neuron of input layer ,vhi1 = num neuron of hidden 1
 Dim wh()
 n = (vin * vhi1) - 1
 Open "c:\project\data\weighthi1.txt" For Output As #3
 For i = 0 To (n - 1)
  ReDim wh(i)
   wh(i) = randweight()
   Print #3, wh(i)
  Next i
 Close #3
End Function

Function writeout(vhi1)  'weight for output,vin = num neuron of input layer ,vhi1 = num neuron of hidden 1
 Dim wh3()
 n = (vin2 - 1)
 Open "c:\project\data\weightout.txt" For Output As #5
 For i = 0 To n
  ReDim wh3(i)
   wh3(i) = randweight()
   Print #5, wh3(i)
  Next i
 Close #5
End Function

Function net_acthi1(vhi1) 'cal of net for hidden1
 Dim neth2()
 Dim net()
 Dim acthi()
 Dim neth()
 hi1 = vhi1 - 1
 Open "c:\project\data\weighthi1.txt" For Input As #7
 Open "c:\project\data\acthi1.txt" For Output As #8
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
For j = 0 To hi1
 ReDim neth2(j)
 ReDim net(j)
 ReDim acthi(j)
 sumnet = 0
 vx = 0
 Open "c;\project\data\binary_input.txt" For Input As #9
 Do While Not EOF(9)
  ReDim neth(vx)
  Line Input #9, vinput
  If EOF(7) Then
   Exit Do
  End If
  Line Input #7, vwin
  wh = vwin
  hinput = vinput
  neth(vx) = nethidden(wh, hinput)
  sumnet = sumnet + neth(vx)
  vx = vx + 1
 Loop
 Close #9
 netih = sumnet
 acthi(j) = acthidden(netih)
 Print #8, acthi(j)
Next j
Close #8
Close #7
End Function

Function net_actout(nvhi1) 'net for output
 Dim output1()
 nn2 = nvhi1 - 1
 Open "c:\project\data\actout.txt" For Output As #13
 Open "c:\project\data\weightout.txt" For Input As #14
 Open "c:\project\data\acthi2.txt" For Input As #50
 sumout = 0
 For l1 = 0 To nn2
  ReDim output1(l1)
  Line Input #14, vwout
  Line Input #50, vhi1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  wo = vwout
  acthiout = vhi1
  output1(l1) = netout(wo, acthiout)
  sumout = sumout + output1(l1)
 Next l1
 Close #14
 Close #50
 neto = sumout
 funcout = actout(neto)
 Print #13, funcout
 Close #13
End Function

Private Sub Command1_Click()
 loaddata.Show
End Sub
Private Sub Command2_Click()
  CreateModel.Show
End Sub

Private Sub Command3_Click()
 Prediction.Show
End Sub

Private Sub Command4_Click(Index As Integer)
 aboutdata.Show
End Sub


Private Sub Command5_Click()
 End
End Sub

Private Sub Command6_Click()
 Kill "c:\project\data\input.txt"
 Kill "c:\project\data\actout.txt"
 Kill "c:\project\data\binary_input.txt"
 Kill "c:\project\data\binary_input_test.txt"
 Kill "c:\project\data\target.txt"
 Kill "c:\project\data\target_test.txt"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Kill "c:\project\data\train_result.txt"
Kill "c:\project\data\test.txt"
Kill "c:\project\data\test_result.txt"
Kill "c:\project\data\train.txt"
Kill "c:\project\data\weighthi1.txt"
Kill "c:\project\data\weightout.txt"
Kill "c:\project\data\predict.txt"
Kill "c:\project\data\predict_result.txt"
End Sub

Private Sub Command7_Click()
 i = 12
 Do
  i = i - 1
  MsgBox (i)
 Loop Until i < 10
End Sub

Private Sub Command1_Click()
Dim Col1, Col2 As Column
Dim SQL As String
Dim db As Database
Dim rec As Recordset
Dim mem
Dim Entry, Msg
Dim tdfNew As TableDef
Dim i, j As Integer

Set db = OpenDatabase(databasename.Text)

mem = TableName.List(TableName.listindex)

 SQL = "select * from " & mem & ""
 Adodc1.RecordSource = SQL

 Set DataGrid1.DataSource = Adodc1

 For i = 0 To Data1.Database.TableDefs.count - 1
  If(Data1.Database.TableDefs(i).Attributes) = 0 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
For j = 0 To Data1.Database.TableDefs(mem).Fields.count - 1
  List1.AddItem Data1.Database.TableDefs(mem).Fields(j).name

Next j
Exit For
End If
Next i


With Adodc1

  .RecordSource = mem
End With
End Sub

Private Sub Command2_Click()
Dim choice As Integer
Dim Col1, Col2 As Column
Dim sSql As String
Dim db As Database
Dim rs As Recordset
Dim mem
Dim i As Integer
Dim listindex

 Set db = OpenDatabase(databasename.Text, False, False)
 Open "c:\project\data\data.txt" For Output As #1

 mem = List2.List(listindex)
  sSql = "Select * from " & TableName & ""
   Set rs = db.OpenRecordset(sSql, dbOpenDynaset)
   With rs
       Do While Not .EOF
         Print #1, rs(mem)
         .MoveNext
       Loop
   End With
 MsgBox ("จัดเก็บข้อมูลเรียบร้อยแล้ว")
 Close #1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub Command3_Click()
 Hide
 List1.Clear
 List2.Clear
 DataGrid1.ClearFields
 DataGrid1.ClearSelCols
 DataGrid1.Refresh
End Sub

Private Sub Command4_Click()
Dim Entry, i, Msg
Dim mem
 mem = List2.List(List2.listindex)
 List2.RemoveItem List2.listindex
 List1.AddItem mem
End Sub

Private Sub Command5_Click()
Dim Entry, i, Msg
Dim mem
 mem = List1.List(List1.listindex)
 List1.RemoveItem List1.listindex

 List2.AddItem mem

End Sub

Private Sub Command6_Click()
Dim i, j As Integer
Data1.databasename = databasename.Text
Data1.Refresh

For i = 0 To Data1.Database.TableDefs.count - 1
If (Data1.Database.TableDefs(i).Attributes) = 0 Then

  Print "Table Name", Data1.Database.TableDefs(i).name
  For j = 0 To Data1.Database.TableDefs(i).Attributes - 1
    Print "Fields", Data1.Database.TableDefs(i).Fields(j).name
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Next j
End If
Next i
End Sub

Private Sub Form_Load()
Dim i, j As Integer
If loaddata.databasename.Text <> "" Then
  databasename.Text = loaddata.databasename.Text
  Data1.databasename = databasename.Text
  Data1.Refresh
End If

If Prediction.databasename_test <> "" Then
  databasename.Text = Prediction.databasename_test
  Data1.databasename = databasename.Text
  Data1.Refresh
End If

For i = 0 To Data1.Database.TableDefs.count - 1
If (Data1.Database.TableDefs(i).Attributes) = 0 Then
  TableName.AddItem Data1.Database.TableDefs(i).name
End If
Next i
End Sub

Dim vin
Dim vhi1
Dim vhi2
Dim vy
Dim h
Dim v_hin
Dim v_hili
Dim numrnd
Dim n
Dim vm
Public num_data As Integer
Public LEARNNING_RATE As Double
Public ErrorLimit As Double
Dim Error_no As Double
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Public data_Max1 As Double
Dim deltahi1(100)
Public num As Integer
Public s_stop As Boolean
Public filename As String
Public status_test As Boolean
Const b = 0
Option Explicit
'Option Base 1                    ' start array from 1
Public FirstLoop As Boolean
Dim Weights() As Double
Dim inputs() As Double
Dim Outputs() As Double
Dim TrainSet(12000) As Double   'for training data within 3 months
Dim Trainval() As Double
Dim TestSet(12000) As Double    'for testing data within 3 months
Dim TestVal() As Double
'Public LearningRate As Double
Public ActualOutput As Double
Public Inodes As Integer
Public Jnodes As Integer
Public Knodes As Integer
Public NumInputs As Integer
Public NumOutputs As Integer
Public MaxLine As Integer
Public SSE As Double
Public MaxCycles As Integer
Dim LayerPointer()  As Integer
Dim Opj() As Double
Dim Opj1() As Double
Dim Opj2() As Double
Dim Opk() As Double
Dim Opk1() As Double
Dim Opk2() As Double
Dim Opk3() As Double
Dim target As Double
Dim Epj() As Double
Dim Epk() As Double
Dim Predic() As Double
Dim Pre As Double
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Public HiddenLayerOutput As Double
Public OutputLayerOutput As Double
Dim HiddenError() As Double
Public OutputError As Double


Function nethidden(weight, hinput)
 nethidden = Val(weight) * Val(hinput)
End Function

Function acthidden(netih)
 acthidden = 1 / (1 + Exp(-netih))
End Function

Function netout(wo, acthi)
 netout = (Val(wo) * Val(acthi))
End Function

Function actout(neto)
 actout = 1 / (1 + Exp(-neto))
End Function

Function deltaout(deltar, act_out)
 deltaout = (Val(deltar) - Val(act_out)) * (Val(act_out) * (1 - Val(act_out)))
End Function

Function randweight()
 Randomize
 'numrnd = (2 * Rnd) - 1
 numrnd = Rnd
 randweight = numrnd
End Function
Function deltaweighthi(delh, inhi)
 deltaweighthi = (LEARNNING_RATE * Val(delh) * Val(inhi))
End Function
Function writewhi1(vin, vhi1)
Dim i
'vin is number of input layer
'vhi1 is number of hidden layer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim wh
n = (vin * vhi1) - 1
Open "c:\project\data\weighthi1.txt" For Output As #1
 For i = 0 To (n)
  'ReDim wh(i)
  wh = randweight()
  Print #1, wh
 Next i
 Close #1
End Function

Function writewhi2(vhi1, vhi2)
'vin is number of input layer
'vhi1 is number of hidden layer
 Dim wh()
 Dim i
 'Dim FileNumber

 'FileNumber = FreeFile

 n = (Val(vhi1) * Val(vhi2)) - 1
 Open "c:\project\data\weighthi2.txt" For Output As #76
  For i = 0 To (n)
   ReDim wh(i)
   wh(i) = randweight()
   Print #76, wh(i)
  Next i
 Close #76
End Function
Function writewout(vhi1)
  Dim wh3
  Dim i
  n = (Val(vhi1) - 1)

  Open "c:\project\data\weightout.txt" For Output As #26
  For i = 0 To n
   'ReDim wh3(i)
   wh3 = randweight()
   Print #26, wh3
  Next i
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Close #26
End Function
Function convert_data()
Dim data_Max As Double
'Dim num_data As Integer
Dim train_data As Double
Dim test_data As Double
Dim data
Dim Data1
Dim data2
Dim i
Dim train_d
Dim test_d
Dim j

 Open CreateModel.filename For Input As #1
 data_Max = 0
 num_data = 0
 Do While Not EOF(1)

  If EOF(1) Then
  Exit Do
  End If
  Line Input #1, data
  num_data = num_data + 1

  If data > data_Max Then
   data_Max = Val(data)
  Else
   data_Max = data_Max
  End If

 Loop

  data_Max1 = data_Max + data_Max * 0.2

  Open CreateModel.filename For Input As #2
  Open "c:\project\data\input.txt" For Output As #36
  Do While Not EOF(2)
  If EOF(2) Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Exit Do
End If
Line Input #2, Data1
data2 = Data1 / data_Max1
Print #36, data2
Loop
Close #2
Close #36
Close #1

train_data = Int(num_data * (Val(amount_train(3).Text) / 100) + 1)
test_data = Int(num_data * (Val(amount_test(2).Text) / 100) - 1)

Open "c:\project\data\input.txt" For Input As #37
Open "c:\project\data\train.txt" For Output As #38
For i = 1 To train_data
 Line Input #37, train_d
 Print #38, train_d
Next i
 Close #38

Open "c:\project\data\test.txt" For Output As #39
For j = (Val(train_data) + 1) To num_data
 Line Input #37, test_d
 Print #39, test_d
 Next j
 Close #37
 Close #39
End Function

Function net_acthi1(vhi1)
 Dim neth2()
 Dim net()
 Dim acthi()
 Dim neth
 Dim hi
 Dim j
 Dim sumnet
 Dim vx
 Dim vwin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim vinput
Dim wh
Dim hinput

hi = Val(vhi1) - 1
 Open "c:\project\data\weighthi1.txt" For Input As #35
 Open "c:\project\data\acthi1.txt" For Output As #4
For j = 0 To hi
 ReDim neth2(j)
 ReDim net(j)
 ReDim acthi(j)
 sumnet = 0
 vx = 0

 If status_test = True Then
 Open "c:\project\data\binary_input_test.txt" For Input As #5
 End If

 If status_test = False Then
 Open "c:\project\data\binary_input.txt" For Input As #5
 End If

 Do While Not EOF(5)
 ReDim neth(vx)


 Line Input #5, vinput
 If EOF(35) Then
  Exit Do
 End If
 Line Input #35, vwin
 wh = vwin
 hinput = vinput
 neth(vx) = nethidden(wh, hinput)
 sumnet = sumnet + neth(vx)
 vx = vx + 1
Loop
Close #5
neth = sumnet
acthi(j) = acthidden(neth)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Print #4, acthi(j)
Next j
Close #4
Close #35
End Function

Function net_acthi2(vhi2)
 Dim v_acthi2(1000)
 Dim v_neth()
 Dim l
 Dim n2
 Dim k
 Dim sumnet2
 Dim vwhi2
 Dim wh2
 Dim hinput2
 Dim nethi2
 'Dim neth2()
 'Dim net()
 'Dim acthi()
 'Dim neth
 'Dim sumnet
 l = 0
 n2 = vhi2 - 1
  Open "c:\project\data\weighthi2.txt" For Input As #10
  Open "c:\project\data\acthi2.txt" For Output As #11
 For k = 0 To n2
  ReDim v_achi2(k)
  'ReDim net(j)
  'ReDim acthi(j)
  sumnet2 = 0
  'vx = 0
  Open "c:\project\data\acthi1.txt" For Input As #12
  l = 0
  Do While Not EOF(12)
   ReDim v_neth(l)
   Line Input #12, vhi1
   If EOF(10) Then
    Exit Do
   End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Line Input #10, vwhi2
 wh2 = vwhi2
 hinput2 = vhi1
 v_neth(l) = nethidden(wh2, hinput2)
 sumnet2 = sumnet2 + v_neth(l)
 l = l + 1
Loop
Close #12
nethi2 = sumnet2
v_acthi2(k) = acthidden(nethi2)
Print #11, v_acthi2(k)
Next k
Close #10
Close #11
End Function

Function net_actout(nvhi2)
 Dim output1()
 Dim train_result(1200)
 Dim b As Integer
 Dim nn2
 Dim sumout
 Dim i1
 Dim vwout
 Dim nnvhi2
 Dim wo
 Dim acthiout
 Dim neto
 Dim funcout

 nn2 = nvhi2 - 1
 If status_test = True Then
   Open "c:\project\data\all_actout.txt" For Append As #10
 End If
 If status_test = False Then
 Open "c:\project\data\actout.txt" For Output As #11
 End If
  Open "c:\project\data\weightout.txt" For Input As #12
 If Hidden2 <> "0" Then
  Open "c:\project\data\acthi2.txt" For Input As #18
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If
sumout = 0

For i1 = 0 To nn2
 ReDim output1(i1)
 Line Input #12, vwout
 If (Hidden1 <> "0") And (Hidden2 = "0") Then
  If status_test = True Then
  Open "c:\project\data\target_test.txt" For Input As #19
  Line Input #19, nnvhi2
  Close #19
  End If

  If status_test = False Then
  Open "c:\project\data\target.txt" For Input As #19
  Line Input #19, nnvhi2
  Close #19
  End If
 End If
 If (Hidden1 <> "0") And (Hidden2 <> "0") Then
  Line Input #18, nnvhi2
 End If
 wo = vwout
 acthiout = nnvhi2
 output1(i1) = netout(wo, acthiout)
 sumout = sumout + output1(i1)
Next i1
Close #12
If Hidden2 <> "0" Then
 Close #18
End If
neto = sumout
funcout = actout(neto)
If status_test = True Then
 Print #10, funcout
 Close #10
End If
If status_test = False Then
 Print #11, funcout
 Close #11
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

If status_test = True Then
 Open "c:\project\data\test_result.txt" For Append As #12
 Print #12, funcout
 Close #12
End If

If status_test = False Then
 Open "c:\project\data\train_result.txt" For Append As #12
 Print #12, funcout
 Close #12
End If
End Function

Function errorp()
Dim finerror As Double
Dim p_error As Double
Dim terror
Dim vtargen
Dim v_actout
Dim fout
Dim tar

 terror = 0
 Open "c:\project\data\actout.txt" For Input As #21
 Open "c:\project\data\target.txt" For Input As #22
 Open "c:\project\data\errorp.txt" For Output As #23
 Line Input #22, vtargen
 Line Input #21, v_actout
 fout = v_actout
 tar = vtargen
 If tar = "" Or fout = Empty Then
  Close #21
  Close #22
  Close #23
  GoTo exitloop
 Else
  p_error = (tar - fout) ^ 2
  finerror = p_error / 2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Print #23, finerror
  'error.Text = finerror
  'DoEvents
  Close #21
  Close #22
  Close #23
 End If
 Error_no = finerror
Close #21
  Close #22
  Close #23
exitloop:
End Function

Function errorp_test()
Dim finerror As Double
Dim p_error As Double
Dim terror
Dim vtarget
Dim v_actout
Dim four
Dim tar
 terror = 0
 Open "c:\project\data\actout.txt" For Input As #21
 Open "c:\project\data\target.txt" For Input As #22
 Open "c:\project\data\errorp.txt" For Output As #23
 Line Input #22, vtarget
 Line Input #21, v_actout
 four = v_actout
 tar = vtarget
 If tar = "" Or four = Empty Then
  Close #21
  Close #22
  Close #23
  GoTo exitloop
 Else
  p_error = (tar - four) ^ 2
  finerror = (tar - four) ^ 2
  Print #23, finerror
  DoEvents
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Close #21
  Close #22
  Close #23
 End If
Close #21
  Close #22
  Close #23
exitloop:
End Function

Function sumerror()
Dim s_errp
Dim errp


 s_errp = 0
 Open "c:\project\data\errorp.txt" For Input As #121
 Do While Not EOF(121)
  Line Input #121, errp
  s_errp = s_errp + errp
 Loop
 Close #121
 sumerror = s_errp
End Function

Function delta_out()
Dim vtg
Dim deltar
Dim act_out
Dim dtout
Dim vacto

 Open "c:\project\data\target.txt" For Input As #31
 Open "c:\project\data\actout.txt" For Input As #32
 Open "c:\project\data\delta_out.txt" For Output As #33
 Line Input #31, vtg
 deltar = vtg
 Line Input #32, vacto
 act_out = vacto
 dtout = deltaout(deltar, act_out)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Print #33, dtout
Close #31
Close #32
Close #33,
End Function

Function deltaweightout(delo, fhidden)
deltaweightout = LEARNNING_RATE * Val(delo) * Val(fhidden)
End Function

Function delta_wout()
Dim del_w()
Dim i
Dim fhidden
Dim delo
Dim vd_out
Dim v_achi2
Dim v_achi1
If Hidden1.Text <> "0" And Hidden2.Text = "0" Then
i = 0
Open "c:\project\data\delta_out.txt" For Input As #41
Line Input #41, vd_out
delo = vd_out
Close #41
Open "c:\project\data\delta_wout.txt" For Output As #42
Open "c:\project\data\acthi1.txt" For Input As #43

Do While Not EOF(43)
ReDim del_w(i)
Line Input #43, v_achi1
fhidden = v_achi1
del_w(i) = deltaweightout(delo, fhidden)
Print #42, del_w(i)
i = i + 1
Loop
Close #42
Close #43
End If

If Hidden1.Text <> "0" And Hidden2.Text <> "0" Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
i = 0
Open "c:\project\data\delta_out.txt" For Input As #41
Line Input #41, vd_out
delo = vd_out
Close #41
Open "c:\project\data\delta_wout.txt" For Output As #42
Open "c:\project\data\acthi2.txt" For Input As #43

Do While Not EOF(43)
 ReDim del_w(i)
 Line Input #43, v_achi2
 fhidden = v_achi2
 del_w(i) = deltaweightout(delo, fhidden)
 Print #42, del_w(i)
 i = i + 1
Loop
Close #42
Close #43

End If
End Function

Function delta_hi2()
 Dim deltahi()
 Dim hn
 Dim v_dout, delout
 Dim w
 Dim fhi1
 Dim v_achi2
 Dim wdel
 Dim vwout, wdelout
 hn = 0
 Open "c:\project\data\delta_hi2.txt" For Output As #51
 'Open "c:\project\data\acthi1.txt" For Input As #52
 Open "c:\project\data\delta_out.txt" For Input As #9
 Line Input #9, v_dout
 delout = v_dout
 Close #9
 Open "c:\project\data\acthi2.txt" For Input As #52
 Open "c:\project\data\weightout.txt" For Input As #54
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Do While Not EOF(52)
  ReDim deltahi(hn)
  Line Input #52, v_achi2
  Line Input #54, vwout
  w = vwout
  fhi1 = v_achi2
  wdelout = w * delout
  deltahi(hn) = wdelout * (fhi1 * (1 - fhi1))
  Print #51, deltahi(hn)
  hn = hn + 1
Loop
Close #54
Close #52
'Close #9
Close #51
End Function

Function delta_whi2()
 Dim delwhi2()
 Dim j
 Dim inhi1
 Dim delh
 Dim vachi1
 Dim inhi
 Dim vdthi2
 j = 0
 Open "c:\project\data\delta_whi2.txt" For Output As #62
 Open "c:\project\data\acthi1.txt" For Input As #63
 Do While Not EOF(63)
   Line Input #63, vachi1
   inhi = vachi1
  Open "c:\project\data\delta_hi2.txt" For Input As #61
  Do While Not EOF(61)
   ReDim delwhi2(j)
   Line Input #61, vdthi2
   delh = vdthi2
   delwhi2(j) = deltaweighthi(delh, inhi)
   Print #62, delwhi2(j)
   j = j + 1
  Loop
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Close #61
 Loop
 Close #62
 Close #63
End Function

Function delta_hi1(vhi2, vhi1)
 Dim deltahi1(1000)
 Dim sumerror
 Dim m1, m2
 Dim wdelout, fhi
 Dim t, i, j
 Dim v_dout, delout, v_achi1, vwhi2, w
 Dim k
 If Hidden1 <> "0" And Hidden2 = "0" Then
 m1 = (vhi1 - 1)              '3
 'm2 = (vhi2 - 1)             '3
 t = ((vhi1) * (vhi2)) - 1    '3
 Open "c:\project\data\delta_hi1.txt" For Output As #51
 Open "c:\project\data\acthi1.txt" For Input As #52
 wdelout = 0
 Open "c:\project\data\delta_out.txt" For Input As #9
  Line Input #9, v_dout
  delout = v_dout
  Close #9
 Open "c:\project\data\weightout.txt" For Input As #54
  wdelout = 0
 For h = 0 To m1
  'ReDim deltahi1(h)
  Line Input #52, v_achi1
  Line Input #54, vwhi2
  w = vwhi2
  wdelout = (Val(w) * Val(delout))
   fhi = v_achi1
   deltahi1(h) = wdelout * (Val(fhi) * (1 - Val(fhi)))
   Print #51, deltahi1(h)
   wdelout = 0
  'Close #52
 Next h
 Close #54
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Close #52
Close #51
End If

If Hidden1.Text <> "0" And Hidden2.Text <> "0" Then
m1 = (vhi1 - 1)              '3
m2 = (vhi2 - 1)              '3
t = ((vhi1) * (vhi2)) - 1    '3
Open "c:\project\data\delta_hi1.txt" For Output As #51
Open "c:\project\data\acthi1.txt" For Input As #52
wdelout = 0
For h = 0 To m1
 'ReDim deltahi1(h)
 Line Input #52, v_achi1
 Open "c:\project\data\delta_hi2.txt" For Input As #9
 For i = 0 To m2
 Line Input #9, v_dout
 delout = v_dout
 Open "c:\project\data\weighthi2.txt" For Input As #54
 k = vhi1 * i + h

 For j = 0 To t
  Line Input #54, vwhi2
  If j = k Then
   w = vwhi2
  End If
 Next j
  Close #54

   wdelout = wdelout + (Val(w) * Val(delout))
 Next i
Close #9
  fhi = v_achi1
  deltahi1(h) = wdelout * (Val(fhi) * (1 - Val(fhi)))
  Print #51, deltahi1(h)
  wdelout = 0
 'Close #52
 Next h
 Close #52
 Close #51
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If
End Function
Function delta_whi1()
 Dim inputh1()
 Dim deltawhi1()
 Dim j
 Dim inhi, vinh1
 Dim vdelh1, delh, vdelhi1

 If Hidden1 <> "0" And Hidden2.Text = "0" Then
 j = 0
 Open "c:\project\data\delta_whi1.txt" For Output As #62
 Open "c:\project\data\binary_input.txt" For Input As #63
 Do While Not EOF(63)
  Line Input #63, vinh1
  inhi = vinh1
 Open "c:\project\data\delta_hi1.txt" For Input As #61
 Do While Not EOF(61)
  ReDim deltawhi1(j)
  Line Input #61, vdelh1
  delh = vdelh1
  deltawhi1(j) = deltaweighthi(delh, inhi)
  Print #62, deltawhi1(j)
  j = j + 1
 Loop
 Close #61
 Loop
 Close #62
 Close #63
 End If

 If Hidden1 <> "0" And Hidden2 <> "0" Then
 j = 0
 Open "c:\project\data\binary_input.txt" For Input As #62
 Open "c:\project\data\delta_whi1.txt" For Output As #63
 Do While Not EOF(62)
  Line Input #62, vinh1
  inhi = vinh1
 Open "c:\project\data\delta_hi1.txt" For Input As #61
 Do While Not EOF(61)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ReDim deltawhi1(j)
    Line Input #61, vdelhi1
    delh = vdelhi1
    deltawhi1(j) = deltaweighthi(delh, inhi)
    Print #63, deltawhi1(j)
    j = j + 1
   Loop
Close #61
  Loop
  Close #62

  Close #63
  End If
End Function

Function upwout()
  Dim rw()
  Dim i
  Dim w_out, delwout, vwout
  Dim dw_out
  i = 0
  Open "c:\project\data\weightout.txt" For Input As #71
  Open "c:\project\data\upweightout.txt" For Output As #72
  Open "c:\project\data\delta_wout.txt" For Input As #73
  Do While Not EOF(71)
   ReDim rw(i)
   Line Input #71, w_out
   Line Input #73, dw_out
   delwout = dw_out
   vwout = w_out
   rw(i) = Val(vwout) + Val(delwout)
   Print #72, rw(i)
   i = i + 1
  Loop
  Close #71
  Close #72
  Close #73
End Function

Function upwhi2()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim rw()
Dim i
Dim wold, delta_weight, volw, vdw
'Dim delta_weight(1000)
i = 0

Open "c:\project\data\upweighthi2.txt" For Output As #82
Open "c:\project\data\delta_whi2.txt" For Input As #84
Open "c:\project\data\weighthi2.txt" For Input As #80
  Do While Not EOF(80)
  ReDim rw(i)
  Line Input #80, wold
  Line Input #84, delta_weight
  volw = wold
  vdw = delta_weight
  rw(i) = Val(volw) + Val(vdw)
  Print #82, rw(i)
  i = i + 1
  Loop
  Close #80
  Close #82
  Close #84
End Function

Function upwhi1()
 Dim upwh()
 Dim i, cc
 Dim del_whi, whi_old, vdhi, voldw, v_whi1
 i = 0
 Open "c:\project\data\upweighthi1.txt" For Output As #82
 Open "c:\project\data\delta_whi1.txt" For Input As #83
 Open "c:\project\data\weighthi1.txt" For Input As #80

  Do While Not EOF(80)
  ReDim upwh(i)
  If EOF(83) Then
   Exit Do
   GoTo app_whi1
  End If
  Line Input #83, del_whi
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Line Input #80, whi_old
    vdhi = del_whi
    voldw = whi_old
    upwh(i) = Val(voldw) + Val(vdhi)
    Print #82, upwh(i)
    i = i + 1
   Loop
   Close #80
   Close #82
   Close #83

app_whi1:
   Close #80
   Close #82
   Close #83
   Open "c:\project\data\weighthi1.txt" For Input As #50
   Open "c:\project\data\upweighthi1.txt" For Append As #51
   cc = 0
   Do While Not EOF(50)
    Line Input #50, v_whi1
    If cc = i Then
     Print #51, v_whi1
     i = i + 1
    End If
    cc = cc + 1
   Loop
   Close #50
   Close #51
End Function

Function re_wout()
Dim vup
 Open "c:\project\data\upweightout.txt" For Input As #101
 'Kill "c:\project\data\weightout.txt"
 Open "c:\project\data\weightout.txt" For Output As #102
 Do While Not EOF(101)
  Line Input #101, vup
  Print #102, vup
 Loop
 Close #101
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Close #102
 Kill "c:\project\data\upweightout.txt"
End Function

Function re_whi2()
Dim vup
 Open "c:\project\data\upweighthi2.txt" For Input As #111
 Kill "c:\project\data\weighthi2.txt"
 Open "c:\project\data\weighthi2.txt" For Output As #112
 Do While Not EOF(111)
  Line Input #111, vup
  Print #112, vup
 Loop
 Close #111
 Close #112
End Function

Function re_whi1()
Dim vup
 Open "c:\project\data\upweighthi1.txt" For Input As #114
 Kill "c:\project\data\weighthi1.txt"
 Open "c:\project\data\weighthi1.txt" For Output As #112
 Do While Not EOF(114)
  Line Input #114, vup
  Print #112, vup
 Loop
 Close #114
 Close #112
 Kill "c:\project\data\upweighthi1.txt"
End Function

Private Sub BuildModel_Click()
Dim mem As Integer
Dim count As Integer
Dim vin1
Dim k
  vin = Val(Lag(0).Text)  'no of neuron in input layer
  vhi1 = Val(Hidden1.Text)
  vhi2 = Val(Hidden2.Text)
  vin1 = Val(Lag(0).Text)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mem = vhi1
k = 0

If loaddata.filename <> "" Then
    filename = loaddata.filename.Text
    GoTo continue
ElseIf loaddata.databasename <> "" Then
        filename = "c:\project\data\data.txt"
        GoTo continue
Else
    MsgBox ("Please Input youe data file")
End If

continue:
Call convert_data
If Hidden1.Text <> "0" Then
 Call writewhi1(vin, vhi1)
End If

If Hidden2.Text <> "0" Then
 Call writewhi2(vhi1, vhi2)
End If

If (Hidden1.Text <> "0") And (Hidden2.Text = "0") Then
Call writewout(vhi1)
End If

If Hidden2.Text <> "0" Then
Call writewout(vhi2)
End If

s_stop = False
StatusBar1.SimpleText = "Runing....."
status_test = False
Call train
Call display_train
status_test = True
Call test
Call display_test
StatusBar1.SimpleText = "Complete"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub display_test()
Dim count As Integer
Dim vin As Integer
Dim vhi1, j, i, p As Integer
Dim data(10000)
Dim Data1(10000)
Dim mem_value(1000)
Dim q(10000)
Dim mem
Dim mem1
Dim v
 StatusBar1.SimpleText = "Runing....."
  status_test = True
  j = 1
  Open "c:\project\data\test.txt" For Input As #13
  Do While Not EOF(13)
   Line Input #13, mem
   data(j) = mem
   j = j + 1
  Loop
  Close #13
  'TrainGrid.Row = 0
  count = 0
  TestGrid.Cols = 1       ' Two strings per row.
  For i = 1 To j ' Count from 1 to 100.
    TestGrid.AddItem data(i)
    count = count + 1
  Next i
  p = 1
  Open "c:\project\data\test_result.txt" For Input As #35
  Do While Not EOF(35)
   Line Input #35, mem1
   Data1(p) = mem1
   p = p + 1
  Loop
  Close #35
  vin = Val(Lag(0).Text)
  v = vin + 2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
TestGrid.Cols = 3
TestGrid.Row = v
TestGrid.Col = 1
For i = 1 To p ' Count from 1 to 100.
  TestGrid.Text = Data1(i)
  v = v + 1
  If v = count + 2 Then
   Exit For
  End If
  TestGrid.Row = v
Next i
StatusBar1.SimpleText = "Complete"
End Sub

Private Sub display_train()
Dim count As Integer
Dim vin As Integer
Dim vhi1, j, i, p As Integer
Dim data(10000)
Dim Data1(10000)
Dim mem_value(1000)
Dim q(10000)
Dim mem
Dim mem1
Dim v
Dim status_train As Boolean
 status_train = True
 j = 1
 Open "c:\project\data\train.txt" For Input As #1
 Do While Not EOF(1)
  Line Input #1, mem
  data(j) = mem
  j = j + 1
 Loop
 Close #1
 TrainGrid.Row = 0
 count = 0
 TrainGrid.Cols = 1      ' Two strings per row.
 For i = 1 To j ' Count from 1 to 100.
   TrainGrid.AddItem data(i)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  count = count + 1
Next i
p = 1
Open "c:\project\data\train_result.txt" For Input As #2

Do While Not EOF(2)
  Line Input #2, mem1
  Data1(p) = mem1
  p = p + 1
Loop
Close #2

vin = Val(Lag(0).Text)
v = vin + 2
TrainGrid.Cols = 2
TrainGrid.Row = v
TrainGrid.Col = 1

For i = 1 To p ' Count from 1 to 100.
   TrainGrid.Text = Data1(i)
   v = v + 1
   If v = count + 2 Then
     Exit For
   End If
   TrainGrid.Row = v
 Next i
End Sub

Private Sub test()
Dim mem, num As Integer
Dim count As Integer
Dim q(1200000)
Dim mem_value(1)
Dim vin1
Dim use As Boolean
Dim E_Error
Dim e1
Dim k
Dim i, p, q1 As Integer
Dim m_tar
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
StatusBar1.SimpleText = "Runing....."
 vin = Val(Lag(0).Text)  'no of neuron in input layer
 vhi1 = Val(Hidden1.Text)
 vhi2 = Val(Hidden2.Text)
 vin1 = Val(Lag(0).Text)
 mem = vhi1
 k = 0
 use = False
 e1 = 0
 E_Error = 0
      Open "c:\project\data\test.txt" For Input As #1
      count = 1
      k = 1
      Do While Not EOF(1)
       Line Input #1, p
       q(k) = p
       k = k + 1
      Loop
      Close #1
 Open "c:\project\data\test.txt" For Input As #1
 SSE = 0
 p = 0
 q1 = 0
 Do While Not EOF(1)
 vin = Val(Lag(0).Text)
 Open "c:\project\data\binary_input_test.txt" For Output As #22
 num = count + vin
 For i = count To num
    If i = num Then
     Exit For
    Else
     If q(i) = "" Then
      GoTo continue
     End If
     Print #22, q(i)
    End If
  Next i
     Close #22
     Open "c:\project\data\target_test.txt" For Output As #38
     Print #38, q(i)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Close #38
Open "c:\project\data\target_test.txt" For Input As #33
Line Input #33, m_tar
If m_tar = "" Then
  Close #33
  GoTo continue
End If
Close #33
 Call ForwardPropagate_test
 SSE = SSE + ((m_tar - ActualOutput) * (m_tar - ActualOutput))
 count = count + 1
Loop

continue:
 Close #1
End Sub

Public Sub CalculateOutput(NumInput As Integer, NumOutput1 As Integer, NumOutput2 As Integer)
   Dim i, j, c As Integer
   Dim net As Double
   Dim w
   Dim inputs
    Dim t As Boolean
   Dim w_hi1
   Dim w_out
   Dim w_hi2
   vin = Val(Lag(0).Text)  'no of neuron in input layer
   vhi1 = Val(Hidden1.Text)
   vhi2 = Val(Hidden2.Text)
   t = False

If NumInput <> 0 Then                                    ' For hidden layer output
      ReDim Opj(NumOutput1)
      ReDim Opk1(NumOutput1)
       c = 0
      Open "c:\project\data\weighthi1.txt" For Input As #17
      For j = 1 To NumOutput1
         net = 0
         Open "c:\project\data\binary_input.txt" For Input As #29
          For i = 1 To NumInput
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Line Input #17, w
            Line Input #29, inputs
            Opj(j) = w * inputs
            net = net + Opj(j)
        Next i
        Opk1(j) = 1 / (1 + Exp(-net))
        Close #29
    Next j
    Close #17
End If

If NumOutput2 <> 0 Then
    ReDim Opk1(NumOutput1)
    ReDim Opk2(NumOutput2)
    c = 0
     net = 0
    For j = 1 To NumOutput2    'Outputlayer' nodes =1
      Open "c:\project\data\weighthi2.txt" For Input As #39
      For i = 1 To NumOutput1
        Line Input #39, w_hi1
        Opk1(i) = (w_hi1 * Opj(i))
        net = net + Opk1(i)
      Next i
      Close #39
      Opk2(j) = 1 / (1 + Exp(-net))
      t = True
    Next j
  End If

  If t = True Then
    ReDim Opk3(NumOutput2)
    c = 0
     net = 0
    For j = 1 To 1    'Outputlayer' nodes =1
      Open "c:\project\data\weightout.txt" For Input As #39
      For i = 1 To NumOutput2
        Line Input #39, w_out
        net = net + (w_out * Opk2(i))
      Next i
      Close #39
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Opk3(j) = 1 / (1 + Exp(-net))
      ActualOutput = Opk3(j)
      Pre = Opk3(j)   'for testing predic
    Next j
  End If

  If t = False Then
    ReDim Opk3(NumOutput1)
    c = 0
    net = 0
    For j = 1 To 1    'Outputlayer' nodes =1
      Open "c:\project\data\weightout.txt" For Input As #39
      For i = 1 To NumOutput1
        Line Input #39, w_out
        net = net + (w_out * Opk1(i))
      Next i
      Close #39
      Opk3(j) = 1 / (1 + Exp(-net))
      ActualOutput = Opk3(j)
      Pre = Opk3(j)   'for testing predic
    Next j
  End If

  Open "c:\project\data\actout.txt" For Output As #33
  Print #33, ActualOutput
  Close #33
  Close #39
End Sub

Public Sub CalculateOutput_test(NumInput As Integer, NumOutput1 As Integer, NumOutput2 As Integer)
    Dim i, j, c As Integer
    Dim net As Double
    Dim w
    Dim inputs
    Dim t As Boolean
    Dim w_hi1
    Dim w_out
    Dim w_hi2
    vin = Val(Lag(0).Text)   'no of neuron in input layer
    vhi1 = Val(Hidden1.Text)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
vhi2 = Val(Hidden2.Text)
t = False

If NumInput <> 0 Then                                    ' For hidden layer output
    ReDim Opj(NumOutput1)
    ReDim Opk1(NumOutput1)
    c = 0
    Open "c:\project\data\weighthi1.txt" For Input As #17
    For j = 1 To NumOutput1
      net = 0
      Open "c:\project\data\binary_input_test.txt" For Input As #29
      For i = 1 To NumInput
        Line Input #17, w
        Line Input #29, inputs
        Opj(j) = w * inputs
        net = net + Opj(j)
      Next i
      Opk1(j) = 1 / (1 + Exp(-net))
      Close #29
    Next j
    Close #17
End If

If NumOutput2 <> 0 Then
    ReDim Opk1(NumOutput1)
    ReDim Opk2(NumOutput2)

    c = 0
    net = 0
    For j = 1 To NumOutput2    'Outputlayer' nodes =1
      Open "c:\project\data\weighthi2.txt" For Input As #39
      For i = 1 To NumOutput1
        Line Input #39, w_hi1
        Opk1(i) = (w_hi1 * Opj(i))
        net = net + Opk1(i)
      Next i
      Close #39
      Opk2(j) = 1 / (1 + Exp(-net))
      t = True
    Next j
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If

If t = True Then
  ReDim Opk3(NumOutput2)
  c = 0
   net = 0
  For j = 1 To 1    'Outputlayer' nodes =1
    Open "c:\project\data\weightout.txt" For Input As #39
    For i = 1 To NumOutput2
      Line Input #39, w_out
      net = net + (w_out * Opk2(i))
    Next i
    Close #39
    Opk3(j) = 1 / (1 + Exp(-net))
    ActualOutput = Opk3(j)
    Pre = Opk3(j)   'for testing predic
  Next j
End If

If t = False Then
  ReDim Opk3(NumOutput1)
  c = 0
   net = 0
  For j = 1 To 1    'Outputlayer' nodes =1
    Open "c:\project\data\weightout.txt" For Input As #39
    For i = 1 To NumOutput1
      Line Input #39, w_out
      net = net + (w_out * Opk1(i))
    Next i
    Close #39
    Opk3(j) = 1 / (1 + Exp(-net))
    ActualOutput = Opk3(j)
    Pre = Opk3(j)   'for testing predic
  Next j
End If

Open "c:\project\data\test_result.txt" For Append As #33
Print #33, ActualOutput
Close #33
Close #39
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub


Public Sub ForwardPropagate()
    Dim I As Integer
    Dim NumOutputs1 As Integer
    Dim NumOutputs2 As Integer
   NumInputs = Val(Lag(0).Text)
   NumOutputs1 = Val(Hidden1.Text)
   NumOutputs2 = Val(Hidden2.Text)
   'FirstLoop = True
   For I = 2 To 2
      Call CalculateOutput(NumInputs, NumOutputs1, NumOutputs2)
      NumInputs = Val(Lag(0).Text)
      NumOutputs1 = Val(Hidden1.Text)
      NumOutputs2 = Val(Hidden2.Text)
      'FirstLoop = False
    Next I
End Sub


Public Sub ForwardPropagate_test()
Dim NumOutputs1 As Integer
Dim NumOutputs2 As Integer
Dim NumInputs As Integer
   Dim I As Integer
    NumInputs = Val(Lag(0).Text)
    NumOutputs1 = Val(Hidden1.Text)
    NumOutputs2 = Val(Hidden2.Text)
    FirstLoop = True
    For I = 2 To 2
       Call CalculateOutput_test(NumInputs, NumOutputs1, NumOutputs2)
       NumInputs = Val(Lag(0).Text)
       NumOutputs1 = Val(Hidden1.Text)
       NumOutputs2 = Val(Hidden2.Text)
       FirstLoop = False
     Next I
End Sub


Public Sub Backwardpropagate()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Dim l As Integer
    Dim NumOutputs1 As Integer
    Dim NumOutputs2 As Integer
    NumInputs = Val(Lag(0).Text)
    NumOutputs1 = Val(Hidden1.Text)
    NumOutputs2 = Val(Hidden2.Text)
    FirstLoop = True
    'For l = 2 To 3
        Call CalculateError(NumInputs, NumOutputs1, NumOutputs2)
        NumInputs = Val(Lag(0).Text)
        NumOutputs1 = Val(Hidden1.Text)
        NumOutputs2 = Val(Hidden2.Text)
        FirstLoop = False
    'Next l
End Sub

Public Sub update(NumInput As Integer, NumOutput1 As Integer, NumOutput2 As Integer)
    Dim i, j, c As Integer
    Dim weighthi1
    Dim inputs
    Dim weightout
    Dim new_weight
    Dim we
    Dim weighthi2
    LearningRate = LEARNNING_RATE
    If NumInput <> 0 Then
        c = 0
        Open "c:\project\data\weighthi1.txt" For Input As #15
        Open "c:\project\data\upweighthi1.txt" For Output As #3
        For j = 1 To NumOutput1
            Open "c:\project\data\binary_input.txt" For Input As #2
            For i = 1 To NumInput
                Line Input #15, weighthi1
                Line Input #2, inputs
                new_weight = weighthi1 + (LearningRate * deltahi1(j) * inputs)
                Print #3, new_weight

            Next i
            Close #2
        Next j
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Close #15
  Close #3
  Call re_whi1
 End If

If NumOutput2 <> 0 Then
  c = 0
 Open "c:\project\data\weighthi2.txt" For Input As #15
 Open "c:\project\data\upweighthi2.txt" For Output As #3
 For j = 1 To NumOutput2
      For i = 1 To NumOutput1
       Line Input #15, weighthi2
       new_weight = weighthi2 + (LearningRate * Epj(j) * Opk1(i))
       Print #3, new_weight
     Next i
    Close #2
  Next j
  Close #15
  Close #3
  Call re_whi2
    Open "c:\project\data\weightout.txt" For Input As #3
    Open "c:\project\data\upweightout.txt" For Output As #4
    For i = 1 To NumOutput2
       Line Input #3, weightout
       we = weightout + (LearningRate * Epk(1) * Opk2(i))
       Print #4, we
     Next i
     Close #4
     Close #3
     Call re_wout
  Else
     Open "c:\project\data\weightout.txt" For Input As #3
     Open "c:\project\data\upweightout.txt" For Output As #4
     For i = 1 To NumOutput1
       Line Input #3, weightout
       we = weightout + (LearningRate * Epk(1) * Opk1(i))
       Print #4, we
     Next i
     Close #4
     Close #3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Call re_wout
    End If
    Close #3
    Close #4
End Sub

Public Sub UpdateWeights()
    Dim l As Integer
    Dim NumOutputs1 As Integer
    Dim NumOutputs2 As Integer
    NumInputs = Val(Lag(0).Text)
    NumOutputs1 = Val(Hidden1.Text)
    NumOutputs2 = Val(Hidden2.Text)
    FirstLoop = True
    For l = 2 To 2
        Call update(NumInputs, NumOutputs1, NumOutputs2)
        FirstLoop = False
        NumInputs = Val(Lag(0).Text)
        NumOutputs1 = Val(Hidden1.Text)
        NumOutputs2 = Val(Hidden2.Text)
    Next l
End Sub

Public Sub CalculateError(NumInput As Integer, NumOutput1 As Integer, NumOutput2 As Integer)
    Dim i, j, c As Integer
    Dim target
    Dim weightout
    Dim f As Boolean
    Dim wdelout
    Dim weighthi2
    Dim weighthi1
    f = False
    If NumInput <> 0 Then 'Process of calculate output error in output layer
     Open "c:\project\data\target.txt" For Input As #16
     Line Input #16, target
       ReDim Epk(1)
       ReDim Opk3(1)
       For i = 1 To 1
                Epk(1) = (target - ActualOutput) * (ActualOutput * (1 - ActualOutput))
       Next i
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Close #16
End If

If NumOutput2 <> 0 Then 'Process of calculate output error in output layer
ReDim Epj(NumOutput2)
    Open "c:\project\data\weightout.txt" For Input As #2
  For j = 1 To NumOutput2
    Line Input #2, weightout
    Epj(j) = (Opk2(j) * (1 - Opk2(j)) * (Epk(1) * weightout))
  Next j
  Close #2
  f = True
Close #2
End If

If (NumOutput1 <> 0) And (f = False) Then 'Process of calculate output error in output layer
  Open "c:\project\data\weightout.txt" For Input As #2
  For j = 1 To NumOutput1
    Line Input #2, weightout
    deltahi1(j) = (Opk1(j) * (1 - Opk1(j)) * (Epk(1) * weightout))
  Next j
  Close #2
End If

If (NumOutput1 <> 0) And (f = True) Then 'Process of calculate output error in output layer
  Open "c:\project\data\weighthi2.txt" For Input As #28
  For i = 1 To NumOutput1
   For j = 1 To NumOutput2
    Line Input #28, weighthi2
    wdelout = wdelout + (weighthi2 * Epk(1))
  Next j
  deltahi1(i) = wdelout * (Opk1(i) * (1 - Opk1(i)))
  Next i
 End If
 Close #28
 Close #2
End Sub

Private Sub train()
Dim mem As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim count As Integer
Dim q(100000)
Dim mem_value(1)
Dim vin1
Dim t As Integer
Dim e1
Dim E_Error
Dim use As Boolean
Dim p
Dim i
Dim m_tar
Dim target
Dim SSE As Double
Dim NumOutputs1
Dim NumOutputs2
Dim stop_p As Boolean
 vin = Val(Lag(0).Text)  'no of neuron in input layer
 vhi1 = Val(Hidden1.Text)
 vhi2 = Val(Hidden2.Text)
 LEARNNING_RATE = Val(LearningRate.Text)
 ErrorLimit = Val(ErrorAccept.Text)
 NumInputs = vin
 NumOutputs1 = vhi1
 NumOutputs2 = vhi2
 vin1 = Val(Lag(0).Text)
 mem = vhi1
  SSE = 0
 use = False
 e1 = 0
 E_Error = 0
 vin = Val(Lag(0).Text)  'no of neuron in input layer
 stop_p = False

 Do
      StatusBar1.SimpleText = "Runing....."
      Open "c:\project\data\train.txt" For Input As #1
      count = 1
       t = 0
       Do While Not EOF(1)
        Line Input #1, p
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        t = t + 1
        q(t) = p
       Loop
      Close #1

Open "c:\project\data\train.txt" For Input As #1
Do While Not EOF(1) And (stop_p = False)

Open "c:\project\data\binary_input.txt" For Output As #2
num = count + vin
For i = count To num
    If i = num Then
     Exit For
    Else
     If q(i) = "" Then
      GoTo continue
     End If
     Print #2, q(i)
    End If
Next i

    Close #2
    If q(i) = "" Then
    Close #1
    Close #2
    GoTo continue
   Else
    Open "c:\project\data\target.txt" For Output As #38
    Print #38, q(i)
    End If
   Close #38

Open "c:\project\data\target.txt" For Input As #33
Line Input #33, m_tar
If m_tar = "" Then
  Close #33
  GoTo continue
End If
Close #33
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Call ForwardPropagate

Open "c:\project\data\train_result.txt" For Append As #26
Print #26, ActualOutput
Close #26

Open "c:\project\data\target.txt" For Input As #34
Line Input #34, target
SSE = SSE + ((target - ActualOutput) * (target - ActualOutput)) / 2
Close #34

Call Backwardpropagate
Call UpdateWeights
count = count + 1
Loop                                                    'until end of file
continue:
Close #1

e1 = Val(SSE) '/ num_data
error.Text = e1
DoEvents
E_Error = Val(SSE) '/ num_data
SSE = 0
use = True

If (e1 > ErrorLimit) Then
    Kill "c:\project\data\train_result.txt"
End If

Loop Until (E_Error <= ErrorLimit) 'And (stop_p = True) 'And stop_p = True
Exit Sub
End Sub

Private Sub cal_error_Click()
Dim count As Integer
Dim n As Integer
Dim vhi1 As Integer
Dim data(10000)
Dim Data1(10000)
Dim mem_value(1000)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim q(10000)
Dim mem
Dim mem1
Dim v
Dim i
Dim r
Dim mean, x, y, total As Double
Dim vin
Dim m
Dim val1, val2
  vin = Val(Lag(0).Text) + 2
   i = 0
  TestGrid.Cols = 3

  Open "c:\project\data\test_result.txt" For Input As #35
  Open "c:\project\data\error_test.txt" For Output As #36
  Do While Not EOF(35)
  Line Input #35, m
  r = vin + i
  If r = 435 Then
    Exit Do
  End If
  TestGrid.Col = 0
  TestGrid.Row = r
  val1 = TestGrid.Text
  TestGrid.Col = 1
  TestGrid.Row = r
  val2 = TestGrid.Text
  TestGrid.Col = 2
  TestGrid.Row = r
  TestGrid.Text = (val1 - val2) ^ 2
  Print #36, (val1 - val2) ^ 2
  i = i + 1
  Loop
  Close #35
  Close #36
  total = 0
  n = 0
  Open "c:\project\data\error_test.txt" For Input As #36
  Do While Not EOF(36)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Line Input #36, x
 total = total + x
 n = n + 1
Loop
mean = total / n
Close #36
Open "c:\project\data\error_test.txt" For Input As #37
Do While Not EOF(37)
 Line Input #37, y
 v = ((y - mean) ^ 2) / n
Loop
Close #37
mean_error.Text = mean
sd_error.Text = v
End Sub

Private Sub Command1_Click()

 Call test
 status_test = False
 Call display_train
 status_test = True
 Call display_test
 Command1.Visible = False
 End Sub

Private Sub Command2_Click()
 TrainGraph.Show
End Sub

Private Sub Command3_Click()
  TestGraph.Show
End Sub

Private Sub Exit_Click()
 Hide
End Sub

Private Sub ShowGraphic_Click()
 'Graphic.Show
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub Form_Load()
StatusBar1.SimpleText = "Ready"
End Sub

Private Sub Stop_Click()
Dim stop_p As Boolean
 stop_p = True
      Close #24
      MsgBox ("Build Neuron Network Not Complete")
Call display_train
status_test = True
Call test
Call display_test
End Sub

Private Sub CommandButton1_Click()
Dim numrnd, randweight As Double
 Randomize
 numrnd = (2 * Rnd) - 1
 randweight = numrnd
 MsgBox (randweight)

End Sub
Private Sub Command3_Click()
 Helpform.Show
End Sub

Private Sub Command4_Click()
 Hide
End Sub

Private Sub Command5_Click()
 Hide
 CreateModel.Show
End Sub

Private Sub databasebtn_Click(Index As Integer)
 Dim Col1, Col2 As Column
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim db_neural As Database
Dim rec As Recordset
 type_mem = 2
 CommonDialog2.Filter = "Database| *.mdb"
 CommonDialog2.FilterIndex = 2
 CommonDialog2.ShowOpen
 databasename = CommonDialog2.filename
 ConvertDatabase.Show
 loaddata.Hide
End Sub

Private Sub textbtn_Click(Index As Integer)
 Dim Col1, Col2 As Column
 type_mem = 1
 CommonDialog1.Filter = "Text Files|*.txt"
 CommonDialog1.FilterIndex = 2
 CommonDialog1.ShowOpen
 filename = CommonDialog1.filename
 Const ole_crete_link = 1
 End Sub

Private Sub Graph_Click()
 'PrepareGraph.Show
 ShowGraph.Show
End Sub

Function convert_data()
Dim mem
Dim data, Data1, data2
   If filename_test <> "" Then
    mem = filename_test.Text
 End If
 If databasename_test <> "" Then
    mem = "c:\project\data\data.txt"
 End If
 If Save1 = True Then
    mem = "c:\project\data\userinput1.txt"
 End If
 Open mem For Input As #1
 data_Max = 0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Do While Not EOF(1)
 If EOF(1) Then
Exit Do
End If
Line Input #1, data
 If data > data_Max Then
 data_Max = Val(data)
Else
 data_Max = data_Max
 End If
 Loop

Max = data_Max + (data_Max * 0.2)
data_Max = data_Max + (data_Max * 0.2)

Open mem For Input As #2
Open "c:\project\data\predict.txt" For Output As #36
Do While Not EOF(2)
If EOF(2) Then
Exit Do
End If
Line Input #2, Data1
data2 = Data1 / data_Max
Print #36, data2
Loop
Close #2
Close #36
Close #1
End Function

Private Sub Save_Click()
Dim i
grdTranslate.Row = 1
grdTranslate.Col = 1
Save1 = True
 i = 1
 Open "c:\project\data\userinput1.txt" For Output As #3
 For i = 2 To grdTranslate.Rows
    Print #3, grdTranslate.Text
    grdTranslate.Row = i
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 Next i
 MsgBox ("Save Complete")
 Close #3
End Sub

Private Sub grdTranslate_KeyDown(KeyCode As Integer, Shift As Integer)
  If KeyCode = 13 Then 'กด Ins
     grdTranslate.Rows = grdTranslate.Rows + 1
     'grdTranslate.Col = 0
     'grdTranslate.Text = grdTranslate.Rows + 1
   ElseIf KeyCode = 46 Then 'กดDel
     If grdTranslate.Row > 1 Then
        grdTranslate.RemoveItem (grdTranslate.Row)
      End If
   End If
End Sub

Option Explicit
Option Base 1

Private Sub Command2_Click()
 Hide
End Sub

Private Sub Command3_Click()
Dim app As Object
Dim s1, s, x As Object
  Set s1 = CreateObject("Excel.Application")
  Set s1 = CreateObject("Excel.Sheet")
   s1.Cells(1, 1).Value = Format(Now, "mm:ss")
   s1.SaveAs "c:\project\data\graph.xls"
   s1.Parent.Parent.Visible = True

  Set x = CreateObject("Excel.Chart")
    x.Parent.Parent.Visible = True
  Set s = GetObject("c:\project\data\test.xls", "Excel.Chart")
  End Sub

  Private Sub CommandButton1_Click()
  Dim NumberEntries, i, j As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim a(10000), b(10000)
Dim MaxY%
Dim data
Dim loop_index As Integer
i = 0
Open "c:\project\data\predict.txt" For Input As #1
Do While Not EOF(1)
 Line Input #1, data
 i = i + 1
 a(i) = data
 If i <> Val(Prediction.lag.Text) + 1 Then
  b(i) = data
 End If
Loop
Close #1
  NumberEntries = i

j = Val(Prediction.lag.Text) + 0
Open "c:\project\data\predict_result.txt" For Input As #2
Do While Not EOF(2)
 Line Input #2, data
 j = j + 1
 b(j) = data
Loop
Close #2
  Line (0.05 * ScaleWidth, 0)-(0.05 * ScaleWidth, 0.9 * ScaleHeight)
  Line (0.05 * ScaleWidth, 0.9 * ScaleHeight)-(0.95 * ScaleWidth, 0.9 * ScaleHeight)
  FontSize = 20
  FontName = "Times New Roman"
  CurrentX = 0.25 * ScaleWidth
  CurrentY = 0.9 * ScaleHeight
  MaxY% = 0

  For loop_index = 1 To NumberEntries
   If a(loop_index) > MaxY% Then
    MaxY% = a(loop_index)
   End If
  Next loop_index

  Scale (0, 1.1 * MaxY%)-(1.1 * NumberEntries, 0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
For loop_index = 1 To NumberEntries
  Line (loop_index, MaxY%)-(loop_index, MaxY%), RGB(255, 0, 0)
Next loop_index
CurrentX = 1
CurrentY = a(1)

For loop_index = 1 To NumberEntries
  Line -(loop_index, a(loop_index)), RGB(255, 0, 0)
Next loop_index
CurrentX = 1
CurrentY = b(1)
For loop_index = 1 To NumberEntries
  Line -(loop_index, b(loop_index)), RGB(0, 255, 0)
Next loop_index
End Sub
Private Sub Form_Load()
Call CommandButton1_Click
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน	นางสาวกัลยาณี ทองเจริญสกุล
วันเดือนปีเกิด	17 ธันวาคม 2518
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษาระดับปริญญาตรี	วท.บ.วิทยาการคอมพิวเตอร์
สถานที่สำเร็จการศึกษา	มหาวิทยาลัยเกษตรศาสตร์
ปีที่สำเร็จการศึกษา	2540
ประวัติการทำงาน	2540 ถึง ปัจจุบัน เป็นนักวิเคราะห์ระบบ บริษัทปูนซิเมนต์ไทย จำกัด

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้