ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การพัฒนาระบบพยากรณ์ราคาหลักทรัพย์โดยใช้ Recurrent Neural Network

Stock Price Forecasting System with Recurrent Neural Network

โดย

ชัชรี ปิ่นแก้ว

รหัส 43067065

H002003

อาจารย์ที่ปรึกษา

ผศ.ดร.วรพจน์ กรีสุระเดช

วัน เดือน ปี 21 ม.ค. 2550
เลขทะเบียน 02003
เลขเรียกหนังสือ วิทย. ช351ก 2545
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2545
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	การพัฒนาระบบพยากรณ์ราคาหลักทรัพย์โดยใช้ Recurrent Neural Network
นักศึกษา	นางสาว ชัชรี ปิ่นแก้ว
อาจารย์ที่ปรึกษา	ผศ.ดร. วรพจน์ กรีสุระเดช
ระดับการศึกษา	วิทยาศาตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2545

บทคัดย่อ

ในปัจจุบันมีการนำเทคนิคของดาต้าไมนิ่งมาใช้ศึกษาวิจัยและพัฒนาเพื่อวิเคราะห์ข้อมูลในงานด้านต่างๆ อย่างแพร่หลายมากขึ้น ซึ่งข้อมูลในตลาดหลักทรัพย์ก็เป็นอีกข้อมูลหนึ่งที่ได้รับความสนใจที่จะนำเทคนิคนี้มาประยุกต์ใช้เพื่อช่วยในการวิเคราะห์ และหาข้อมูลที่สามารถใช้เป็นแนวทางตัดสินใจเพื่อการลงทุนที่มีประสิทธิภาพมากขึ้น

สำหรับโครงการนี้เป็นการประยุกต์ใช้ดาต้าไมนิ่งในการพัฒนาระบบพยากรณ์ราคาหลักทรัพย์ เพื่อหาข้อมูลประกอบการวิเคราะห์การลงทุนในตลาดหลักทรัพย์ โดยในการศึกษาจะเป็นการนำข้อมูลราคาหลักทรัพย์ในอดีตจำนวนหนึ่งมาพิจารณาในลักษณะของอนุกรมเวลา (Time Series) และใช้หลักการของโครงข่ายประสาทเทียม หรือ นิวรอลเน็ตเวิร์ก (Neural Network) แบบป้อนกลับอย่างง่าย (Simple Recurrent Neural Network: Elman Network) ในการสร้างแบบจำลองเพื่อใช้พยากรณ์ราคาหลักทรัพย์ที่จะเกิดขึ้นในอนาคตได้อย่างน่าเชื่อถือ ซึ่งผลลัพธ์ที่ได้นี้สามารถนำมาวิเคราะห์และพิจารณาความเหมาะสมร่วมกับองค์ประกอบอื่นๆ เพื่อนำไปใช้ให้เกิดประโยชน์ในการลงทุนได้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Stock Price Forecasting System with Recurrent Neural Network
Student	Miss Chatcharee Pinkaew
Advisor	Asst. Prof. Dr. Worapoj Kreesuradej
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2002

ABSTRACT

At this time, Data mining is the very useful technique of various applications for knowledge discovery. Information in stock market is the one that has been analyzed with this technique to find the appropriate model for effective investment.

In this project, the system development for stock price forecasting is introduced by using data mining concept with neural network technique to generate model for further stock price prediction in the stock market. Within this application, the historical data of stock price was gathered in time-series and analyzed with simple recurrent neural network model (or "elman" network). The result from this model will be used with another factor for properly decision to get useful information for stock market investment.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ในการพัฒนาระบบงานพยากรณ์ราคาหลักทรัพย์นี้ ผู้จัดทำขอขอบพระคุณ ผศ. ดร. วรพจน์ กรีสุระเดช ที่ได้กรุณาให้แนวทางการศึกษา และให้คำปรึกษาตลอดช่วงการดำเนินงาน จนสามารถพัฒนาโครงการได้สำเร็จลุล่วงด้วยดี

ขอขอบคุณ ทุกๆ กำลังใจจากบุคคลในครอบครัวที่เป็นส่วนหนึ่งที่ช่วยให้สามารถแก้ไขปัญหา และผ่านอุปสรรคต่างๆ ได้

สุดท้ายขอขอบคุณเพื่อนๆ IS 9.2 ที่ให้คำแนะนำในการศึกษาเป็นอย่างดี

ชัชรี ปิ่นแก้ว

20 กุมภาพันธ์ 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	VI
สารบัญตาราง	VIII
บทที่	
1. บทนำ	1
1.1 ความเป็นมา	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตการดำเนินงาน	3
1.4 ขั้นตอนและวิธีการดำเนินงาน	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ	4
2. ทฤษฎีเบื้องต้นของคาต้าไมนิ่ง (Data Mining)	5
2.1 หลักการทั่วไปของคาต้าไมนิ่ง	5
2.2 ประเภทของการใช้งานคาต้าไมนิ่ง	9
2.2.1 งานด้านการพยากรณ์ (Predictive Data Mining)	10
2.2.2 งานด้านการแบ่งกลุ่ม (Descriptive Data Mining)	10
2.3 การประยุกต์ใช้คาต้าไมนิ่งกับงานด้านการพยากรณ์ราคาหลักทรัพย์	11
2.3.1 การวิเคราะห์ข้อมูลในอนุกรมเวลา (Time series analysis)	12
2.3.2 การประมวลผลข้อมูลเบื้องต้น (Data Preprocessing)	13
3. หลักการของนิวรอลเน็ตเวิร์ก (Neural Network)	15
3.1 หลักการทำงานของนิวรอลเน็ตเวิร์ก	16
3.2 โครงสร้างพื้นฐานของนิวรอลเน็ตเวิร์ก	17
3.3 ประเภทของนิวรอลเน็ตเวิร์ก	25
3.4 การประยุกต์ใช้นิวรอลเน็ตเวิร์กแบบ Elman ในการพัฒนาระบบ	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การพัฒนาระบบพยากรณ์ราคาหลักทรัพย์ 31
4.1 องค์ประกอบของระบบพยากรณ์ราคาหลักทรัพย์ 31
4.1.1 การสร้างแบบจำลองใหม่ (New Model) 31
4.1.2 การใช้งานแบบจำลอง (Load Model) 32
4.2 อัลกอริทึมในระบบพยากรณ์ราคาหลักทรัพย์ 33
4.3 การใช้งานระบบพยากรณ์ราคาหลักทรัพย์ 36
5. การทดลองและสรุปผล 44
5.1 การทดลอง 44
5.2 สรุปผลการศึกษา 46

บรรณานุกรม 47
ภาคผนวก 50
ประวัติผู้เขียน 63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ภาพที่	หน้า
2.1 แสดงขั้นตอนต่างๆ ของกระบวนการค้นหาลักษณะแฝงในข้อมูล	6
2.2 แสดงตัวอย่างของนิวรอลเน็ตเวิร์ก (Neural Network)	8
2.3 แสดงตัวอย่างของโครงสร้างต้นไม้ (Decision Tree)	8
2.4 แสดงประเภทของงานที่มีการนำเทคนิคต่างๆ ของดาต้าไมนิ่งมาใช้	9
2.5 แสดงตัวอย่างผลที่ได้จากแบบจำลองด้านการพยากรณ์	10
2.6 แสดงตัวอย่างผลที่ได้จากแบบจำลองด้านการแบ่งกลุ่ม	11
3.1 แสดงโครงสร้างเซลล์ประสาทของมนุษย์	16
3.2 แสดงลักษณะโครงสร้างพื้นฐานของแบบจำลองนิวรอลเน็ตเวิร์ก	17
3.3 แสดงแบบจำลองของการคำนวณของโหนดภายในโครงข่ายนิวรอลเน็ตเวิร์ก	19
3.4 แสดงรูปแบบของฟังก์ชันกระตุ้นที่นำมาใช้ในการคำนวณค่าในโหนดของเน็ตเวิร์ก	19
3.5 แสดงรูปแบบของค่าความผิดพลาดแบบ Prediction Of Charge In Direction	23
3.6 แสดงโครงสร้างของนิวรอลเน็ตเวิร์กแบบส่งสัญญาณไปข้างหน้า	26
3.7 แสดงโครงสร้างของนิวรอลเน็ตเวิร์กแบบป้อนกลับอย่างง่าย - Jordan Network	27
3.8 แสดงโครงสร้างของนิวรอลเน็ตเวิร์กแบบป้อนกลับอย่างง่าย - Elman Network	27
3.9 แสดงโครงสร้างของนิวรอลเน็ตเวิร์กแบบป้อนกลับอย่างสมบูรณ์- Hopfield Network	28
4.1 แสดงหน้าจอหลักของการรับข้อมูลเพื่อใช้สร้างแบบจำลอง	37
4.2 แสดงขั้นตอนการเลือกไฟล์ข้อมูลเพื่อใช้สร้างแบบจำลอง	37
4.3 การแสดงผลเมื่อตรวจสอบพบข้อมูลที่ไม่สมบูรณ์	38
4.4 แสดงขั้นตอนการประมวลผลข้อมูลเบื้องต้น	38
4.5 แสดงตัวอย่างขั้นตอนการรับข้อมูลเพื่อคำนวณค่าเฉลี่ยสำหรับแทนค่าข้อมูล	39
4.6 แสดงรายละเอียดของตัวอย่างข้อมูลที่ได้หลังการประมวลผลเบื้องต้น	39
4.7 แสดงตัวอย่างผลจากการสร้าง และการทดสอบแบบจำลอง	40
4.8 แสดงตัวอย่างผลการทดสอบแบบจำลองในรูปแบบรายงาน	40
4.9 แสดงตัวอย่างผลการทดสอบแบบจำลองในรูปแบบกราฟ	41
4.10 แสดงขั้นตอนการเลือกไฟล์แบบจำลองเพื่อใช้งาน	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.11 แสดงรายละเอียดของตัวอย่างแบบจำลองที่เลือกมาใช้งาน 42
4.12 แสดงตัวอย่างการป้อนค่าข้อมูลอินพุตเข้าสู่ระบบ 42
4.13 แสดงการรายงานผลราคาหลักทรัพย์ของช่วงเวลาถัดไปที่คำนวณได้ 43
4.14 แสดงตัวอย่างไฟล์ที่เก็บรายละเอียดของแบบจำลอง 43
5.1 แสดงผลที่ได้จากการเรียนรู้โดยใช้จำนวนอินพุต = 2 และจำนวนโหนดในชั้นกลาง = 7 45
5.2 แสดงผลการทดสอบจากการเรียนรู้โดยจำนวนอินพุต=2 และจำนวนโหนดในชั้นกลาง = 7 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
5.1 แสดงผลที่ได้จากการสร้าง และทดสอบแบบจำลองโดยใช้ 2 อินพุต	44
5.2 แสดงผลที่ได้จากการสร้าง และทดสอบแบบจำลองโดยใช้ 3 อินพุต	44
5.3 แสดงผลที่ได้จากการสร้าง และทดสอบแบบจำลองโดยใช้ 5 อินพุต	45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมา

ตลาดหลักทรัพย์ (Stock Market) เป็นแหล่งทุนที่มีผู้สนใจเข้าไปลงทุนเป็นจำนวนมาก เนื่องจากผลตอบแทนในการลงทุนค่อนข้างสูง ดังนั้นจึงมีความต้องการที่จะนำวิธีการที่มีประสิทธิภาพเข้ามาใช้ในการวิเคราะห์ข้อมูลต่างๆ ที่เป็นตัวแปรสำคัญเพื่อช่วยในการตัดสินใจลงทุน ซึ่งการพยากรณ์ข้อมูลต่างๆในตลาดหลักทรัพย์นั้นก็นับว่าเป็นงานด้านหนึ่งที่ได้รับความสนใจนำมาศึกษาค่อนข้างมากในปัจจุบัน เช่น การพยากรณ์ราคาของหลักทรัพย์ที่มีการซื้อขายในตลาดหลักทรัพย์

สำหรับการพยากรณ์ข้อมูล หรือตัวแปรใดๆ นั้นควรมีการพิจารณาปัจจัยที่เกี่ยวข้องประกอบกันหลายๆ ด้าน โดยเฉพาะอย่างยิ่งในการตัดสินใจลงทุนในหลักทรัพย์นั้นควรจะพิจารณาทั้งปัจจัยหรือตัวชี้วัดทางด้านเทคนิค (Technical) และ ปัจจัยพื้นฐาน (Fundamental) เพื่อบ่งชี้ถึงสัญญาณต่างๆ ที่สามารถนำมาช่วยในการวิเคราะห์การลงทุนได้อย่างมีประสิทธิภาพ ซึ่งตัวอย่างของปัจจัยพื้นฐานที่มีความสัมพันธ์กับราคาหลักทรัพย์นั้น ได้แก่ ปริมาณการซื้อขายหลักทรัพย์ ราคาทองคำ ดัชนีทางด้านการผลิตและอุตสาหกรรม และอัตราแลกเปลี่ยนเงินตราต่างประเทศ เป็นต้น ส่วนการวิเคราะห์โดยใช้ปัจจัยทางเทคนิคนั้นจะใช้ข้อมูลของตัวแปรที่นำมาพยากรณ์ เช่น ราคาหลักทรัพย์ ย้อนหลังในลักษณะของอนุกรมเวลา (Time series) มาเป็นข้อมูลอินพุต แล้วหาแบบจำลองที่เหมาะสมเพื่อใช้ในการพยากรณ์ราคาหลักทรัพย์ที่เป็นไปได้ในอนาคต แต่เนื่องจากราคาหลักทรัพย์นั้นเป็นข้อมูลหนึ่งที่มีความสำคัญมาก และการเปลี่ยนแปลงของราคาหลักทรัพย์ในตลาดหลักทรัพย์นั้นจะได้รับผลกระทบจากหลายๆ ปัจจัย ทั้งปัจจัยทางด้านเศรษฐกิจ การเมือง หรือแม้แต่ปัจจัยในเชิงจิตวิทยา โดยปัจจัยทั้งหมดเหล่านี้มีความสัมพันธ์ซึ่งกันและกันอย่างสลับซับซ้อน ดังนั้นการพยากรณ์ลักษณะความเปลี่ยนแปลงของราคาหลักทรัพย์จึงเป็นงานที่ได้รับความสนใจในการทำวิจัย และมีการพัฒนาอัลกอริทึมที่ใช้สร้างแบบจำลองมาอย่างต่อเนื่องตลอดในช่วงเวลาหลายปีที่ผ่านมาเพื่อหาแบบจำลองที่เหมาะสมที่สุด โดยในปัจจุบันผู้ลงทุนไม่ได้พิจารณาข้อมูลโดยใช้เพียงเทคนิคใดเทคนิคหนึ่งเท่านั้น แต่จะใช้เทคนิคที่เหมาะสมหลายๆ วิธีมาใช้เพื่อสร้างผลตอบแทนจากตลาดหลักทรัพย์ให้ได้มากที่สุด

ปัจจุบันดาต้าไมนิ่ง (Data Mining) ก็นับว่าเป็นอีกหลักการหนึ่งที่มีการนำมาประยุกต์ใช้กับการวิเคราะห์ข้อมูลราคาหลักทรัพย์ เพื่อหาความสัมพันธ์ของข้อมูลต่างๆ ของหลักทรัพย์ และสามารถนำไปใช้ประกอบการพิจารณาหาแนวโน้มเพื่อการลงทุนที่เหมาะสมต่อไปได้ ตัวอย่างเช่น

เอกสารนี้มีลิขสิทธิ์ ห้ามมิให้ทำซ้ำ ดัดแปลง เผยแพร่ หรือนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การหาช่วงเวลาที่เหมาะสมสำหรับการลงทุนในหลักทรัพย์ต่างๆ การพิจารณาเลือกหลักทรัพย์ในการลงทุน หรือการพยากรณ์แนวโน้มของข้อมูลเพื่อใช้การลงทุน เป็นต้น ซึ่งในโครงการพัฒนาระบบงานนี้จะนำเสนอการใช้ดาต้าไมนิ่งเพื่อพัฒนาระบบวิเคราะห์ข้อมูลราคาหลักทรัพย์ในลักษณะงานด้านการพยากรณ์ข้อมูล (Predictive Modeling) โดยเป็นการพัฒนาระบบเพื่อใช้พยากรณ์ราคาหลักทรัพย์ของช่วงเวลาถัดไปในอนุกรมเวลา

สำหรับแบบจำลองที่ใช้ในการพยากรณ์ราคาหลักทรัพย์นั้น ได้เริ่มมีการพัฒนาจากการใช้วิธีการทางสถิติในการพยากรณ์ แต่ยังมีข้อจำกัดกับข้อมูลที่มีลักษณะที่ไม่เป็นเชิงเส้น (Non-linear Data) ต่อมาจึงได้มีการนำอัลกอริทึมในลักษณะของโครงข่ายประสาทเทียม หรือ นิวรอลเน็ตเวิร์ก (Neural Network) มาใช้ในงานด้านนี้ ซึ่งหลังจากการนำไปประยุกต์ใช้ และมีการพัฒนาอย่างต่อเนื่อง ทำให้ได้รับการยอมรับว่าเป็นอัลกอริทึมที่สามารถเรียนรู้ได้ทั้งการวิเคราะห์ข้อมูลที่เป็นปัจจัยเชิงเทคนิคและปัจจัยพื้นฐาน ดังนั้นจึงมีความเหมาะสมที่จะนำมาใช้ในสร้างระบบสำหรับการวิเคราะห์ข้อมูลของราคาหลักทรัพย์ในตลาดหลักทรัพย์ สำหรับโครงการพัฒนาระบบงานนี้จะใช้อัลกอริทึมของนิวรอลเน็ตเวิร์กแบบป้อนกลับ (Recurrent Neural Network)ในการสร้างแบบจำลองเพื่อพยากรณ์ราคาหลักทรัพย์รายวันของหลักทรัพย์ในตลาดหลักทรัพย์แห่งประเทศไทย โดยในการทำงานนั้นระบบจะใช้ข้อมูลในช่วงเวลาก่อนหน้าจำนวนหนึ่งเป็นข้อมูลอินพุต (Input) และใช้ข้อมูลของช่วงเวลาถัดไปในอนุกรมเวลาเป็นเอาต์พุต (Output) ที่ได้จากระบบ แล้วทำการสร้างแบบจำลองขึ้น ซึ่งหลังจากได้แบบจำลองที่เหมาะสมแล้วก็จะสามารถนำระบบไปใช้ในการพยากรณ์ราคาหลักทรัพย์ของช่วงเวลาต่อไปในอนาคตได้

1.2 วัตถุประสงค์

1.2.1 เพื่อศึกษาหลักการเบื้องต้นของทฤษฎีดาต้าไมนิ่งและการนำมาประยุกต์ใช้ในการพัฒนาระบบงานด้านการพยากรณ์ข้อมูล (Predictive Modeling)

1.2.2 เพื่อศึกษาทฤษฎี และการประยุกต์ใช้อัลกอริทึมของนิวรอลเน็ตเวิร์กแบบป้อนกลับ (Recurrent Neural Network) ในการวิเคราะห์ข้อมูลในลักษณะที่เป็นอนุกรมเวลา (Time Series)

1.2.3 เพื่อศึกษาวิธีการ และขั้นตอนของการพัฒนาระบบงานที่มีประสิทธิภาพในการสร้างระบบเพื่อการพยากรณ์ราคาหลักทรัพย์

1.2.4 เพื่อใช้เป็นพื้นฐานในการศึกษาการประยุกต์ใช้ทฤษฎีดาต้าไมนิ่งสำหรับงานในด้านอื่นๆ ได้อย่างมีประสิทธิภาพต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตการดำเนินงาน

โครงการนี้เป็นการศึกษาและพัฒนาระบบที่ใช้ในการพยากรณ์ราคาหลักทรัพย์โดยมีขอบเขตการศึกษาดังต่อไปนี้

1.3.1 เป็นการพัฒนาระบบตามหลักการของดาต้าไมนิ่งโดยใช้อัลกอริทึมแบบป้อนกลับอย่างง่าย (Partial Recurrent Neural Network) ในการสร้างแบบจำลองของระบบเพื่อใช้พยากรณ์ราคาหลักทรัพย์

1.3.2 ข้อมูลที่นำมาใช้ในการวิเคราะห์เป็นข้อมูลราคาหลักทรัพย์จากตลาดหลักทรัพย์แห่งประเทศไทยในช่วงเดือนมกราคม พ.ศ. 2539 ถึงช่วงเดือนธันวาคม พ.ศ. 2543 โดยใช้ข้อมูลของธนาคารกรุงเทพจำกัด (มหาชน) เป็นตัวอย่างในการทดสอบระบบ

1.3.3 เป็นการพัฒนาระบบโดยใช้โปรแกรมภาษา DELPHI โดยการทำงานของระบบจะเริ่มต้นจากการรับข้อมูลราคาหลักทรัพย์ในลักษณะอนุกรมเวลา ทำการตรวจสอบและประมวลผลข้อมูลเบื้องต้นแล้วนำข้อมูลเหล่านั้นมาใช้เพื่อสร้างและทดสอบหาแบบจำลองที่เหมาะสมให้กับระบบ หลังจากนั้นจึงนำแบบจำลองที่ได้มาใช้ในการพยากรณ์หาราคาหลักทรัพย์ที่เป็นไปได้ในอนาคตต่อไป

1.4 ขั้นตอน และวิธีการดำเนินงาน

ในการสร้างระบบพยากรณ์ราคาหลักทรัพย์ของโครงการพัฒนาระบบงานที่จัดทำขึ้นนี้มีขั้นตอนการดำเนินงานดังต่อไปนี้

1.4.1 ศึกษาหลักการของดาต้าไมนิ่งที่ใช้ในงานด้านการพยากรณ์ข้อมูล (Prediction)

1.4.2 ศึกษาทฤษฎีของอัลกอริทึมนิวรอลเน็ตเวิร์กแบบป้อนกลับอย่างง่าย (Partial Recurrent Neural Network) เพื่อใช้ในการสร้างแบบจำลองให้กับระบบ

1.4.3 รวบรวมข้อมูลราคาหลักทรัพย์ที่ต้องการใช้ในการวิเคราะห์

1.4.4 ดำเนินการพัฒนาระบบงาน โดยแบ่งส่วนการทำงานเป็น 4 ขั้นตอนย่อย คือ

1.4.4.1 การนำข้อมูลเข้าสู่ระบบ และการประมวลผลเบื้องต้น

1.4.4.2 การสร้างแบบจำลองให้กับข้อมูล และการทดสอบแบบจำลองที่ได้

1.4.4.3 การทดสอบแบบจำลอง และทำการปรับเปลี่ยนค่าตัวแปรที่ใช้กำหนดและควบคุมการสร้างแบบจำลองให้กับระบบ เพื่อหาแบบจำลองที่เหมาะสมกับข้อมูลแต่ละชุด โดยพิจารณาจากค่าความผิดพลาดเฉลี่ย (Mean Square Error: MSE) ของเอาต์พุตที่ได้จากแบบจำลองที่สร้างขึ้น

1.4.4.4 การบันทึกแบบจำลองที่เหมาะสมเพื่อนำไปใช้ในการพยากรณ์ราคาหลักทรัพย์ตามความต้องการต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 สามารถเรียนรู้ และเข้าใจทฤษฎีเบื้องต้นของดาต้าไมนิ่งโดยใช้อัลกอริทึมแบบนิวรอลเน็ตเวิร์ก และนำไปประยุกต์ใช้ในงานด้านการพยากรณ์ข้อมูลได้อย่างเหมาะสม

1.5.2 สามารถพัฒนาระบบพยากรณ์ราคาหลักทรัพย์เพื่อช่วยในการตัดสินใจด้านการลงทุนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีเบื้องต้นของดาต้าไมนิ่ง (Data Mining)

ในปัจจุบันการเติบโตอย่างรวดเร็วของกระบวนการจัดการกับข้อมูลที่อยู่ในรูปแบบทางอิเล็กทรอนิกส์นั้นนำไปสู่ช่วงเวลาที่เรียกกันว่า ยุคของข่าวสารข้อมูล (Information Age) ซึ่งองค์กรทั้งขนาดใหญ่ และขนาดกลางส่วนมากมีการนำระบบฐานข้อมูลมาใช้ในการเก็บรวบรวมข้อมูลและกิจกรรมขององค์กรที่เกิดขึ้น โดยในแต่ละวันจำนวนของข้อมูลจากการปฏิบัติงานขององค์กรก็มีเพิ่มขึ้นเรื่อยๆ และจากข้อมูลทั้งหมดนี้อาจมีลักษณะบางอย่างแฝงอยู่ในข้อมูล เช่น แนวโน้ม (Trend) หรือ รูปแบบเฉพาะ (Pattern) ของข้อมูล เป็นต้น ซึ่งลักษณะแฝงเหล่านี้อาจจะสามารถนำไปใช้ประโยชน์ในการตัดสินใจในเชิงธุรกิจเพื่อปรับปรุงแก้ไข หรือหาวิธีสำหรับการดำเนินงานที่เหมาะสมต่อไปได้ แต่อย่างไรก็ตามในระบบฐานข้อมูลนั้นมีจำนวนข้อมูลค่อนข้างมาก ทำให้การวิเคราะห์หาข้อมูลที่มีประโยชน์เหล่านี้โดยใช้การคำนวณด้วยมือนั้นทำได้ยาก เพราะเนื่องจากในหลายๆ กรณีนั้นตัวแปรที่เป็นอิสระต่อกันจะต้องถูกนำมาพิจารณาพร้อมๆ กันเพื่อสร้างแบบจำลองที่มีความแม่นยำมากขึ้น ดังนั้นจึงเกิดความต้องการในการเพิ่มขีดความสามารถของการวิเคราะห์ข้อมูลต่างๆ เพื่อให้สามารถนำเอาประโยชน์จากข้อมูลในฐานข้อมูลขนาดใหญ่มาใช้ได้เพิ่มมากขึ้น และนำไปสู่การพัฒนาอย่างรวดเร็วในเชิงเทคโนโลยีของระบบที่ใช้ในการวิเคราะห์ และค้นหาประโยชน์ของข้อมูลอย่างมีประสิทธิภาพนี้

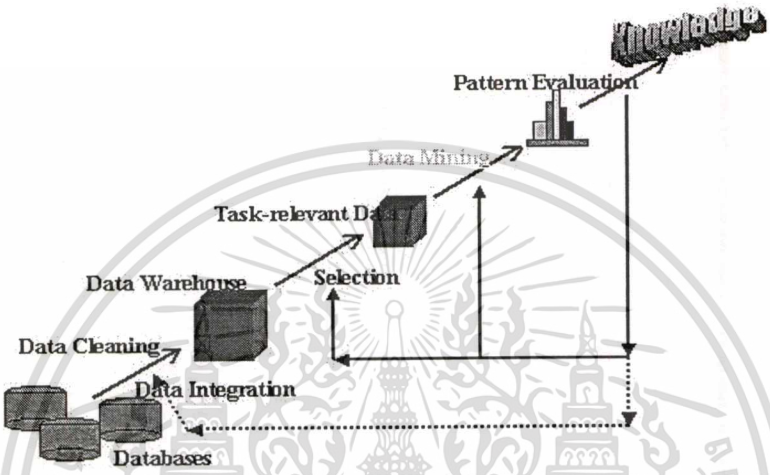
2.1 หลักการทั่วไปของดาต้าไมนิ่ง (Data Mining)

ดาต้าไมนิ่ง (Data Mining) เป็นกระบวนการที่สำคัญกระบวนการหนึ่งในการค้นหาลักษณะแฝงของข้อมูลที่อยู่ในกลุ่มข้อมูลจำนวนมาก (Knowledge Discovery in Database: KDD) ซึ่งลักษณะที่น่าสนใจของข้อมูลเหล่านี้ ได้แก่ รูปแบบ ความสัมพันธ์ การเปลี่ยนแปลงของข้อมูล โครงสร้างที่เด่นชัด หรือ ลักษณะที่ผิดปกติของข้อมูลจากข้อมูลจำนวนมากที่เก็บอยู่ในฐานข้อมูลหรือแหล่งที่เก็บข้อมูลอื่นๆ เป็นต้น ซึ่งวิธีการที่นำมาใช้ในการทำไมนิ่งข้อมูลเหล่านี้ก็มีวัตถุประสงค์แตกต่างกันขึ้นอยู่กับผลลัพธ์ของกระบวนการโดยรวมที่ต้องการ ดังนั้นจึงมีการนำเสนอวิธีการหลายอย่างสำหรับแต่ละงานที่มีเป้าหมายแตกต่างกันไปเพื่อให้ได้ผลลัพธ์ที่เหมาะสมตามที่ต้องการ และเนื่องจากความแพร่หลายของการจัดเก็บข้อมูลในลักษณะที่เป็นรูปแบบทางอิเล็กทรอนิกส์ และความต้องการในการเปลี่ยนข้อมูลเหล่านี้ให้เป็นข้อมูลที่มีประโยชน์ต่อการนำไปประยุกต์ใช้ในงานด้านต่างๆ เช่น การวิเคราะห์ด้านการตลาด การบริหารธุรกิจ รวมถึงระบบที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วยสนับสนุนการตัดสินใจ เป็นต้น ดังนั้นจึงทำให้การนำดาต้าไมนิ่งมาใช้ได้รับความสนใจมากในช่วง 2-3 ปีที่ผ่านมา

สำหรับกระบวนการค้นหาลักษณะแฝงของข้อมูลนั้นประกอบด้วยขั้นตอนต่างๆ ตามรูปที่ 2.1 ดังนี้



รูปที่ 2.1 แสดงขั้นตอนต่างๆ ของกระบวนการค้นหาลักษณะแฝงของข้อมูล

2.1.1 การคัดเลือกข้อมูล (Data Selection)

2.1.2 การกรองข้อมูล (Data Cleaning)

2.1.3 การแปลงรูปแบบข้อมูล (Data Transformation)

2.1.4 การทำไมนิ่งข้อมูล (Data Mining)

2.1.5 การวิเคราะห์และประเมินผลลัพธ์ที่ได้ (Result Analysis and Evaluation)

2.1.1 การคัดเลือกข้อมูล (Data Selection)

เป็นการระบุถึงแหล่งข้อมูลที่จะนำมาใช้ในการทำไมนิ่งรวมถึงการนำข้อมูลที่ต้องการออกมาจากฐานข้อมูลเพื่อทำการพิจารณาความเหมาะสมในเบื้องต้นต่อไป

2.1.2 การกรองข้อมูล (Data Cleaning)

เป็นกระบวนการที่ทำให้เกิดความมั่นใจในคุณภาพของข้อมูลที่จะนำมาใช้วิเคราะห์ ซึ่งโดยทั่วไปในขั้นตอนนี้จะคำนึงถึงข้อมูลใน 2 ลักษณะ คือ Noisy Data ซึ่งหมายถึงข้อมูลที่มีค่าผิดไปจากค่าปกติ และ Missing Value ซึ่งหมายถึงข้อมูลที่ขาดหายไป โดยมีการพิจารณาว่าควรจะตัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลเหล่านั้นทิ้งไป หรือควรหาวิธีการที่เหมาะสมในการคาดคะเนค่าสำหรับข้อมูลในตำแหน่งนั้นๆ เพื่อนำมาใช้วิเคราะห์ร่วมกับข้อมูลอื่นๆ ได้ต่อไป

2.1.3 การแปลงรูปแบบข้อมูล (Data Transformation)

เป็นการแปลงข้อมูลที่เลือกมาให้อยู่ในรูปแบบที่สามารถนำไปใช้วิเคราะห์ตามอัลกอริทึม (Algorithm) ของแบบจำลองที่ใช้ในการทำดาต้าไมนิ่งได้อย่างมีประสิทธิภาพต่อไป

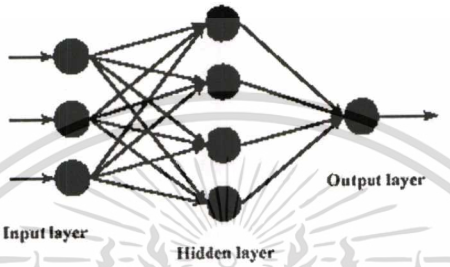
2.1.4 การทำไมนิ่งข้อมูล (Data Mining)

จากลักษณะงานหลายประเภทในการทำดาต้าไมนิ่งนั้นทำให้มีการพัฒนาเทคนิค หรือวิธีการที่เหมาะเพื่อใช้กับงานแต่ละประเภท ซึ่งวิธีการที่ต่างกันก็จะมีความเหมาะสมกับวัตถุประสงค์ของผู้ใช้งานในด้านต่างๆกันด้วย โดยแต่ละวิธีก็มีข้อดี และข้อด้อยที่แตกต่างกัน และนอกจากนี้ในงานแต่ละอย่างนั้นไม่ได้ระบุว่าจะต้องใช้เทคนิคใดเทคนิคหนึ่งโดยเฉพาะเท่านั้น แต่ขึ้นอยู่กับลักษณะของปัญหาที่นำมาวิเคราะห์ ซึ่งในบางครั้งการนำเทคนิคหลายๆ อย่างมาใช้เพื่อเปรียบเทียบความเหมาะสมกับแต่ละงาน หรือการนำหลายๆ วิธีการมาใช้ร่วมกันอาจจะมีส่วนช่วยในการค้นหาความหมาย หรือความสัมพันธ์จากข้อมูลออกมาได้อย่างมีประสิทธิภาพมากกว่าการใช้เพียงวิธีการเดียว ซึ่งตัวอย่างของเทคนิคที่สามารถนำมาใช้ในการทำดาต้าไมนิ่ง ได้แก่

- **Statistical Method** : เป็นการวิเคราะห์ข้อมูลในเชิงตัวเลข โดยมีวัตถุประสงค์หลักเพื่อแยกความแตกต่าง หรือหาความสัมพันธ์ของข้อมูลภายในกลุ่มข้อมูล โดยใช้วิธีการคำนวณตามหลักการทางสถิติ เช่น การสร้างและทดสอบสมมติฐาน (Hypothesis Testing) และ การวิเคราะห์สมการการถดถอย (Regression Analysis) เพื่อหาความสัมพันธ์ของข้อมูล เป็นต้น ซึ่งสิ่งสำคัญสำหรับวิธีนี้ คือ ข้อมูลที่นำมาวิเคราะห์ต้องอยู่ในรูปตัวเลข และมีการแบ่งกลุ่มเพื่อใช้ในการวิเคราะห์เปรียบเทียบกัน
- **Case-Based Reasoning** : เป็นการพยายามนำเทคโนโลยีมาช่วยแก้ปัญหาโดยการใช้ข้อมูลจากประสบการณ์และผลลัพธ์ที่ได้จากในช่วงที่ผ่านมาโดยตรง ซึ่งตามปกติในแต่ละครั้งของการเกิดปัญหาเฉพาะต่างๆ นั้น ส่วนมากมักเป็นปัญหาที่เคยเกิดขึ้น และได้รับการแก้ไขสำเร็จผ่านมาแล้วในอดีต ดังนั้นในการพิจารณาปัญหาใหม่ที่เกิดขึ้นนั้นจะมีการนำรายละเอียดเฉพาะของปัญหามาวิเคราะห์ แล้วระบบก็จะค้นหากรณีต่างๆ จากข้อมูลที่มีการเก็บรวบรวมไว้ โดยหาลักษณะที่ใกล้เคียงกันมากที่สุด และเมื่อพบกรณีดังกล่าวแล้ว ก็จะนำวิธีการแก้ไขนั้นมาใช้กับปัญหาใหม่ที่เกิดขึ้น และผลลัพธ์ของกรณีใหม่นี้ก็จะถูกเก็บไว้เพื่อใช้อ้างอิงต่อไป
- **Neural Networks** : เป็นวิธีการที่ใช้ในการระบุรูปแบบของข้อมูล หรือ จัดกลุ่มข้อมูล โดยในแบบจำลองจะประกอบด้วยโหนดที่มีการเชื่อมต่อกันเป็นโครงข่าย ทำงานในลักษณะคล้ายกับระบบ

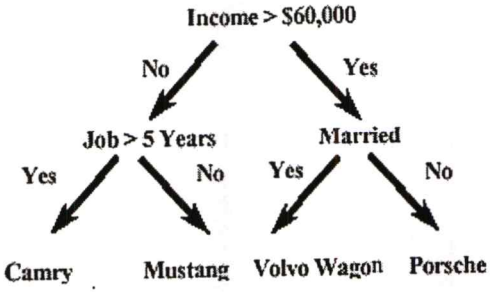
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประสาทของมนุษย์ ดังแสดงตัวอย่างโครงสร้างได้ดังรูปที่ 2.2 โดยที่แต่ละโหนดมีการตอบสนองต่อตัวแปร หรือข้อมูล ที่เข้ามากระตุ้นระบบ โดยในช่วงเวลาใดๆ อาจมีโหนดที่ได้รับการกระตุ้นให้มีการทำงานได้พร้อมกันหลายๆ โหนด ซึ่งลักษณะของการทำงานที่ตอบสนองต่อข้อมูลที่เข้าสู่ระบบในแต่ละครั้งนั้นจะเสมือนเป็นตัวแทนของข้อมูลที่สร้างให้ระบบมีการเรียนรู้ลักษณะดังกล่าวของแต่ละข้อมูล ต่อจากนั้นจึงทำการจัดกลุ่มข้อมูลตามลักษณะบางอย่างที่คล้ายกัน



รูปที่ 2.2 แสดงตัวอย่างของนิวรอลเน็ตเวิร์ก

• **Decision Trees** : เป็นการวิเคราะห์เพื่อค้นหากฎเกณฑ์ หรือความสัมพันธ์ในลักษณะที่แบ่งย่อยลงไปตามของโครงสร้างแบบต้นไม้โดยที่แต่ละโหนดที่ไม่ใช่โหนดสุดท้ายของโครงสร้างจะแทนจุดที่มีการตรวจสอบเงื่อนไข หรือเป็นการตัดสินใจเกี่ยวกับลักษณะของข้อมูลที่นำมาพิจารณา และจากผลลัพธ์ที่ได้ในแต่ละครั้งก็จะมีการเลือกเส้นทางที่แน่นอน โดยในการจัดแบ่งประเภทของข้อมูลนั้นจะเริ่มต้นพิจารณาจากโหนดที่เป็นโหนดแรก (Root) และอ้างอิงลงมาตามโครงสร้างต้นไม้เรื่อยๆ จนกระทั่งถึงโหนดสุดท้ายของโครงสร้าง (Leaf) ซึ่งแสดงว่ากระบวนการตัดสินใจ หรือวิเคราะห์ข้อมูลนั้นๆ เสร็จสิ้นลงแล้ว และสามารถบอกลักษณะ หรือประเภทของข้อมูลนั้นได้ตามโหนดที่ข้อมูลผ่านการตรวจสอบ ดังแสดงตัวอย่างโครงสร้างในรูปที่ 2.3



รูปที่ 2.3 แสดงตัวอย่างของโครงสร้างต้นไม้ (Decision Tree)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• **Association Rule** : เป็นวิธีการเพื่อหาข้อมูลที่มีความสัมพันธ์กันหรือข้อมูลที่มีลักษณะการเกิดขึ้นร่วมกันภายในกลุ่มข้อมูลที่นำมาพิจารณา แต่จากผลที่ได้นั้นไม่ได้หมายความว่าความสัมพันธ์ของข้อมูลที่พบนั้นจะเป็นเป็นสิ่งที่มีความสำคัญเสมอไป ดังนั้นจึงต้องนำผลที่ได้ไปพิจารณาโดยใช้วิธีที่เหมาะสมอีกครั้งหนึ่ง

2.1.5 การวิเคราะห์และประเมินผลลัพธ์ที่ได้ (Result Analysis and Evaluation)

เป็นขั้นตอนการแปลความหมาย และการประเมินผลลัพธ์ที่ได้ว่ามีความเหมาะสม หรือตรงกับวัตถุประสงค์ที่ต้องการนำไปใช้งานหรือไม่ ซึ่งโดยทั่วไปควรจะมีการแสดงผลในรูปแบบที่สามารถเข้าใจได้โดยง่าย

2.2 ประเภทของการใช้งานดาต้าไมนิ่ง

ดาต้าไมนิ่งเป็นวิธีการที่ถูกนำมาใช้อย่างกว้างขวางในการวิเคราะห์หาความหมาย หรือลักษณะเด่นที่แฝงอยู่ในข้อมูลจำนวนมากๆ โดยการเลือกเทคนิค หรือแบบจำลองต่างๆ เพื่อนำมาประยุกต์ใช้ก็ขึ้นอยู่กับประเภทของงานและวัตถุประสงค์ของผู้ใช้เป็นหลัก ซึ่งลักษณะงานที่มีการนำดาต้าไมนิ่งมาใช้สามารถจำแนกได้ดังรูปที่ 2.4

Applications	Market Management	Risk Management	Fraud Management
	✓ Target Marketing ✓ Customer Relationship Management ✓ Market Basket Analysis ✓ Cross Selling ✓ Market Segmentation	✓ Forecasting ✓ Customer Retention ✓ Quality Control ✓ Competitive Analysis	✓ Fraud Detection

Opertions	Predictive Modeling	Database Segmentation	Link Analysis	Deviation Detection
Techniques	✓ Classification ✓ Value Prediction	✓ Demographic Clustering ✓ Neural Clustering	✓ Association Discovery ✓ Sequential Pattern Discovery ✓ Similar Time Sequence Discovery	✓ Visualization ✓ Statistics

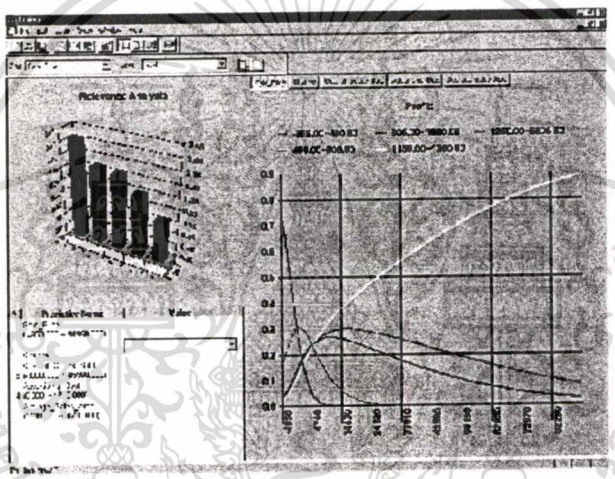
รูปที่ 2.4 แสดงประเภทของงานที่มีการนำเทคนิคต่างๆ ของดาต้าไมนิ่งมาใช้

โดยทั่วไปประเภทของงานตามลักษณะของแบบจำลองที่ใช้ในการทำดาต้าไมนิ่งนั้นสามารถแบ่งกลุ่มได้เป็น 2 ประเภท คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 งานด้านการพยากรณ์ (Predictive Data Mining) เป็นการคาดคะเนลักษณะ หรือประมาณค่าที่ชัดเจนของข้อมูลที่จะเกิดขึ้นโดยใช้พื้นฐานจากข้อมูลที่ผ่านมาในอดีต ดังตัวอย่างในรูปที่ 2.5 ซึ่งสามารถแบ่งลักษณะงานได้ดังต่อไปนี้

- **Classification:** เป็นการจัดกลุ่มให้กับแต่ละข้อมูลในฐานข้อมูลโดยมีการระบุค่า หรือลักษณะที่เป็นไปได้ของข้อมูลภายในแต่ละกลุ่ม เช่น การจัดกลุ่มของผู้ป่วยตามผลของการใช้ยารักษาเพื่อระบุรูปแบบการรักษาให้กับผู้ป่วยใหม่ที่เข้ารับการรักษา เป็นต้น
- **Prediction:** เป็นการพยากรณ์ค่าที่เป็นไปได้ หรือการกระจายของค่าที่เป็นไปได้ของตัวแปรใดๆ ในกลุ่มข้อมูล



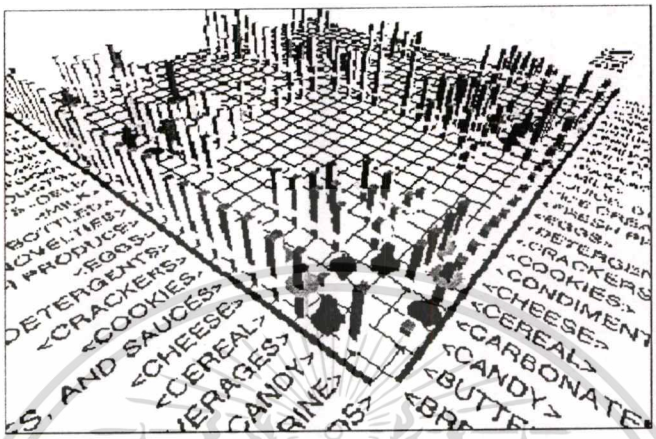
รูปที่ 2.5 แสดงตัวอย่างผลที่ได้จากแบบจำลองด้านการพยากรณ์ (Han. 2000: 45)

2.2.2 งานด้านการแบ่งกลุ่ม (Descriptive Data Mining) เป็นการหาแบบจำลองเพื่ออธิบายลักษณะบางอย่างของข้อมูลที่มีอยู่ ซึ่งโดยส่วนมากจะเป็นลักษณะการแบ่งกลุ่มให้กับข้อมูล ดังตัวอย่างในรูปที่ 2.6 สามารถแบ่งลักษณะของงานได้ดังต่อไปนี้

- **Link Analysis (Associations) :** เป็นการหาความสัมพันธ์ของข้อมูลภายในกลุ่มข้อมูล เพื่อใช้ลักษณะของข้อมูลหนึ่งๆ ในการบอกลักษณะที่จะเกิดขึ้นกับข้อมูลอีกตัวหนึ่ง ซึ่งอาจจะเป็นการหาความสัมพันธ์ของข้อมูลในกลุ่มเดียวกัน เช่น การระบุว่าในกลุ่มของลูกค้าที่ซื้อนมนั้นจะมีลูกค้า 64% ที่ซื้อขนมปังด้วย หรือ อาจจะเป็นการหาความสัมพันธ์ของตัวแปรระหว่างกลุ่มข้อมูลก็ได้ เช่น ในทุกๆ ครั้งที่ดัชนีของตลาดหุ้นหนึ่งลดลง 5% ดัชนีของตลาดหุ้นอื่นจะเพิ่มขึ้น 13% ภายในช่วง 2-6 เดือนหลังจากนั้น เป็นต้น ซึ่งลักษณะของการหาความสัมพันธ์นั้นอาจแบ่งได้เป็น 3 กลุ่ม คือ การหาความสัมพันธ์ระหว่างข้อมูล (Association Discovery) การหาความสัมพันธ์ในลักษณะที่เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับของข้อมูล (Sequential Pattern discovery) และการหาความสัมพันธ์ของข้อมูลกับช่วงเวลาใดๆ (Similar Time Sequence discovery)



รูปที่ 2.6 แสดงตัวอย่างผลที่ได้จากแบบจำลองด้านการแบ่งกลุ่ม (Han. 2000: 28)

- **Clustering :** เป็นการแบ่งกลุ่มของข้อมูลโดยที่ในแต่ละกลุ่มนั้นประกอบด้วยข้อมูลที่มีลักษณะคล้ายกันโดยในการแบ่งกลุ่มของข้อมูลที่ดีนั้น จะทำให้ได้กลุ่มที่มีคุณภาพ นั่นคือ มีความแตกต่างกันระหว่างกลุ่มมาก (High Heterogeneity) และ มีความแตกต่างกันภายในกลุ่มน้อย (High Homogeneity) เช่น การแบ่งกลุ่มของลูกค้าโดยใช้พฤติกรรมการซื้อเป็นเกณฑ์
- **Deviation Detection :** เป็นเทคนิคที่ใช้ในการแสดงลักษณะของข้อมูลที่ผิดปกติ หรือผิดไปจากที่คาดไว้ โดยมีการแสดงผลอยู่ในลักษณะที่สามารถทำความเข้าใจและแปลความหมายได้ง่าย เช่น การใช้กราฟ เป็นต้น

2.3 การประยุกต์ใช้ดาต้าไมนิ่งกับการพยากรณ์ราคาหลักทรัพย์

ในปัจจุบันบริษัทหลักทรัพย์ต่างๆ ได้ใช้การพยากรณ์เพื่อการหาผลกำไรจากการลงทุนเพิ่มขึ้นจากตลาดหลักทรัพย์ โดยมีการพัฒนาแบบจำลองที่มีประสิทธิภาพมากขึ้นเรื่อยๆ และมีการปรับเปลี่ยนจากการใช้เทคนิคเดียวมาใช้หลายๆ เทคนิคประกอบกันเพื่อสร้างสัญญาณที่จะบอกแนวทางของตลาดในอนาคตหลายๆ แบบได้อย่างน่าเชื่อถือ นอกจากนี้ยังมีการนำปัจจัยหลายประการมาพิจารณาร่วมกัน เช่น ปริมาณการซื้อขาย อัตราแลกเปลี่ยนเงินตราต่างประเทศ ดัชนีหลักทรัพย์ในกลุ่มต่างๆ ราคาทองคำ เป็นต้น รวมถึงการพิจารณาข้อมูลของหลักทรัพย์ที่เปลี่ยนไปตามช่วงเวลาต่างๆ กันด้วย โดยปัจจัยดังกล่าวจะถูกนำมาใช้เป็นอินพุตให้กับแบบจำลองที่สร้างขึ้น สำหรับในโครงการพัฒนาระบบงานนี้จะพิจารณาปัจจัยในลักษณะทางเทคนิค คือการนำราคาหลักทรัพย์ในรูป

เอกสารนี้มีลิขสิทธิ์ ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบของอนุกรมเวลามาใช้เพื่อพยากรณ์ค่าราคาหลักทรัพย์ของช่วงเวลาถัดไป และเนื่องจากลักษณะข้อมูลของราคาหลักทรัพย์เป็นข้อมูลในแบบไม่เป็นเชิงเส้น (Non-Linear) ดังนั้นแบบจำลองที่เหมาะสมที่เลือกมาใช้ในการพัฒนาระบบงานนี้ จะใช้หลักการของนิวรอลเน็ตเวิร์กเป็นหลัก ซึ่งโครงข่ายของนิวรอลเน็ตเวิร์กจะถูกสอนให้เลียนแบบแนวคิด และเรียนรู้ลักษณะการเคลื่อนไหวของข้อมูลราคาหลักทรัพย์ที่ผ่านมาในอดีต โดยพิจารณาจากตัวแปรหลายๆ ค่าในอนุกรมเวลาที่นำมาใช้เป็นอินพุตร่วมกัน และหลังจากที่เน็ตเวิร์กมีการเรียนรู้แล้ว ก็จะสามารถนำแบบจำลองมาใช้ในการพยากรณ์ค่าของราคาหลักทรัพย์ที่จะเกิดขึ้นในช่วงเวลาต่อๆ ไปได้ ซึ่งทฤษฎีและหลักการของนิวรอลเน็ตเวิร์กนั้นจะได้อธิบายไว้ในบทที่ 3 ต่อไป

2.3.1 การวิเคราะห์ข้อมูลในอนุกรมเวลา (Time series analysis)

ลักษณะของข้อมูลที่แสดงความสัมพันธ์กับเวลานั้น โดยทั่วไปจะอยู่ในรูปของ t(1), t(2), ….t(i), ….t(n) โดยที่ n เป็นค่าล่าสุดของข้อมูลในช่วงเวลาใดๆ ซึ่งจากข้อมูลที่รวบรวมได้ในช่วงเวลาที่ผ่านมา การวิเคราะห์ข้อมูลในลักษณะชุดข้อมูลตามช่วงเวลาดังกล่าวนี้มุ่งเน้นในการพยายามหาแบบจำลองใดๆ เพื่อใช้พยากรณ์ค่าของข้อมูลในลำดับที่ t(n+1) จากค่าของข้อมูลที่ t(n)

ในการวิเคราะห์ชุดของข้อมูลที่มีความสัมพันธ์กันในอนุกรมเวลานั้นเป็นการสร้างแบบจำลองอีกแบบหนึ่งที่มีการนำไปประยุกต์ใช้ในเชิงธุรกิจอย่างกว้างขวาง เนื่องจากมีการพิจารณาว่าค่าของข้อมูลหรือลักษณะต่างๆ ที่เกิดขึ้นกับข้อมูลนั้นมีรูปแบบเฉพาะที่มีความสัมพันธ์กันในเชิงเวลา ดังนั้นจึงมีการนำข้อมูลจากช่วงเวลาที่ผ่านมาก่อนหน้า มาทำการวิเคราะห์เพื่อให้สามารถหาวิธีพยากรณ์ค่าข้อมูลที่จะเกิดขึ้นในอนาคตได้ ซึ่งจากลักษณะของข้อมูลราคาหลักทรัพย์โดยทั่วไปนั้นก็จะมีการรวบรวมไว้ในลักษณะของชุดข้อมูลต่างๆ เทียบกับช่วงเวลา เช่นเดียวกัน และเมื่อพิจารณาลักษณะของการวิเคราะห์ข้อมูลแบบที่สัมพันธ์กับเวลานี้แล้ว สามารถแบ่งได้เป็น 2 ลักษณะ

2.3.1.1 การวิเคราะห์โดยพิจารณาตัวแปรเดียว (Univariate time - series analysis) นั่นคือเป็นการนำข้อมูลของตัวแปรที่สนใจเพียงตัวเดียวมาใช้ในการวิเคราะห์ เช่น การพยากรณ์ค่าราคาหลักทรัพย์ในอนาคต โดยใช้ข้อมูลในอดีตของราคาหลักทรัพย์ที่ต้องการพยากรณ์มาพิจารณาเพียงตัวแปรเดียว

2.3.1.2 การวิเคราะห์โดยพิจารณาจากหลายตัวแปร (Multivariate time-series analysis) เป็นการพัฒนามาจากการพยากรณ์โดยใช้ตัวแปรเดียว โดยแทนที่จะพิจารณาเพียงตัวแปรเดียว ก็จะมีการนำตัวแปรอื่นๆ ณ ช่วงเวลาเดียวกันมาพิจารณาร่วมด้วย เช่นในการพยากรณ์ราคาหลักทรัพย์อาจจะมีการนำค่าอัตราแลกเปลี่ยนของเงินสกุลต่างๆ ราคาทองคำ หรือราคาน้ำมัน มาพิจารณาร่วมกับราคาหลักทรัพย์ในแต่ละตัวด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อพิจารณาจากวัตถุประสงค์ในการวิเคราะห์ข้อมูลลักษณะนี้แล้ว จะพบว่าในบางครั้งไม่จำเป็นที่จะต้องสร้างแบบจำลองที่สามารถรู้ค่าที่แน่นอนในช่วงเวลาที่กำหนดในอนาคตได้อย่างชัดเจนแต่ ต้องการใช้เพียงแค่สังเกตแนวโน้มของข้อมูลว่ามีโอกาสเพิ่มขึ้นหรือลดลงมากน้อยแค่ไหนเท่านั้น โดยใช้ข้อมูลในอดีตมาเป็นพื้นฐาน ซึ่งตัวอย่างของงานที่มีการวิเคราะห์ข้อมูลในลักษณะนี้ได้แก่

แบบที่ 1: การพยากรณ์ค่าข้อมูลที่น่าจะเกิดขึ้นอย่างแน่นอน ของตัวแปรหนึ่งๆ ณ ช่วงเวลาใดๆ ที่กำหนดในอนาคต

แบบที่ 2: การพยากรณ์ข้อมูลเพื่อดูแนวโน้มที่จะเกิดขึ้นในอนาคต

แบบที่ 3: การพยากรณ์ค่าความเปลี่ยนแปลงของข้อมูลในช่วงเวลาใดๆ

ดังนั้นการสร้างแบบจำลองจึงต้องพิจารณาจากวัตถุประสงค์ที่ต้องการเป็นสำคัญ สำหรับในการพัฒนาโครงการนี้จะสร้างระบบเพื่อการพยากรณ์ราคาหลักทรัพย์ตามลักษณะที่ 1 คือเป็นการพยากรณ์ค่าของราคาหลักทรัพย์ในช่วงเวลาถัดไปของอนุกรมเวลาโดยใช้ค่าข้อมูลราคาหลักทรัพย์ในช่วงที่ผ่านมาระยะเวลาหนึ่งเป็นเกณฑ์ในการสร้างแบบจำลองให้กับระบบ

2.3.2 การประมวลผลข้อมูลเบื้องต้น (Data Preprocessing)

ในการประมวลผลข้อมูลเบื้องต้นก่อนการสร้างแบบจำลองที่เหมาะสมให้กับระบบพยากรณ์ราคาหลักทรัพย์นั้น มีการพิจารณาใน 2 ลักษณะ คือ การพิจารณาข้อมูลส่วนที่ขาดหายไปในอนุกรมเวลา (Data Cleaning) และการแปลงค่าข้อมูลให้เหมาะสมกับรูปแบบอัลกอริทึมที่ใช้ในการสร้างแบบจำลอง (Data Transformation) ซึ่งมีวิธีการหลักที่นำมาใช้ ดังต่อไปนี้

2.3.2.1 การแทนค่าข้อมูล (Data Cleaning)

สำหรับเทคนิคที่นิยมนำมาใช้ในการหาข้อมูลเพื่อแทนค่าข้อมูลที่ขาดหายไป ได้แก่

- การหาค่า Moving Average ซึ่งจะเป็นการคำนวณหาค่าเฉลี่ยของข้อมูลจากช่วงเวลาหนึ่งก่อนหน้าแล้วจึงนำมาใช้เป็นอินพุตแทนข้อมูลส่วนที่หายไปให้กับระบบแทน เพื่อช่วยเพิ่มประสิทธิภาพให้กับการสร้างแบบจำลองในพยากรณ์ข้อมูล โดยส่วนใหญ่จะใช้ช่วงในการหาค่าเฉลี่ยที่ประมาณ 2-5 ช่วงข้อมูล โดยมีสูตรที่ใช้ในการคำนวณ ดังต่อไปนี้

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• การใช้สมการถดถอยแบบเชิงเส้น (Linear Regression Model) เป็นการหาสมการเส้นตรงที่เหมาะสมให้กับชุดของข้อมูล โดยที่สามารถนำสมการที่ได้มาใช้คำนวณค่าของตัวแปรหนึ่ง (Y) ได้จากค่าตัวแปรอีกตัวหนึ่ง (X) โดยมีสมการแสดงความสัมพันธ์ของข้อมูลคือ

$$Y = \alpha + \beta X \qquad \text{.....2.2}$$

โดย Y = ตัวแปรตาม

X = ตัวแปรนำ

α และ β = ค่าสัมประสิทธิ์ของสมการถดถอย โดยสามารถคำนวณได้จากสมการดังต่อไปนี้

$$\beta = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(X_i - \bar{X})^2} \qquad \text{.....2.3}$$

$$\alpha = \bar{Y} - \beta\bar{X} \qquad \text{.....2.4}$$

2.3.2.2 การแปลงค่าข้อมูล (Data Transformation)

เป็นกระบวนการในการปรับขอบเขตของข้อมูลให้อยู่ในช่วงที่เหมาะสมต่อการนำไปใช้งานในการสอนให้นิวรอลเน็ตเวิร์กเกิดการเรียนรู้ สำหรับในแบบจำลองที่ใช้หลักการของนิวรอลเน็ตเวิร์กนั้น กระบวนการแปลงค่าข้อมูลที่ต้องนำมาใช้ คือ การนอมัลไลซ์ข้อมูล (Normalization) ซึ่งเป็นการลดค่าของข้อมูลให้อยู่ในขอบเขตที่น้อยลง เพื่อให้เหมาะสมกับฟังก์ชันที่ใช้งานในอัลกอริทึมที่นำมาใช้ในการเรียนรู้ของเน็ตเวิร์ก ซึ่งวิธีการนอมัลไลซ์ข้อมูลนั้นมีหลายวิธี แต่ที่มีการนำมาใช้กันอย่างกว้างขวางได้แก่ การแปลงค่าข้อมูลในลักษณะเป็นเชิงเส้น (Min-max Normalization)

โดยถ้ากำหนดให้ $\min_A$ และ $\max_A$ เป็นค่าต่ำสุด และค่าสูงสุดของตัวแปร A ตามลำดับ วิธีการของแปลงค่าข้อมูลแบบเชิงเส้น จะทำการแปลงค่าข้อมูลใดๆ (V) ของตัวแปร A ให้เป็นค่าข้อมูลใหม่ (V') ที่อยู่ในช่วง [new_min_A,new_max_A) โดยการคำนวณจากสูตรดังต่อไปนี้

$$V' = [(\frac{V - \min_A}{\max_A - \min_A})(new_\max_A - new_\min_A)] + new_\min_A \qquad \text{....2.5}$$

ซึ่งหลังจากการประมวลผลข้อมูลเบื้องต้นแล้ว จะได้ข้อมูลที่มีความสมบูรณ์มากขึ้น และสามารถนำไปใช้สร้างแบบจำลองได้อย่างมีประสิทธิภาพต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

หลักการของนิวรอลเน็ตเวิร์ก (Neural Network)

ในช่วงเริ่มต้นของการสร้างแบบจำลองเพื่อใช้พยากรณ์ข้อมูลลักษณะที่เป็นอนุกรมเวลานั้นได้มีการนำแบบจำลองทางสถิติ เช่น การวิเคราะห์สมการแสดงความถดถอย (Linear auto-regressive model) มาใช้ในการศึกษา ซึ่งวิธีการที่นำมาใช้ส่วนใหญ่นั้นจะเป็นรูปแบบของการตั้งสมมติฐานในลักษณะที่ข้อมูลที่นำมาใช้นั้นมีความสัมพันธ์กันในลักษณะที่เป็นเชิงเส้น ซึ่งในความเป็นจริงแล้วข้อมูลในรูปอนุกรมเวลาส่วนมากจะไม่ใช่ลักษณะที่เป็นเชิงเส้นตามที่ได้ตั้งสมมติฐานไว้ ดังนั้นจึงทำให้การวิเคราะห์ และพยากรณ์ข้อมูลโดยใช้แบบจำลองทางสถิตินั้นทำได้ค่อนข้างยาก ซึ่งจากผลการศึกษาได้มีการระบุข้อด้อยของแบบจำลองที่เป็นเชิงเส้นไว้ เช่น แบบจำลองในลักษณะเชิงเส้นนั้นไม่สามารถจัดการกับข้อมูลที่มีค่าเปลี่ยนแปลงอย่างมากในช่วงเวลาที่ไม่แน่นอนได้ (Sudden burst of amplitude) และยังไม่สามารถจัดการกับข้อมูลอินพุตที่มีลักษณะแปรปรวนมากๆ (Noisy signal) ได้ด้วย

ดังนั้นเพื่อแก้ปัญหาของข้อจำกัดของแบบจำลองในลักษณะเชิงเส้นตามที่กล่าวมาแล้วข้างต้น จึงได้มีความพยายามนำแบบจำลองในลักษณะที่ไม่เป็นเชิงเส้นมาประยุกต์ใช้งาน เช่น แบบจำลองที่ใช้ทฤษฎีของโครงข่ายประสาทเทียม หรือนิวรอลเน็ตเวิร์ก (Neural Network) ซึ่งเป็นแบบจำลองที่สามารถปรับเปลี่ยนโครงสร้าง หรือตัวแปรในแบบจำลองให้ใช้กับสภาพงานที่แตกต่างกันไปได้ง่ายกว่าการปรับแบบจำลองทางสถิติที่มักจะต้องอาศัยผู้ที่มีความรู้ความชำนาญในงานด้านนี้โดยเฉพาะมาเป็นผู้สร้าง และจากผลการพัฒนาและประยุกต์ใช้งานอย่างต่อเนื่องพบว่านิวรอลเน็ตเวิร์กนั้นให้ผลลัพธ์ที่ดีกว่ากับการนำไปใช้วิเคราะห์ข้อมูลที่มีลักษณะไม่เป็นเชิงเส้นถึงแม้ว่าจะไม่สามารถใช้ทฤษฎีที่เกี่ยวกับสภาวะนั้นๆ อธิบายผลลัพธ์ที่ได้จากแบบจำลองนั้นๆ โดยตรงก็ตาม ดังนั้นจึงมีอาจจะพิจารณาแบบจำลองนิวรอลเน็ตเวิร์กได้ในลักษณะเดียวกับกล่องดำ (Black box) ที่ไม่สามารถอธิบายกระบวนการทำงานภายในได้

ถึงแม้ว่าในช่วงเวลาต่อมาได้มีการพัฒนาวิธีการทางสถิติเพื่อสร้างแบบจำลองที่ไม่เป็นเชิงเส้นสำหรับใช้ในงานด้านการพยากรณ์ขึ้นอีกมาก แต่จากการเปรียบเทียบผลของนิวรอลเน็ตเวิร์กและแบบจำลองที่เป็นเชิงเส้น (Linear Model) ที่พัฒนาขึ้นแล้วพบว่า ในการพยากรณ์ข้อมูลระยะยาวแบบจำลองทั้งสองลักษณะสามารถใช้งานได้อย่างมีประสิทธิภาพใกล้เคียงกัน แต่สำหรับการใช้งานเพื่อการพยากรณ์ข้อมูลในระยะเวลาสั้นๆ นั้น นิวรอลเน็ตเวิร์กสามารถใช้งานได้ดีกว่าอย่างเห็นได้ชัด อย่างไรก็ตามการนำนิวรอลเน็ตเวิร์กมาใช้งานนั้นไม่ได้รับรองว่าจะให้ผลลัพธ์ที่ดีกว่าแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำลองที่เป็นเชิงเส้นทุกครั้ง ในบางกรณีที่เป็นการพยากรณ์ในระยะยาว หรือในกรณีที่โครงสร้างของนิวรอลเน็ตเวิร์กไม่เพียงพอที่จะครอบคลุมปัญหานั้นๆ แบบจำลองทางสถิติก็อาจจะถูกนำมาใช้โดยให้ประสิทธิภาพที่ดีกว่าได้ ดังนั้นในการใช้งานจึงต้องพิจารณาลักษณะของการพยากรณ์ตลอดจนโครงสร้างของนิวรอลเน็ตเวิร์กให้เหมาะสมกับงานที่นำไปใช้ด้วย

สำหรับในบทนี้จะอธิบายถึงหลักการทั่วไปของทฤษฎีนิวรอลเน็ตเวิร์ก ประเภทของนิวรอลเน็ตเวิร์กที่มีการนำมาใช้ในงานด้านการพยากรณ์ข้อมูล โครงสร้างพื้นฐานของนิวรอลเน็ตเวิร์กที่นำมาใช้พัฒนาระบบพยากรณ์ราคาหลักทรัพย์ทั้งในด้านองค์ประกอบของการสร้างแบบจำลอง การคำนวณของโหนด (Node หรือ Unit) ภายในเน็ตเวิร์ก และการปรับโครงสร้างของเน็ตเวิร์กตามอัลกอริทึมที่เลือกนำมาใช้ทำให้เน็ตเวิร์กเกิดการเรียนรู้และสร้างแบบจำลองที่เหมาะสมสำหรับระบบพยากรณ์ราคาหลักทรัพย์ที่พัฒนาขึ้นด้วย

3.1 หลักการทำงานของนิวรอลเน็ตเวร์ก

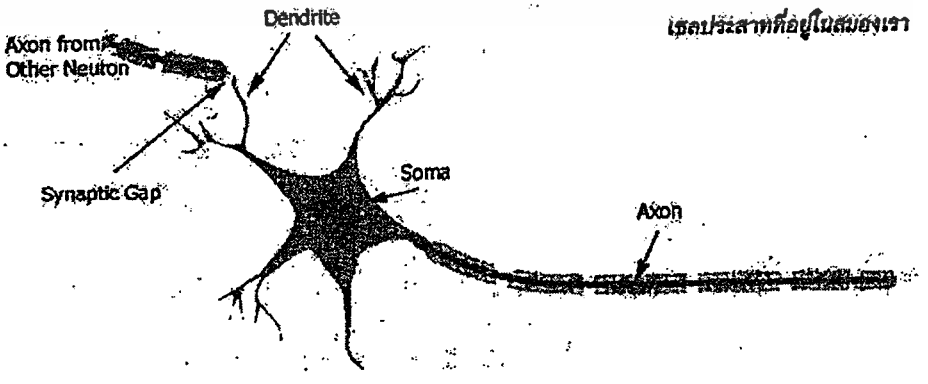
นิวรอลเน็ตเวิร์ก (Neural Network) หรือ โครงข่ายประสาทเทียมนั้นเป็นวิธีการที่นำพื้นฐานการคำนวณของแบบจำลองทางคณิตศาสตร์มาใช้ในการสร้างให้เน็ตเวิร์ก (Network) ให้เกิดการเรียนรู้ ซึ่งจะมีการทำงานในลักษณะเช่นเดียวกับโครงข่ายประสาทของมนุษย์ซึ่งประกอบด้วยเซลล์ประสาท หรือนิวรอล (Neural) ตามโครงสร้างในรูปที่ 3.1 จำนวนมากเชื่อมต่อกันเป็นโครงข่าย โดยในแต่ละเซลล์ประกอบด้วย

3.1.1 เดนไดรต์ (Dendrite) ทำหน้าที่รับสัญญาณอินพุต

3.1.2 ตัวเซลล์ (Soma) ทำหน้าที่เป็นจุดรวมสัญญาณ

3.1.3 แอกซอน (Axon) ทำหน้าที่ส่งสัญญาณเอาต์พุต

3.1.4 ซินแนปติก (Synaptic) ทำหน้าที่เป็นจุดเชื่อมต่อระหว่างเซลล์



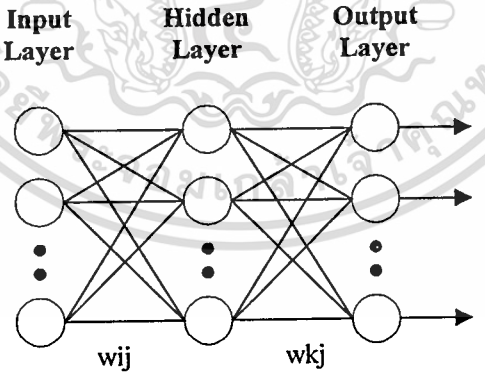
รูปที่ 3.1 แสดงโครงสร้างเซลล์ประสาทของมนุษย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

โดยการทำงานของเซลล์เริ่มจาก เดนไดรต์ (Dendrite) ที่ทำหน้าที่เหมือนอินพุตจะรับข้อมูลมาจากช่องว่างซินแนปติก (Synaptic) โดยวิธีการทางเคมี ซึ่งเราจะเรียกว่าเป็นการถ่วงน้ำหนัก (Weight) ต่อจากนั้นตัวเซลล์ (Soma) จะทำหน้าที่รวมอินพุตต่างๆ ที่ผ่านการถ่วงน้ำหนักแล้วมาประมวลผลสัญญาณโดยใช้วิธีการกระตุ้น ซึ่งเรียกว่าเป็นฟังก์ชันกระตุ้น (Activation Function) ของเซลล์ทำให้ได้เอาต์พุตออกมา และเอาต์พุตที่ได้จากตัวเซลล์นี้ก็ส่งเป็นอินพุตให้กับเซลล์ประสาทหรือนิวรอลอีกตัวหนึ่งต่อเนื่องกันไป ซึ่งภายในร่างกายมนุษย์จะเกิดปฏิกิริยาเช่นนี้นับเป็นล้านๆ ครั้งใน 1 วินาที

สำหรับในโครงสร้างของนิวรอลเน็ตเวิร์กนั้นก็มีการทำงานในลักษณะเดียวกัน คือ มีการส่งข้อมูลการเรียนรู้ของระบบเชื่อมต่อกันเป็นโครงข่ายจากอินพุตจนถึงเอาต์พุต โดยในแบบจำลองของนิวรอลเน็ตเวิร์กนั้นจะมีการเชื่อมต่อกันของข้อมูลในลักษณะที่เป็นลำดับชั้น (Layer) ดังแสดงโครงสร้างพื้นฐานได้ดังรูปที่ 3.2 แต่อย่างไรก็ตามถึงแม้ว่าแบบจำลองนิวรอลเน็ตเวิร์กนี้จะมีข้อดีตรงที่มีความยืดหยุ่นกับข้อมูลที่มีลักษณะผิดปกติไปจากกลุ่มได้มาก และมีความสามารถในการสร้างแบบจำลองจากข้อมูลที่ไม่เคยมีการเรียนรู้มาก่อนได้เป็นอย่างดี แต่ก็ยังมีส่วนที่เป็นข้อด้อย คือ ต้องเสียเวลาในการสอนให้เน็ตเวิร์กเกิดการเรียนรู้ก่อนที่จะนำแบบจำลองไปใช้ในพยากรณ์ค่าข้อมูลได้ และนอกจากนี้ยังไม่สามารถอธิบายวิธีการสร้างแบบจำลองของเน็ตเวิร์กภายในได้อย่างชัดเจนดังที่ได้กล่าวมาแล้วด้วย



รูปที่ 3.2 แสดงโครงสร้างพื้นฐานของแบบจำลองนิวรอลเน็ตเวิร์ก

3.2 โครงสร้างพื้นฐานของนิวรอลเน็ตเวิร์ก

จากรูปที่ 3.2 แสดงให้เห็นว่าโครงสร้างของนิวรอลเน็ตเวิร์ก เป็นลักษณะของโครงข่ายที่มีการเชื่อมต่อกันของกลุ่มอินพุต (ชั้นแรก) และกลุ่มของเอาต์พุต (ชั้นสุดท้าย) โดยผ่านโหนดต่างๆ ตามลำดับชั้นของการเชื่อมต่อ ซึ่งในการเชื่อมต่อกันระหว่างแต่ละโหนดจากอินพุตจนถึงเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั้นจะมีค่าการถ่วงน้ำหนัก (Weight) เฉพาะที่แตกต่างกันไป โดยในระหว่างการสอนให้เน็ตเวิร์กเกิดการเรียนรู้นั้น ค่าถ่วงน้ำหนักเหล่านี้ก็จะถูกปรับเปลี่ยนไปตามค่าที่ได้จากการคำนวณตามอัลกอริทึมที่ใช้จนกระทั่งได้ค่าที่เหมาะสมที่สุด ซึ่งองค์ประกอบที่สำคัญของการพัฒนาแบบจำลองนิวรอลเน็ตเวิร์กมีดังนี้

- **โครงสร้างของเน็ตเวิร์ก** (Architecture) ประกอบด้วยจำนวนของชั้นภายในเน็ตเวิร์ก จำนวนของโหนดในชั้นที่เป็นอินพุต (Input Layer) และชั้นเอาต์พุต (Output Layer) รวมถึงจำนวนโหนดในชั้นกลางแต่ละชั้น (Hidden Layer) ที่เชื่อมต่ออยู่ระหว่างชั้นอินพุต และชั้นเอาต์พุตด้วย
- **ฟังก์ชันกระตุ้น** (Transfer Function หรือ Activation Function) ที่ใช้คำนวณค่าที่ได้จากแต่ละโหนดภายในเน็ตเวิร์กจากอินพุตจนถึงเอาต์พุต
- **ฟังก์ชันที่ใช้ทดสอบหาค่าความผิดพลาดของผลลัพธ์ที่ได้** (Cost Function) โดยเป็นการเปรียบเทียบค่าผลลัพธ์ที่คำนวณได้จากแบบจำลองกับค่าข้อมูลจริงที่เกิดขึ้น
- **อัลกอริทึมที่ใช้สอนให้เน็ตเวิร์กเกิดการเรียนรู้** (Training Algorithm) โดยมีการกำหนดวิธีการปรับค่าตัวแปรต่างๆ ภายในเน็ตเวิร์ก เพื่อให้ค่าความผิดพลาดที่คำนวณได้จากฟังก์ชันทดสอบความผิดพลาดมีค่าน้อยที่สุด

3.2.1 การคำนวณในโครงสร้างนิวรอลเน็ตเวิร์ก (Neural Computation)

คุณสมบัติที่น่าสนใจของนิวรอลเน็ตเวิร์ก คือ ความสามารถของเน็ตเวิร์กในการเรียนรู้ ซึ่งปกติพฤติกรรม หรือตัวแปรต่างๆ ของเน็ตเวิร์กนั้นจะถูกพัฒนาในระหว่างที่มีกระบวนการเรียนรู้หรือกระบวนการสอน โดยในระหว่างกระบวนการนี้จะมีการคำนวณในลักษณะซ้ำๆ กับข้อมูลที่นำมาใช้เป็นอินพุตหลายๆ ชุด ร่วมกับลักษณะของเอาต์พุตที่เกิดขึ้นสำหรับอินพุตแต่ละชุด โดยเน็ตเวิร์กจะมีการเปลี่ยนแปลงโดยทำการปรับค่าของตัวแปรต่างๆ ภายในโครงสร้างตามอัลกอริทึมที่นำมาใช้ในการเรียนรู้จนกระทั่งสามารถสร้างเอาต์พุตได้ตามที่ออกแบบไว้ได้ หรือภายใต้เงื่อนไขที่ยอมรับได้ ซึ่งกลุ่มข้อมูลอินพุตและเอาต์พุตที่นำมาใช้นี้เรียกว่า กลุ่มข้อมูลที่ใช้สอน (Training Set)

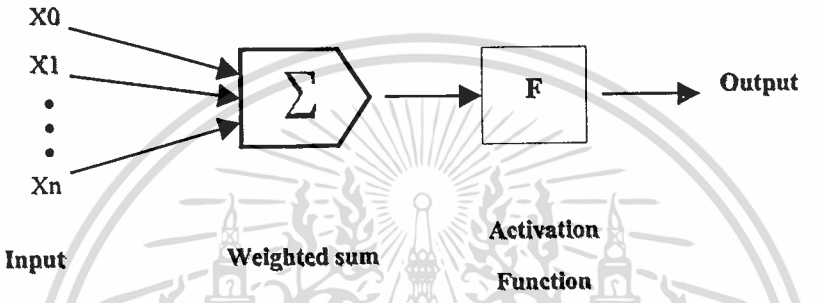
ในการสอนให้เน็ตเวิร์กมีการเรียนรู้นั้นจะเกิดประโยชน์ก็ต่อเมื่อข้อมูลที่นำมาใช้สอนต้องมีลักษณะครอบคลุมรูปแบบที่เป็นไปได้ของข้อมูลให้มากที่สุด ดังนั้นจึงมีความสำคัญมากที่ต้องเลือกข้อมูลที่เป็นตัวแทนของทุกรูปแบบของงานที่จะนำแบบจำลองที่ได้มาใช้ โดยลักษณะ 2 ประการที่ควรต้องคำนึงถึงในการพิจารณาเลือกข้อมูลที่นำมาใช้ในการสอน คือ ข้อมูลที่ใช้สอนต้องเป็นส่วนหนึ่งของข้อมูลที่คาดหวังว่าจะนำแบบจำลองไปใช้ในการประมวลผล และข้อมูลที่ใช้สอนต้องถูกเลือกจากช่วงเวลาที่มีการกระจายของข้อมูลครอบคลุมรูปแบบของอินพุตที่เป็นไปได้ทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมด เนื่องจากเน็ตเวิร์กจะไม่สามารถใช้พยากรณ์ข้อมูลได้อย่างมีประสิทธิภาพ ถ้ามีการสอนด้วยข้อมูลที่มีค่าผิดปกติ (Outlier) จำนวนมาก

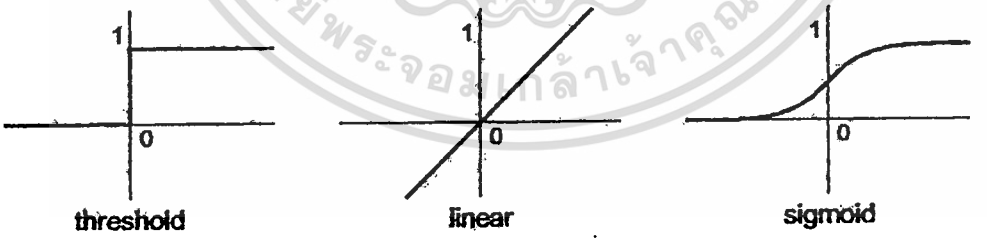
3.2.2 การคำนวณของโหนด (Node หรือ Unit) ในนิวรอลเน็ตเวิร์ก

แบบจำลองอย่างง่ายของการคำนวณในแต่ละโหนดภายในนิวรอลเน็ตเวิร์กแสดงได้ดังรูปที่ 3.3



รูปที่ 3.3 แสดงแบบจำลองของการคำนวณของโหนดภายในโครงข่ายนิวรอลเน็ตเวิร์ก

ในการคำนวณนั้นสามารถแบ่งประเภทของโหนดตามรูปแบบของฟังก์ชันกระตุ้น (Transfer Function) ที่ใช้ได้เป็น 3 แบบ ดังแสดงรูปที่ 3.4 คือ



รูปที่ 3.4 แสดงรูปแบบของฟังก์ชันกระตุ้นที่นำมาใช้ในการคำนวณค่าในโหนดของเน็ตเวิร์ก

<u>แบบที่ 1</u>: Threshold unit: ฟังก์ชันกระตุ้นจะถูกจำกัดค่าเอาต์พุตไว้เป็น 0 หรือ 1 ตามสถานะก่อนหน้า

<u>แบบที่ 2</u>: Linear unit: ใช้ฟังก์ชัน f(h) = h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบที่ 3: Non-linear unit: โดยทั่วไปจะใช้การเรียนรู้แบบ Gradient Descent ซึ่งส่วนมากโหนดประเภทนี้จะใช้ฟังก์ชันกระตุ้นของซิกมอยด์ (Sigmoid Function) เช่น $f(h) = 1/(1+e^{-h})$

โดยทั่วไปฟังก์ชันกระตุ้นที่นิยมใช้มากที่สุด คือฟังก์ชันของซิกมอยด์ เนื่องจากสามารถหาค่า Differentiate และคำนวณค่า Gradient ได้ง่าย ส่วนฟังก์ชันแบบเชิงเส้นนั้นนิยมใช้ในส่วนที่เป็นโหนดของชั้นเอาต์พุตของเน็ตเวิร์กในการนำไปประยุกต์ใช้กับข้อมูลในลักษณะอนุกรมเวลา

3.2.3 อัลกอริทึมที่ใช้ในการสอน (Training Algorithm)

อัลกอริทึมที่ใช้สอนนั้นเป็นการกำหนดวิธีการคำนวณภายในเน็ตเวิร์กเพื่อใช้ในการปรับค่าของตัวแปรภายในโครงสร้างให้ได้แบบจำลองที่เหมาะสมที่สุดสำหรับข้อมูลแต่ละชุด โดยอัลกอริทึมที่ได้รับการยอมรับในการนำมาใช้กับงานด้านการพยากรณ์ข้อมูล ได้แก่ อัลกอริทึมแบบ Backpropagation

ลักษณะของอัลกอริทึมแบบ Backpropagation นี้ เป็นวิธีการที่ทำให้เน็ตเวิร์กมีกระบวนการเรียนรู้ซ้ำๆ จากข้อมูลในอดีต โดยที่เอาต์พุตที่ได้จากเน็ตเวิร์กในแต่ละครั้งที่ป้อนชุดข้อมูลอินพุตให้กับเน็ตเวิร์กนั้นจะถูกนำมาเปรียบเทียบกับเอาต์พุตที่เกิดขึ้นจริงในอดีต เพื่อหาค่าความผิดพลาดสำหรับแต่ละชุดของข้อมูลอินพุต และค่าความผิดพลาดที่ได้นี้จะนำมาใช้ในการคำนวณเพื่อปรับค่าถ่วงน้ำหนักระหว่างแต่ละโหนดในโครงสร้างของเน็ตเวิร์กต่อไป โดยที่การปรับค่าถ่วงน้ำหนักนี้จะมีการคำนวณในทิศทางย้อนกลับจากทางด้านเอาต์พุตผ่านชั้นกลาง (Hidden layer) จนปรับค่ามาถึงชั้นแรกที่ต่อจากชั้นของอินพุต จากนั้นจึงคำนวณค่าของเอาต์พุตสำหรับข้อมูลอินพุตชุดต่อๆ ไปโดยใช้ค่าถ่วงน้ำหนักที่คำนวณใหม่นี้ เพื่อเปรียบเทียบหาค่าความผิดพลาดในลักษณะเช่นเดียวกัน โดยในกระบวนการเรียนรู้จะมีการคำนวณเช่นนี้เรื่อยไปจนกระทั่งค่าความผิดพลาดที่ได้มีค่าน้อยที่สุด หรือเป็นค่าที่ยอมรับได้เมื่อเทียบค่าเอาต์พุตที่คำนวณได้กับค่าที่เกิดขึ้นจริง ซึ่งจากโครงสร้างในรูปที่ 3.2 สามารถสรุปขั้นตอนการคำนวณของอัลกอริทึมได้ดังต่อไปนี้

ขั้นตอนที่ 1: การกำหนดค่าถ่วงน้ำหนักเริ่มต้น (Initial weight)

เป็นขั้นตอนการกำหนดค่าตัวเลขสุ่มที่มีค่าน้อยๆ ให้กับค่าถ่วงน้ำหนัก (Weight) สำหรับการเชื่อมต่อระหว่างแต่ละโหนดจากชั้นอินพุตถึงชั้นเอาต์พุต รวมถึงการสุ่มค่า Bias ให้กับแต่ละโหนดด้วย (ในกรณีที่ต้องใช้)

ขั้นตอนที่ 2: การหาค่าของอินพุตในทิศทางไปข้างหน้า (Propagate the input forward)

ขั้นตอนนี้เป็นการคำนวณค่าอินพุต และค่าเอาต์พุตให้กับแต่ละโหนดในเน็ตเวิร์กโดยข้อมูลในอดีตที่นำมาใช้สอนให้เกิดการเรียนรู้ภายในโครงสร้างนั้นจะถูกป้อนเข้าสู่ชั้นอินพุตของเน็ตเวิร์ก และมีการคำนวณค่าอินพุตรวม และค่าเอาต์พุตของแต่ละโหนดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคำนวณค่าในชั้นอินพุต:

สำหรับค่าเอาต์พุตของโหนด j ใดๆ ในชั้นของอินพุตมีค่าเท่ากับค่าของอินพุตของโหนดนั้นๆ

$$O_j = I_j \qquad \text{.....3.1}$$

โดยที่ O_j = ค่าเอาต์พุตของโหนด j และ

I_j = ค่าอินพุตของโหนด j

การคำนวณค่าในชั้นกลาง (Hidden Layer) และชั้นเอาต์พุต:

สำหรับค่าอินพุตของโหนด j ใดๆที่อยู่ในชั้นกลางหรือชั้นเอาต์พุตจะมีค่าเท่ากับการบวกค่า Bias ของโหนด j กับผลรวมทั้งหมดของการคูณค่าเอาต์พุตของทุกๆ โหนดที่เชื่อมต่อกับโหนด j กับค่าถ่วงน้ำหนักของการเชื่อมต่อนั้นๆ ดังสมการต่อไปนี้

$$I_j = \sum w_{ij} O_i + \theta_j \qquad \text{.....3.2}$$

โดยที่ θ_j = ค่า bias ของโหนด j

w_{ij} = ค่าถ่วงน้ำหนักของการเชื่อมต่อระหว่างโหนด i กับโหนด j

ส่วนค่าของเอาต์พุตสำหรับโหนดใดๆในชั้นกลาง หรือชั้นเอาต์พุตนั้น ได้จากการคำนวณค่าอินพุตรวม สำหรับแต่ละโหนดผ่านฟังก์ชันกระตุ้นที่ใช้ทำการคำนวณผลที่ได้จากโหนดนั้นๆ ซึ่งในอัลกอริทึมนี้จะใช้ฟังก์ชันของซิกมอยด์โดยได้ค่าของเอาต์พุตตามสมการต่อไปนี้

$$Oj = \frac{1}{1 + e^{-Ij}} \qquad \text{.....3.3}$$

<u>ขั้นตอนที่ 3</u>: การคำนวณค่าความผิดพลาดย้อนกลับ (Back-propagate the error)

เมื่อได้ค่าของเอาต์พุตครบทุกๆโหนดแล้ว ก็จะคำนวณค่าความผิดพลาดของเอาต์พุตที่ได้ในทิศทางย้อนกลับเพื่อปรับค่าถ่วงน้ำหนักสำหรับการเชื่อมต่อระหว่างโหนดใดๆในเน็ตเวิร์กรวมทั้งปรับค่า bias ให้กับแต่ละโหนดด้วย

การคำนวณค่าความผิดพลาดในชั้นเอาต์พุต:

สำหรับโหนด k ใดๆในชั้นเอาต์พุต สามารถคำนวณค่าความผิดพลาด (Err_k) ได้ดังสมการต่อไปนี้

$$Err_k = O_k(1 - O_k)(T_k - O_k) \qquad \text{.....3.4}$$

โดยที่ O_k = ค่าเอาต์พุตของโหนด k ที่ได้จากเน็ตเวิร์ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T_k = ค่าเอาต์พุตจริงของโหนด k จากข้อมูลในอดีต

การคำนวณค่าผิดพลาดในชั้นกลาง (Hidden Layer):

ในการคำนวณสำหรับโหนดในชั้นกลางนี้จะมีการนำค่าถ่วงน้ำหนักของการเชื่อมต่อระหว่างโหนด j ใดๆ ที่พิจารณา กับโหนดอื่นๆ ที่อยู่ในลำดับชั้นที่สูงกว่าในเน็ตเวิร์กมาคำนวณร่วมกัน ดังสมการต่อไปนี้

$$Err_j = O_j(1 - O_j)\sum Err_k w_{jk} \quad \text{.....3.5}$$

โดยที่ Err_k = ค่าความผิดพลาดของโหนด k ในลำดับชั้นที่สูงกว่า

w_{jk} = ค่าถ่วงน้ำหนักของการเชื่อมต่อระหว่างโหนด j กับโหนด k ใดๆในชั้นถัดไปที่สูงกว่าในเน็ตเวิร์ก

ค่าถ่วงน้ำหนักของการเชื่อมต่อระหว่างโหนดใดๆ จะถูกปรับเปลี่ยนด้วยค่าความผิดพลาดที่เกิดขึ้นต่อเนื่องกันไปตามสมการต่อไปนี้

$$\Delta w_{ij} = (l) Err_j O_i \quad \text{.....3.6}$$

$$w_{ij} = w_{ij} + \Delta w_{ij} \quad \text{.....3.7}$$

และ ค่า bias ใหม่ของโหนดใดๆ คำนวณได้จาก

$$\Delta \theta_j = (l) Err_j \quad \text{.....3.8}$$

$$\theta_j = \theta_j + \Delta \theta_j \quad \text{.....3.9}$$

โดยที่ Δw_{ij} = ค่าการเปลี่ยนแปลงของ w_{ij}

$\Delta \theta_j$ = ค่าการเปลี่ยนแปลงของ θ_j

l = ค่าอัตราการเรียนรู้ (Learning rate)

สำหรับค่า "l" หมายถึง อัตราการเรียนรู้ (Learning Rate) ของเน็ตเวิร์กซึ่งเป็นค่าที่ใช้ควบคุมความเร็วในการตอบสนองต่อค่าความผิดพลาดที่เกิดขึ้นในแต่ละครั้ง โดยในกรณีที่ "l" มีค่ามากแสดงว่าเน็ตเวิร์กมีการตอบสนองต่อค่าความผิดพลาดมาก โดยการปรับค่าถ่วงน้ำหนักให้เปลี่ยนแปลงไปอย่างชัดเจน แต่ถ้า "l" มีค่าน้อยๆ แสดงว่ามีการปรับค่าถ่วงน้ำหนักอย่างค่อยเป็นค่อยไปจนกระทั่งได้ค่าที่เหมาะสมที่ทำให้เกิดค่าความผิดพลาดของเอาต์พุตน้อยที่สุด

3.2.4 การวัดค่าความผิดพลาด (Error)

การเรียนรู้ของเน็ตเวิร์กนั้นขึ้นอยู่กับค่าความผิดพลาด (Error) ที่วัดได้โดยยึดหลักแนวความคิดที่ว่า ยิ่งค่าความผิดพลาดที่วัดได้ หรือค่าจากฟังก์ชันทดสอบความผิดพลาด (Cost Function) ที่คำนวณได้มีค่าน้อยเท่าไรนั่นหมายความว่าค่าถ่วงน้ำหนักต่างๆ ภายในโครงสร้างของเน็ตเวิร์กยิ่งดีขึ้น หรือมีความเหมาะสมมากขึ้นนั่นเอง โดยการที่มีฟังก์ชันการวัดค่าความผิดพลาดนั้นทำให้เรา

เอกสารนี้มีลิขสิทธิ์ ห้ามนำไปใช้ในเชิงพาณิชย์ ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถปรับปรุงค่าถ่วงน้ำหนักได้ในลักษณะคล้ายๆ กับการค่อยๆไต่ระดับลงมาตามพื้นผิวของหุบเขาเพื่อหาตำแหน่งที่ต่ำที่สุดของหุบเขา หรือ ภาวะที่ค่าความผิดพลาดของเน็ตเวิร์กน้อยที่สุดนั่นเอง

ส่วนมากในการคำนวณค่าความผิดพลาดนั้นมักจะใช้ค่า Mean Squared Error (MSE)โดยสามารถคำนวณได้จากสมการต่อไปนี้

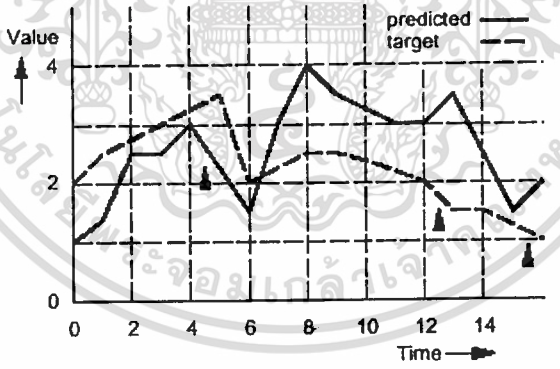
$$MSE = \frac{1}{N}\sum_{i=1}^{n}(O_j - T_j)^2 \qquad \text{.....3.10}$$

โดยที่ N = จำนวนของชุดข้อมูลที่ใช้ในการสอน

O_j = ค่าเอาต์พุตของโหนด j ที่ได้จากเน็ตเวิร์ก

T_j = ค่าเอาต์พุตจริงของโหนด j จากข้อมูลในอดีต

อย่างไรก็ตามถ้าการสอนโดยการวัดค่าความผิดพลาดแบบมาตรฐานที่กล่าวไว้ข้างต้นนั้นสามารถทำได้ช้าหรือไม่เพียงพอก็สามารถใช้การตรวจวัดค่าความผิดพลาดแบบอื่นๆร่วมด้วย หรือใช้แทนกันก็ได้ เช่น การใช้ค่า POCID (Prediction Of Change In Direction Error) โดยที่ POCID Error จะเกิดขึ้นก็ต่อเมื่อค่าของเอาต์พุตจริง และค่าที่คำนวณได้จากเน็ตเวิร์กมีการเปลี่ยนแปลงในทิศทางที่ตรงข้ามกัน เช่น ค่าหนึ่งเพิ่มขึ้นแต่อีกค่ากลับลดลง ดังรูปที่ 3.5



รูปที่ 3.5 แสดงรูปแบบของค่าความผิดผลาดแบบ Prediction Of Charge In Direction

ดังนั้นถ้าค่าของเอาต์พุตจริงมีค่ามากกว่าค่าของเอาต์พุตจริงก่อนหน้าในอนุกรมเวลาแล้วค่าของเอาต์พุตที่คำนวณได้ต้องมีค่าอย่างน้อยเท่ากับค่าเอาต์พุตที่ได้จากการคำนวณค่าก่อนหน้านั้นจึงจะทำให้เกิด POCID Error ขึ้น โดยสมการการคำนวณสามารถแสดงได้ดังนี้

$$POCID = \frac{1}{N}\sum_{i=1}^{n} D_i \qquad \text{.....3.11}$$

โดยที่ $D_i = 1$ ถ้า $(tar(t) - tar(t-1)) * (0(t) - 0(t-1)) < 0$ และ $D_i = 0$ สำหรับกรณีอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความผิดพลาดของ POCID จะอยู่ในช่วง 0 ถึง 1 โดยในการสอนนั้นมีเป้าหมายเพื่อลดค่า POCID Error ให้น้อยที่สุด ค่าความผิดพลาดที่กล่าวถึงข้างต้นนั้นสามารถนำมาใช้ในการสอนเน็ตเวิร์กได้เฉพาะในกรณีที่เอาต์พุตมีการเปลี่ยนแปลงในทิศทางที่ตรงกันข้ามกับค่าจริง (Target) โดยมีค่า D_i เท่ากับ 1 และค่าความผิดพลาดที่ได้ทั้งหมดนั้นจะถูกนำมาใช้คำนวณเพื่อปรับค่าถ่วงน้ำหนักให้เหมาะสมต่อไป

3.2.5 การพิจารณาจำนวนโหนดในชั้นอินพุต (Input Layer) และชั้นกลาง (Hidden Layer)

แม้ว่าจะมีความพยายามในการพัฒนาแบบจำลองและอัลกอริทึมในการเรียนรู้ของนิวรอลเน็ตเวิร์กแล้ว แต่การออกแบบเพื่อให้ได้โครงสร้างของเน็ตเวิร์กที่เหมาะสมกับงานใดๆ นั้น ก็ยังมีปัญหาอยู่บ้าง ในการหาโครงสร้างที่มีขนาดของเน็ตเวิร์กเล็กที่สุด (Parsimonious Network) ที่สามารถสร้างความผิดพลาดของเอาต์พุตสำหรับข้อมูลที่นำมาใช้ในการสอน และรวมถึงค่าความผิดพลาดของข้อมูลที่ไม่ใช่กรณีที่นำมาใช้สอนในระดับที่เหมาะสมและยอมรับได้ (Optimal Error) ซึ่งกฎโดยทั่วไปจะมีการพิจารณาให้ค่าถ่วงน้ำหนักทั้งหมดภายในเน็ตเวิร์กนั้นควรมีจำนวนน้อยกว่า 1/10 ของจำนวนชุดข้อมูลที่นำมาใช้ในการสอน (Training Record)

ในการเลือกจำนวนโหนดของชั้นอินพุตนั้น ถ้าใช้จำนวนอินพุตน้อยเกินไปจะมีผลให้เกิดแบบจำลองที่ไม่ครอบคลุมลักษณะของข้อมูล (Inadequate Modeling) ในขณะที่ถ้าใช้จำนวนอินพุตมากเกินไปก็จะทำให้โครงสร้างของเน็ตเวิร์กมีความซับซ้อนมากเกินไปทำให้เกิดผลกระทบทั้งในเรื่องของเวลาที่ใช้ในการสอนมากขึ้น และมิติของเน็ตเวิร์กที่ซับซ้อนขึ้นด้วย ซึ่งวิธีการหนึ่งที่ถูกนำเสนอเพื่อใช้พิจารณาจำนวนของโหนดในชั้นอินพุตของเน็ตเวิร์ก คือการใช้หลักการของ Auto Correlation โดยการคำนวณค่าสัมประสิทธิ์ระหว่างข้อมูลอินพุตในช่วงเวลาที่เหลื่อมกัน (Lag) เพื่อหาระยะเวลาที่ข้อมูลอินพุตยังมีความสัมพันธ์กันว่ามีช่วงเวลาที่เหลื่อมกันอยู่กี่ช่วงในอนุกรมเวลานั่นเอง

ในส่วนของจำนวนโหนดในชั้นกลางนั้น หลังจากที่พิจารณาหาจำนวนโหนดในชั้นอินพุตแล้วก็อาจจะใช้การทดลองซ้ำๆ กับกลุ่มข้อมูลที่นำมาใช้ในการสอนโดยการเปลี่ยนจำนวนโหนดของชั้นกลางสำหรับการทดลองในแต่ละครั้งเพื่อหาจำนวนโหนดที่เหมาะสมที่สุด นอกจากนี้อาจพิจารณาได้จากกฎของ Baum-Hausster ซึ่งระบุวิธีการหาจำนวนของโหนดในชั้นกลางของเน็ตเวิร์กไว้ดังสมการต่อไปนี้

$$N_{hidden} = \frac{N_{train}E_{tolerance}}{N_{pts} + N_{output}} \qquad .....3.12$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ N_{hidden} = จำนวนโหนดในชั้นกลาง

N_{train} = จำนวนของชุดข้อมูลที่ใช้ในการสอน

$E_{tolerance}$ = ความแปรปรวนของค่าความผิดพลาด (error tolerance) โดยทั่วไปจะพิจารณาที่ 0.01

N_{pts} = จำนวนข้อมูลในแต่ละชุดข้อมูลที่นำมาสอน

N_{output} = จำนวนของโหนดในชั้นเอาต์พุต

ในการเลือกโครงสร้างของเน็ตเวิร์กสามารถพิจารณาจากแนวทางต่างๆ ได้ 3 แบบ คือ

แบบที่ 1: Ad hoc approach เป็นการออกแบบโครงสร้างโดยอาศัยพื้นฐานจากประสบการณ์การทดลองที่มีมาในครั้งก่อนๆ

แบบที่ 2: Dynamic approach เป็นขั้นตอนการเลือกแบบอัตโนมัติซึ่งมีแนวคิดว่าให้มีการปรับเพิ่มหรือลดจำนวนตัวแปรต่างๆในเน็ตเวิร์กลงในระหว่างกระบวนการสอน โดยพิจารณาจากค่าความผิดพลาดของระบบ เช่น อัลกอริทึมแบบ Cascade Correlation

แบบที่ 3: Distribution approach เป็นแนวคิดที่พยายามค้นหาหรือตั้งสมมติฐานของ Distribution functionในการประมาณและหาวิธีการทางสถิติเพื่อวัดประสิทธิภาพของเน็ตเวิร์ก

3.3 ประเภทของนิวรอลเน็ตเวิร์ก

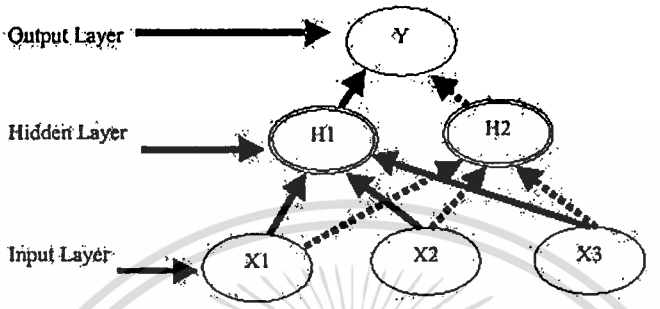
จากการพัฒนาอย่างต่อเนื่องในช่วงที่ผ่านมามีนิวรอลเน็ตเวิร์กหลายประเภทที่ถูกนำมาประยุกต์ใช้กับงานด้านการพยากรณ์ ซึ่งโครงสร้างของนิวรอลเน็ตเวิร์กเหล่านี้ก็มีความแตกต่างกันในเรื่องต่างๆ เช่น ประเภทของโหนดที่นำมาใช้ในเน็ตเวิร์ก ลักษณะโครงสร้างของเน็ตเวิร์ก และอัลกอริทึมที่นำมาใช้ในการเรียนรู้ภายในเน็ตเวิร์ก เป็นต้น โดยในส่วนต่อไปนี้จะอธิบายถึงรูปแบบโครงสร้างของนิวรอลเน็ตเวิร์กแต่ละประเภทที่แตกต่างกัน พร้อมทั้งผลกระทบในแง่ประสิทธิภาพของเน็ตเวิร์กอย่างสังเขป ซึ่งโครงสร้างหลักๆ ของนิวรอลเน็ตเวิร์กนั้นมีอยู่ 2 รูปแบบ คือ แบบที่มีการคำนวณส่งสัญญาณไปข้างหน้า (Feed Forward Neural Network) และแบบที่มีการป้อนกลับ (Recurrent Neural Network)

3.3.1 โครงสร้างแบบส่งสัญญาณไปข้างหน้า (Feed Forward Neural Network)

โครงสร้างแบบที่มีการคำนวณสัญญาณไปข้างหน้า หรือ Feed Forward Network นั้นเป็นโครงสร้างที่มีการจัดเน็ตเวิร์กในลักษณะที่เป็นลำดับชั้น (Layer) ดังแสดงตัวอย่างในรูปที่ 3.6 โดยภายในเน็ตเวิร์กนั้น ประกอบด้วยกลุ่มของอินพุตที่มีหน้าที่ส่งสัญญาณอินพุตต่อไปให้กับเน็ตเวิร์กในส่วนที่เหลือ ซึ่งหลังจากชั้นแรกที่เป็นอินพุตนี้อาจมีชั้นกลางถัดไปอีกหนึ่ง หรือหลายๆ ชั้นก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และชั้นสุดท้ายจะเป็นชั้นของเอาต์พุต ซึ่งเป็นค่าผลลัพธ์ของการคำนวณจากโหนดต่างๆ ภายในเน็ตเวิร์กทุกโหนด โดยในการคำนวณของโหนดต่างๆ ในแต่ละชั้นนั้นจะส่งค่าที่ได้ต่อไปให้กับโหนดที่อยู่ในชั้นถัดไปเท่านั้น



รูปที่ 3.6 แสดงโครงสร้างของนิวรอลเน็ตเวิร์กแบบส่งสัญญาณไปข้างหน้า

3.3.2 โครงสร้างแบบมีการป้อนกลับ (Recurrent Neural Network: RNN) สามารถแบ่งได้เป็น 2 ประเภท คือ

3.3.2.1 โครงสร้างแบบมีการป้อนกลับอย่างง่าย (Simple Recurrent Network: SRN)

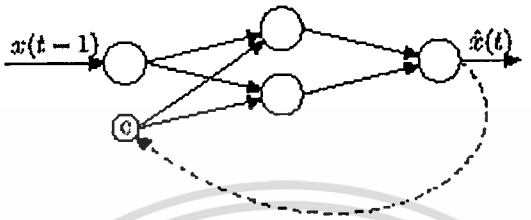
รูปแบบโครงสร้างของเน็ตเวิร์กแบบนี้นั้นจะใกล้เคียงกับลักษณะของโครงสร้างแบบส่งสัญญาณไปข้างหน้า แต่จะมีส่วนที่เพิ่มขึ้นมาคือมีการป้อนกลับ (Feedback Connection) จากบางชั้นของเน็ตเวิร์กมายังชั้นในลำดับก่อนหน้าเพื่อช่วยทำให้เน็ตเวิร์กสามารถเรียนรู้ และสร้างแบบจำลองที่มีความเหมาะสมกับข้อมูลในลักษณะที่เป็นอนุกรมเวลาได้มากขึ้น

โครงสร้างแบบมีการป้อนกลับนี้ได้เริ่มมีการศึกษาครั้งแรกโดย Jordan ซึ่งแสดงโครงสร้างได้ดังรูปที่ 3.7 โดยจะเห็นได้ว่าส่วนที่เป็นอินพุตของเน็ตเวิร์กนี้จะรับค่าทั้งจากอินพุตภายนอกเน็ตเวิร์ก (External Input) และจากอินพุตที่แสดงสถานะภายในของโหนดภายในเน็ตเวิร์ก (State Unit)โดยข้อมูลในส่วนหลังนี้จะเป็นอินพุตที่รับข้อมูลมาจากการป้อนกลับค่าจากชั้นเอาต์พุตของเน็ตเวิร์ก

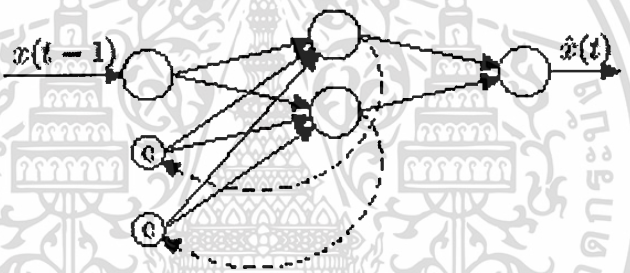
ในปัจจุบันได้มีการพัฒนาโครงสร้างอีกรูปแบบหนึ่ง คือ โครงสร้างแบบ Elman ดังแสดงไว้ในดังรูปที่ 3.8 ซึ่งโครงสร้างนี้เป็นที่ยอมรับโดยทั่วไปว่าเป็นโครงสร้างแบบหนึ่งที่นำไปประยุกต์ใช้งานได้ง่าย และมีประสิทธิภาพ โดยในส่วนของชั้นอินพุตของโครงสร้างจะประกอบด้วยค่าที่ได้จากค่าเอาต์พุตจากชั้นกลางของการคำนวณในครั้งที่ผ่านมา โดยเก็บไว้ในส่วนของชั้น Context หรือ State Unit ที่ทำหน้าที่เสมือนเป็นชั้นความจำระยะสั้นของสถานะภายในเน็ตเวิร์ก (Short Term

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Memory: STM) ซึ่งวิธีการง่ายๆ ในการเพิ่มความจำระยะสั้นให้กับโครงสร้างแบบ Elman คือการเพิ่มจำนวนโหนดในชั้นกลางของเน็ตเวิร์กนั่นเอง แต่ก็จะส่งผลให้การเรียนรู้มีความซับซ้อน และใช้เวลามากขึ้นด้วย



รูปที่ 3.7 แสดงโครงสร้างของนิวรอลเน็ตเวิร์กแบบป้อนกลับอย่างง่าย - Jordan Network



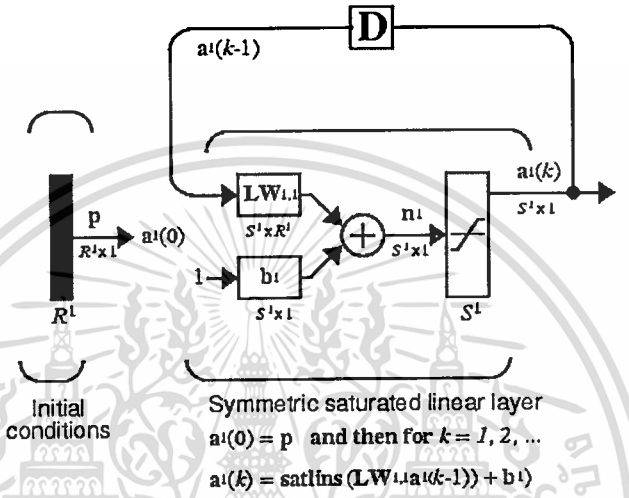
รูปที่ 3.8 แสดงโครงสร้างนิวรอลเน็ตเวิร์กแบบป้อนกลับอย่างง่าย - Elman Network

3.3.2.2 โครงสร้างแบบมีการป้อนกลับแบบสมบูรณ์ (Fully Recurrent Neural Network)

ความแตกต่างระหว่างโครงสร้างแบบมีการป้อนกลับอย่างง่าย และโครงสร้างที่มีการป้อนกลับแบบสมบูรณ์ (Fully Recurrent Network) คือ โครงสร้างที่มีการป้อนกลับแบบสมบูรณ์นั้นจะมีการเชื่อมต่อในทุกรูปแบบทั้งแบบทางตรง (Direct loop) และทางอ้อม (Indirect loop) และค่าถ่วงน้ำหนักของการเชื่อมต่อระหว่างแต่ละโหนดภายในเน็ตเวิร์กนี้ สามารถมีการเรียนรู้ได้อย่างเป็นอิสระ ดังนั้นจึงทำให้การสอนเน็ตเวิร์กนั้นทำได้ยากและต้องใช้เวลานานขึ้น แต่อย่างไรก็ตามในกรณีที่โครงสร้างแบบส่งสัญญาณไปข้างหน้านั้นไม่สามารถให้ผลลัพธ์ตามที่ต้องการได้อย่างเหมาะสมแล้ว โครงสร้างที่มีการป้อนกลับแบบนี้ก็เป็นอีกทางเลือกหนึ่งที่สามารถนำไปใช้ได้ ตัวอย่างของโครงสร้างแบบนี้ได้แก่ โครงสร้างของ Hopfield Network ดังแสดงในรูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในโครงการพัฒนาระบบงานนี้จะใช้หลักการของนิวรอลเน็ตเวิร์กแบบที่มีการป้อนกลับอย่างง่ายโดยใช้โครงสร้างแบบ Elman ในการสร้างระบบเพื่อใช้พยากรณ์ราคาหลักทรัพย์ โดยที่มีอินพุตเป็นค่าของราคาหลักทรัพย์ของช่วงเวลาในอดีต และมีค่าเอาต์พุตเป็นราคาหลักทรัพย์ในช่วงเวลาถัดไปในอนุกรมเวลา ซึ่งจะได้อธิบายขั้นตอนในการสร้างเน็ตเวิร์กไว้ในส่วนต่อไป



รูปที่ 3.9 แสดงโครงสร้างนิวรอลเน็ตเวิร์กแบบมีการป้อนกลับอย่างสมบูรณ์ - Hopfield Network

3.4 การประยุกต์ใช้นิวรอลเน็ตเวิร์กแบบ Elman ในการพัฒนาระบบพยากรณ์ราคาหลักทรัพย์

3.4.1 โครงสร้างของเน็ตเวิร์กแบบ Elman

รูปแบบของนิวรอลเน็ตเวิร์กแบบป้อนกลับที่นำมาใช้ในการพัฒนาระบบงานพยากรณ์ราคาหลักทรัพย์ คือ โครงสร้างแบบ Elman ซึ่งจะใช้อัลกอริทึมแบบ Backpropagation ในการสอนให้เน็ตเวิร์กมีการเรียนรู้ โดยโครงสร้างของเน็ตเวิร์กแบบ Elman นั้น ประกอบด้วยชั้นต่างๆ ดังนี้

- ชั้นของอินพุต (Input Layer)
- ชั้นของการป้อนกลับ (Context หรือ Copy Layer)
- ชั้นกลางที่เชื่อมระหว่างชั้นอินพุต และเอาต์พุต (Hidden Layer)
- ชั้นของเอาต์พุต (Output Layer)

โดยชั้นของการป้อนกลับ หรือชั้น Context (Context Layer) นั้น จะเป็นกลุ่มของโหนดที่รับค่าที่ป้อนกลับมาจากชั้นกลางซึ่งเป็นชั้นในลำดับที่สูงกว่า หรือเป็นชั้นที่อยู่ใกล้กับชั้นเอาต์พุตมากกว่านั่นเอง โดยในชั้น Context จะทำหน้าที่รับข้อมูลของค่าจากฟังก์ชันกระตุ้นของโหนดภายในในครั้งก่อนหน้าส่งให้กับเน็ตเวิร์ก อย่างไรก็ตามค่าถ่วงน้ำหนักของการป้อนกลับจะถูกกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นค่าที่แน่นอนไว้เท่ากับ 1 โดยมีเพียงค่าถ่วงน้ำหนักของการเชื่อมต่อในทิศทางไปข้างหน้า (Forward Connection) เท่านั้น ที่ได้รับการปรับเปลี่ยนในช่วงที่มีการเรียนรู้ ซึ่งจะใช้อัลกอริทึมมาตรฐานแบบ Backpropagation ตามวิธีการที่ได้อธิบายไว้ข้างต้นในการปรับโครงสร้างของเน็ตเวิร์ก โดยขั้นตอนการคำนวณตามหลักการของ Elman จะได้อธิบายในส่วนต่อไป

สำหรับโครงสร้างของเน็ตเวิร์กแบบ Elman ที่นำมาใช้นั้น เรียกว่าเป็นโครงสร้างที่มีการป้อนกลับบางส่วน (Partial Recurrent Network) ประเภทหนึ่งที่ประกอบด้วยชั้นกลาง 1 ชั้น โดยที่เอาต์พุตของชั้นกลางนี้จะถูกป้อนกลับมาเป็นอินพุตให้กับชั้นนั้นๆ โดยผ่านชั้น Context โดยแต่ละโหนดในชั้นกลางจะต่อกับโหนด 1 โหนดในชั้น Context โดยผ่านการเชื่อมต่อที่มีค่าถ่วงน้ำหนักเป็นค่าคงที่เท่ากับ 1 ซึ่งชั้น Context นี้จะเป็นส่วนประกอบของเน็ตเวิร์กที่บันทึกและจดจำสถานะของชั้นกลางจากการคำนวณในครั้งที่ผ่านมา ดังนั้นจำนวนของโหนดในชั้น Context จะเท่ากับจำนวนโหนดในชั้นกลาง โดยทุกๆ โหนดในชั้นกลางจะรับค่าจากเอาต์พุตของชั้น Context เข้ามาเป็นอินพุตของชั้นกลางนั้นๆ ด้วยนอกเหนือจากการรับอินพุตที่เป็นค่าที่ป้อนมาจากภายนอกของเน็ตเวิร์ก

ในแต่ละครั้งของการสอนให้เน็ตเวิร์กมีการเรียนรู้นั้นจะมีการบันทึกค่าเอาต์พุตจากโหนดในชั้นกลางไปยังส่วนที่เป็นชั้น Context เพื่อใช้เป็นส่วนหนึ่งของอินพุตในการสร้างแบบจำลองให้กับระบบโดยมีขั้นตอนคำนวณโดยสรุปดังนี้

ขั้นตอนที่ 1: นำค่าข้อมูลอินพุต สำหรับช่วงเวลาหนึ่งเข้าสู่แต่ละโหนดในชั้นอินพุต

ขั้นตอนที่ 2: คำนวณค่าตามฟังก์ชันกระตุ้นของโหนดในชั้นกลาง โดยใช้อินพุตจากโหนดในชั้นอินพุต และอินพุตจากโหนดในชั้น Context

ขั้นตอนที่ 3: คำนวณค่าตามฟังก์ชันกระตุ้นของโหนดในชั้นเอาต์พุต

ขั้นตอนที่ 4: บันทึกค่าใหม่ของโหนดในชั้นกลางไปยังชั้น Context

ขั้นตอนที่ 5: นำข้อมูลที่ใช้ในการสอนชุดต่อๆ ไป เข้าสู่โหนดในชั้นอินพุต แล้วทำการคำนวณซ้ำตามขั้นตอนที่ 1-4 จนกระทั่งครบทุกชุดข้อมูลที่นำมาใช้สอนเน็ตเวิร์ก

ในแต่ละช่วงเวลาที่มีอินพุตเข้าสู่เน็ตเวิร์ก แต่ละโหนดในเน็ตเวิร์กจะทำการคำนวณค่าตามฟังก์ชันกระตุ้นในลักษณะเช่นเดียวกับโครงสร้างแบบส่งสัญญาณไปข้างหน้า อย่างไรก็ตามอินพุตโดยรวมของระบบจะประกอบด้วยส่วนที่ป้อนสถานะของเน็ตเวิร์กซึ่งเป็นค่าเอาต์พุตจากโหนดในชั้นกลางร่วมกับส่วนของอินพุตชุดใหม่ที่ป้อนให้กับเน็ตเวิร์ก ดังนั้นในแต่ละครั้งเอาต์พุตที่ได้จะเป็นฟังก์ชันของลักษณะทุกอย่างที่เน็ตเวิร์กเคยเจอมาก่อนหน้าแล้ว และถึงแม้ว่าการเพิ่มการป้อนกลับให้กับเน็ตเวิร์กนั้นเป็นการช่วยปรับปรุงประสิทธิภาพของการพยากรณ์ค่าข้อมูลในลักษณะที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นอนุกรมเวลาได้ แต่อย่างไรก็ตามการกำหนดจำนวนอินพุต (Window of time delay input) และการประมวลผลข้อมูลเบื้องต้นอย่างเหมาะสมก็นับว่ามีความสำคัญมากต่อการสร้างแบบจำลอง

3.4.2 ขั้นตอนการพัฒนานิวรอลเน็ตเวิร์กสำหรับระบบพยากรณ์ราคาหลักทรัพย์

ในการสร้างแบบจำลองเพื่อนำไปใช้ในการพยากรณ์ราคาหลักทรัพย์นั้น สามารถแบ่งกระบวนการหลักออกได้เป็นขั้นตอนต่างๆ ดังนี้

ขั้นตอนที่ 1: กำหนดตัวแปรของข้อมูลที่ต้องการพยากรณ์ซึ่งอาจจะพิจารณาข้อมูลของตัวแปรเพียงตัวเดียว หรือนำตัวแปรหลายๆ ตัวมาพิจารณาร่วมกันก็ได้

ขั้นตอนที่ 2: รวบรวมข้อมูลที่เกี่ยวข้องในลักษณะที่เป็นอนุกรมเวลา (Time-series) โดยจำนวนของข้อมูลต้องมากพอที่จะสอนให้เน็ตเวิร์กเกิดการเรียนรู้ได้อย่างถูกต้อง

ขั้นตอนที่ 3: กรองข้อมูล หรือประมวลผลข้อมูลเบื้องต้น เพื่อลดความแปรปรวนที่ผิดปกติของข้อมูลซึ่งอาจทำให้ได้ประโยชน์จากการใช้ข้อมูลบางตัวมากขึ้น หรือเพื่อให้สามารถนำข้อมูลไปใช้ในการสร้างแบบจำลองได้อย่างเหมาะสม หลังจากนั้นจึงแปลงข้อมูลให้อยู่ในรูปแบบที่เหมาะสมกับฟังก์ชันกระตุ้นที่ใช้ภายในเน็ตเวิร์ก

ขั้นตอนที่ 4: แบ่งข้อมูลออกเป็น 2 ส่วน คือ ข้อมูลที่ใช้ในการสอนให้เน็ตเวิร์กเกิดการเรียนรู้ (Training Data) และข้อมูลที่ใช้ทดสอบความถูกต้องของแบบจำลอง (Testing Data)

ขั้นตอนที่ 5: เลือกโครงสร้างที่เหมาะสมโดยกำหนดจำนวนชั้น (Layer) ภายในเน็ตเวิร์ก และกำหนดจำนวนโหนดในแต่ละชั้นของเน็ตเวิร์ก

ขั้นตอนที่ 6: เลือกอัลกอริทึมที่นำมาใช้สอนให้นิวรอลเน็ตเวิร์กเกิดการเรียนรู้

ขั้นตอนที่ 7: เริ่มสอนเน็ตเวิร์กโดยใช้ข้อมูลจากช่วงที่ผ่านมาในอดีตเพื่อสร้างให้เกิดการเรียนรู้ตามรูปแบบของอัลกอริทึมที่กำหนดไว้ แล้วทำการปรับค่าถ่วงน้ำหนักที่เชื่อมอยู่ระหว่างแต่ละโหนดให้เหมาะสมมากขึ้น

ขั้นตอนที่ 8: ทำการทดสอบแบบจำลองโดยใช้ข้อมูลที่จักแบ่งไว้ในขั้นตอนที่ 5 เพื่อทำการพิจารณาความถูกต้องของแบบจำลองที่สร้างได้ ซึ่งถ้าค่าข้อมูลที่ได้จากการทดสอบมีความผิดพลาดสูงก็อาจจะต้องพิจารณาความเหมาะสมของข้อมูลที่ใช้ในการสอนเน็ตเวิร์ก รวมถึงค่าของตัวแปรอื่นๆ ที่ใช้ในการกำหนดให้เน็ตเวิร์กเกิดการเรียนรู้แล้วทำการปรับค่าเหล่านั้นให้เหมาะสม เพื่อทำให้การสอนครั้งต่อไปมีความถูกต้องมากยิ่งขึ้น

ขั้นตอนที่ 9: การนำแบบจำลองของเน็ตเวิร์กที่ได้ไปใช้งานโดยการกำหนดค่าข้อมูลอินพุตในช่วงปัจจุบันเพื่อพยากรณ์ค่าข้อมูลในช่วงเวลาถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การพัฒนาระบบพยากรณ์ราคาหลักทรัพย์

จากทฤษฎีของดาต้าไมนิ่ง และหลักการของนิวรอลเน็ตเวิร์ก ที่กล่าวไว้ในบทที่ 2 และ 3 นั้น สามารถนำมาใช้ในการพัฒนาระบบเพื่อพยากรณ์ราคาหลักทรัพย์ได้ ซึ่งในโครงการพัฒนาระบบนี้ เป็นการนำดาต้าไมนิ่งมาใช้ในลักษณะของงานด้านการพยากรณ์ค่าข้อมูล (Value Prediction) โดยใช้อัลกอริทึมของนิวรอลเน็ตเวิร์กแบบป้อนกลับอย่างง่ายตามโครงสร้างของ Elman (Simple Recurrent Neural Network: Elman Network) ในการสร้างแบบจำลองให้กับระบบโดยใช้ภาษา Delphi ในการพัฒนาระบบ

การทำงานของระบบนั้นสามารถแบ่งได้เป็น 2 แบบ คือ การสร้างแบบจำลองใหม่ให้กับระบบ (New model) และการเรียกแบบจำลองที่สร้างไว้แล้วมาใช้งานเพื่อพยากรณ์ค่าข้อมูล (Open model) ซึ่งในบทนี้จะอธิบายถึงองค์ประกอบของระบบ อัลกอริทึมที่สำคัญภายในระบบ รายละเอียดการทำงาน รวมถึงขั้นตอนการใช้งานระบบที่สร้างขึ้น

4.1 องค์ประกอบของระบบพยากรณ์ราคาหลักทรัพย์

4.1.1 ส่วนสร้างแบบจำลองใหม่ (New Model)

4.1.1.1 ส่วนรับข้อมูล: เป็นการเลือกไฟล์ข้อมูลที่จะนำมาใช้ในการสร้างแบบจำลองให้กับระบบโดยเป็นข้อมูลที่อยู่ในรูปของ Text File (*.txt)

4.1.1.2 ส่วนประมวลผลข้อมูลเบื้องต้น: เป็นการตรวจสอบข้อมูลที่นำมาใช้สร้างแบบจำลอง สามารถแบ่งการทำงานได้เป็น 2 ขั้นตอน คือ

ขั้นตอนที่ 1: การจัดการกับข้อมูลที่ขาดหายไป (Missing Value)

สำหรับในระบบนี้จะมีวิธีการจัดการกับข้อมูลที่ขาดหายไปใน 4 รูปแบบ โดยให้ผู้ใช้งานเลือกตามความเหมาะสม คือ

1. การลบข้อมูล ณ ช่วงเวลาที่หายไปออกจากอนุกรมเวลาแล้วเลื่อนข้อมูลถัดไปเข้ามาแทน

2. การแทนค่าข้อมูล ณ ช่วงเวลาที่หายไปโดยมีวิธีการคำนวณค่าที่นำมาแทนใน 3 รูปแบบ คือ

- ค่าเฉลี่ยของข้อมูลที่ใกล้เคียงกับตำแหน่งข้อมูลที่หายไปในอนุกรมเวลา
- ค่า Moving Average ของข้อมูลในช่วงเวลาก่อนหน้าของอนุกรมเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ค่าจากสมการแสดงความถดถอยแบบเชิงเส้นของข้อมูลในช่วงเวลาก่อนหน้าในอนุกรมเวลา

ขั้นตอนที่ 2: การแปลงค่าข้อมูล (Transformation)

เนื่องจากข้อมูลที่จะนำมาใช้ในแบบจำลองนิวรอลเน็ตเวิร์กนั้นต้องอยู่ในขอบเขตที่เหมาะสมตามลักษณะของฟังก์ชันกระตุ้นที่นำมาใช้ในแบบจำลอง สำหรับในระบบนี้จะใช้ฟังก์ชันของซิกมอยด์ (Sigmoid Function) ในการคำนวณ ดังนั้นจึงต้องทำการแปลงค่าของข้อมูลให้อยู่ในช่วง 0 ถึง 1 โดยเลือกใช้วิธีการแปลงค่าข้อมูลแบบเชิงเส้น

4.1.1.3 ส่วนการสร้างแบบจำลอง: ใช้อัลกอริทึมของนิวรอลเน็ตเวิร์กแบบ Elman ในการสอนให้ระบบเกิดการเรียนรู้ และมีการสร้างแบบจำลองขึ้น โดยผู้ใช้สามารถกำหนดตัวแปรบางตัวที่ใช้ควบคุมการเรียนรู้ได้ เช่น จำนวนโหนดในชั้นอินพุต (Input unit) จำนวนโหนดในชั้นกลาง (Hidden unit) จำนวนรอบของการเรียนรู้ และค่าความผิดพลาดที่ยอมรับได้ เพื่อช่วยให้ระบบมีการเรียนรู้อย่างมีประสิทธิภาพมากขึ้น

สำหรับการสร้างแบบจำลองของระบบนั้นประกอบด้วยขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1: การสอนให้ระบบเกิดการเรียนรู้ (Training Phase) เป็นการนำข้อมูลแต่ละชุดที่ใช้สอนระบบมาทำการคำนวณตามอัลกอริทึมแบบ Elman เพื่อปรับตัวแปรภายในโครงข่ายจนกระทั่งได้แบบจำลองที่เหมาะสม โดยระบบจะใช้ค่าความผิดพลาด (Mean square error) ตามที่ผู้ใช้กำหนดไว้เป็นเงื่อนไขในการจบการเรียนรู้ของแบบจำลองโดยที่จำนวนรอบในการเรียนรู้ยังไม่เกินค่าที่กำหนดไว้ หรืออาจกำหนดเงื่อนไขการจบการเรียนรู้โดยพิจารณาจากจำนวนรอบของการเรียนรู้ก็ได้

ขั้นตอนที่ 2: การทดสอบแบบจำลองที่ได้จากการสอน (Testing Phase) เป็นการประเมินแบบจำลองโดยใช้ข้อมูลส่วนที่เก็บไว้เพื่อทดสอบ (Test set) โดยในการทำงานขั้นตอนนี้จะเป็นการนำข้อมูลเข้าสู่แบบจำลองที่ได้จากกาสอน แล้วทำการคำนวณหาเอาต์พุตของโครงข่ายเพื่อเปรียบเทียบกับค่าจริงที่เกิดขึ้น แล้วหาค่าความผิดพลาดเพื่อประเมินประสิทธิภาพของแบบจำลองที่ได้

หลังจากการทดสอบแล้ว ถ้าประสิทธิภาพของระบบเป็นที่น่าพอใจผู้ใช้สามารถบันทึกแบบจำลองนี้เก็บไว้เพื่อนำไปใช้ในการพยากรณ์ได้ต่อไป

4.1.2 ส่วนการใช้งานแบบจำลอง (Load Model) เป็นการเรียกแบบจำลองที่สร้างไว้แล้วมาใช้งานซึ่งมีขั้นตอนหลักๆ ดังนี้

4.1.2.1การเลือกแบบจำลอง: โดยเลือกไฟล์ที่เก็บไว้ในรูปแบบ *.mod ตามความเหมาะสมกับข้อมูลที่จะนำมาพยากรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2.2การรับข้อมูล: เป็นส่วนที่รับข้อมูลอินพุตจากผู้ใช้ตามจำนวนของข้อมูลในชั้นอินพุตที่ใช้สร้างแบบจำลองให้กับระบบ

4.1.2.3การประมวลผล: เป็นการคำนวณหาค่าเอาต์พุตโดยใช้ค่าตัวแปรของแบบจำลองคำนวณร่วมกับค่าอินพุตที่ป้อนจากผู้ใช้งาน ตามลำดับชั้นของโครงสร้างเน็ตเวิร์กภายในระบบ

4.2 อัลกอริทึมภายในระบบ

สำหรับอัลกอริทึมของการทำงานหลักๆ ของระบบสามารถแสดงได้ดังต่อไปนี้

อัลกอริทึม 4.2.1: Pre-Processing ทำการตรวจสอบความสมบูรณ์ของข้อมูลที่นำมาใช้สร้างแบบจำลองให้กับระบบ มีการทำงานดังนี้

อัลกอริทึม 4.2.1.1: Data Cleaning ทำการจัดการกับข้อมูลบางส่วนที่ขาดหายไป

```
Step 1: Read data (z_price) from file
Step 2: Calculate detail of z_price: MinValue, MaxValue, AvgValue, No. of
        TotalRecord, and No. of MissingValue
Step 3: If MissingValue <> 0 then do Step4
Step 4: Select method for pre-processing
    Step 4.1: If iMethod=1 do Deleted all MissingValue
    Step 4.2: If iMethod=2 do
            Read window size (z) for Average value calculation
            Calculate Average Value
            Replace MissingValue with Average Value
    Step 4.3: If iMethod=3 do
            Read window size (z) for MA (Moving Average) calculation
            Calculate MAValue by equation 2.1
            Replace MissingValue with MAValue
    Step 4.4: If iMethod=4 do
            Read window size (z) for Regression model calculation
            Calculate Regression model by equation 2.2-2.4
            Replace MissingValue with value from regression equation
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัลกอริทึม 4.2.1.2: Data Transformation (Encode, Decode) ทำการแปลงค่าข้อมูลให้เหมาะสมกับอัลกอริทึมที่ใช้สร้างแบบจำลอง (Encode) และ แปลงค่าข้อมูลจากแบบจำลองกลับเป็นค่าจริง (Decode)

Encode:

Step 1: Read z_price data

Step 2: for all New_z_price do

Calculate encoded data by equation 2.5

Decode:

Step 1: Read z_price data

Step 2: for all New_z_price do

Calculate decoded data by equation 2.5

อัลกอริทึม 4.2.2: Training ทำการสร้างแบบจำลองให้กับระบบโดยใช้นิวรอลเน็ตเวิร์กแบบป้อนกลับอย่างง่ายของ Elman (Simple Recurrent Neural Network: Elman Network)

Step 1: Identify no. of input, no. of hidden node, error (Err), and maximum cycle (Cycle)

Step 2: Initialize all weights in network

Step 3: While terminate condition is not satisfied (ErrAvg < Err or CycleCount > Cycle)

Do Step 3.1 to 3.8

Step 3.1: For each training sample S in samples do

// Propagate the input forward

Step 3.2: For each hidden layer units do

For first training sample S do

$$I_j = \sum w_{ij} O_i$$

For other training sample S do

$$I_j = \sum w_{ij} O_i + \sum w_{(j-1)j} O_{j-1}$$

$$O_j = 1/(1+e^{-Ij})$$

Step 3.3: For each output layer units k do

$$I_k = \sum w_{jk} O_j$$

$$O_k = I_k$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

//Backpropagate errors

Step 3.4: Calculate Error for each output layer unit k

$$Err_k = O_k(1-O_k)(T_k-O_k)$$

Step 3.5: Calculate Error for each hidden layer units j

$$Err_j = O_j(1-O_j)\sum Err_k w_{jk}$$

//Update weight

Step 3.6: for each hidden layer weight (w_{ij}) do

$$\Delta w_{ij} = (l)Err_j O_i$$

$$w_{ij} = w_{ij}+\Delta w_{ij}$$

Step 3.7: for each output layer weight (w_{jk}) do

$$\Delta w_{jk} = (l)Err_k O_j$$

$$w_{jk} = w_{jk}+\Delta w_{jk}$$

Step 3.8: Increase CycleCount and check for terminate condition

อัลกอริทึม 4.2.3: Testing ทำการทดสอบแบบจำลองที่ได้จากการสร้างโดยใช้อัลกอริทึมของนิวรอลเน็ตเวิร์กแบบ Elman

Step 1: For each testing sample S in samples do

Step 2: For each hidden layer units do

$$I_j = \sum w_{ij} O_i$$

$$O_j = 1/(1+e^{-Ij})$$

Step 3: For each output layer units k do

$$I_k = \sum w_{jk} O_j$$

$$O_k = I_k$$

Step 4: Calculate Error for each output layer unit k

Step 5: Calculate Average Error for all samples

อัลกอริทึม 4.2.4: Calculation ทำการคำนวณค่าเอาต์พุตจากแบบจำลองที่เลือกมาใช้

Step 1: Read network parameter from model file (InputUnit, MinValue, MaxValue, All weights in network)

Step 2: For i:=1 to InputUnit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Read Input data from user

Step 3: Calcualte Result

For each hidden layer units do

$$I_j = \sum w_{ij} O_i$$

$$O_j = 1/(1+e^{-Ij})$$

For each output layer units k do

$$I_k = \sum w_{jk} O_j$$

$$O_k = I_k$$

Step 4: Report calculated value

4.3 การใช้งานระบบพยากรณ์ราคาหลักทรัพย์

สำหรับขั้นตอนการใช้งานระบบพยากรณ์ราคาหลักทรัพย์สามารถแสดงตัวอย่างได้ดังรูปที่ 4.1 ถึง 4.14 โดยมีวิธีการใช้งานโดยสรุปดังต่อไปนี้

4.3.1 การสร้างแบบจำลองใหม่

ขั้นตอนที่ 1: เลือกไฟล์ข้อมูลที่ใช้ในการสร้างแบบจำลอง

ขั้นตอนที่ 2: เลือกวิธีการประมวลผลข้อมูลเบื้องต้น (กรณีที่มีข้อมูลบางค่าขาดหายไป)

ขั้นตอนที่ 3: ป้อนค่าตัวแปรให้กับระบบเพื่อใช้ในการสร้างแบบจำลอง ดังนี้ จำนวนโหนดในชั้นอินพุต (Input Node) จำนวนโหนดในชั้นกลาง (Hidden Node) จำนวนรอบของการเรียนรู้ (Training Cycle) และ ค่าความผิดพลาดที่ยอมรับได้

ขั้นตอนที่ 4: เริ่มสร้างแบบจำลอง (New Train)

ขั้นตอนที่ 5: ทดสอบแบบจำลองที่ได้จากขั้นตอนที่ 4 (Test)

ขั้นตอนที่ 6: วิเคราะห์ผลของแบบจำลอง และบันทึกไฟล์แบบจำลองเพื่อนำไปใช้งานต่อไป

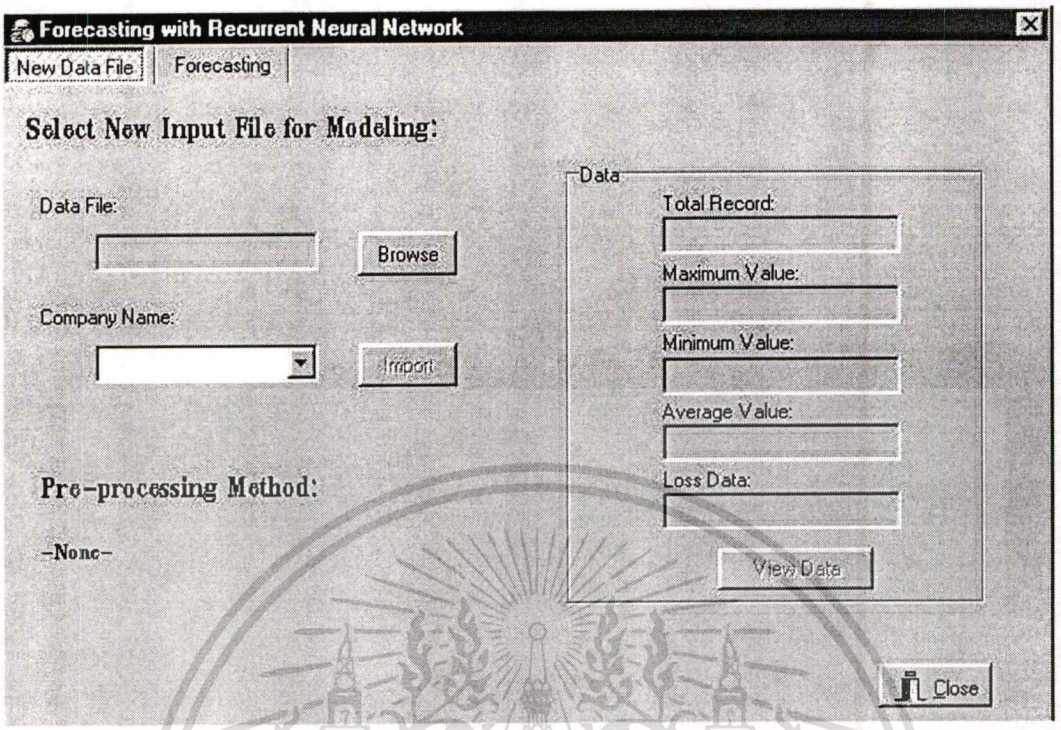
4.3.2 การใช้งานแบบจำลอง มีขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1: เลือกแบบจำลองเพื่อใช้งาน

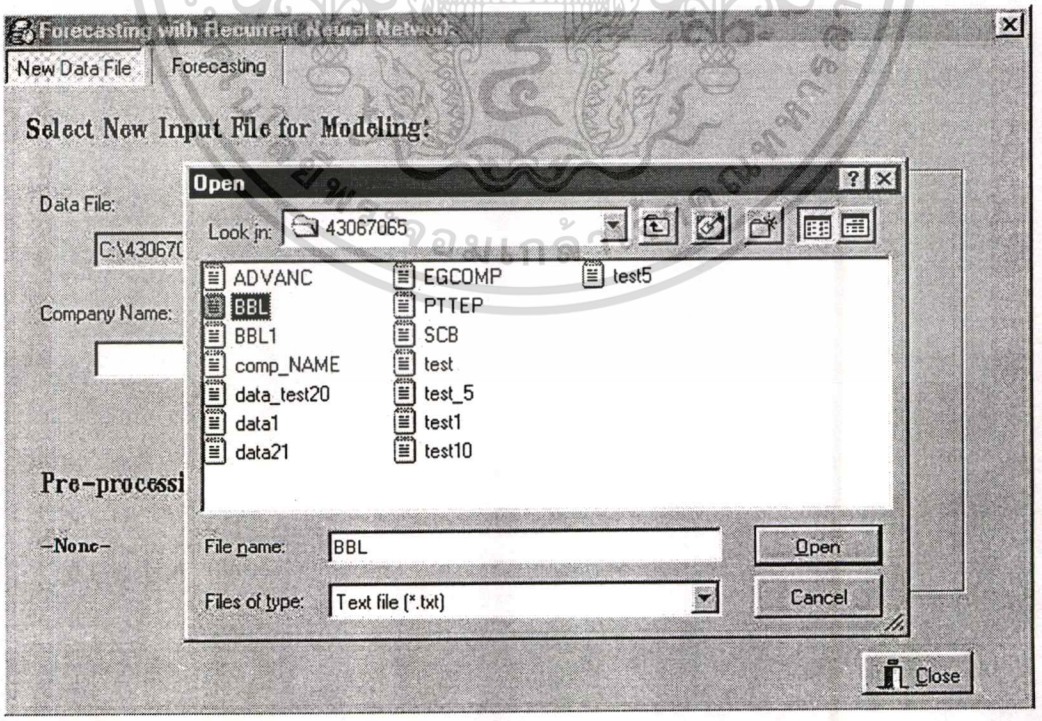
ขั้นตอนที่ 2: ป้อนข้อมูลเข้าสู่ระบบตามจำนวนที่กำหนด

ขั้นตอนที่ 3: ทำการคำนวณหาผลลัพธ์จากระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

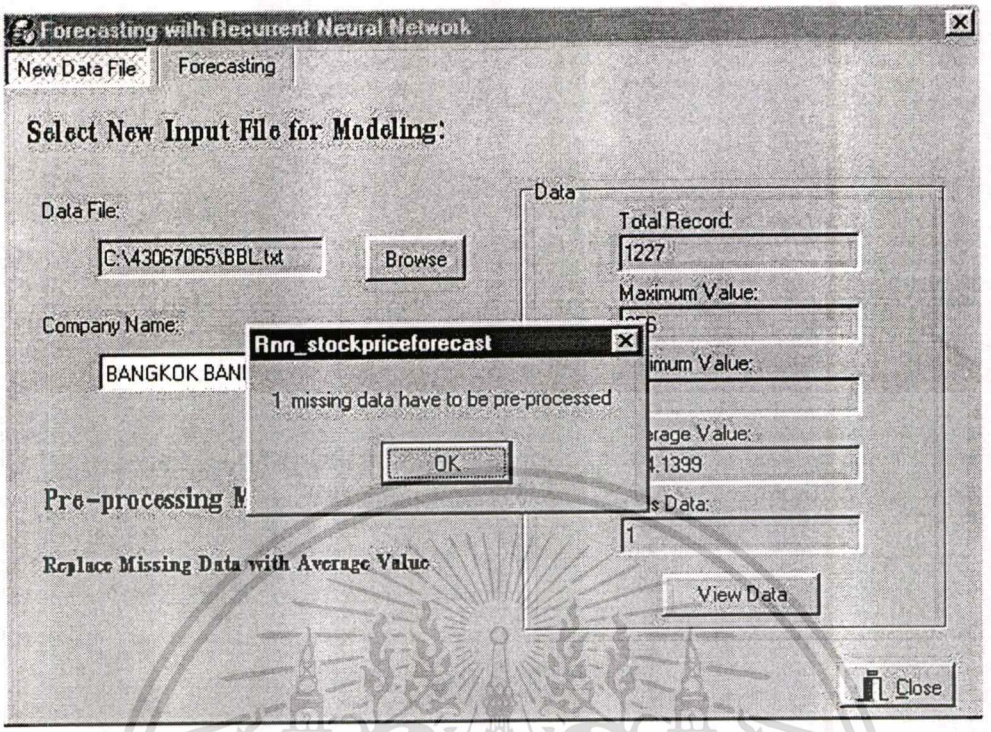


รูปที่ 4.1 แสดงหน้าจอหลักของการรับข้อมูลเพื่อใช้สร้างแบบจำลอง

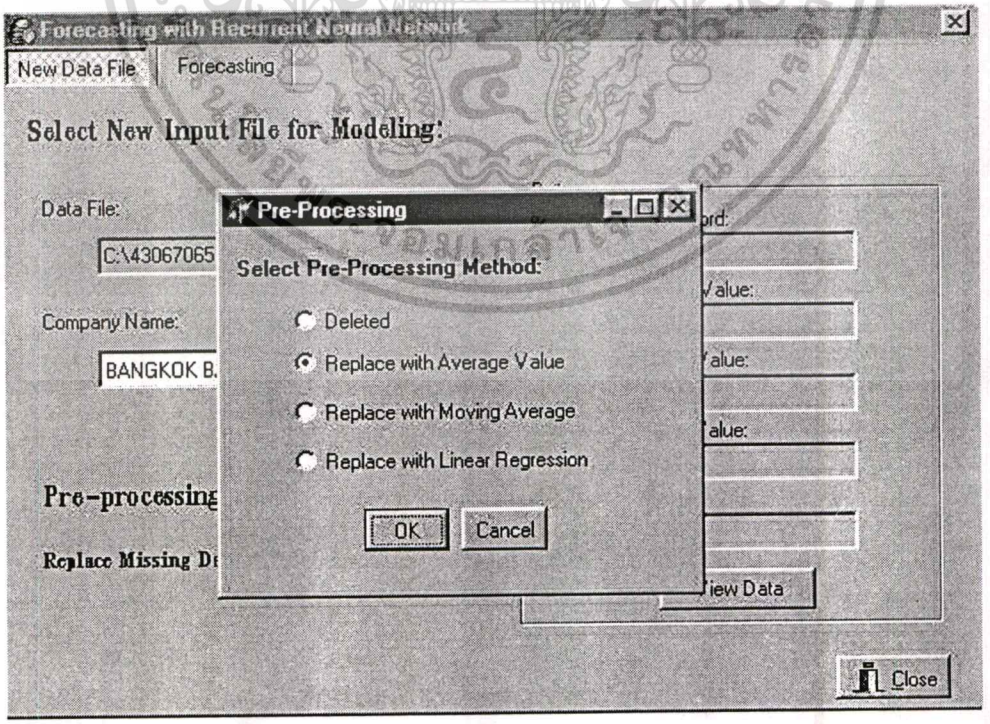


รูปที่ 4.2 แสดงขั้นตอนการเลือกไฟล์ข้อมูลเพื่อใช้สร้างแบบจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

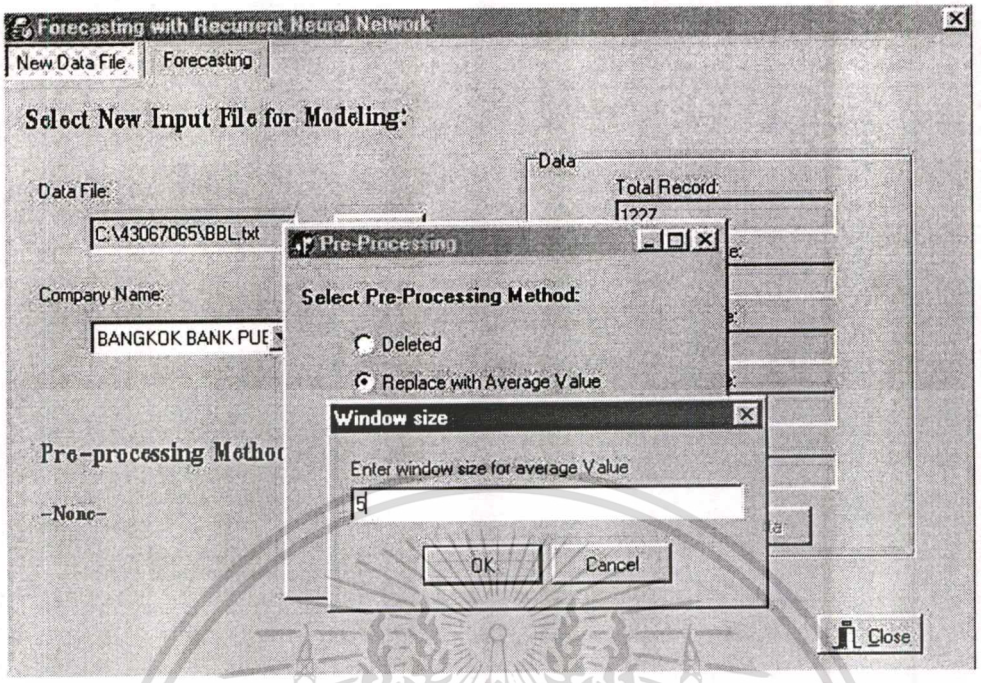


รูปที่ 4.3 การแสดงผลเมื่อตรวจสอบพบข้อมูลที่ไม่สมบูรณ์

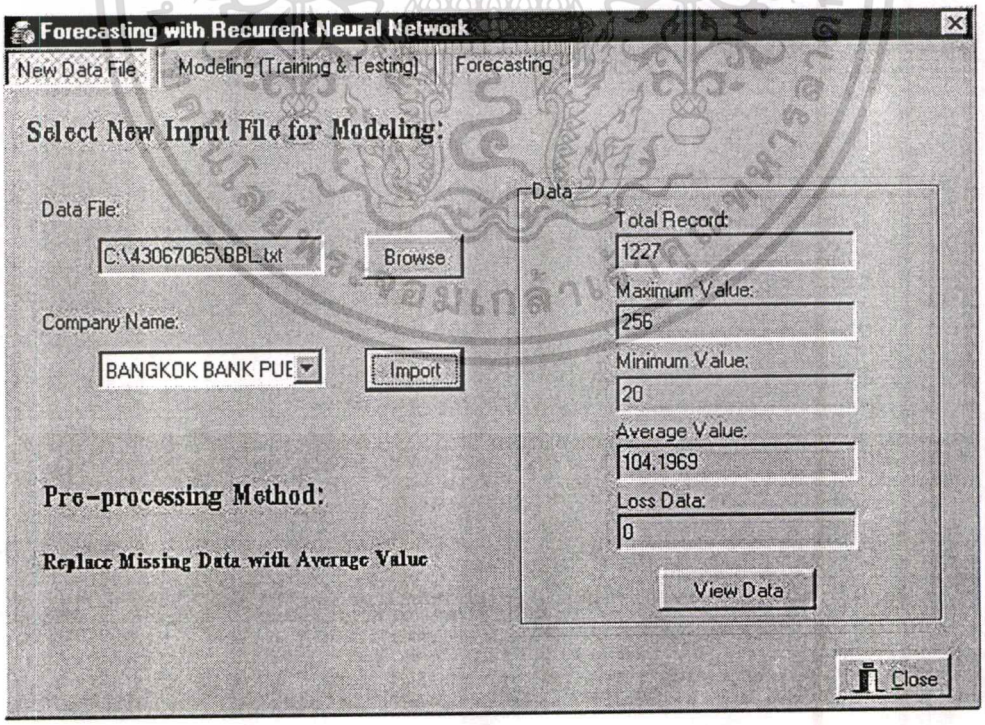


รูปที่ 4.4 แสดงขั้นตอนการประมวลผลข้อมูลเบื้องต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

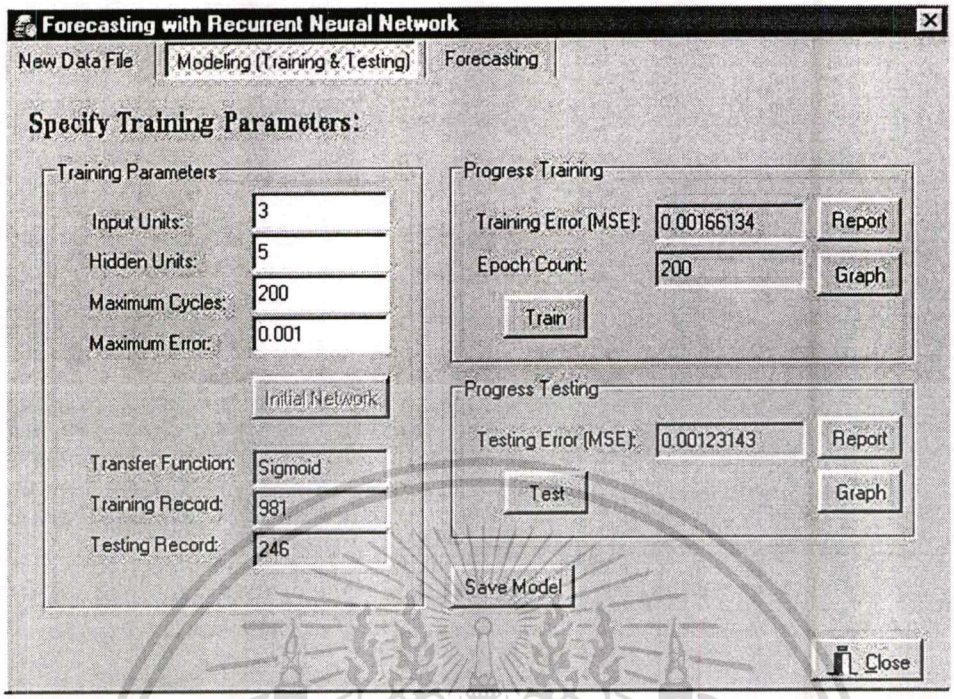


รูปที่ 4.5 แสดงตัวอย่างขั้นตอนการรับข้อมูลเพื่อคำนวณค่าเฉลี่ยสำหรับแทนค่าข้อมูลที่หายไป

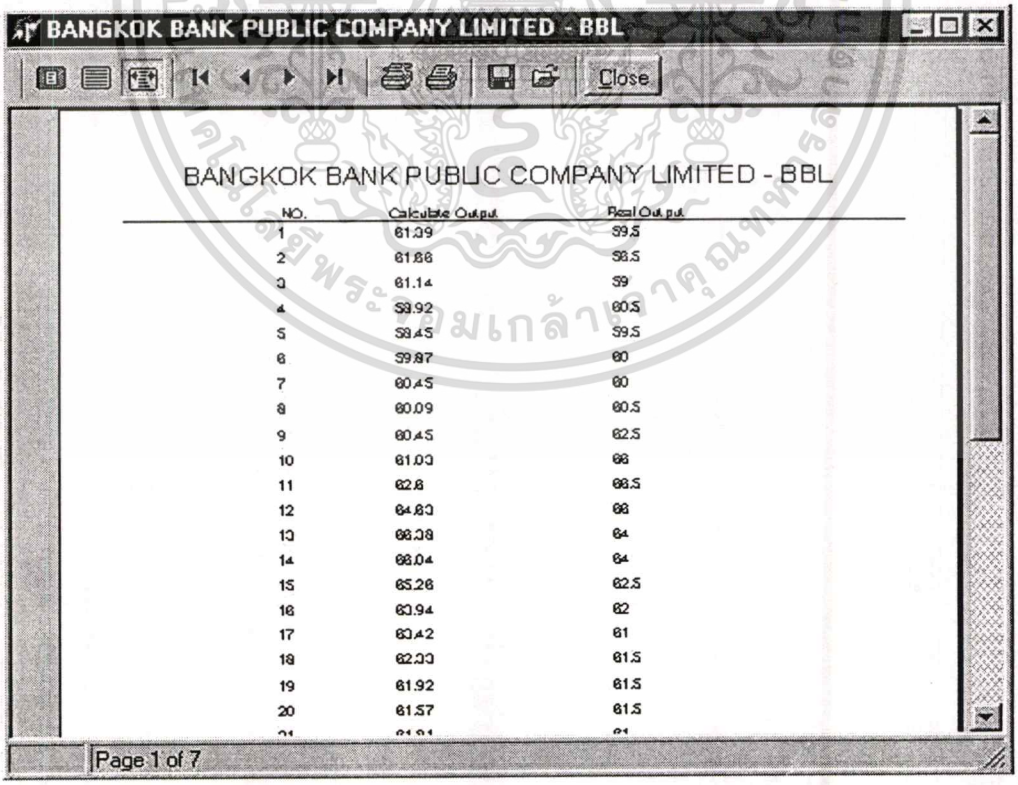


รูปที่ 4.6 แสดงรายละเอียดของตัวอย่างข้อมูลที่ได้หลังการประมวลผลเบื้องต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

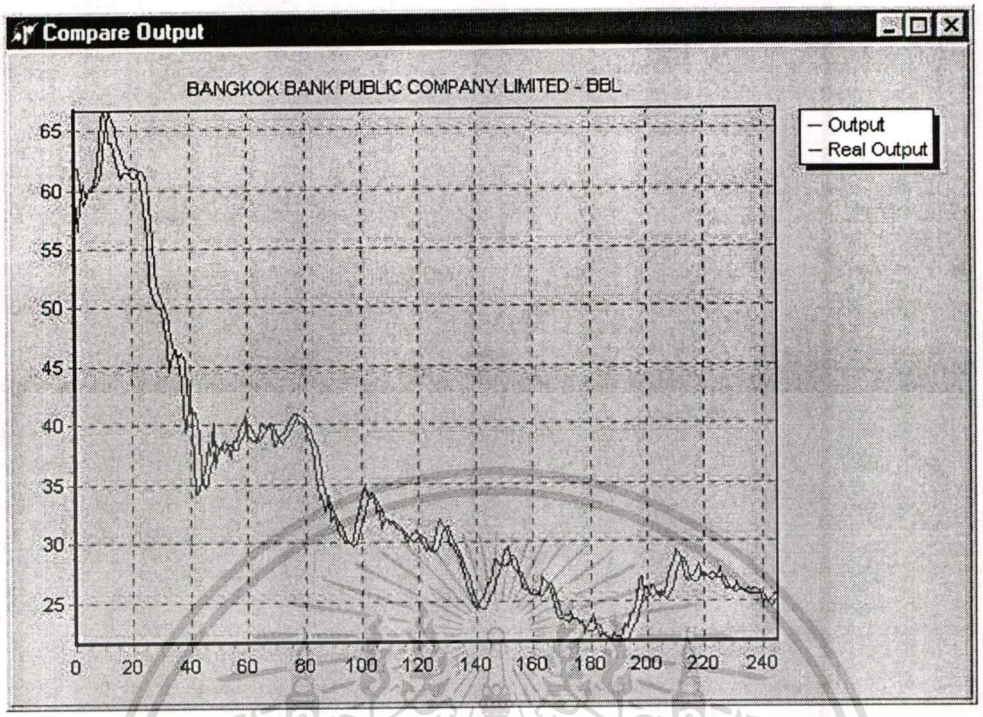


รูปที่ 4.7 แสดงตัวอย่างผลจากการสร้าง และการทดสอบแบบจำลอง

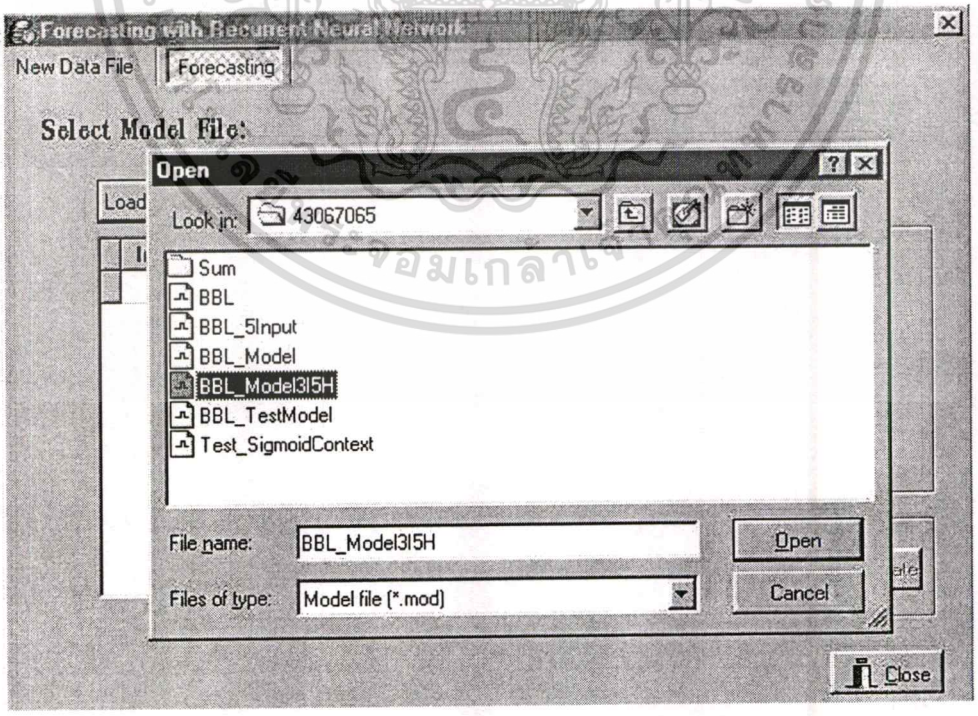


รูปที่ 4.8 แสดงตัวอย่างผลการทดสอบแบบจำลองในรูปแบบรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

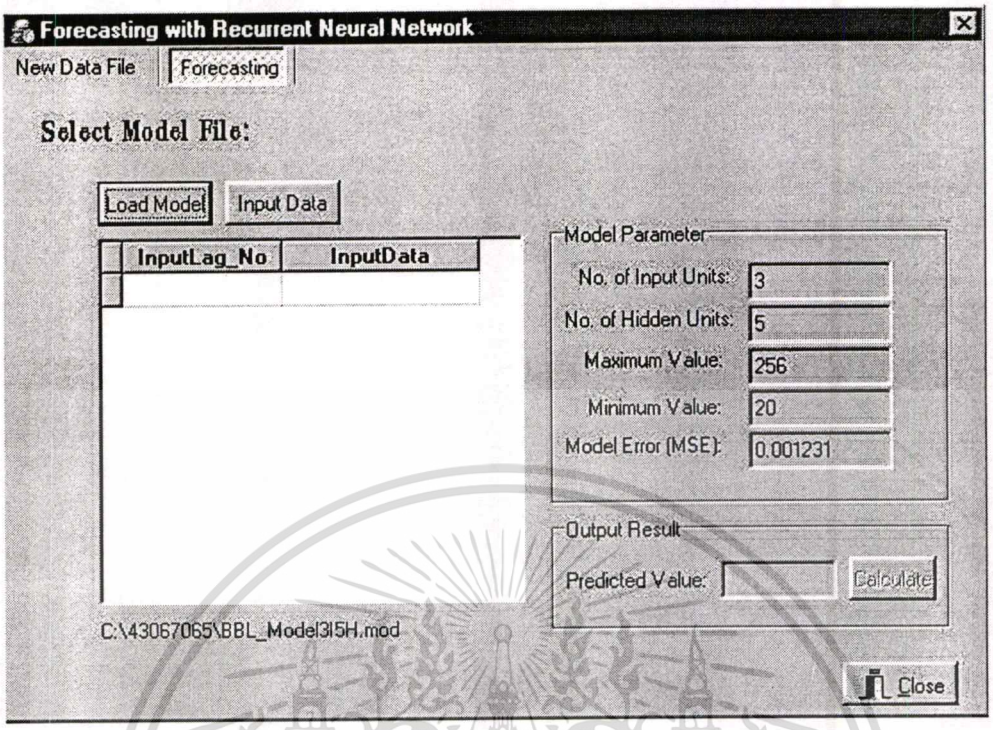


รูปที่ 4.9 แสดงตัวอย่างผลการทดสอบแบบจำลองในรูปแบบกราฟ

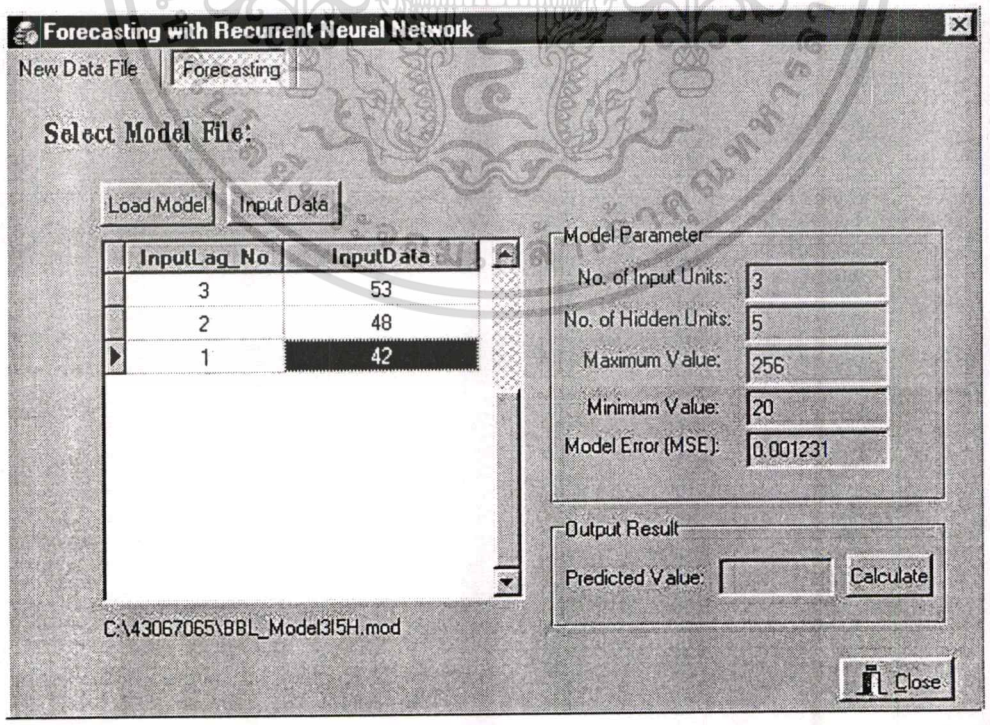


รูปที่ 4.10 แสดงขั้นตอนการเลือกไฟล์แบบจำลองเพื่อใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

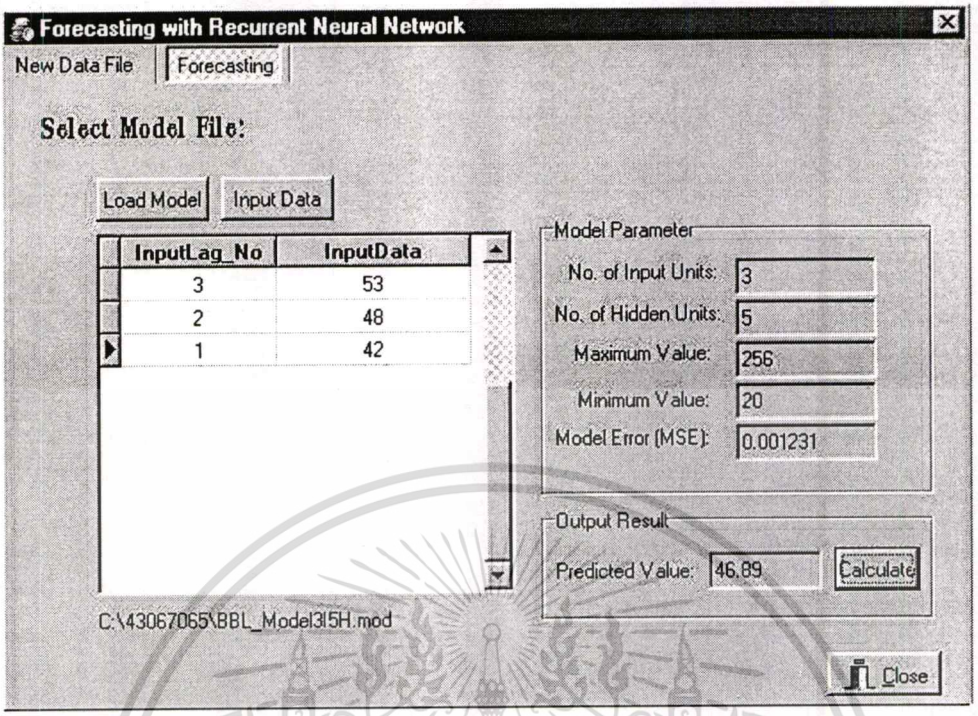


รูปที่ 4.11 แสดงรายละเอียดของตัวอย่างแบบจำลองที่เลือกมาใช้งาน

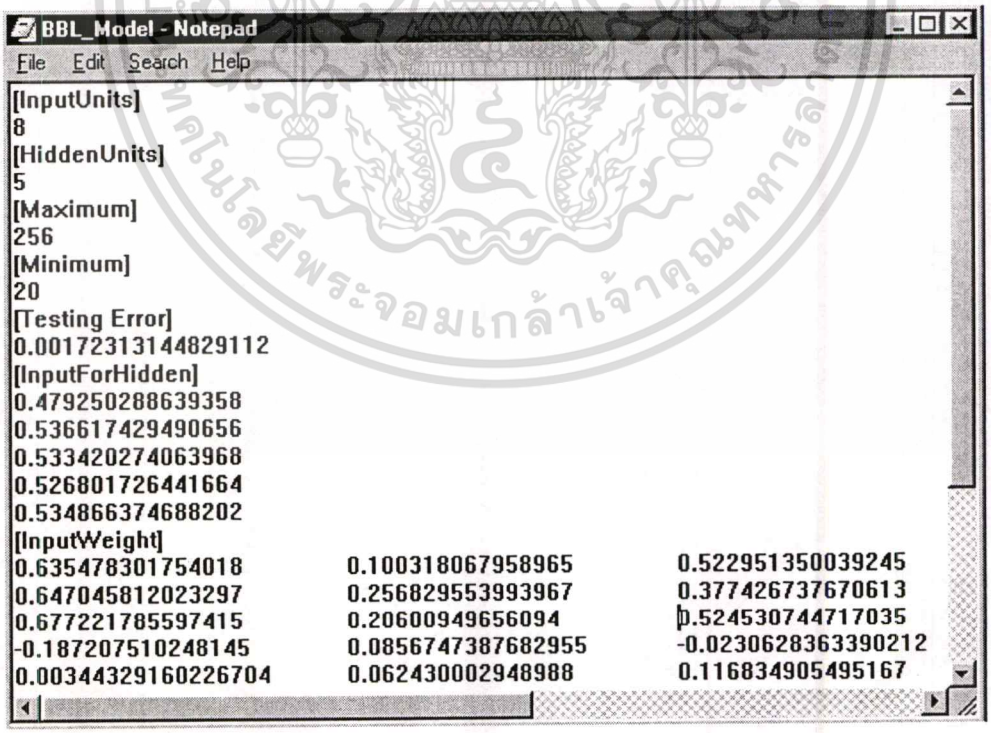


รูปที่ 4.12 แสดงตัวอย่างการป้อนค่าข้อมูลอินพุตเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 แสดงการรายงานผลราคาหลักทรัพย์ของช่วงเวลาถัดไปที่คำนวณได้



รูปที่ 4.14 แสดงตัวอย่างไฟล์ที่เก็บรายละเอียดของแบบจำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดสอบระบบ และสรุปผล

5.1 การทดสอบระบบ

หลังจากการพัฒนาระบบพยากรณ์ราคาหลักทรัพย์โดยใช้อัลกอริทึมของนิวรอลเน็ตเวิร์ก ตามขั้นตอนการทำงานในบทที่ 4 แล้ว สามารถทำการศึกษา และทดสอบระบบได้โดยมีรายละเอียดของการทดสอบระบบดังนี้

แหล่งข้อมูล: ข้อมูลราคาหลักทรัพย์ของธนาคารกรุงเทพ จำกัด (มหาชน) จากตลาดหลักทรัพย์แห่งประเทศไทยในช่วงเวลา 5 ปีตั้งแต่เดือนมกราคม 2539 ถึงเดือนธันวาคม 2543 จำนวน 1227 ข้อมูล

จำนวนข้อมูลที่ใช้ในการสร้างแบบจำลอง: 981 ข้อมูล

จำนวนข้อมูลที่ใช้ในการทดสอบแบบจำลอง: 246 ข้อมูล

จำนวนรอบที่ใช้ในการสร้างแบบจำลอง: 200 รอบ

จากการศึกษาโดยการปรับโครงสร้างภายในเน็ตเวิร์ก ซึ่งได้แก่ จำนวนอินพุต และจำนวนโหนดในชั้นกลาง สามารถแสดงผลการทดสอบได้ดังตารางต่อไปนี้

ตารางที่ 5.1 แสดงผลที่ได้จากการสร้าง และทดสอบแบบจำลองโดยใช้ 2 อินพุต

	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4
จำนวนโหนดในชั้นอินพุต	2	2	2	2
จำนวนโหนดในชั้นกลาง	1	3	5	7
ค่าความผิดพลาดจากการเรียนรู้ :MSE	0.15300	0.00437	0.00240	0.00230
ค่าความผิดพลาดจากการทดสอบ :MSE	0.20726	0.02022	0.00280	0.00268

ตารางที่ 5.2 แสดงผลที่ได้จากการสร้าง และทดสอบแบบจำลองโดยใช้ 3 อินพุต

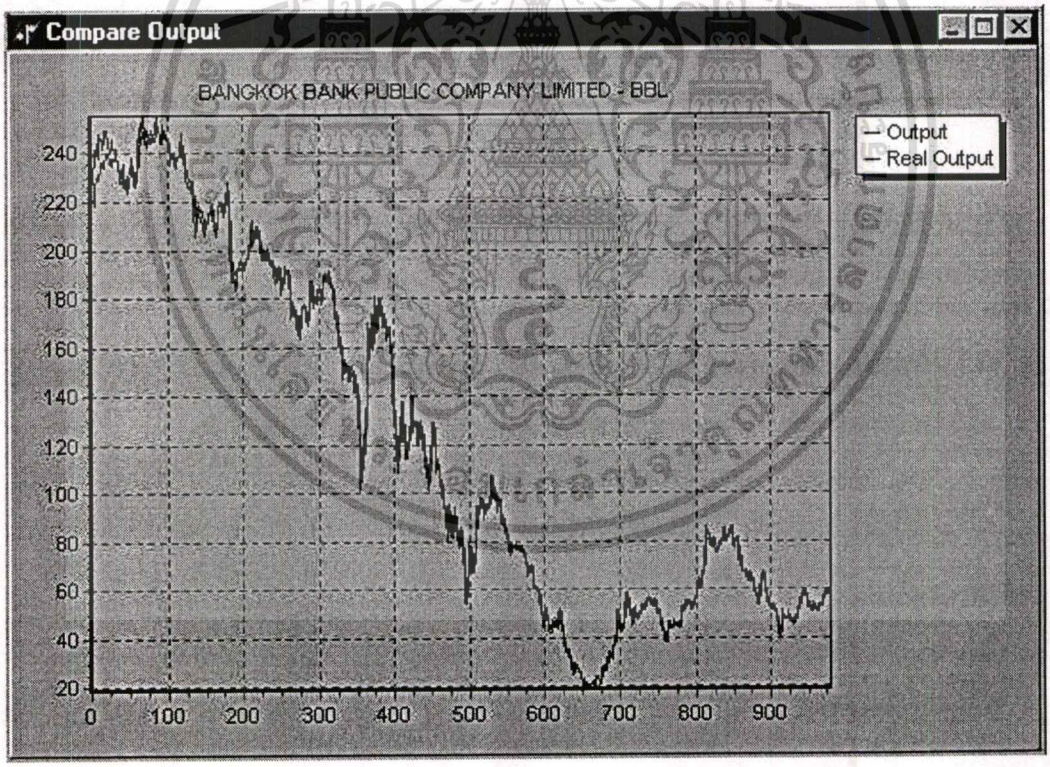
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4
จำนวนโหนดในชั้นอินพุต	3	3	3	3
จำนวนโหนดในชั้นกลาง	1	3	5	7
ค่าความผิดพลาดจากการเรียนรู้ :MSE	0.15404	0.01404	0.00422	0.00306
ค่าความผิดพลาดจากการทดสอบ :MSE	0.19932	0.11580	0.01416	0.00350

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.3 แสดงผลที่ได้จากการสร้าง และทดสอบแบบจำลองโดยใช้ 5 อินพุต

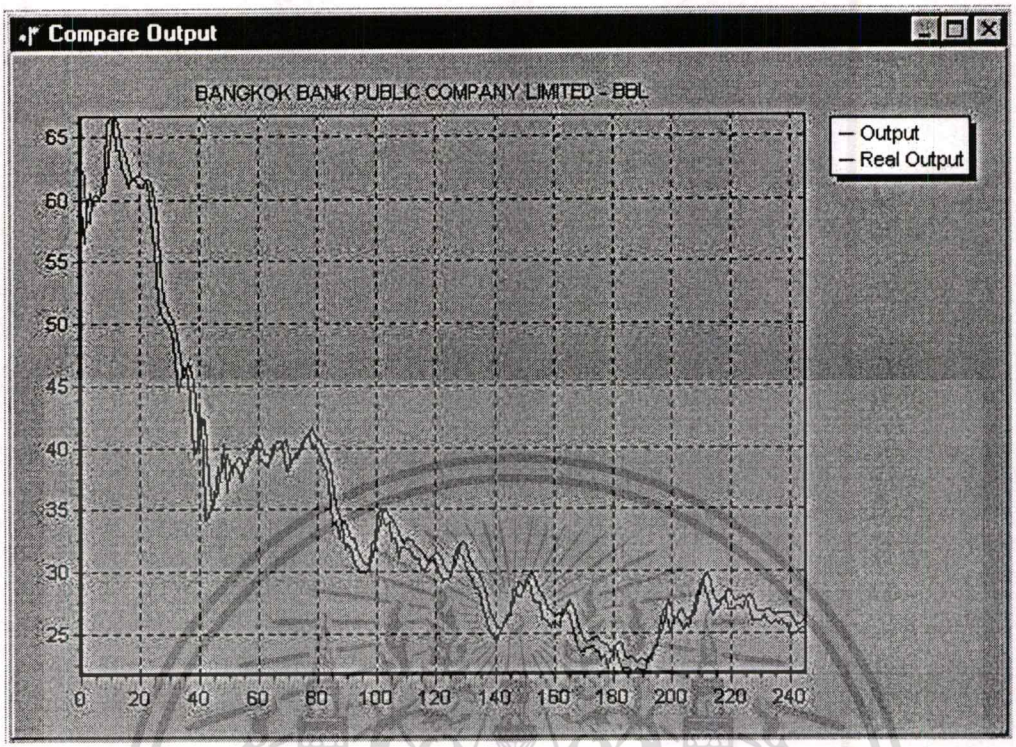
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4
จำนวนโหนดในชั้นอินพุต	5	5	5	5
จำนวนโหนดในชั้นกลาง	1	3	5	7
ค่าความผิดพลาดจากการเรียนรู้ :MSE	0.16520	0.00963	0.00523	0.00389
ค่าความผิดพลาดจากการทดสอบ :MSE	0.27044	0.05703	0.01460	0.00750

จากผลการศึกษาสามารถแสดงข้อมูลเอาต์พุตจากการสร้าง (Train) และทดสอบ (Test) แบบจำลองเปรียบเทียบกับค่าจริงของราคาหลักทรัพย์ (Real Output)ได้ดังรูปที่ 5.1 และ 5.2 ตามลำดับ โดยตัวอย่างของผลที่ได้จากการทดสอบแบบจำลอง แสดงไว้ในภาคผนวก ข



รูปที่ 5.1 แสดงผลที่ได้จากการเรียนรู้โดยใช้จำนวนอินพุต = 2 และจำนวนโหนดในชั้นกลาง = 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 แสดงผลการทดสอบจากการเรียนรู้โดยจำนวนอินพุต=2 และจำนวนโหนดในชั้นกลาง=7

5.2 สรุปผลการทดสอบ

จากผลการพยากรณ์ราคาหลักทรัพย์โดยใช้อัลกอริทึมนิวรอลเน็ตเวิร์กแบบ Elman พบว่าผลที่ได้ขึ้นอยู่กับตัวแปรที่กำหนดให้ในการสร้างแบบจำลองให้กับระบบ เช่น จำนวนโหนดในชั้นอินพุต และชั้นกลางของเน็ตเวิร์ก เป็นต้น และจากผลการทดสอบแบบจำลองที่ได้พบว่าค่าของราคาหลักทรัพย์ที่ได้นั้นจะมีความผิดพลาดอยู่บ้าง ซึ่งอาจเป็นผลมาจากปัจจัยที่กำหนดใช้สร้างแบบจำลองนั้นไม่เหมาะสม ข้อมูลที่นำมาใช้สร้างแบบจำลองไม่เหมาะสม หรืออาจเกิดจากมีปัจจัยพื้นฐานภายนอกอื่นๆ ที่มีอิทธิพลต่อราคาหลักทรัพย์นั้นๆ ก็ได้ แต่อย่างไรก็ตามแม้ว่าราคาหลักทรัพย์จะได้รับผลกระทบจากปัจจัยภายนอก แต่ถ้าหากกำหนดตัวแปรที่ใช้สร้างแบบจำลองได้อย่างเหมาะสมแล้ว ผลที่ได้จากแบบจำลองสามารถนำไปใช้วิเคราะห์ร่วมกับปัจจัยพื้นฐานอื่นๆ เพื่อใช้เป็นแนวทางสำหรับการเลือกลงทุนในตลาดหลักทรัพย์ได้อย่างมีประสิทธิภาพต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Andrew D. Martin and Daniel B. Pemstein. September 18, 2001. "Forecasting Conflict with Recurrent Artificial Neural Networks".

Amir F. Atiya and Alexander G. Parlos. May 2000. "New Results on Recurrent Network Training: Unifying the Algorithms and Accelerating Convergence". IEEE Transactions on Neural Networks. 11(3). 697-709.

Barbara Hammer. 1999. "Learning with Recurrent Neural Networks". Technical Report, University of Osnabruck.

Bill G. Horne and C. Lee Giles. 1995. "An Experimental Comparison of Recurrent Neural Networks". Neural Information Proceeding Systems 7, eds. G. Tesauro, D. Touretzky, T. Leen, P.697, MIT Press.

Bruce D. Baker. 1998. "A Comparison of Linear Regression and Neural Network Methods for Forecasting Educational Spending". Annual Meeting of the American Education Finance Association Mobile, Alabama. University of Kansas.

Cabena, Peter. Et al. 1997. Discovering Data Mining: from Concept to Implementation. New Jersey: Prentice- Hall.

Chung-Ming Kuan and Tung Liu. Sept. 10, 1994. "Forecasting Exchange Rates using Feedforward and Recurrent Neural Networks". Department of Economics, National Taiwan University and Ball State University. North American Winter Meeting of the Economic Society in New Orleans, Louisiana.

Devid Noelle. 1996. "A Brief Introduction to Artificial Neural Network". CSE 151 – Introduction to Artificial Intelligence. University of California, Sandiego.

Dietrich, J. et. Al. 2000. "Data Mining: Are You Ready?". Kellogg TechVenture 2000 Anthology.

Elman, Jeffrey L. 1990. "Finding Structure in Time". Cognitive Science 14: 179-211.

Gately, E. 1996. Neural Network for Financial Forecasting. New York: John Wiely and Sons.

Han, J. and Kamber, M. 2001. Data Mining: Concepts and Techniques. San Diego: Morgan Kaufmann.

Han, J. 2000. **From Data Mining to Web Mining: An Overview.** [Online]. Available: ftp://ftp.fas.sfu.ca/pub/cs/han/slides/hkw00.ppt.

Han, J. 2000. **Towards Integrated, Intelligent Tools for Data Mining.** [Online]. Available: ftp://ftp.fas.sfu.ca/pub/cs/han/slides/tai00.ppt.

Kevin Swingler. July 14, 1994. "Financial Prediction, Some Pointers, Pitfalls, and Common Error". Center for Cognitive and Computational Neuroscience. Stirling University. Stirling FK9 4LA.

Kuo, Jyh-Ming, 1992. "Prediction of Chaotic Time Series Using Recurrent Neural Networks". 1992 IEEE Workshop NN for SP. University of Florida.

Lawrence, S. et. al. 1996. "What Size Neural Network Give Optimal Generalization? Convergence Properties of Backpropagation". Technical Report, UMIACS-TR-96-22 and CS-TR-3617. Institute of Advance Computer Studies, University of Maryland, College Park.

Morten With Pedersen. "Training Recurrent networks". CONNECT, Department of Mathematical Modelling, Technical University of Denmark. DK-2800 Lingby, Denmark.

Morten With Pedersen and Jan Larsen. "Interpretation of Recurrent Neural Networks". CONNECT, Department of Mathematical Modelling, Technical University of Denmark. DK-2800 Lingby, Denmark.

Peter C. McCluskey. 1993. "Feedforward and Recurrent Neural Networks and Genetic Programs for Stock Market and Time Series Forecasting". Department of Computer Science. Brown University.

Pyle, D. 1999. Data Preparation for Data Mining. San Francisco: Morgan Kaufmann.

Ronald J. Williams and David Zipser. 1995. "Gradient-Based Learning Algorithms for Recurrent Networks and their Computational Complexity". Northeastern University Boston and University of California, San Diego. – Appear in Y. Chauvin and D. E. Rumelhart (Eds.): Back-propagation: Theory, Architectures and Applications. Hillsdale, NJ: Erlbaum. 1995.

Roy Schwarzel. 1995. "Prediction of Currency Exchange Rates by Using a Multi-Neural Network System". Technical Report.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thomas Hellstrom and Kenneth Homstrom. August 1998. "Predicting the Stock Market". Center of Mathematical Modeling (CMM) –Technical Report. Department of Mathematics and Physics. Malardalen University. Sweden.

Tomasz J. Cholewo and Jacek M. Zurada. 1997. "Neural Network Tools for Stellar Light Prediction". February 1-8. 415-422. Proceedings of IEEE Aerospace Conference. Snowmass, Colo., USA

Williams, Ronald J. and Devid Zipser. 1989. "A Learning Algorithm for Continually Running Fully Recurrent Networks". Neural Computation 1, p.270.

Wright, P. 1998. "The Significance of the Missing Data Problem in Knowledge Discovery". Proceedings of the First Southern Symposium on Computing. The University of Southern Mississippi, December 4-5.

Yao, Jingtao and POH,Hean-Lee. "Forecasting the KLSE Index Using Neural Networks". Department of Information Systems&Computer Science, National University of Singapore, Kent Ridge, Singapore 0511.

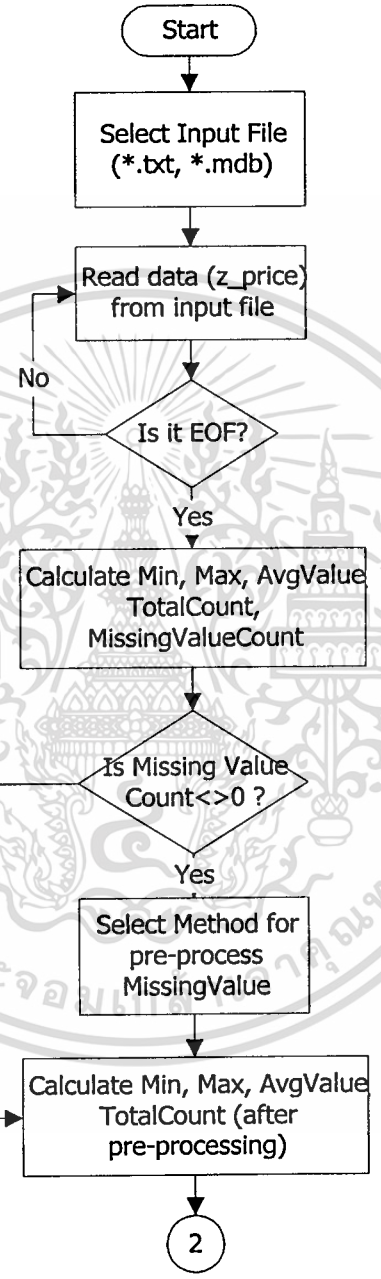
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

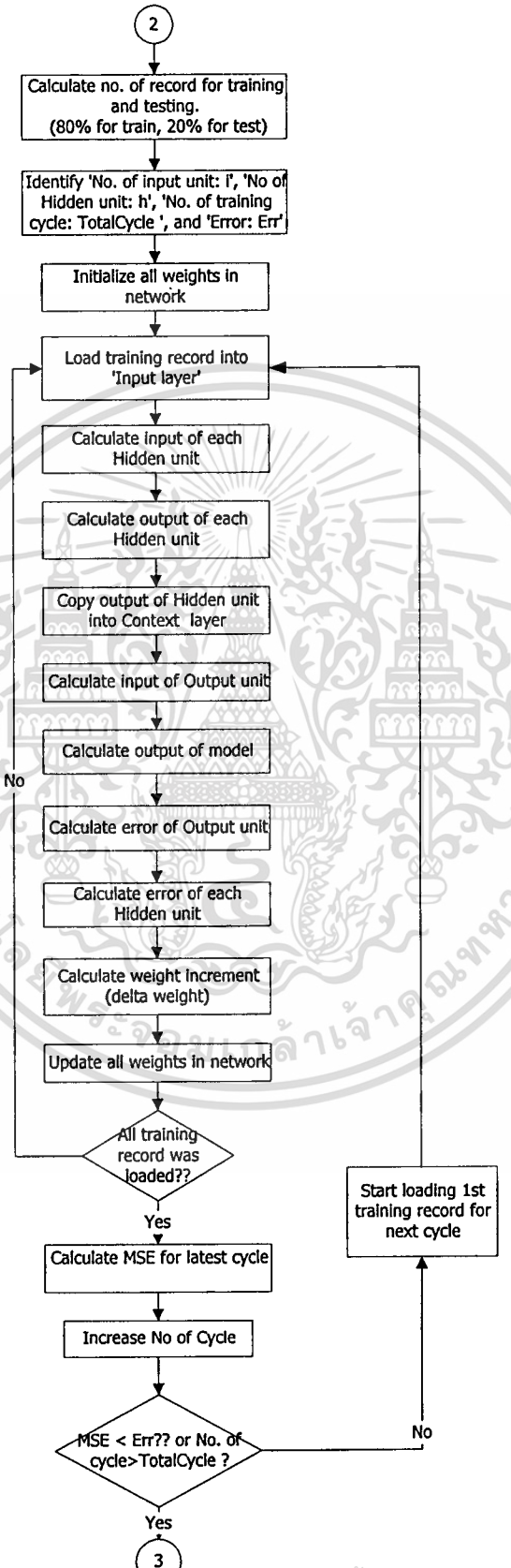
ภาคผนวก ก. แผนภาพแสดงลำดับงานของโปรแกรม

ส่วนรับข้อมูล:



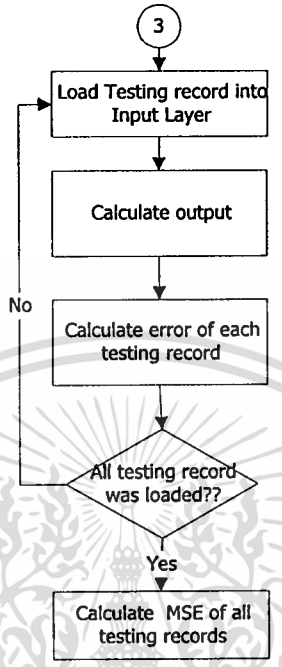
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนสร้างแบบจำลอง:

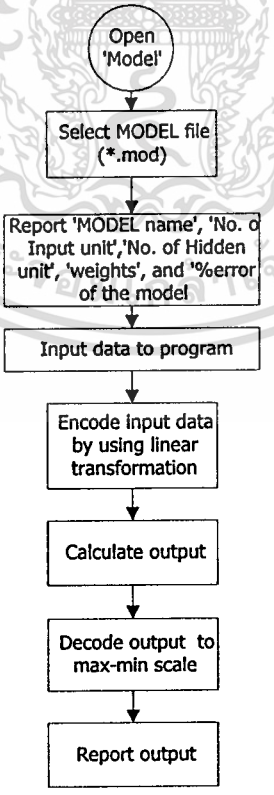


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนทดสอบแบบจำลอง:



ส่วนการใช้งานแบบจำลอง:



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข. แสดงผลราคาหลักทรัพย์ที่ได้จากการทดสอบแบบจำลอง (CalculateOutput) ที่สร้างจากข้อมูลของธนาคารกรุงเทพจำกัด (มหาชน) เปรียบเทียบกับค่าข้อมูลจริงของราคาหลักทรัพย์ (RealOutput) โดยใช้จำนวนข้อมูลในการสร้างแบบจำลอง 981 ข้อมูล จำนวนอินพุต = 2 อินพุต และจำนวนโหนดในชั้นกลาง = 7 โหนด

OutputOrder	CalculateOutput	RealOutput
0	62.64	59.50
1	61.86	56.50
2	58.39	59.00
3	58.17	60.50
4	60.15	59.50
5	60.39	60.00
6	60.15	60.00
7	60.39	60.50
8	60.64	62.50
9	61.89	66.00
10	64.62	66.50
11	66.59	66.00
12	66.59	64.00
13	65.34	64.00
14	64.36	62.50
15	63.61	62.00
16	62.62	61.00
17	61.88	61.50
18	61.63	61.50
19	61.88	61.50
20	61.88	61.00
21	61.63	61.00
22	61.38	61.50
23	61.63	60.50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OutputOrder	CalculateOutput	RealOutput
24	61.38	58.50
25	59.89	56.00
26	57.65	52.00
27	54.42	51.00
28	51.96	50.50
29	51.22	50.00
30	50.72	49.75
31	50.35	48.75
32	49.73	46.75
33	48.23	44.50
34	46.13	46.00
35	45.77	46.50
36	46.76	45.75
37	46.63	43.75
38	45.26	39.50
39	42.16	40.00
40	40.32	43.50
41	42.32	39.75
42	42.16	34.25
43	37.57	34.50
44	35.01	35.75
45	35.75	36.00
46	36.49	38.50
47	37.86	37.50
48	38.58	40.25
49	39.47	36.50
50	38.94	38.00
51	37.85	38.50
52	38.84	38.75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OutputOrder	CalculateOutput	RealOutput
53	39.21	38.25
54	39.08	37.25
55	38.34	38.75
56	38.60	38.75
57	39.33	39.50
58	39.70	40.25
59	40.45	40.75
60	41.06	39.00
61	40.44	39.00
62	39.58	38.75
63	39.45	38.50
64	39.21	39.75
65	39.71	40.25
66	40.57	39.75
67	40.56	40.00
68	40.44	40.25
69	40.69	38.25
70	39.82	38.25
71	38.84	39.00
72	39.21	39.25
73	39.70	39.25
74	39.82	40.25
75	40.32	40.50
76	40.94	41.00
77	41.31	41.00
78	41.56	40.00
79	41.06	40.25
80	40.69	39.75
81	40.56	39.00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OutputOrder	CalculateOutput	RealOutput
82	39.95	38.50
83	39.33	38.00
84	38.83	35.75
85	37.47	33.75
86	35.37	34.00
87	34.51	32.75
88	34.01	34.00
89	34.02	31.75
90	33.52	32.25
91	32.66	31.75
92	32.66	30.75
93	31.92	30.50
94	31.31	30.00
95	30.93	30.00
96	30.69	30.00
97	30.69	30.25
98	30.81	32.25
99	31.93	32.50
100	33.03	34.00
101	33.90	35.00
102	35.13	34.00
103	35.13	33.75
104	34.51	34.00
105	34.51	32.50
106	33.89	32.75
107	33.28	31.25
108	32.66	32.00
109	32.30	32.00
110	32.66	31.75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OutputOrder	CalculateOutput	RealOutput
111	32.54	31.75
112	32.42	31.25
113	32.17	31.00
114	31.80	31.00
115	31.68	30.25
116	31.30	30.00
117	30.81	30.75
118	31.06	30.75
119	31.43	31.00
120	31.55	30.50
121	31.43	30.00
122	30.93	29.25
123	30.32	29.50
124	30.08	29.25
125	30.07	30.00
126	30.32	31.00
127	31.19	32.00
128	32.17	31.50
129	32.41	31.25
130	32.04	30.25
131	31.43	29.75
132	30.69	29.50
133	30.32	28.75
134	29.82	28.75
135	29.46	27.75
136	28.96	27.00
137	28.10	26.00
138	27.24	25.25
139	26.38	25.00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OutputOrder	CalculateOutput	RealOutput
140	25.89	24.25
141	25.40	24.75
142	25.28	25.25
143	25.77	25.75
144	26.26	26.50
145	26.88	26.50
146	27.24	28.50
147	28.24	28.50
148	29.21	28.00
149	28.96	28.00
150	28.72	29.25
151	29.34	29.50
152	30.08	28.75
153	29.82	27.75
154	28.96	27.50
155	28.35	27.50
156	28.23	26.25
157	27.61	26.00
158	26.87	26.00
159	26.75	25.50
160	26.50	26.00
161	26.51	25.75
162	26.63	25.50
163	26.38	27.00
164	27.00	26.75
165	27.61	26.50
166	27.37	25.75
167	26.87	24.75
168	26.01	24.00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OutputOrder	CalculateOutput	RealOutput
169	25.15	23.50
170	24.54	23.50
171	24.3	23.75
172	24.42	23.75
173	24.54	24.00
174	24.67	23.50
175	24.54	23.50
176	24.30	23.00
177	24.05	23.25
178	23.93	22.25
179	23.56	23.00
180	23.44	23.25
181	23.93	23.75
182	24.30	22.75
183	24.05	22.00
184	23.19	22.25
185	22.95	22.25
186	23.07	22.00
187	22.95	21.75
188	22.70	22.25
189	22.83	22.00
190	22.95	22.00
191	22.82	21.75
192	22.70	23.00
193	23.20	22.75
194	23.68	23.75
195	24.05	24.00
196	24.67	25.25
197	25.41	26.75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OutputOrder	CalculateOutput	RealOutput
198	26.76	27.00
199	27.61	25.50
200	26.99	25.00
201	26.01	26.25
202	26.39	26.25
203	27.00	25.75
204	26.75	25.25
205	26.26	25.75
206	26.26	25.50
207	26.38	26.50
208	26.76	27.75
209	27.86	28.25
210	28.72	29.25
211	29.46	29.00
212	29.83	28.00
213	29.21	27.00
214	28.22	26.50
215	27.49	27.00
216	27.49	27.00
217	27.74	27.75
218	28.11	28.00
219	28.6	27.00
220	28.22	27.00
221	27.74	27.00
222	27.74	27.25
223	27.86	27.25
224	27.98	27.00
225	27.86	27.75
226	28.11	26.75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OutputOrder	CalculateOutput	RealOutput
227	27.98	26.00
228	27.12	26.00
229	26.75	26.00
230	26.75	26.00
231	26.75	26.50
232	27	26.00
233	27	25.75
234	26.63	25.75
235	26.51	26.00
236	26.63	25.75
237	26.63	26.00
238	26.63	25.75
239	26.63	25.75
240	26.51	24.75
241	26.01	25.00
242	25.65	25.00
243	25.77	25.25
244	25.89	25.50
245	26.14	25.50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน	นางสาวชัชรี ปิ่นแก้ว
วันเดือนปีเกิด	21 เมษายน พ.ศ.2517
สถานที่เกิด	จ. ภูเก็ต
วุฒิการศึกษาระดับปริญญาตรี	วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมอิเล็กทรอนิกส์
สถานที่สำเร็จการศึกษา	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีที่สำเร็จการศึกษา	ปีการศึกษา 2539
ประวัติการทำงาน	2543-ปัจจุบัน วิศวกร บริษัท ฟิลิปส์ เซมิคอนดัคเตอร์ (ประเทศไทย) จำกัด 2541-2543 วิศวกรควบคุมกระบวนการ บริษัท รีด-ไรท์ (ประเทศไทย) จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้