

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การทำนายอนุกรมเวลาโดยใช้โครงข่ายประสาทเทียมร่วมกับ

พาทิคลัสตวอมพ์ออปติไมซ์เซชัน

**Time Series Forecasting Using Artificial Neural Network and
Particle Swarm Optimization**

โดย

พิชักษณ์ ศิริมาจันทร์

รหัสประจำตัว 46066729

วัน เดือน ปี.....	21 ก.พ. 2550
เลขทะเบียน.....	02323
เลขเรียกหนังสือ.....	อพ. ๗ ๖32๓ 2548
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

อาจารย์ที่ปรึกษา

รศ.ดร. อาริต ธรรมโน



H002323

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2548
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การทำนายอนุกรมเวลาโดยใช้โครงข่ายประสาทเทียมร่วมกับ พหุคูณกำลังสองไม่เชิงเส้น
นักศึกษา	พิชญ์ ศิริมาจันทร์
อาจารย์ที่ปรึกษา	รศ.ดร.อาริต ธรรมโน
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2548

บทคัดย่อ

ความสามารถในการทำนายเหตุการณ์ในอนาคต มีความสำคัญอย่างมากต่อองค์กรของระบบธุรกิจในปัจจุบัน โครงข่ายประสาทเทียมเป็นอัลกอริทึมรูปแบบของการเรียนรู้สำหรับคอมพิวเตอร์ โดยจะอาศัยหลักการของการค้นหาค่าถ่วงน้ำหนักที่ถูกต้องสำหรับค่านำเข้าแต่ละค่า โดยใช้วิธีการทำซ้ำเพื่อแก้ไขค่าถ่วงน้ำหนักจนกว่าจะได้รับค่าความถูกต้องที่น่าพอใจ ซึ่งในที่นี้เราจะใช้เทคนิค Particle Swarm Optimization นำมาใช้ร่วมด้วยเพื่อช่วยในการค้นหาค่าที่เหมาะสมให้มีประสิทธิภาพดียิ่งขึ้น

Title	Time Series Forecasting Using Artificial Neural Network and Particle Swarm Optimization
Student	Mr. Pijak Sirimajun
Advisor	Assoc. Prof. Dr. Arit Thammano
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2005

ABSTRACT

Nowadays, many business organizations want to get the predictive ability, Artificial Neural Network is a one of pattern of Artificial Algorithm to used for computer. In commonly Artificial Neural Network is use a principle of the search for optimum value of weight from data. Particle Swarm Optimization is new way to used for finding the optimize Neural Network weight in concept of social behavior.

กิตติกรรมประกาศ

การเดินทางครั้งนี้ข้าพเจ้าคงไม่สามารถมาถึงจุดหมายปลายทางได้เลย ถ้าปราศจากเสบียงอาหารที่ส่งมาให้อย่างไม่เคยขาดจาก ป้า แม่ และแม่เอ๋ ขอขอบพระคุณท่าน อ.อาริต ซึ่งเปรียบดังดวงอาทิตย์ที่คอยกรุณาแสงสว่างนำทางข้าพเจ้าเสมอ ยามที่การเดินทางของข้าพเจ้าพบกับความมืดมิดและความสับสน ขอขอบคุณเพื่อนๆ IS16.1 ที่เป็นเพื่อนร่วมเดินทางที่แสนดี สุดท้ายนี้ข้าพเจ้าขอขอบคุณทุกคนที่มอบความรักความหวังดีตลอดการเดินทางให้แก่ข้าพเจ้า ถึงแม้ข้าพเจ้าจะไม่ได้กล่าวนามของท่านเหล่านั้นในโอกาสนี้ แต่ทุกๆท่านนั้นจะจารึกอยู่ในความทรงจำของข้าพเจ้าตลอดไป

พิชักษณ์ ศิริมาจันทร์

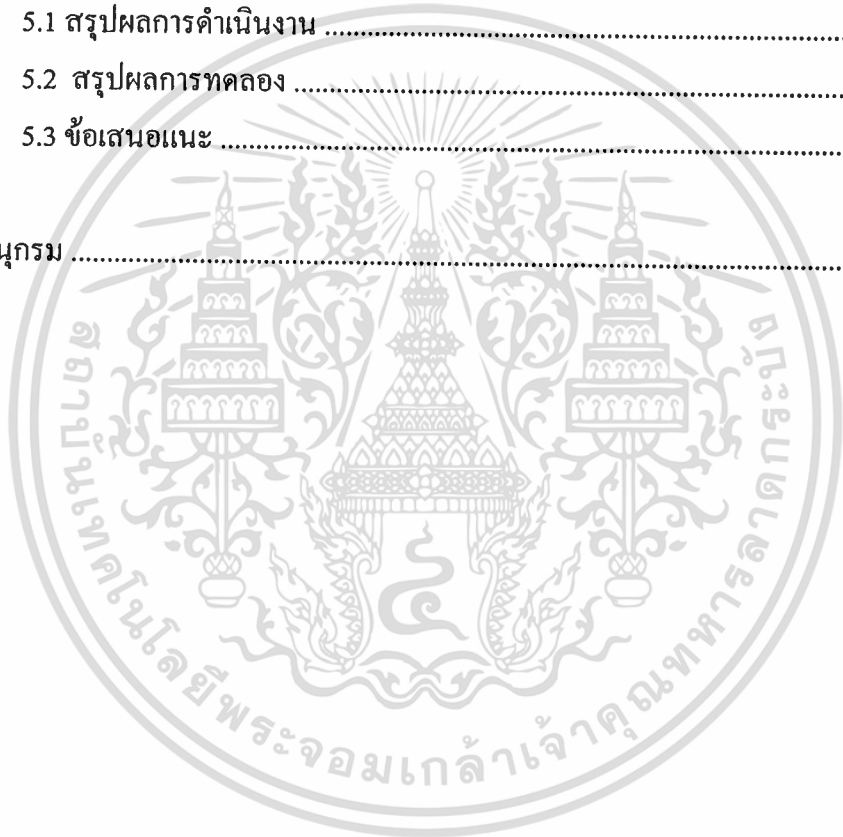


สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญภาพ	VIII
บทที่	
1. บทนำ	1
1.1 ความเป็นมา	1
1.2 วัตถุประสงค์ของ โครงการงาน	2
1.3 ขอบเขตของการศึกษา	2
1.4 ขั้นตอนและวิธีการดำเนินงาน	2
1.5 ผลที่คาดว่าจะได้รับ	3
2. ทฤษฎีที่เกี่ยวข้อง	4
2.1 การพยากรณ์	4
2.2 การวิเคราะห์อนุกรมเวลา	7
2.3 โครงข่ายประสาทเทียม	9
2.4 Particle Swarm Optimization	13
2.5 โครงข่ายประสาทเทียมแบบแบคพรอพากะชัน	18
2.6 ความผิดพลาดของการพยากรณ์	21
3. ระบบการทำนายข้อมูลแบบผสม	23
3.1 สถาปัตยกรรมของระบบ	23
3.2 ขั้นตอนการทำงานของระบบ	25
3.3 แผนภาพสรุปขั้นตอนการทำงานของระบบ	35

สารบัญ (ต่อ)

บทที่	หน้า
4. การทดลอง และผลการทดลอง	36
4.1 การทำงานของโปรแกรม	36
4.2 การทดลองใช้งาน และปรับค่าโปรแกรม	49
5. สรุปผล และข้อเสนอแนะ	67
5.1 สรุปผลการดำเนินงาน	67
5.2 สรุปผลการทดลอง	67
5.3 ข้อเสนอแนะ	68
บรรณานุกรม	70



สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงความสัมพันธ์ระหว่างเซลล์ประสาทจริงกับเซลล์ประสาทเทียม	10
3.1 แสดงค่าสมมุติของข้อมูลนำเข้า	25
3.2 สรุปค่าถ่วงน้ำหนักทั้งหมดของพาดิคัลที่เริ่มสมมุติขึ้น	27
3.3 สรุปค่าผลลัพธ์ที่ออกมาจากโครงข่ายประสาทเทียมเมื่อแทนค่าตำแหน่งปัจจุบันของทั้ง 4 พาดิคัลเข้าไป	28
3.4 สรุปค่า Error ของแต่ละพาดิคัล	28
3.5 สรุปค่า Fitness ของแต่ละพาดิคัล	29
3.6 สรุปค่าความเร็วเมื่อแทนค่าสูตร	30
3.7 สรุปค่าความเร็วเมื่อจำกัดค่าความเร็วสูงสุด	30
3.8 สรุปค่าตำแหน่งใหม่ของแต่ละพาดิคัล	31
3.9 สรุปค่าผลลัพธ์ที่ออกมาจากโครงข่ายประสาทเทียมเมื่อแทนค่าตำแหน่งปัจจุบันของทั้ง 4 พาดิคัลเข้าไปรอบที่ 2	31
3.10 สรุปค่า Error ของแต่ละพาดิคัลรอบที่ 2	32
3.11 สรุปค่า Fitness ของแต่ละพาดิคัลรอบที่ 2	32
3.12 แสดงค่าสมมุติของข้อมูลนำเข้า	33
3.13 ผลลัพธ์ที่ออกมาจากโครงข่ายประสาทเทียมเมื่อแทนค่าถ่วงน้ำหนักในขั้นตอนทดสอบ	34
4.1 แสดงผลการทดลองที่ 1 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรมเพื่อใช้ในการวิเคราะห์	52
4.2 แสดงผลการทดลองที่ 1 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรมเพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า	53
4.3 แสดงผลการทดลองที่ 2 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรมเพื่อใช้ในการวิเคราะห์	55
4.4 แสดงผลการทดลองที่ 2 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรมเพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า	56

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
4.5 แสดงผลการทดลองที่ 3 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรม เพื่อใช้ในการวิเคราะห์	58
4.6 แสดงผลการทดลองที่ 3 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรม เพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า	59
4.7 แสดงผลการทดลองที่ 4 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรม เพื่อใช้ในการวิเคราะห์	61
4.8 แสดงผลการทดลองที่ 4 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรม เพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า	62
4.9 แสดงผลการทดลองที่ 5 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรม เพื่อใช้ในการวิเคราะห์	64
4.10 แสดงผลการทดลองที่ 5 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรม เพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า	65

สารบัญภาพ

ภาพที่	หน้า
2.1 แสดงข้อมูลอนุกรมเวลา แบบ Seasonal	7
2.2 แสดงข้อมูลอนุกรมเวลาแบบ Trend	7
2.3 แสดงข้อมูลอนุกรมเวลาแบบ Cyclical	8
2.4 แสดงข้อมูลอนุกรมเวลา แบบ Irregular	8
2.5 ลักษณะของเซลล์ประสาทในสมองมนุษย์	10
2.6 แสดงแบบจำลองเซลล์ประสาทเทียม	11
2.7 โครงสร้างของโครงข่ายประสาทเทียมแบบ Feed Forward Multilayer	12
2.8 แสดงตัวอย่างภาพการทำงานของ PSO อัลกอริทึม	15
2.9 แสดงตัวอย่างลักษณะของการปรับค่าความเร็วของพาดิคัล	16
2.10 แสดง Pseudocode ของอัลกอริทึม PSO	17
2.11 สถาปัตยกรรมของ Feed forward Neural Network	18
3.1 แสดงสถาปัตยกรรมของระบบที่ใช้ โครงข่ายประสาทเทียมร่วมกับ PSO เพื่อหาค่าที่เหมาะสมให้กับค่า ค่าถ่วงน้ำหนักของ โครงข่ายประสาทเทียม.	24
3.2 แสดงตัวอย่างโครงข่ายที่สมมุติขึ้น	26
3.3 ลักษณะพื้นฐานของพาดิคัล	26
3.4 แสดงตัวอย่างโครงข่ายสำหรับการทดสอบ	33
3.5 แสดงตัวอย่างโครงข่ายสำหรับการทดสอบเมื่อใส่ค่าข้อมูล Pattern ที่ 1 ...	33
3.6 แสดงตัวอย่างโครงข่ายสำหรับการทดสอบเมื่อใส่ค่าข้อมูล Pattern ที่ 2 ...	34
3.7 แสดงแผนภาพสรุปขั้นตอนการทำงานของระบบ	35
4.1 แสดงหน้าที่ 1 : Neural Swarm Prediction	36
4.2 แสดงหน้าที่ 1.1 : About	37
4.3 แสดงหน้าที่ 2 : หน้าจอหลักของโปรแกรม	38
4.4 แสดงหน้าที่ 2.1 : หน้าจอหลักของโปรแกรม (เมนู File)	39
4.5 แสดงหน้าที่ 2.1.1 : หน้าจอนำเข้าค่า	40
4.6 แสดงหน้าที่ 2.2 : หน้าจอหลักของโปรแกรม (เมนู Setup)	41

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
4.7 แสดงหน้าที่ 2.2.1 : หน้าจอ Setup Normal Mode	42
4.8 แสดงหน้าที่ 2.2.2 : หน้าจอ Setup Advance Mode	43
4.9 แสดงหน้าที่ 2.3 : หน้าจอหลักของโปรแกรม (เมนู Run)	44
4.10 แสดงหน้าที่ 2.3.1 : หน้าจอ Training	45
4.11 แสดงหน้าที่ 2.3.2 : หน้าจอ Testing	46
4.12 แสดงหน้าที่ 2.3.2.1 : Graph	47
4.13 แสดงหน้าที่ 2.3.2.2 : Save Best Weight Training	48
4.14 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 1	54
4.15 แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 1	54
4.16 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 2	57
4.17 แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 2	57
4.18 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 3	60
4.19 แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 3	60
4.20 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 4	63
4.21 แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 4	63
4.22 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 5	66

สารบัญญภาพ (ต่อ)

ภาพที่		หน้า
4.23	แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ ของการทดลองที่ 5	66



บทที่ 1

บทนำ

1.1 ความเป็นมา

เนื่องจากในปัจจุบันหน่วยงาน และองค์กรต่างๆทั้งภาครัฐบาล ภาคเอกชน ไม่ว่าจะเป็นธุรกิจการเงิน การขายปลีก ธุรกิจการประกันภัย การรักษาพยาบาล งานบริการของภาครัฐบาล ตลอดจนงานที่เกี่ยวข้องกับความมั่นคงของประเทศได้มีการพัฒนาเทคโนโลยีทางด้านฐานข้อมูลเพื่อใช้ประโยชน์อย่างกว้างขวาง จึงเป็นสิ่งที่ไม่ยากนักที่จะเก็บข้อมูลดิบจำนวนมากลงในคอมพิวเตอร์อย่างมีประสิทธิภาพ และยังสามารถดึงข้อมูลกลับออกมาเมื่อใดก็ได้เท่าที่ต้องการ ประกอบกับปัจจุบันนี้ได้มีองค์กรถือกำเนิดใหม่ขึ้นเป็นจำนวนมาก อันส่งผลให้เกิดแรงกดดันจากการแข่งขันกันระหว่างองค์กร ได้มีเพิ่มขึ้นอย่างรวดเร็ว เป็นเหตุให้หลายๆองค์กรจำเป็นต้องพยายามทำทุกวิถีทางที่จะพัฒนาองค์กร และผลผลิตขององค์กรให้มีศักยภาพมากที่สุด

การพยากรณ์ (Prediction) เป็นวิธีหนึ่งที่ได้รับคามนิยมที่ในหลายๆองค์กรใช้ในการวิเคราะห์จากข้อมูลที่จัดเก็บไว้ ทั้งนี้เนื่องจาก หากว่าองค์กรใดสามารถทราบสิ่งที่จะเกิดขึ้นได้ก่อน ผู้บริหารองค์กรนั้นๆ ก็จะสามารถตัดสินใจ วางแผน และบริหารดำเนินการได้อย่างถูกต้อง และมีประสิทธิภาพมากขึ้น เช่นการทำนายความต้องการสินค้าของลูกค้า เพื่อวางแผนการตลาด การวางมาตรการลดความเสี่ยง การลดต้นทุนการดำเนินงาน หรือการเพิ่มโอกาสในการแสวงหากำไร เป็นต้น ดังนั้นแนวความคิดทางด้านการสร้างโปรแกรมเพื่อการพยากรณ์จึงเกิดขึ้นภายใต้ชื่อว่า การพยากรณ์โดยใช้โครงข่ายประสาทเทียมร่วมกับ Particle Swarm Optimization

การพยากรณ์โดยใช้โครงข่ายประสาทเทียมร่วมกับ Particle Swarm Optimization (PSO) นี้เราจะใช้วิธีการของโครงข่ายประสาทเทียม (Neural Network) โดยแทนที่จะเป็นแบบแพร่ย้อนกลับ (Back Propagation) เพื่อปรับปรุงค่าถ่วงน้ำหนักวนไปเรื่อยๆ จนกว่าจะได้รับความถูกต้องจนเราพอใจ นั้น ในโครงงานนี้เราได้ใช้วิธีการของ Particle Swarm Optimization (PSO) ซึ่งเป็นอีกอัลกอริทึมหนึ่ง เพื่อนำมาประยุกต์เข้าช่วยหาค่าถ่วงน้ำหนักที่ถูกต้องของโครงข่ายประสาทเทียมแทนการแพร่ย้อนกลับ โดยอาศัยหลักการการเลียนแบบพฤติกรรมตามธรรมชาติของกลุ่มสัตว์ที่อยู่กันเป็นกลุ่ม และทางผู้จัดทำหวังว่าวิธีที่เราจะใช้โครงข่ายประสาทเทียมร่วมกับ Particle Swarm Optimization นี้จะเป็นวิธีที่มีประสิทธิภาพที่ตีมากกว่าวิธีหนึ่งที่ใช้ในการพยากรณ์

1.2 วัตถุประสงค์ของโครงการ

การศึกษาโครงการพัฒนาระบบงานการพยากรณ์โดยใช้โครงข่ายประสาทเทียมร่วมกับ Particle Swarm Optimization มีวัตถุประสงค์ คือ

1.2.1 เพื่อศึกษาเปรียบเทียบขบวนการในการเรียนรู้ และการพยากรณ์ จากวิธีการ โครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ (Back Propagation Neural Network) และการพยากรณ์โดยใช้โครงข่ายประสาทเทียมร่วมกับ PSO

1.2.2 เพื่อค้นหาข้อดี-ข้อเสีย ของการเรียนรู้และพยากรณ์จากวิธีการของโครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ และการพยากรณ์โดยใช้โครงข่ายประสาทเทียมร่วมกับ PSO

1.2.3 เพื่อพิจารณาค้นหาตัวแปรต้นที่สำคัญที่ส่งผลต่อขบวนการในการเรียนรู้และการพยากรณ์ จากวิธีการการพยากรณ์โดยใช้โครงข่ายประสาทเทียมร่วมกับ PSO

1.2.4 เพื่อเข้าใจอัลกอริทึม และสามารถเขียน โปรแกรมการใช้งาน โครงข่ายประสาทเทียมได้อย่างถูกต้องและมีประสิทธิภาพ

1.2.5 เพื่อได้ศึกษาเข้าใจ โมเดลของโครงข่ายประสาทเทียมที่นำมาใช้การพยากรณ์ และนำไปใช้ในการวิเคราะห์ข้อมูลต่อไป

1.3 ขอบเขตของการศึกษา

การศึกษาโครงการพัฒนาระบบงานนี้มีขอบเขตของการศึกษา คือ เป็นการพัฒนาโปรแกรมที่ใช้ในการพยากรณ์ผลที่จะเกิดขึ้นหลังที่เกิดเหตุการณ์ตามสถานการณ์ต่างๆ ได้ เช่นการพยากรณ์ดัชนีในตลาดหลักทรัพย์ การพยากรณ์พฤติกรรมของลูกค้า เป็นต้น โดยจะใช้หลักการของโครงข่ายประสาทเทียมร่วมกับ PSO ซึ่งนำมาพัฒนาระบบการพยากรณ์ผลที่จะเกิดขึ้น เพื่อช่วยในการตัดสินใจดำเนินงานให้เป็นอย่างมั่นคง และมีประสิทธิภาพ

1.4 ขั้นตอนและวิธีการดำเนินงาน

การศึกษาโครงการพัฒนาระบบงานนี้มีขั้นตอนและวิธีการดำเนินงาน คือ

1. ศึกษาหลักการและเทคนิคของโครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ
2. ศึกษาหลักการและเทคนิคของ PSO
3. ศึกษาหลักการและเทคนิคของการทำงานร่วมกันของ โครงข่ายประสาทเทียม และ PSO
4. ศึกษาภาษาโปรแกรม JAVA
5. รวบรวมข้อมูลสถิติของข้อมูลที่จะนำมาทดสอบ
6. ออกแบบและพัฒนาระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ทดสอบการใช้งานของระบบ
8. สรุปผลการศึกษาและการดำเนินงาน

1.5 ผลที่คาดว่าจะได้รับ

การศึกษาโครงการพัฒนาระบบงานนี้มีผลที่คาดว่าจะได้รับคือ

1. เพื่อเข้าใจหลักการและเทคนิคของโครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ
2. เพื่อเข้าใจหลักการและเทคนิคของ PSO
3. เพื่อเข้าใจหลักการและเทคนิคของ ภาษาโปรแกรม JAVA ที่ใช้ในการเขียนโปรแกรมโครงข่ายประสาทเทียม และ อัลกอริทึมของ PSO
4. เพื่อนำระบบงานนี้ไปช่วยในการตัดสินใจให้เป็นไปอย่างมั่นคง และมีประสิทธิภาพ



บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

ในการพัฒนาระบบสารสนเทศสำหรับการทำงานอนุกรมเวลาโดยใช้โครงข่ายประสาทเทียมร่วมกับ Particle Swarm Optimization ได้มีการนำทฤษฎีในการวิเคราะห์และออกแบบระบบต่าง ๆ มาใช้ดังต่อไปนี้

2.1 การพยากรณ์

การพยากรณ์ คือการประมาณหรือการคาดคะเนถึงเหตุการณ์ หรือสถานการณ์ต่าง ๆ ในอนาคต การพยากรณ์จะมีบทบาทสำคัญกับทุกด้าน โดยเฉพาะอย่างยิ่งจะมีความสำคัญต่อการวางแผน และการตัดสินใจเกี่ยวกับการดำเนินงานของทุกองค์กร เช่นองค์กรด้านธุรกิจ, องค์กรด้านอุตสาหกรรม รวมถึงองค์กรด้านการเมือง เป็นต้น การกำหนดปัญหา เป็นการกำหนดขอบเขตของปัญหาจากการดำเนินงานในปัจจุบัน ความเป็นไปได้ในการสร้างระบบใหม่ กำหนดความต้องการระหว่างนักวิเคราะห์ระบบ กับผู้ใช้งาน โดยข้อมูลได้จากการสัมภาษณ์ผู้ดูแลระบบและผู้ใช้งานระบบ โดยจะประกอบไปด้วยขั้นตอนหลักๆของการพยากรณ์ดังต่อไปนี้

ขั้นตอนที่ 1 กำหนดปัญหา

การกำหนดปัญหาจะต้องมีการทำความเข้าใจ และกำหนดว่าต้องการพยากรณ์ในเรื่องใด เช่น ต้องการพยากรณ์ความสัมพันธ์ของรายได้ในองค์กร กับค่าโฆษณาที่ทางองค์กรที่ต้องเสียไป หรืออาจเป็นการพยากรณ์ราคาข้าวที่บริโภคภายในประเทศไทย เป็นต้น

ขั้นตอนที่ 2 การศึกษาและรวบรวมข้อมูล

ทำการศึกษาและรวบรวมข้อมูลต่าง ๆ ที่ต้องการใช้ในการพยากรณ์ โดยทำการเลือกข้อมูลที่เกี่ยวข้องกับปัญหาที่ได้กำหนดไว้แล้วในขั้นตอนที่ 1 ดังนั้นจึงต้องมีการทำความเข้าใจข้อมูลเหล่านั้นเนื่องจากในการพยากรณ์บางกรณีนั้นมีการใช้ข้อมูลในอดีต

ขั้นตอนที่ 3 การวิเคราะห์ข้อมูล

การวิเคราะห์ข้อมูล เนื่องจากข้อมูลที่ได้ทำการรวบรวมมานั้นอาจจะมาจากหลายแหล่ง ซึ่งในแต่ละแหล่งนั้นก็อาจจะมึรูปแบบในการจัดเก็บ และการอ้างถึงข้อมูลนั้นแตกต่างกัน ส่งผลให้เมื่อนำข้อมูลที่ไม่ได้ทำการวิเคราะห์ดังกล่าวไปใช้ในการพยากรณ์อาจได้ค่าที่ไม่ถูกต้อง

เมื่อทำการวิเคราะห์ข้อมูลแล้ว จะนำชุดข้อมูลนั้นมาหาความสัมพันธ์ ซึ่งจะเป็นประโยชน์ในการพยากรณ์ เนื่องจากในบางตัวแบบนั้นจะพิจารณาใช้ความสัมพันธ์ดังกล่าวร่วมในการพยากรณ์ด้วย

ขั้นตอนที่ 4 การเลือกตัวแบบ (Model)

ทำ การเลือกตัวแบบ (Model) ที่ใช้ในการพยากรณ์ให้สอดคล้องกับข้อมูล

ขั้นตอนที่ 5 การพยากรณ์

ทำการพยากรณ์ตามตัวแบบ (Model) ที่ได้เลือกไว้กับชุดข้อมูลที่ทำการศึกษาหรือทำการรวบรวมเอาไว้

2.1.1 ทฤษฎีการพยากรณ์

ในปัจจุบันได้มีการพัฒนาตัวแบบ หรือเทคนิคต่างๆ ที่ใช้ในการพยากรณ์โดยที่แต่ละเทคนิคนั้นก็เหมาะสมกับเหตุการณ์หรือสถานการณ์ รวมถึงชุดข้อมูลในรูปแบบที่แตกต่างกัน ดังนั้นผลที่ได้จากการพยากรณ์ก็จะมี ความแตกต่างกันขึ้นอยู่กับเทคนิค หรือตัวแบบที่ผู้พยากรณ์เลือกใช้กับชุดข้อมูลนั้น ๆ

สำหรับขั้นตอนในการสร้างระบบเพื่อใช้ในการพยากรณ์นั้น จะต้องทำการเลือกตัวแบบ (Model Specification) ขึ้นมาเพื่อจัดการกับข้อมูลที่ได้รวบรวมไว้ จากนั้นจะทำการประมาณค่าแบบที่เลือกไว้ (Model Estimation) เพื่อตรวจสอบว่าตัวแปรนั้นเหมาะสมกับข้อมูลที่ต้องการพยากรณ์หรือไม่ ถ้าตัวแบบดังกล่าวยังไม่เหมาะสมก็จะกลับไปในขั้นตอนการระบุตัวแบบใหม่อีกครั้งแต่ถ้าเป็นตัวแบบที่เหมาะสมแล้วก็จะผ่านไปยังขั้นตอนของการพยากรณ์ (Forecast Generation) โดยอาจมีการตรวจสอบว่าเป็นตัวแบบนั้นทำให้ผลลัพธ์ที่ได้นั้นเสถียรหรือไม่ (Stable) ถ้ายังไม่เสถียรก็จะกลับไปสู่ขั้นตอนในการเลือกตัวแบบใหม่อีกครั้ง แต่ถ้าเป็นตัวแบบที่ให้ค่าผลลัพธ์ที่เหมาะสม และถูกต้องแล้วนั้นก็ถือว่าตัวแบบที่เลือกมาใช้ในการพยากรณ์นั้นมีความเหมาะสมกับข้อมูล ทำให้ได้ระบบการพยากรณ์ที่สามารถนำมาใช้งานได้จริง (Forecasting Updating) โดยสำหรับเทคนิคการพยากรณ์ในปัจจุบันที่นิยมใช้ สามารถแบ่งได้ 2 ประเภท ได้แก่

1. เทคนิคด้านสถิติ (Statistic Technique)
2. เทคนิคด้านการเรียนรู้ (Artificial Intelligent Technique)

โดยมีรายละเอียดดังต่อไปนี้

2.1.1.1 เทคนิคด้านสถิติ (Statistic Technique)

เทคนิคประเภทนี้เป็นการพยากรณ์โดยใช้ความรู้ และความเข้าใจทางด้านสถิติ ซึ่งสามารถแบ่งวิธีในการพยากรณ์เป็น 2 แบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การพยากรณ์เชิงคุณภาพ(Qualitative Forecasting) มีลักษณะเป็นการพยากรณ์ที่ใช้ความรู้ความสามารถรวมถึงประสบการณ์ของผู้พยากรณ์โดยไม่ใช้ตัวแบบ (Model) ทางคณิตศาสตร์สามารถแบ่งตัวแบบที่ใช้พยากรณ์ ดังนี้

- การคาดคะเน (Judgment) โดยการพยากรณ์วิธีนี้จะเหมาะสมกับธุรกิจขนาดเล็ก และผู้พยากรณ์ต้องมีประสบการณ์เกี่ยวกับงานด้านนี้เป็นระยะเวลานานพอ
- การระดมความคิด เป็นการระดมความคิด หรือประชุมกลุ่มผู้บริหารขององค์กร ซึ่งผลลัพธ์จะได้จากเสียงส่วนใหญ่ ทำให้เกิดผลเสียที่อาจเกิดจากความเอนเอียง หรือเกรงใจ
- พยากรณ์โดยการสำรวจตลาด เป็นการพยากรณ์โดยทำการสำรวจลูกค้า หรือผู้ที่คาดว่าจะ是客户 เพื่อตรวจสอบความต้องการของกลุ่มลูกค้า ซึ่งวิธีนี้จะสิ้นเปลืองค่าใช้จ่าย และเวลา

2. การพยากรณ์เชิงปริมาณ (Quantitative Forecasting) เป็นการพยากรณ์ที่ใช้ข้อมูลเชิงปริมาณ (ตัวเลข) ที่ทำการเก็บรวบรวมไว้ในอดีตมาพยากรณ์ค่าในอนาคต โดยสร้างตัวแบบทางคณิตศาสตร์ซึ่งสามารถแบ่งตัวแบบที่ใช้พยากรณ์ได้ดังนี้

- Explanatory Model เป็นตัวแบบที่ใช้พยากรณ์หาความสัมพันธ์ระหว่างตัวแปรที่ต้องการพยากรณ์กับตัวแปรอื่นๆ โดยมีจุดมุ่งหมายเพื่อค้นหารูปแบบของความสัมพันธ์ และใช้รูปแบบนั้นพยากรณ์ค่าในอนาคต เช่น การวิเคราะห์ความถดถอย (Regression Analysis)
- Time Series Model เป็นตัวแบบที่ใช้พยากรณ์ข้อมูลในอนาคตโดยใช้พื้นฐานของตัวแปรในอดีต หรือค่าความผิดพลาดที่ผ่านมา ซึ่งตัวแบบนี้มีจุดประสงค์เพื่อค้นหารูปแบบในชุดข้อมูลที่เข้ามา โดยไม่พิจารณาตัวแปรอื่น ๆ แต่จะทำการแยกข้อมูลอนุกรมเวลาออกเป็น ส่วน ๆ แล้วจึงทำการวิเคราะห์แต่ละส่วนนั้น เช่น การวิเคราะห์อนุกรมเวลา (Time Series Analysis) การวิเคราะห์ค่าเฉลี่ยเคลื่อนที่ (Moving Averages Analysis) เป็นต้น

2.1.1.2 เทคนิคด้านการเรียนรู้ (Artificial Intelligent Technique)

เทคนิคประเภทนี้จะใช้วิธีการทำงานแบบการเรียนรู้ กล่าวคือ มีการรับข้อมูลเข้าไปสะสมจากหลายๆ เซลล์แล้วนำข้อมูลเหล่านั้นมาผ่านกระบวนการ เพื่อปรับเปลี่ยนข้อมูลให้ได้ตามต้องการ และส่งออกไปยังเซลล์ประสาทอื่นๆ ซึ่งเรียกกระบวนการนี้ว่า การบวนการเรียนรู้ สำหรับอัลกอริทึมที่นำมาใช้ในการพยากรณ์ด้านนี้เช่น โครงข่ายประสาทเทียม ซึ่งเป็นอัลกอริทึมที่สามารถใช้ในการพยากรณ์ได้หลากหลายรูปแบบนั้น ได้นำกระบวนการเรียนรู้ดังกล่าวทำการเปรียบเทียบค่าเอาต์พุตที่ได้กับค่าเป้าหมายที่ทราบ เพื่อปรับปรุงประสิทธิภาพในการทำงาน

2.2 การวิเคราะห์อนุกรมเวลา (Time Series Data)

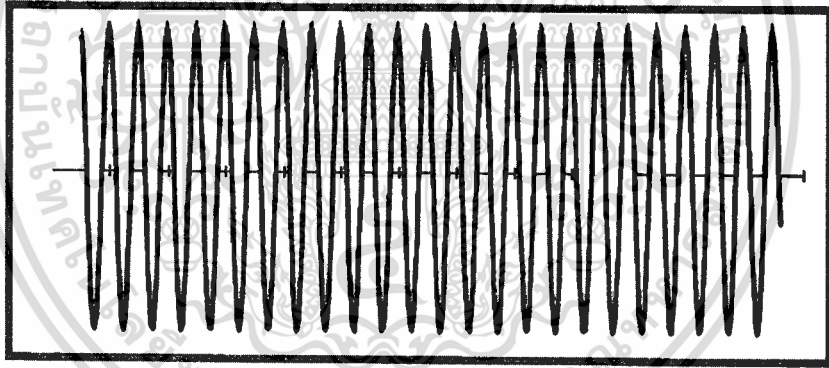
การวิเคราะห์ข้อมูลอนุกรมเวลา (Time Series Data) คือข้อมูลที่เกิดขึ้นในระยะเวลาเท่า ๆ กันและต่อเนื่องกันข้อมูลอนุกรมเวลาจึงถือเป็นค่าที่แสดงการเปลี่ยนแปลงของตัวแปรใดตัวแปรหนึ่งที่เปลี่ยนไปตามเวลา เช่น ราคาหุ้นของธนาคารต่าง ๆ

สำหรับการวิเคราะห์อนุกรมเวลาคือ การศึกษาหารูปแบบการเปลี่ยนแปลงของตัวแปรที่เปลี่ยนไปตามเวลาในอดีตจนถึงปัจจุบัน แล้วนำรูปแบบนั้นมาวิเคราะห์ เพื่อพยากรณ์ค่าของตัวแปรนั้นในอนาคต โดยจะถือว่าปัจจัยที่เกี่ยวข้องในอนาคตเหมือนหรือใกล้เคียงกับของในอดีต

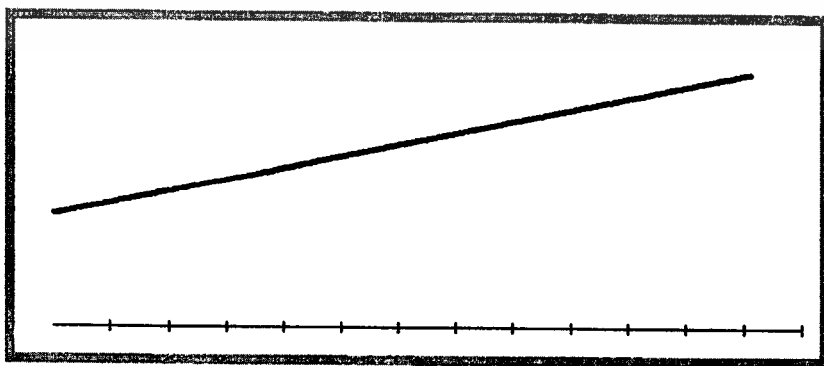
2.2.1 ส่วนประกอบของอนุกรมเวลา

สำหรับการวิเคราะห์อนุกรมเวลานั้นจะมีการพิจารณารูปแบบที่เปลี่ยนแปลงไปตามเวลา โดยสามารถแบ่งแยกข้อมูลอนุกรมเวลาได้เป็น 4 ส่วน ดังต่อไปนี้

1. Seasonal เป็นการเปลี่ยนแปลงของข้อมูลที่เกิดจากอิทธิพลของฤดูกาล ซึ่งจะเกิดขึ้นอย่างสม่ำเสมอในช่องเวลาเดียวกันในแต่ละปี ดังรูปที่ 2.1



รูปที่ 2.1 แสดงข้อมูลอนุกรมเวลาแบบ Seasonal

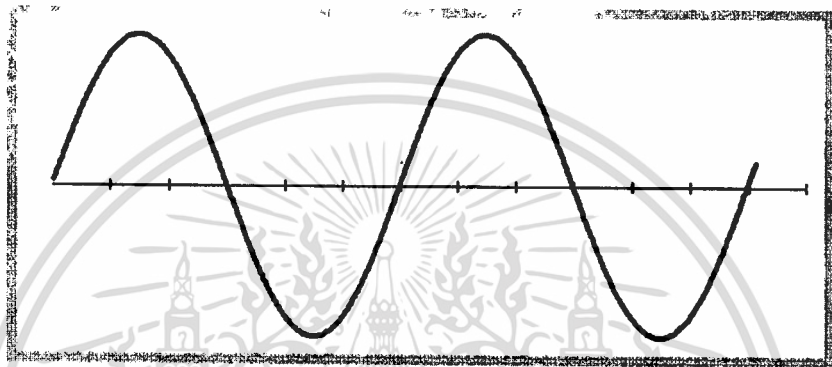


รูปที่ 2.2 แสดงข้อมูลอนุกรมเวลาแบบ Trend

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ผู้เห็นนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

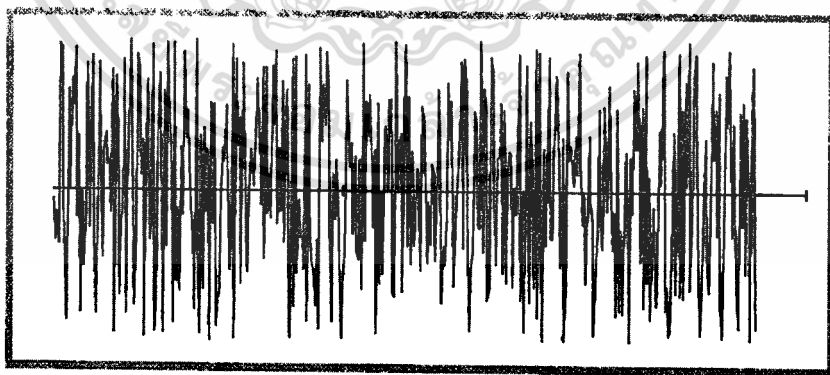
2. Trend เป็นข้อมูลที่มีการเปลี่ยนแปลงระยะยาว และสามารถเห็นแนวโน้มของข้อมูลที่จะเพิ่มขึ้นหรือลดลงในอนาคต ดังรูปที่ 2.2

3. Cyclical มีลักษณะคล้ายกับ Seasonal แต่จะรอบคลุมระยะเวลาหลายปี โดยแต่ละ Cyclical นั้นอาจมีระยะเวลาเท่ากันหรือไม่ก็ได้ รวมถึงการเคลื่อนไหวของข้อมูลนั้นจะไม่แตกต่างกันมากนัก ดังรูปที่ 2.3



รูปที่ 2.3 แสดงข้อมูลอนุกรมเวลาแบบ Cyclical

4. Irregular เป็นการเปลี่ยนแปลงของข้อมูลที่ไม่ได้คาดคิดมาก่อน ดังนั้นจึงเป็นการยากที่จะพยากรณ์ได้จากข้อมูลในอดีต ดังรูปที่ 2.4



รูปที่ 2.4 แสดงข้อมูลอนุกรมเวลา แบบ Irregular

โดยทั่วไปแล้วข้อมูลแต่ละชุดอาจมีข้อมูลไม่ครบทั้ง 4 ส่วนก็ได้ เช่น ข้อมูลรายปี จะไม่มีข้อมูลในส่วน of Seasonal ดังนั้นสำหรับข้อมูลชุดนี้จึงไม่จำเป็นต้องใช้ Seasonal ในการพิจารณา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

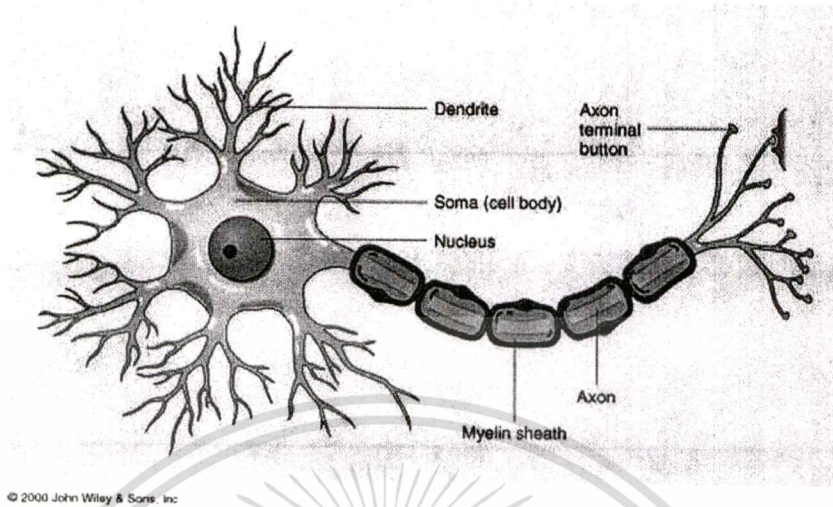
2.2.2 ขั้นตอนการสร้างรูปแบบอนุกรมเวลา

เราสามารถหาค่าพยากรณ์ในอนาคตจากการศึกษาข้อมูลของตัวแปรนั้นในอดีต เช่นการหารูปแบบของข้อมูลแล้วนำรูปแบบนั้นไปพยากรณ์ค่าในอนาคต ขั้นตอนการสร้างรูปแบบมีดังนี้

1. กำหนดหรือเลือกรูปแบบที่เหมาะสมกับข้อมูลอนุกรมเวลาที่ศึกษา โดยใช้ความรู้ทางด้านสถิติและประสบการณ์ในอดีต หรือทำการเปรียบเทียบกับข้อมูลอื่นที่มีลักษณะคล้ายกัน
2. ประมาณค่าพารามิเตอร์ของรูปแบบที่ได้จากขั้นที่ 1
3. ทดสอบความถูกต้องเหมาะสมของรูปแบบที่ได้จากขั้นที่ 1 โดยใช้หลักการทางสถิติมาทำสอบถ้าสรุปผลการทดสอบได้ว่าเหมาะสมก็สามารถนำรูปแบบนั้นไปใช้การพยากรณ์ได้ แต่ถ้าสรุปได้ว่าไม่เหมาะสม ต้องสร้างรูปแบบใหม่ โดยต้องย้อนกลับไปทำขั้นตอนที่ 1 ใหม่

2.3 โครงข่ายประสาทเทียม (Artificial Neural Network)

โครงข่ายประสาทเทียม (Neural Networks) เป็นวิทยาการแขนงหนึ่งทางปัญญาประดิษฐ์ (Artificial Intelligence) ที่พยายามลอกเลียนแบบการทำงานของเซลล์ประสาทในสมองมนุษย์ เรียกว่า เซลล์ประสาทเทียม (Artificial Neural) หรือยูนิต (Unit) ซึ่งเซลล์ประสาทเทียมที่จำลองขึ้นนั้นจะมีความเร็วในการประมวลผลที่สูงกว่าเซลล์ประสาทจริงในสมองมนุษย์ แต่เซลล์ประสาทในสมองมนุษย์สามารถทำงานได้หลากหลายและมีความสลับซับซ้อนได้ดีและเร็วกว่าเซลล์ประสาทเทียมในคอมพิวเตอร์ เนื่องจากโครงสร้างของเซลล์ประสาทมนุษย์นั้นมีโครงสร้างที่ขนานกันอย่างมาก (Massively Parallel Structure) ซึ่งการจำลองการทำงานของเซลล์ประสาทเทียมนั้น ได้อาศัยโครงสร้างที่ประกอบไปด้วยหน่วยประมวลผล (Processing Element) ที่เรียกว่า นิวรอล (Neural) จำนวนมากเชื่อมต่อกัน (Connection) ทำให้การทำงานของโครงข่ายประสาทเทียมนั้นเป็นไปในรูปแบบของการประมวลผลพร้อมๆ กันจำนวนมาก (Massively Parallel Processing) โดยการเชื่อมต่อถึงกันของหน่วยประมวลผลนั้นจะมีค่าน้ำหนัก (Weight) ของแต่ละการเชื่อมต่อและเมื่อมีการให้ตัวอย่างที่ใช้ในการเรียนรู้ โครงข่ายประสาทเทียมก็จะทำการปรับค่าน้ำหนักให้เหมาะสมจนได้ผลลัพธ์ที่ถูกต้องหรือมีข้อผิดพลาดน้อยที่สุด และสามารถนำค่าน้ำหนักที่ถูกปรับนี้ไปใช้ในการเรียนรู้ครั้งต่อไปได้



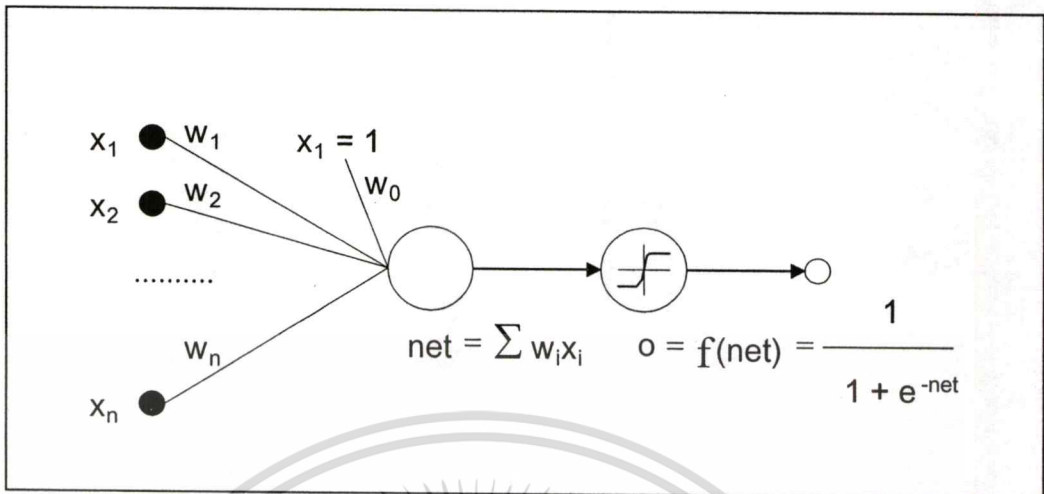
รูปที่ 2.5 ลักษณะของเซลล์ประสาทในสมองมนุษย์

เซลล์ประสาทจริง	เซลล์ประสาทเทียม
ตัวเซลล์ (Cell Body)	ยูนิต (Unit) หรือนิวรอน (Neural)
เดนไดรต์ (Dendrites)	ข้อมูลเข้า (Input Layer)
แอกซอน (Axon)	ข้อมูลออก (Output)
ไซแนปส์ (Synapse)	ค่าถ่วงน้ำหนัก (Weight)
ความเร็วในการประมวลผลช้ากว่า	ความเร็วในการประมวลผลเร็วกว่า
เซลล์จำนวนมาก (ประมาณ 10 ¹¹ เซลล์)	เซลล์จำนวนน้อย (ประมาณ 100 ยูนิต)

ตารางที่ 2.1 ความสัมพันธ์ระหว่างเซลล์ประสาทจริงกับเซลล์ประสาทเทียม

เมื่อเราพิจารณาจากรูปจะพบว่าเราพยายามที่จะจำลองการทำงานของเซลล์ประสาทของสิ่งมีชีวิตโดยการแทนด้วยโครงสร้างของชุดวิธีการคำนวณ โดยที่หนึ่งหน่วยของชุดการคำนวณนั้นจะรับข้อมูลนำเข้ามาจากข้อมูลนำออกของชุดการคำนวณหน่วยอื่นๆและต่อกันไปจนเป็นโครงข่ายเพื่อใช้ในการเรียนรู้และจดจำได้ ซึ่งโดยสรุปก็คือแบบจำลองของเซลล์ประสาทเทียมหนึ่งเซลล์ที่สร้างขึ้นนั้นเพื่อจำลองการทำงานของเซลล์ประสาทของสิ่งมีชีวิตนั้นจะประกอบไปด้วย 3 ส่วนหลักดังต่อไปนี้ ตามรูป 2.6 ได้แก่

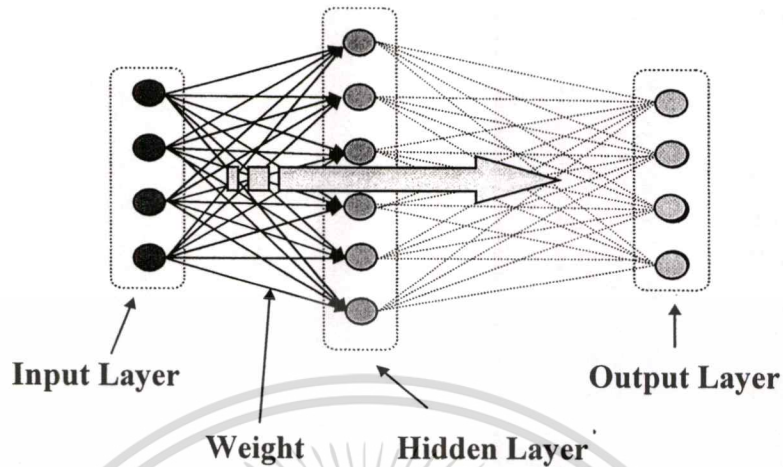
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงแบบจำลองเซลล์ประสาทเทียม

1. เดนไดรต์ (Dendrites) และ ซิแนปส์ (Synapse) คือส่วนที่มีการรับข้อมูลอินพุต $X_1, X_2, X_3, \dots, X_n$ และบริเวณที่ระบุค่าถ่วงน้ำหนัก (Weight) ตัว $W_1, W_2, W_3, \dots, W_n$ ตามรูป
2. ตัวเซลล์ส่วนหาผลบวก (Summation) ทำหน้าที่รวบรวมค่าผลคูณระหว่าง ข้อมูลอินพุตทั้งหมดที่ได้รับเข้ามา และค่าถ่วงน้ำหนักที่มีอยู่
3. ตัวเซลล์ส่วนแปลงค่าเพื่อส่งให้เซลล์อื่นต่อไป โดยการผ่านฟังก์ชันกระตุ้น (Activate Function) จะเปรียบเสมือน เอกซอน (Axon) ที่ทำหน้าที่เหมือนสายส่งสัญญาณที่ทำการนำสัญญาณจากเซลล์ประสาทมาแปลงเป็นสัญญาณกระตุ้นเพื่อส่งต่อไปยังโหนดอื่นต่อไป

โครงข่ายประสาทเทียมนั้นจะเป็นการประกอบกันของเซลล์ประสาทเทียมมากกว่าหนึ่งเซลล์ขึ้นไป มีอยู่หลายแบบจากแบบง่ายไปจนถึงแบบที่ซับซ้อนมากๆ ซึ่งโดยส่วนใหญ่แล้วโครงข่ายประสาทเทียมที่เรานำมาใช้ในการทำนายกันคือโครงข่ายประสาทเทียมแบบ Feedforward และเป็นแบบมีหลายชั้น (Multilayer) ซึ่งตัวเซลล์ประสาทเทียมของระดับชั้นหนึ่งจะเชื่อมต่อและป้อนข้อมูลให้กับเซลล์ประสาทเทียมของชั้นต่อมาโดยไม่มีการย้อนกลับของข้อมูล โดยทั่วไปโครงข่ายจะประกอบไปด้วยระดับชั้นข้อมูลนำเข้า (Input Layer) หนึ่งชั้น ระดับชั้นซ่อนเร้น (Hidden Layer) หนึ่งหรือสองชั้น และระดับชั้นข้อมูลออก (Output Layer) หนึ่งชั้น ซึ่งจะมีรายละเอียดตาม รูปที่ 2.7 ดังนี้



รูปที่ 2.7 โครงสร้างของโครงข่ายประสาทเทียมแบบ Feed Forward Multilayer

1. ระดับชั้นข้อมูลเข้า (Input Layer) จะรับสัญญาณจากภายนอก โดยไม่ได้ดำเนินการกับสัญญาณเอง เพียงแต่ทำหน้าที่ส่งสัญญาณไปยังระดับชั้นของเซลล์ประสาทที่อยู่ถัดมา จำนวนของเซลล์ประสาทเทียมในชั้นนี้
2. ระดับชั้นกลาง หรือเรียกอีกอย่างว่าชั้นซ่อนเร้น (Hidden Layer) โดยที่ชั้นซ่อนเร้นถูกเรียกเช่นนี้เนื่องจากชั้นนี้จะไม่ทำการติดต่อกับโลกภายนอกโดยตรง เป็นชั้นที่เซลล์ประสาทจะทำการคิดคำนวณกับข้อมูลที่ได้รับเข้ามาจากชั้นข้อมูลเข้า จำนวนของเซลล์ประสาทชั้นนี้จะบ่งบอกถึงความซับซ้อนของการทำงานในโครงข่ายประสาทเทียม
3. ระดับชั้นข้อมูลออก (Output Layer) จะส่งคืนสัญญาณผลลัพธ์ไปยังภายนอกหลังจากที่โครงข่ายได้ดำเนินการกับข้อมูลที่เข้ามาแล้ว

2.3.1 วิธีการสอนโครงข่ายประสาทเทียมตามลักษณะการเรียนรู้

สามารถแบ่งการเรียนรู้ได้ 2 แบบ คือการเรียนรู้แบบมีการชี้นำ (Supervised Learning) และการเรียนรู้แบบไม่มีการชี้นำ (Unsupervised Learning)

2.3.1.1 การเรียนรู้แบบมีการชี้นำ (Supervised Learning)

การเรียนรู้จะมีการกำหนดเซตของการฝึกหัดให้กับเครือข่าย ซึ่งเซตนี้จะประกอบด้วย อินพุต และเอาต์พุตที่ต้องการ โดยทำการจับคู่การสอน (Training Pair) เมื่อทำการป้อนอินพุตให้กับเครือข่ายแล้ว เครือข่ายจะทำการประมวลผลจนได้คำตอบและค่าถ่วงน้ำหนัก (Weight) ออกมาชุดหนึ่งสำหรับคำตอบที่ได้จริง (Actual Output) เพื่อนำมาคำนวณค่าผิดพลาด โดยสามารถหาได้จากการนำคำตอบที่เราต้องการ (Target Output) ลบจากคำตอบที่ได้จริงจากการเรียนรู้ของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครือข่าย ถ้าค่าความผิดพลาดที่ได้จากการลบกันระหว่างค่าที่ได้จริงกับค่าที่เราต้องการยังมีค่าที่เราต้องการยังมีค่าสูง หรือในบางกรณีอาจต้องปรับให้มีค่าเป็นศูนย์ จะต้องมีการปรับค่าถ่วงน้ำหนักไปจนกว่าจะได้ค่าที่ยอมรับได้จึงจะหยุดการสอนในเครือข่าย

2.3.1.2 การเรียนรู้แบบไม่มีการชี้นำดูแล (Unsupervised Learning)

การเรียนรู้การสอนโดยที่ไม่ต้องมีการจับคู่สอน เนื่องจากเมื่อเราใส่ค่าอินพุตเข้าสู่เครือข่ายแล้ว เครือข่ายจะพยายามจัดกลุ่มอินพุตเซตที่มีลักษณะเดียวกันให้เอาท์พุตออกมาจากเครือข่ายที่เดียวกัน คือ มีการปรับตัวเองภายใน (Self-Organization) การเรียนรู้โดยวิธีนี้จะป้อนอินพุตเข้าสู่เครือข่าย ภายในเครือข่ายจะมีเอาท์พุตโหนดอยู่หลายโหนด โดยแต่ละโหนดแทนกลุ่มของข้อมูลที่มีคุณสมบัติเหมือนกัน เมื่อป้อนอินพุตเข้าสู่เครือข่าย เครือข่ายจะทำการคำนวณหาค่าความสัมพันธ์ที่มีภายในเซตของอินพุต โดยอาศัยค่าถ่วงน้ำหนักเป็นตัวแยกความแตกต่างของอินพุต ไปยังโหนดเอาท์พุตของเครือข่ายการเรียนรู้ คือ ใช้ค่าน้ำหนักเป็นตัวแยกความแตกต่างของอินพุต โดยวิธีนี้ไม่สามารถทำการระบุเอาท์พุตที่ถูกต้องได้ว่าโหนดใดเป็นข้อมูลของกลุ่มใด ซึ่งผู้ใช้งานต้องเป็นคนกำหนดเอง

2.4 Particle Swarm Optimization

Particle Swarm Optimization (PSO) เป็นเทคนิควิธีการของอัลกอริทึมเพื่อใช้สำหรับค้นหาค่าที่ดีที่สุดของปัญหาใดๆ โดยใช้วิธีของกลุ่มประชากรในการค้นหาซึ่งถูกคิดค้นในปี 1995 โดย ดอกรเตอร์อีเบอร์ฮาร์ด และดอกรเตอร์เคนนาคีซึ่งเป็นนักฟิสิกส์และวิศวกร โดยพวกเขาได้แรงบันดาลใจมาจาก พฤติกรรมการอยู่ร่วมกันเป็นกลุ่มหรือฝูงของสัตว์ ดังเช่น ฝูงนก และฝูงปลา เป็นต้น

เมื่อเราพิจารณา PSO แล้วจะพบว่าเป็น อัลกอริทึม ที่มีแนววิธีการที่คล้ายกันกับเทคนิคของ จีเนติกอัลกอริทึม ซึ่งเป็นเทคนิคของ Evolutionary Computation อยู่พอสมควร โดยจะใช้วิธีการสุ่มสร้างประชากรจำนวนหนึ่งขึ้นมา แล้วทำการหาค่าที่เหมาะสมที่สุดโดยการพัฒนาประชากรที่มีอยู่ไปเรื่อยๆ แต่ในความเป็นจริงก็คือทั้งวิธีการของทั้ง 2 อัลกอริทึม นั้นจะเหมือนกันก็เพียงตรงขั้นตอนการสร้างประชากรเท่านั้น เนื่องจาก PSO ได้ใช้วิธีการให้ประชากรที่สร้างขึ้นมาที่เรียกว่า “Particle” (พาดิคัล) และใช้ประชากรเดินนั้นทำการบินผ่านช่องว่างของปัญหา และปรับทิศทางของตัวมันให้ไปในทิศทาง ของตำแหน่งของพาดิคัลตัวที่มีค่าความเหมาะสมมากกว่าตัวมันเป็นจำนวนหลายๆรอบจนกระทั่งได้ตำแหน่งที่ดีที่สุดที่เท่าที่ พาดิคัล ชุดนี้จะหาได้ออกมา ซึ่งไม่ได้ทำการสร้างประชากรขึ้นใหม่ด้วยการ Crossover และ Mutation เพื่อสร้างประชากรใหม่ๆเหมือนจีเนติกอัลกอริทึมแต่อย่างใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 การทำงานของ Particle Swarm Optimization

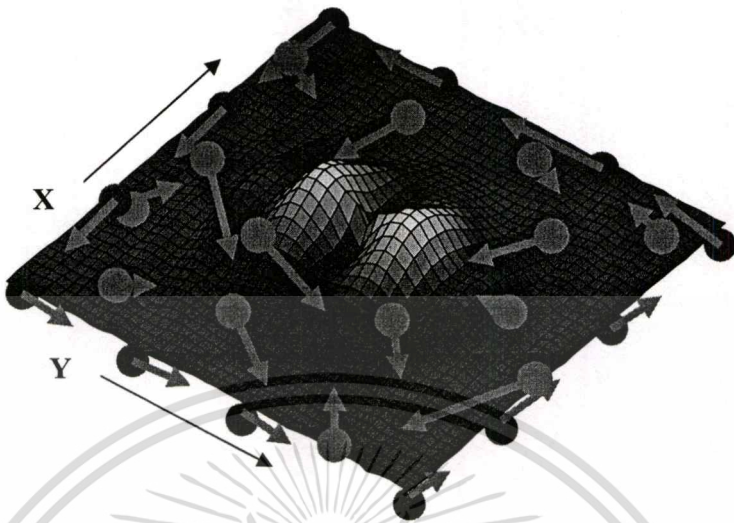
วิธีการทำงานของ PSO นั้นถือได้ว่าเป็นการประยุกต์จากการเลียนแบบธรรมชาติมา โดยให้ลองนึกถึงตัวอย่างของฝูงปลาที่ว่ายอยู่ในน้ำซึ่งกำลังค้นหาอาหาร ปลาแต่ละตัวนั้นไม่ได้รู้มาก่อนว่า ณ.ตำแหน่งไหนในท้องทะเลที่มีอาหารอยู่มาก ดังนั้นปลาแต่ละตัวจึงต่างว่ายไปเพื่อหาตำแหน่งที่มีอาหารอุดมสมบูรณ์ แต่ปลาก็ต้องมีวิธีการที่จะคิดว่าตัวมันนั้นจะว่ายไปซ้าย ขวา หรือ ขึ้นบน ลงล่างดี โดยปกติแล้ววิธีการของปลาก็คือมันก็จะตัดสินใจจาก ปลาตัวนั้นจะจดจำตำแหน่งที่ดีที่สุดที่ตัวปลานั้นได้ว่ายผ่านมา (พบว่ามีความสุขที่สุดเท่าที่ตัวมันว่ายผ่านมา) เอาไว้ และปลาก็จะดูว่าเพื่อนตัวมันนั้นส่วนใหญ่ว่ายไปทางไหนกัน (ว่ายตามเพื่อนที่เจอตำแหน่งที่มีอาหารมาก) มันก็จะว่ายไปโดยจะว่ายตามตำแหน่งของ ทั้ง 2 ตำแหน่งมาคำนวณเพื่อหาทิศที่มันควรจะว่ายไปนั่นเอง

2.4.1.1 อธิบายการทำงาน PSO อัลกอริทึม

อัลกอริทึม Particle Swarm Optimization นั้นถ้าหากเราแปลความหมายของมันตามชื่อก็คือการหาค่าที่เหมาะสมโดยใช้กลุ่มของจุด (พาดิคล) จำนวนกลุ่มหนึ่ง ซึ่งขั้นตอนการทำงานของอัลกอริทึมนี้ก็จะเริ่มจาก การทำความเข้าใจกับ พาดิคล และค่าความเหมาะสม (Fitness Value) กันก่อน โดยเมื่อเราได้ลองพิจารณา ตัวอย่างในการหาค่าที่เหมาะสม (ต้องการหาค่า X และ Y ที่ทำให้ $f(X,Y)$ มีค่า Maximum) ของสมการ 2 ตัวแปร ใดๆ เช่น $40X - X^2 + 20Y - Y^2 = f(X,Y)$ ซึ่งอาจจะเปรียบได้กับรูปที่ 2.8 ที่แสดงอยู่ด้านล่างซึ่งมีแกน X และแกน Y โดยค่าที่เหมาะสมมากที่สุดโดยปกติแล้วเราจะหมายถึงจุดที่ค่ามากที่สุด ซึ่งถ้าในที่นี้ก็เปรียบเสมือนจุดที่มีความสูงมากที่สุด (ค่า X และ Y ที่ทำให้ $f(X,Y)$ มีค่า Maximum) ในรูปที่ 2.8 นั่นเอง ซึ่งพาดิคล ก็คือจุดที่เรา Random ตำแหน่งซึ่งก็คือค่า X และ Y ขึ้นมาให้กับมันซึ่งก็คือจุด 1 จุดในรูปที่ 2.8 ตามปกติแล้วการ Random น่าจะทำให้จุดกระจายอย่างสม่ำเสมอในพื้นที่ปัญหาดังกล่าว ดังนั้นพาดิคล 1 ตัวก็จะมีค่าความเหมาะสมออกมา 1 ค่า โดยในที่นี้เราจะขอยกตัวอย่างลักษณะของค่าว่า ตำแหน่ง และค่า Fitness ดังต่อไปนี้ เช่นถ้าพาดิคล A มีตำแหน่งของ $X = 5$, $Y = 8$ ก็จะมีค่าความเหมาะสมเท่ากับ $40(5) - (5)^2 + 20(8) - (8)^2 = 271$ ถ้า พาดิคล B มีตำแหน่งของ $X = 7$, $Y = 2$ ก็จะมีค่าความเหมาะสมเท่ากับ $40(7) - (7)^2 + 20(2) - (2)^2 = 267$ แสดงว่าเมื่อเราลองพิจารณาจากค่าความเหมาะสมของพาดิคล A และพาดิคล B เมื่ออยู่ในรูปที่ 2.8 ตามที่เราได้สมมุติขึ้นนั้นพาดิคล A จะอยู่ในจุดที่มีระดับที่สูงกว่าพาดิคล B (ในกรณีตามรูปที่ 2.8 ค่าที่เราต้องการก็คือตำแหน่งของจุดที่สูงที่สุด หรือเราสามารถจะเรียกอีกอย่างได้ว่าพาดิคล A มีค่าความเหมาะสมที่ดีกว่าพาดิคล B) โดยเราจะทำการเรียกฟังก์ชัน $f(X,Y)$ หรือ ฟังก์ชันใดๆที่ต้องการหาค่า X และ Y ที่ต้องการหาค่าความเหมาะสม (ค่าที่มากที่สุด) นี้ว่า Fitness Function และเราจะเรียกค่าที่เป็นผลลัพธ์ (Output) จากฟังก์ชันนี้ว่า ค่าความเหมาะสม (Fitness) ซึ่งในที่นี้ค่า Fitness ของพาดิคล A และ พาดิคล B ก็คือ 271 และ 267 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงตัวอย่างภาพการทำงานของ PSO อัลกอริทึม 2 ตัวแปร (X , Y)

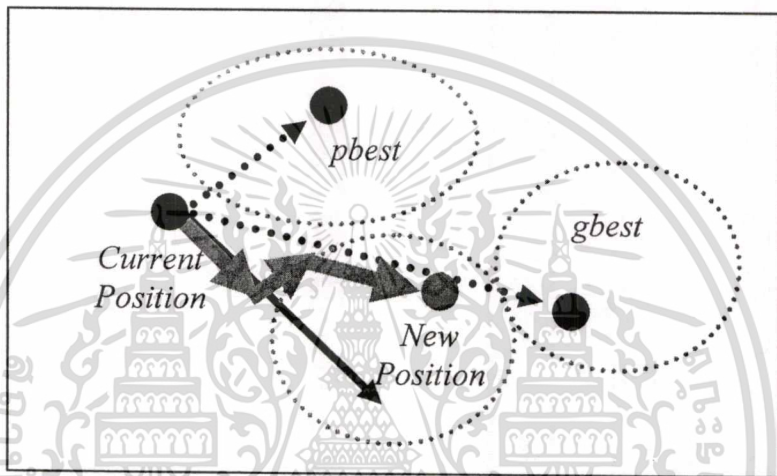
พาคัลแต่ละตัวนั้นยังต้องสามารถที่จะจดจำตำแหน่ง (Position) ที่ดีที่สุดของตัวมันที่เคยวิ่งผ่านมาแล้วทั้งหมดซึ่งเราจะเรียกค่าของตำแหน่งนี้ว่า pbest และต้องทราบตำแหน่งที่ดีที่สุดของทั้งหมดของทั้งกลุ่มของ พาคัล ซึ่งเราเรียกตำแหน่งนี้ว่า gbest เอาไว้เพื่อใช้ในการปรับตำแหน่งถัดไปของแต่ละพาคัลเองด้วยโดยหลังจากที่เราสร้างพาคัลขึ้นมาจำนวนหนึ่งโดย Random ตำแหน่งให้กับพาคัล และคำนวณค่า Fitness ทั้งหมดเรียบร้อยแล้วขั้นตอนต่อไปก็คือการปรับตำแหน่งใหม่ของ พาคัล แต่ละตัว ซึ่งจะมีวิธีการเลือกตำแหน่งใหม่จากค่า pbest และ gbest ที่พาคัลได้จำเอาไว้จากสมการดังต่อไปนี้

$$\text{present}[] = \text{present}[] + v[] \tag{a}$$

$$v[] = v[] + c1 * \text{rand}() * (\text{pbest}[] - \text{present}[]) + c2 * \text{rand}() * (\text{gbest}[] - \text{present}[]) \tag{b}$$

- โดยที่ **present[]** คือ ค่าตำแหน่งของ พาคัล ขณะปัจจุบัน
- v[]** คือ ค่าความเร็วที่จะใช้ในการเปลี่ยนตำแหน่ง
- c1** คือ ค่าคงที่เพื่อใช้ในการปรับค่าพจน์แรก นิยมใช้ค่าประมาณ 2.00 โดยถ้าค่านี้มีค่ามากหมายถึงผู้ใช้งานให้ความสำคัญกับค่า pbest มาก

- c2** คือ ค่าคงที่เพื่อใช้ในการปรับค่าพจน์ที่สอง นิยมใช้ค่าประมาณ 2.00 โดยถ้า ค่านี้นี้มากหมายถึงผู้ใช้งานให้ความสำคัญกับค่า *gbest* มาก
- rand()** คือ ค่าที่ถูก Random ขึ้นมามีค่า 0.00 - 1.00
- pbest[]** คือ ค่าตำแหน่งที่มีค่าความเหมาะสมสูงสุดเท่าที่พาดิคลนั้นได้วิ่งผ่านมา
- gbest[]** คือ ค่าตำแหน่งที่มีค่าความเหมาะสมสูงสุดที่ที่ทุกพาดิคลเคยเจอ



รูปที่ 2.9 แสดงตัวอย่างลักษณะของการปรับค่าความเร็วของพาดิคล

พิจารณาการหาตำแหน่งใหม่ของแต่ละพาดิคลจะประกอบไปด้วย 2 ขั้นตอนหลักคือ ขั้นตอนการหาความเร็วใหม่ และขั้นตอนการปรับตำแหน่งดังรูปที่ 2.9 (เอาความเร็วใหม่ไปบวกกับตำแหน่งเดิม) โดยเริ่มจากสมการ เราจะใช้สมการ (b) ในการคำนวณหาความเร็วที่จะต้องเปลี่ยนให้กับพาดิคลก่อนแล้วจึงนำค่าที่ได้มารวมกับตำแหน่งเดิมของตัวพาดิคลนั้นๆตามสมการ (a) เราจะได้ค่าตำแหน่งใหม่ของพาดิคล และหลังจากนั้นทุกพาดิคลก็จะได้ตำแหน่งใหม่ และก็เริ่มทำการคำนวณหาค่าความเหมาะสมอีกวนไปเรื่อยๆจนกว่าจะได้ค่าความเหมาะสมที่เราพอใจ ซึ่งถ้าเทียบกับรูปที่ 2.8 ก็คือจุดที่ยอดที่มีความสูงมากที่สุดนั่นเอง

นอกจากนี้ในการการปรับความเร็ว และตำแหน่งของแต่ละพาดิคลนั้นเรายังต้องพิจารณา ค่าอีก 1 ค่าด้วยซึ่งก็คือ ค่าความเร็วสูงสุดที่เราจะอนุญาต โดยค่านี้เป็นค่าที่มีความสำคัญอย่างมาก เนื่องจากหากเรากำหนดค่านี้ให้มีค่ามากเกินไปก็จะทำมีโอกาสมากที่จะทำให้พาดิคลวิ่งผ่านจุดที่มีค่าความเหมาะสมมากที่สุดไปได้

2.4.1.2 สรุปการทำงาน PSO อัลกอริทึม

จากที่ได้กล่าวมาในหัวข้อ การทำงานของ Particle Swarm Optimization เราสามารถสรุปขั้นตอนต่างๆได้ดังต่อไปนี้

ขั้นที่ 1 สร้างประชากรของพาดิคัลขึ้นมาพร้อมทั้งสุ่มค่าตำแหน่งให้กับแต่ละพาดิคัล

ขั้นที่ 2 คำนวณหาค่าความเหมาะสมแต่ละพาดิคัลและจัดเก็บค่า pbest ไว้

ขั้นที่ 3 จัดเก็บค่า gbest ไว้

ขั้นที่ 4 คำนวณค่าความเร็วใหม่ ตามค่า pbest และ gbest ของแต่ละ พาดิคัล (ซึ่งก็คือการปรับค่าความเร็วให้ชี้ไปในทิศทางของตำแหน่งของจุด pbest และ gbest ตามรูปที่ 2.9 นั่นเอง)

ขั้นที่ 5 ปรับค่าตำแหน่งของทุกๆพาดิคัลใหม่

ขั้นที่ 6 กลับไปทำขั้นที่ 2 วนไปเรื่อยๆจนกว่าจะได้ค่าที่เราพอใจ

หรือเราสามารถจะเขียนเป็น Pseudocode ได้ดังรูปที่ 2.10 คือ

```

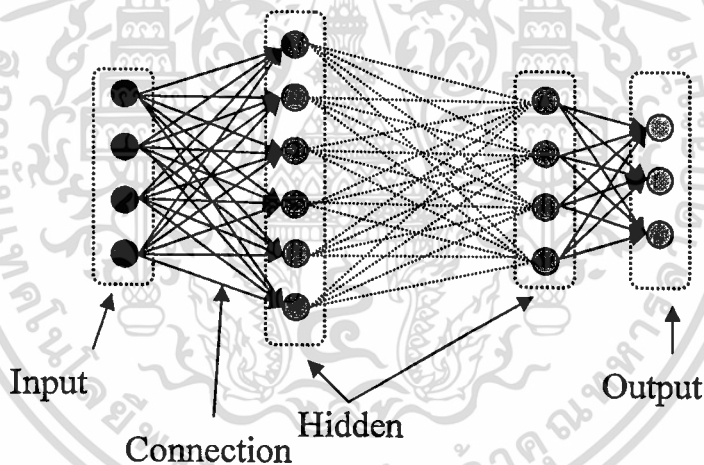
For each particle
  Initialize particle
END
Do
  For each particle
    Calculate fitness value
    If the fitness value is better than its personal best
      set current value as the new pBest
  End
  Choose the particle with the best fitness value of all as gBest
  For each particle
    Calculate particle velocity according equation (b)
    Update particle position according equation (a)
  End
While maximum iterations or minimum error criteria is not attained
    
```

รูปที่ 2.10 แสดง Pseudocode ของอัลกอริทึม PSO

2.5 โครงข่ายประสาทเทียมแบบแบคพรอพพาเกชัน

แบคพรอพพาเกชันอัลกอริทึม ซึ่งมีลักษณะของโครงข่ายประสาทเทียมแบบหนึ่ง ซึ่งสามารถนำมาใช้ในการแก้ปัญหาในลักษณะเชิงเส้น (Linear) และลักษณะไม่เชิงเส้น (Nonlinear) ได้ โดยมีกระบวนการเรียนรู้จากข้อมูลในอดีต โดยที่ Output ที่ได้จากโครงข่ายในแต่ละครั้งของชุดข้อมูลที่ป้อนให้กับโครงข่ายนั้นจะถูกนำมาเปรียบเทียบกับ Output ที่เกิดขึ้นจริงในอดีต เพื่อหาค่าความผิดพลาดในแต่ละครั้ง แล้วนำค่าความผิดพลาดที่ได้นั้นมาใช้ในการคำนวณเพื่อปรับค่าถ่วงน้ำหนักระหว่างแต่ละโหนดในโครงข่ายต่อไป โดยที่การปรับค่าถ่วงน้ำหนักจะมีการคำนวณใน

ทิศทางย้อนกลับจากทางด้าน Output ผ่านชั้นกลาง (Hidden Layer) จนปรับค่ามาถึงชั้นแรกที่อยู่ถัดจากชั้นของ Input จากนั้นจึงคำนวณค่าของ Output ใหม่ที่ได้เพื่อเปรียบเทียบหาค่าความผิดพลาดใหม่โดยในกระบวนการเรียนรู้จะมีการคำนวณเช่นนี้เรื่อยไป จนกระทั่งค่าความผิดพลาดน้อยที่สุดเมื่อเทียบกับค่าที่เกิดขึ้นจริง



รูปที่ 2.11 สถาปัตยกรรมของ Feed forward Neural Network

2.5.1 ลักษณะการเรียนรู้ที่ใช้ในการพยากรณ์

ใช้โครงข่ายประสาทเทียมที่มีลักษณะการเชื่อมต่อภายในเหมือนกับเซลล์ประสาทในสมองคน โดยรับข้อมูลเข้าไปในหลาย ๆ เซลล์ แล้วนำข้อมูลเหล่านั้นมาผ่านกระบวนการการเรียนรู้ โดยปรับเปลี่ยนข้อมูลให้ได้ตามต้องการ เพื่อส่งออกไปยังเซลล์ประสาทอื่น ๆ ในโครงข่าย คือ ใช้กระบวนการเรียนรู้เพื่อทำการเปรียบเทียบค่าเอาท์พุทที่ได้จากโครงข่ายกับค่าเป้าหมาย เพื่อให้ประสิทธิภาพในการทำงานของโครงข่ายดีขึ้น โดยให้ค่าน้ำหนักเริ่มต้นจะอยู่ระหว่าง -0.5 ถึง 0.5

พฤติกรรมการเรียนรู้ของโครงข่ายประสาทเทียมจะขึ้นอยู่กับค่าน้ำหนัก (Weight) และ ฟังก์ชันการโยกย้าย (Transfer Function) ซึ่งเป็นกลไกในการย้ายค่าข้อมูลอินพุตไปยังข้อมูล เอาท์พุต เช่น ฟังก์ชันกระตุ้น (Sigmoid Function) โดยมีขั้นตอนการเปลี่ยนแปลง ดังนี้

- ขั้นตอนการคูณค่าอินพุตกับค่าน้ำหนักทุกโหนดในโครงข่าย
- ขั้นตอนการหาผลรวมจากการบวกผลลัพธ์ในขั้นตอนแรก

ขั้นตอนการนำผลรวมที่ได้มาผ่านฟังก์ชันโยกย้าย จะได้ผลลัพธ์เป็นค่าเอาท์พุตของโครงข่าย

2.5.1.1 การแพร่ย้อนกลับ (Backpropagation)

การเรียนรู้ของโครงข่ายประสาทเทียมแบบการแพร่ย้อนกลับ (Backpropagation) สามารถนำมาใช้ในการแก้ปัญหาในลักษณะเชิงเส้น และไม่เชิงเส้น โดยมีสถาปัตยกรรมพื้นฐาน ดังนี้

1. สามารถแบ่งออกเป็นชั้นอินพุต ชั้นซ่อน และชั้นเอาท์พุต ซึ่งในชั้นซ่อนสามารถมีได้มากกว่า 1 ชั้น ที่ชั้นอินพุตจะไม่มีการประมวลผล จะรับค่าอินพุตแล้วกระจายไปยังโครงข่ายในชั้นต่อไป ที่ชั้นซ่อน และชั้นเอาท์พุตจะมีการประมวลผล

2. มีลักษณะการเชื่อมต่อแบบ Fully Connected Feedforward ในแต่ละชั้นคือ ทุกโหนดที่ชั้นอินพุตจะส่งสัญญาณผ่านไปยังทุก ๆ โหนดในชั้นต่อไป แล้วทุกโหนดในชั้นซ่อนสุดท้ายก็จะส่งสัญญาณไปยังทุกโหนดที่ชั้นเอาท์พุต

ในการฝึกหัดโครงข่ายที่ชั้นอินพุต ชั้นซ่อน และชั้นเอาท์พุต โดยหลักการการแพร่ย้อนกลับสามารถแบ่งออกเป็น 3 ขั้นตอน ดังนี้

1. แพร่สัญญาณจากชั้นอินพุตไปยังชั้นซ่อน ถ้ามีชั้นซ่อนหลายชั้นก็แพร่ไปยังชั้นซ่อนชั้นแรก และแพร่ไปยังชั้นซ่อนถัดไป จนถึงชั้นเอาท์พุต โดยจะต้องผ่านฟังก์ชันกระตุ้น

2. คำนวณหาค่าความผิดพลาดของค่าเอาท์พุตกับค่าเป้าหมาย (Target Value) และแพร่ย้อนกลับค่าความผิดพลาด ไปยังชั้นซ่อน

3. ปรับเปลี่ยนค่าน้ำหนักที่ชั้นเอาท์พุต และชั้นซ่อนเพื่อลดค่าความผิดพลาด เพื่อให้ค่าเอาท์พุตมีค่าใกล้เคียงหรือเท่ากับค่าเป้าหมาย

ในกระบวนการแพร่ย้อนกลับใช้กฎเดลต้า (Delta Rules) เป็นพื้นฐานเพื่อทำการลดค่าความผิดพลาดระหว่างค่าเอาท์พุตที่คำนวณได้กับค่าเป้าหมาย โดยค่าความผิดพลาดที่ลดลงนี้จะได้จากการปรับเปลี่ยนค่าน้ำหนัก

2.5.2 ขั้นตอนการฝึกหัดโครงข่ายโดยใช้วิธีแบคพรอพพาเกชัน

ขั้นตอนในการนำแบคพรอพพาเกชัน มาสอนให้ Neural Network เรียนรู้ จะแบ่งออกเป็น 3 layer ประกอบด้วย ชั้นอินพุต (Input Layer) ชั้นซ่อน (Hidden Layer) และชั้นเอาท์พุต (Output Layer) ซึ่งมีขั้นตอนการทำงาน ดังนี้

- กำหนดอินพุตเวกเตอร์ให้แก่โหนดอินพุตในโครงข่าย โดยให้

$$I_p = (x_{p1}, x_{p2}, \dots, x_{pm}; y)$$

- คำนวณหาค่าของเอาต์พุต ของแต่ละโหนดในโครงข่าย โดยให้

$$out_j = f(sum_j)$$

$$sum_j = \sum_{i=1}^n w_{ji} x_{pi}$$

เมื่อ w_{ji} คือ ค่าน้ำหนักระหว่างโหนด j ของชั้นปัจจุบัน และโหนด i ของชั้นก่อนหน้า

x_{pi} คือ ค่าอินพุต Pattern ชุดที่ p และค่าของข้อมูลค่าที่ i

$$f(sum_j) = (1 + e^{-sum_j})^{-1}$$

- คำนวณหาค่าความผิดพลาดของแต่ละโหนดที่อยู่ในชั้นเอาต์พุต (output layer) โดย

$$\delta_k = (y - out_k)[out_k(1 - out_k)]$$

เมื่อ y คือ ค่าเป้าหมาย

- คำนวณหาค่าความผิดพลาดของแต่ละโหนดในชั้นซ่อน (hidden layer) โดยจะมีค่า

$$\delta_j = [f'(sum_j)] \sum_{k=1}^n \delta_k w_{kj}$$

- คำนวณหาค่าน้ำหนักที่เปลี่ยนของชั้น อินพุตโดย

$$\Delta w_{kj} = \eta \delta_k I_j$$

เมื่อ η คือ อัตราการเรียนรู้ซึ่งมีค่าระหว่าง 0 ถึง 1

เมื่อ I_j คือ เวกเตอร์ที่มาจาก hidden J

6. คำนวณหาค่าน้ำหนักที่เปลี่ยนไปในชั้นซ่อนโดย

$$\Delta w_{ji} = \eta \delta_{ji} I_{ji}$$

เมื่อ I_{ji} คือเวกเตอร์ที่มาจากอินพุต i ไปยัง hidden j

7. ปรับปรุงค่าน้ำหนักใหม่ให้สอดคล้องกับค่าน้ำหนักที่เปลี่ยน โดย

$$w = w + \Delta w$$

เมื่อได้ค่าน้ำหนักที่ปรับปรุงใหม่แล้วให้ทำซ้ำจนกระทั่งได้ค่าความผิดพลาดที่เข้าใกล้ 0 หรือได้ค่าที่สามารถยอมรับได้

2.6. ความผิดพลาดของการพยากรณ์

ความถูกต้องจากการพยากรณ์ นับว่าเป็นสิ่งสำคัญสำหรับผู้พยากรณ์ โดยความถูกต้องขึ้นอยู่กับผลต่างของค่าความผิดพลาดที่เกิดขึ้นในการพยากรณ์ ซึ่งเกิดจากค่าจริงลบกับค่าพยากรณ์ ซึ่งถ้าผลต่างนี้ไม่มีความต่างกันมากนัก ถือเป็นพยากรณ์ที่สามารถควบคุมได้ คือ ตัวแบบนั้นมีความเหมาะสมกับชุดข้อมูลนั้นๆ แต่ถ้าผลต่างนั้นมีค่าความต่างกันมาก แสดงว่าการพยากรณ์นั้นไม่สามารถควบคุมได้ นั่นคือตัวแบบที่ใช้ในการพยากรณ์นั้นไม่เหมาะสม จึงควรทำการปรับเปลี่ยนตัวแบบพยากรณ์นั้นให้เหมาะสมก่อนที่จะนำค่าพยากรณ์นั้นไปใช้งาน

2.6.1 ค่าสถิติที่ใช้วัดความถูกต้องของพยากรณ์

ค่าวัดความถูกต้องของพยากรณ์ที่ใช้กันมากคือ

1. SSE (Sum Square Error) เป็นค่าวัดความถูกต้องของการพยากรณ์ที่วัดจากค่าความคลาดเคลื่อนในแต่ละครั้งยกกำลังสอง

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

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. RMSE (Root Mean Square Error) เป็นค่าเบี่ยงเบนมาตรฐานความคลาดเคลื่อนกำลังสองเฉลี่ย

$$\text{RMSE} = \sqrt{\left(\frac{1}{n} \sum_{i=1}^n e_i^2\right)}$$

เนื่องจากค่าวัดความถูกต้องของการพยากรณ์มีหลายค่า ในทางปฏิบัติอาจจะพิจารณา รวมกันหลายค่าจะพิจารณาจากกราฟของความคลาดเคลื่อนควบคู่กันไปด้วย



บทที่ 3

ระบบการทำนายข้อมูลแบบผสม

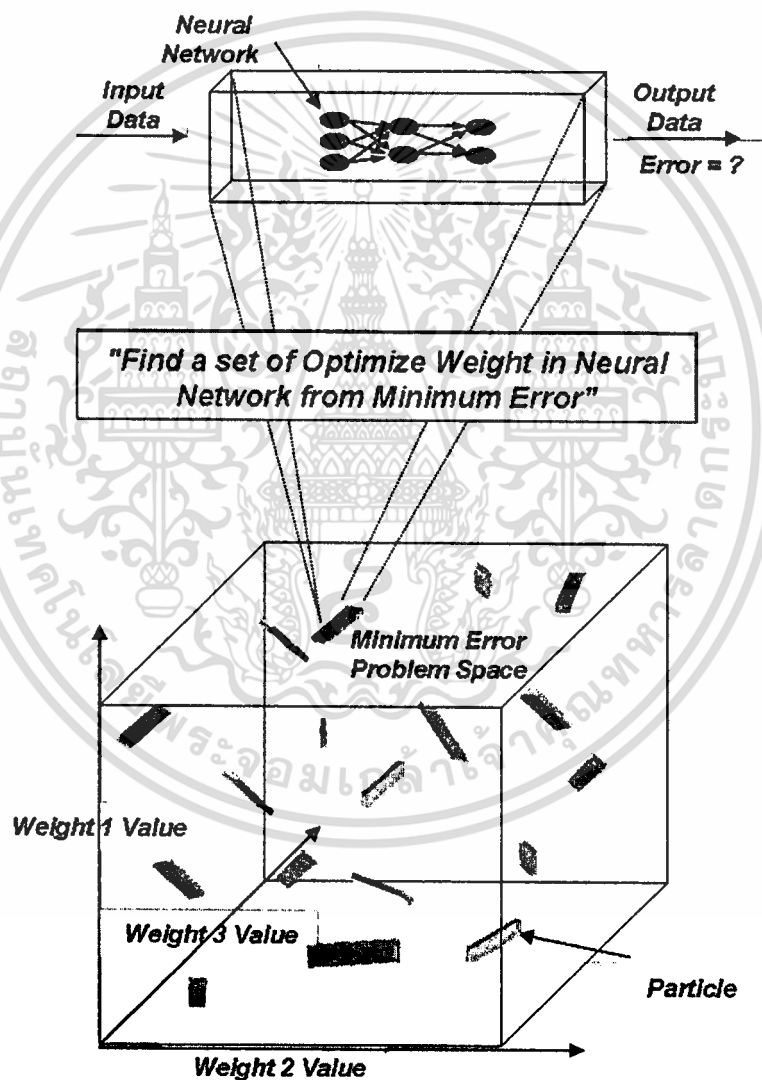
เนื่องจากในปัจจุบันกระบวนการทำงานของ PSO เป็นอัลกอริทึมที่มีประสิทธิภาพและได้รับการยอมรับอย่างแพร่หลายที่ใช้ในการค้นหาค่าที่เหมาะสมสำหรับฟังก์ชันใดๆ ประกอบด้วยคุณสมบัติของโครงข่ายประสาทเทียมซึ่งมีความสามารถในการใช้โครงข่ายประสาทเทียมแทนฟังก์ชันใดๆที่เราต้องการ ซึ่งแม้เราจะมองไม่เห็นลักษณะของฟังก์ชันภายในก็ตามแต่เราก็สามารถใช้งานโครงข่ายประสาทเทียมได้อย่างมีประสิทธิภาพโดยการทดลองใส่ค่านำเข้า และตรวจสอบการทำงานจากค่าที่เราได้รับจากผลของโครงข่ายประสาทเทียม ระบบการทำนายข้อมูลโดยใช้โครงข่ายประสาทเทียมร่วมกับ PSO จึงได้เกิดขึ้น

3.1 สถาปัตยกรรมของระบบ

สถาปัตยกรรมของโมเดลแบบผสมสำหรับพยากรณ์ข้อมูลในอนาคตนี้จะประกอบด้วยส่วนที่สำคัญ 2 ส่วนได้แก่ ส่วนที่เรานำมาใช้แทนฟังก์ชันใดๆที่เราต้องการจะพยากรณ์ซึ่งก็คือโครงข่ายประสาทเทียม โดยเราจะนำมาใช้แทนฟังก์ชันต่างๆของข้อมูลที่เราต้องการจะนำมาพยากรณ์ และอีกส่วนก็คือส่วนของอัลกอริทึม PSO ที่ใช้สำหรับหาค่าที่เหมาะสมที่สุด ซึ่งจะใช้สำหรับปรับ และค้นหาถ่วงน้ำหนักที่ดีที่สุดภายในสถาปัตยกรรมโครงข่ายประสาทเทียมที่ได้จำลองขึ้นมาตามความเหมาะสมของข้อมูลของผู้ใช้งาน สำหรับใช้งานแทนฟังก์ชันที่เราต้องการให้โครงข่ายประสาทเทียมทำการเลียนแบบ ดังรูปที่ 3.1

ในการทำงานของระบบการทำนายโดยใช้โครงข่ายประสาทเทียมร่วมกับ Particle Swarm Optimization นั้นเริ่มจากเราจะต้องใช้พาดิคัลหนึ่งพาดิคัลทำงาน โดยใช้โครงข่ายประสาทเทียมหนึ่งชุดทำงานเป็นส่วนของการค้นหาค่าความเหมาะสมที่ดีที่สุดอยู่ภายในพาดิคัลแต่ละตัว โดยเราจะต้องสร้างพาดิคัลแต่ละตัวให้มีจำนวน Unknown Dimensions เท่ากับจำนวนค่าถ่วงน้ำหนักทั้งหมดที่มีอยู่ในโครงข่ายประสาทเทียม เช่นถ้าโครงข่ายประสาทเทียมที่มีชั้นซ่อนเพียงแค่หนึ่งชั้น และมีจำนวนอินพุทโหนดเท่ากับ 2 โหนด , จำนวนโหนดของชั้นซ่อนเท่ากับ 3 โหนด, จำนวนเอาต์พุทโหนดเท่ากับ 1 โหนด เราจะมี Unknown Dimensions = $(2 \times 3) + (3 \times 1) = 9$ ตัวแปร ซึ่งจุดประสงค์ของหลักของระบบก็คือทำการค้นหาค่าถ่วงน้ำหนักทั้งหมดนี้โดยใช้อัลกอริทึม PSO โดยเราจะต้องใส่ค่าข้อมูลทั้งค่าข้อมูลเข้าและข้อมูลผลลัพธ์ให้กับโครงข่ายประสาทเทียมนี้เพื่อที่จะให้โครงข่ายประสาทเทียมสามารถคำนวณ

ค่าความคลาดเคลื่อน (ในที่นี่ใช้การคำนวณค่าความคลาดเคลื่อนแบบ SSE) ที่เกิดขึ้นได้เพื่อคำนวณค่าความเหมาะสมต่อไป และเราจะพิจารณาว่าค่าถ่วงน้ำหนักแต่ละชุดนั้นเป็นค่าที่ดีพอแล้วหรือยังจากค่าความเหมาะสม ถ้ายังไม่ดีพอก็จะเป็นหน้าที่ของ PSO อัลกอริทึมต่อไปที่จะต้องทำการย้ายตำแหน่งของพาทิคล์ของมันตามค่า pbest และ gbest ของชุดของพาทิคล์ชุดนั้นเพื่อทำการค้นหาค่าถ่วงน้ำหนักสำหรับการเลียนแบบฟังก์ชันที่เราได้ฝึกสอนให้กับโครงข่ายประสาทเทียม



รูปที่ 3.1 แสดงสถาปัตยกรรมของระบบที่ใช้ โครงข่ายประสาทเทียมร่วมกับ PSO เพื่อหาค่าที่เหมาะสมให้กับค่า ค่าถ่วงน้ำหนักของโครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ขั้นตอนการทำงานของระบบ

ระบบการทำนายโดยใช้โครงข่ายประสาทเทียมร่วมกับ PSO นั้นมีจุดประสงค์เพื่อใช้ในการทำนายข้อมูลที่จะเกิดขึ้นในอนาคตโดยใช้ข้อมูลที่เกิดขึ้นแล้วในอดีต และปัจจุบันเป็นค่าอินพุทโดยที่เราจะต้องบอกให้กับระบบ จากที่กล่าวมาแล้วว่าลักษณะการทำงานของระบบจะใช้ลักษณะการทำงานคือให้อัลกอริทึม PSO ค้นหาค่าถ่วงน้ำหนัก ที่มีค่าที่ทำให้การทำนายเกิดค่าความคลาดเคลื่อนให้น้อยที่สุด โดยมีขั้นตอนการทำงานที่สำคัญหลักๆประมาณ 4 ขั้นตอนหลักด้วยกัน ได้แก่ การเลือกข้อมูลที่จะนำมาใช้สำหรับระบบ, การปรับตั้งค่าตัวแปรที่สำคัญให้กับระบบ, การฝึกหัดโครงข่ายของระบบ และการทดสอบโครงข่ายของระบบ

3.2.1 การเลือกข้อมูลที่จะนำมาใช้สำหรับระบบ

ข้อมูลที่จะนำมาสอนระบบนั้น ไม่ได้หมายความว่าเราจะเอาข้อมูลอะไรก็ได้หรือข้อมูลชนิดไหนก็ได้มาป้อนเพื่อสอนให้กับระบบ ซึ่งถ้าข้อมูลที่นำเข้านั้นไม่มีคุณภาพแล้วก็ไม่ต้องสงสัยเลยว่าระบบจะสามารถทำนายผลลัพธ์ที่มีคุณภาพมาให้กับเราหรือไม่ การเลือกข้อมูลมีเรื่องใหญ่ๆที่เราจะต้องพิจารณาอยู่หลายประเด็นหลักด้วยกัน ได้แก่

- ชนิดข้อมูลที่จะนำมาใช้ ข้อมูลบางชนิดมีนัยสำคัญบางชนิดไม่มีนัยสำคัญเช่น เบอร์โทรศัพท์, บ้านเลขที่ และหมายเลขประจำตัวข้อมูลพวกนี้เป็นข้อมูลที่ไม่น่าจะมีนัยสำคัญเพื่อที่จะนำมาใช้กับระบบการทำนายได้
- ปริมาณข้อมูลที่มีจำนวนน้อยเกินไปก็มีโอกาสที่จะทำให้เกิดความผิดพลาดได้สูงเนื่องจากโครงข่ายประสาทเทียมมีการเรียนรู้บ่อยครั้ง
- ต้องมีการปรับค่าโดยการนอร์มอลไลเซชันข้อมูลให้มีค่าอยู่ระหว่าง 0 - 1 ก่อนเสมอ

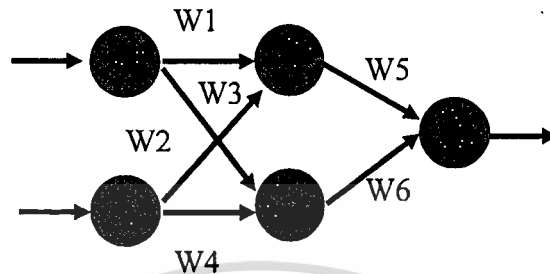
3.2.2 การกำหนดค่าเบื้องต้นให้กับตัวแปร

- ขั้นตอนการกำหนดค่าเบื้องต้นให้กับตัวแปร มีดังนี้
1. กำหนดค่าจำนวนพาดิคัลที่จะใช้ สมมุติใช้ 4 พาดิคัล
 2. กำหนดค่าจำนวนรอบที่จะทำ สมมุติใช้ 10 รอบ
 3. กำหนดความเร็วสูงสุดให้กับพาดิคัล สมมุติใช้ 0.04
 4. สมมุติข้อมูลที่ใช้ในการฝึกสอนโครงข่ายมีค่า

Pattern	Input 1	Input 2	Output
1	0.1	0.2	0.2
2	0.3	0.4	0.8

ตารางที่ 3.1 แสดงค่าสมมุติของข้อมูลนำเข้า

5. กำหนดลักษณะของโครงข่ายประสาทเทียม สมมุติให้เป็นลักษณะดังรูปซึ่งจะประกอบไปด้วย ชั้นอินพุต 2 โหนด, ชั้นซ่อน 2 โหนด และชั้นเอาต์พุต 1 โหนด ดังรูปที่ 3.2



รูปที่ 3.2 แสดงตัวอย่างโครงข่ายที่สมมุติขึ้น

3.2.3 ขั้นตอนการทำงานฝึกสอนโครงข่ายประสาทเทียม

- สร้างพาคัลขึ้นมา 4 พาคัลตามที่ได้สมมุติเอาไว้โดยแต่ละพาคัลจะมีข้อมูลบรรจุไว้ดังนี้

พาคัล เบอร์ 1	ตำแหน่งปัจจุบัน = (0.1,0.2,-0.1,-0.2,0.7,0.5) ตำแหน่งที่ดีที่สุด = (0.1,0.2,-0.1,-0.2,0.7,0.5) ความเร็ว = (0.0,0.0,0.0,0.0,0.0,0.0) ค่า Fitness ที่ดีที่สุด = 0
พาคัล เบอร์ 2	ตำแหน่งปัจจุบัน = (-0.8,-0.7,0.3,0.5,-0.5,0.9) ตำแหน่งที่ดีที่สุด = (-0.8,-0.7,0.3,0.5,-0.5,0.9) ความเร็ว = (0.0,0.0,0.0,0.0,0.0,0.0) ค่า Fitness ที่ดีที่สุด = 0
พาคัล เบอร์ 3	ตำแหน่งปัจจุบัน = (0.2,0.4,0.5,-0.4,-0.8,-0.3) ตำแหน่งที่ดีที่สุด = (0.2,0.4,0.5,-0.4,-0.8,-0.3) ความเร็ว = (0.0,0.0,0.0,0.0,0.0,0.0) ค่า Fitness ที่ดีที่สุด = 0
พาคัล เบอร์ 4	ตำแหน่งปัจจุบัน = (-0.3,0.6,0.8,0.2,0.1,-0.1) ตำแหน่งที่ดีที่สุด = (-0.3,0.6,0.8,0.2,0.1,-0.1) ความเร็ว = (0.0,0.0,0.0,0.0,0.0,0.0) ค่า Fitness ที่ดีที่สุด = 0

รูปที่ 3.3 ลักษณะพื้นฐานของพาคัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าตำแหน่งนั้นได้มาจากการสุ่มค่าขึ้นมาซึ่งจากการสมมุติค่าถ่วงน้ำหนักของโครงข่ายประสาทเทียม 6 ค่า ($w_1, w_2, w_3, w_4, w_5, w_6$) ด้วยกันตามรูป โดยค่าที่เราสุ่มจะอยู่ระหว่างค่า $[-1,1]$ และกำหนดค่าความเร็วเริ่มต้นให้มีค่าเท่ากับ 0 ทั้งหมดจากพาดิคัลทั้ง 4 พาดิคัลเราสามารถสรุปข้อมูลของพาดิคัลได้ดังนี้

พาดิคัล	w_1	w_2	w_3	w_4	w_5	w_6
1	0.1	0.2	-0.1	-0.2	0.7	0.5
2	-0.8	-0.7	0.3	0.5	-0.5	0.9
3	0.2	0.4	0.5	-0.4	-0.8	-0.3
4	-0.3	0.6	0.8	0.2	0.1	-0.1

ตารางที่ 3.2 สรุปค่าถ่วงน้ำหนักทั้งหมดของพาดิคัลที่เริ่มสมมุติขึ้น

2. คำนวณหาค่าความเหมาะสมโดยจะแบ่งเป็นขั้นตอนใหญ่ 2 ขั้นตอนด้วยกันได้แก่ ขั้นตอนคำนวณค่าผ่านโครงข่ายประสาทเทียม และขั้นตอนของการคำนวณค่า Fitness ของแต่ละพาดิคัลออกมา โดยเราจะเริ่มจากนำค่าตำแหน่งของแต่ละพาดิคัลไปแทนค่าในค่าถ่วงน้ำหนักตามตำแหน่ง ในรูปที่ 3.2 และคำนวณค่าเอาท์พุท โดยใช้สูตร

$$out_j = f(sum_j)$$

$$sum_j = \sum_{l=1}^n w_{jl} x_{pl}$$

ณ. โหนดที่มีการรับข้อมูลเข้ามาโดยจะกระทำกับข้อมูลทุกค่าที่เข้ามายังเซลล์ประสาทเทียม และหลังจากนั้นเอาค่า sum ที่ได้มาแทนค่าในสูตร

$$f(sum_j) = (1 + e^{-sum_j})^{-1}$$

โดยจะทำได้จากชั้นอินพุทเข้าไปเรื่อยๆ จนทำครั้งสุดท้ายที่โหนดสุดท้ายซึ่งมีอยู่โหนดเดียวของชั้นเอาท์พุท ซึ่งจะทำให้เราได้รับค่าเอาท์พุทจากตำแหน่ง ณ. ปัจจุบันของแต่ละพาดิคัลดังตารางที่ 3.3 ซึ่งค่าที่ได้นั้นจะไม่ตรง และเมื่อนำมาเทียบกับผลลัพธ์ที่เกิดขึ้นจริงแล้วก่อนข้างจะมีค่าความคลาดเคลื่อน (Error) ก่อนข้างสูง

พาดิกล	Input Pattern 1 ที่คำนวณได้	Input Pattern 2 ที่คำนวณได้
1	0.644683	0.644683
2	0.552122	0.552983
3	0.361013	0.354639
4	0.500187	0.499815

ตารางที่ 3.3 สรุปค่าผลลัพธ์ที่ออกมาจากโครงข่ายประสาทเทียมเมื่อแทนค่าตำแหน่งปัจจุบันของทั้ง 4 พาดิกลเข้าไป

หลังจากนั้นให้นำค่าที่ได้มาหาค่า Error ที่เกิดขึ้นเมื่อเทียบกับค่าจากข้อมูลจริง โดยจะใช้วิธีการหาค่า Error แบบ SSE (Sum Square Error) ซึ่งมีสูตรคำนวณดังต่อไปนี้

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

และทำการคำนวณซึ่งจะได้ผลดังต่อไปนี้

พาดิกล	Pattern 1 (e)	Pattern 2 (e)	Pattern 1 (e ²)	Pattern 2 (e ²)	SSE
1	0.444683	-0.15532	0.197743	0.024123	0.221866
2	0.352122	-0.24702	0.12399	0.061017	0.185007
3	0.161013	-0.44536	0.025925	0.198346	0.224272
4	0.300187	-0.30019	0.090112	0.090111	0.180223

ตารางที่ 3.4 สรุปค่า Error ของแต่ละพาดิกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่า SSE มาใช้คำนวณหาค่า Fitness ต่อจากสูตร

$$\text{Fitness} = 1 / \text{SSE}$$

เนื่องจากค่า SSE ยิ่งมากถือว่าตำแหน่งของพาคัลนั้นไม่ดีดังนั้นเราจะได้ค่า Fitness ของแต่ละพาคัลดังต่อไปนี้

พาคัล	Fitness
1	4.507225
2	5.405201
3	4.458871
4	5.548681

ตารางที่ 3.5 สรุปค่า Fitness ของแต่ละพาคัล

3. ขั้นตอนการปรับตำแหน่งของพาคัล ซึ่งขั้นตอนนี้เป็นหัวใจของอัลกอริทึม PSO ซึ่งจะทำโดย
 - 3.1 หาค่า pbest ซึ่งก็คือค่าตำแหน่งของพาคัลที่มีค่า Fitness มากที่สุดเท่าที่พาคัลนั้นๆ ได้วิ่งผ่านมาของแต่ละพาคัลเอง ซึ่งในการทำรอบแรกค่า pbest ก็คือค่าตำแหน่งแรกที่ถูกสุ่มค่ามานั่นเอง
 - 3.2 หาค่า gbest ซึ่งก็คือค่าตำแหน่งของพาคัลที่มีค่า Fitness มากที่สุดของทุกพาคัลภายในกลุ่มเท่าที่ทุกพาคัลเคยวิ่งผ่านมา ซึ่งในตอนนี้ค่า gbest = ค่าตำแหน่งของพาคัลที่ 4 = (-0.3, 0.6, 0.8, 0.2, 0.1, -0.1)
 - 3.3 ทำการคำนวณหาความเร็วใหม่ของทุกพาคัลซึ่งในปัจจุบันมีค่าเป็น 0 อยู่ และโดยตามปกติแล้วเราควรที่จะกำหนดค่า ความเร็วสูงสุดไว้ด้วยเนื่องจากถ้าไม่กำหนดจะทำให้ พาคัลวิ่งเร็วเกินไปจนทำให้การทำงานของอัลกอริทึมไม่มีประสิทธิภาพ โดยการคำนวณความเร็วใหม่ของแต่ละพาคัลเราจะคำนวณได้ดังต่อไปนี้

จากสมการ

$$v[] = v[] + c1 * \text{rand}() * (\text{pbest}[] - \text{present}[]) + c2 * \text{rand}() * (\text{gbest}[] - \text{present}[])$$

ในตัวอย่างนี้จะสมมติให้ตัวแปรต่างๆที่ใช้ในสมการมีค่าดังนี้

$$\text{ความเร็วสูงสุด} = 0.4, c1 = 2, c2 = 2, \text{rand}() = 0.5$$

แทนค่าสูตร สำหรับ พาดิคัล 1

$$v[] = 0 + 2 * 0.5 * ((0.1, 0.2, -0.1, -0.2, 0.7, 0.5) - (0.1, 0.2, -0.1, -0.2, 0.7, 0.5))$$

$$+ 2 * 0.5 * ((-0.3, 0.6, 0.8, 0.2, 0.1, -0.1) - (0.1, 0.2, -0.1, -0.2, 0.7, 0.5))$$

$$v[] = (-0.4, 0.4, 0.9, 0.4, -0.6, -0.6)$$

ทำงานครบทุกพาดิคัลจะได้ค่าตามตาราง

พาดิคัล	v1	v2	v3	v4	v5	v6
1	-0.4	0.4	0.9	0.4	-0.6	-0.6
2	0.5	1.3	0.5	-0.3	0.6	-1
3	-0.5	0.2	0.3	0.6	0.9	0.2
4	0	0	0	0	0	0

ตารางที่ 3.6 สรุปค่าความเร็วเมื่อแทนค่าสูตร

แต่เนื่องจากมีค่าความเร็วสูงสุดกำหนดค่าขอบเขตเอาไว้ว่าไม่ให้ความเร็วสูงสุดมีค่าเกิน 0.4 ดังนั้นค่าความเร็วที่นำไปใช้ได้จะเป็นไปตามตารางที่ 3.7

พาดิคัล	v1	v2	v3	v4	v5	v6
1	-0.4	0.4	0.4	0.4	-0.4	-0.4
2	0.4	0.4	0.4	-0.3	0.4	-0.4
3	-0.4	0.2	0.3	0.4	0.4	0.2
4	0	0	0	0	0	0

ตารางที่ 3.7 สรุปค่าความเร็วเมื่อจำกัดค่าความเร็วสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ทำการหาค่าตำแหน่งใหม่ของแต่ละพาคีลจากสูตร

$$\text{present}[] = \text{persent}[] + v[]$$

แทนค่า จะได้ค่าตามตารางที่ 3.8

พาคีล	W1	W2	W3	W4	W5	W6
1	-0.3	0.6	0.3	0.2	0.3	0.1
2	-0.4	-0.3	0.7	0.2	-0.1	0.5
3	-0.2	0.6	0.8	0	-0.4	-0.1
4	-0.3	0.6	0.8	-0.2	0.1	-0.1

ตารางที่ 3.8 สรุปค่าตำแหน่งใหม่ของแต่ละพาคีล

วนกลับไปทำขั้นตอนที่ 2 อีก และวนทำไปจนกว่าจะได้ค่า Fitness ที่เราพอใจ หรือ วนทำไปจนกว่าจะครบจำนวนรอบที่เรากำหนดไว้ โดยในที่นี้ถ้าเราจะลองหาค่า Fitness คู่อีกครั้งจะได้ค่าดังต่อไปนี้

พาคีล	Input Pattern 1 ที่คำนวณได้	Input Pattern 2 ที่คำนวณได้
1	0.551009	0.55199
2	0.549525	0.548536
3	0.434018	0.43037
4	0.500187	0.499815
ข้อมูลจริง	0.2	0.8

ตารางที่ 3.9 สรุปค่าผลลัพธ์ที่ออกมาจากโครงข่ายประสาทเทียมเมื่อแทนค่าตำแหน่งปัจจุบันของทั้ง 4 พาคีลเข้าไปรอบที่ 2

พาดิกล	Pattern 1 (e)	Pattern 2 (e)	Pattern 1 (e ²)	Pattern 2 (e ²)	SSE
1	0.351009	-0.24801	0.123207	0.061509	0.184716
2	0.349525	-0.25146	0.122168	0.063234	0.185402
3	0.234018	-0.36963	0.054764	0.136626	0.191391
4	0.300187	-0.30019	0.090112	0.090111	0.180223

ตารางที่ 3.10 สรุปค่า Error ของแต่ละพาดิกลรอบที่ 2

พาดิกล	Fitness
1	5.413716
2	5.393685
3	5.224906
4	5.548681

ตารางที่ 3.11 สรุปค่า Fitness ของแต่ละพาดิกลรอบที่ 2

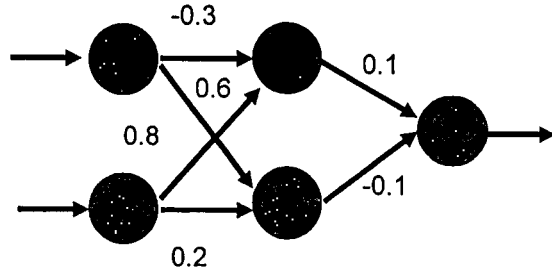
ดังนั้นเราจะพอสังเกตได้ว่าพาดิกลจะทำการปรับความเร็ว และตำแหน่งของตัวมันเอง ภายใต้งี้อเนกไขของฟังก์ชันที่เราต้องการจะสอนให้โครงข่ายประสาทเทียมเลียนแบบให้มีค่า Fitness สูงขึ้นเรื่อยๆ โดยการปรับย้ายตำแหน่งของแต่ละพาดิกลให้เข้าใกล้จุดที่มีค่า Fitness สูงสุด

3.2.4 ขั้นตอนการทำงานทดสอบโครงข่ายประสาทเทียม

จากการทำงานในขั้นตอนการฝึกสอนโครงข่ายจะทำให้เราสามารถหาค่าถ่วงน้ำหนักที่ดีที่สุดได้ออกมา 1 ชุด หลังจากนั้นเราต้องใช้ค่าถ่วงน้ำหนักชุดนี้มาใช้ในการทดสอบโครงข่ายประสาทเทียมของเรา หากเราพิจารณาจากตัวอย่างที่แล้วจะได้ว่า

1. นำค่าถ่วงน้ำหนักที่ดีที่สุดหามาได้มาแทนลงในค่าถ่วงน้ำหนักของโครงข่ายประสาทเทียมที่เราได้กำหนดไว้ในตอนแรกในที่นี่จะแสดงใช้ข้อมูลของพาดิกลหมายเลข 4 โดยจะได้ผลลัพธ์ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



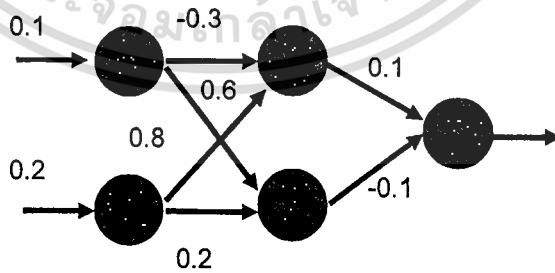
รูปที่ 3.4 แสดงตัวอย่างโครงข่ายสำหรับการทดสอบ

- นำค่าข้อมูลทำการป้อนค่าเข้าไปในโครงข่าย แล้วทำการหาค่าที่เอาท์พุท โหนดตามวิธีการของโครงข่ายประสาทเทียมดังนี้ จากข้อมูล

Pattern	Input1	Input2	Output
1	0.1	0.2	0.2
2	0.3	0.4	0.8

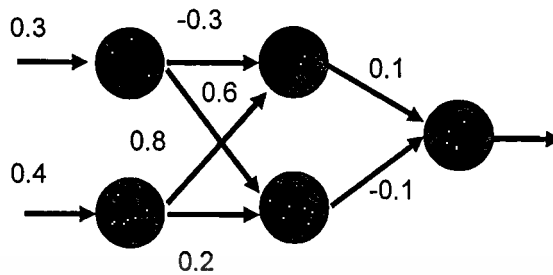
ตารางที่ 3.12 แสดงค่าสมมุติของข้อมูลนำเข้า

จะได้



รูปที่ 3.5 แสดงตัวอย่างโครงข่ายสำหรับการทดสอบเมื่อใส่ค่าข้อมูล Pattern ที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แสดงตัวอย่าง โครงข่ายสำหรับการทดสอบเมื่อใส่ค่าข้อมูล Pattern ที่ 2

คำนวณ โดยใช้สูตร

$$out_j = f(sum_j)$$

$$sum_j = \sum_{i=1}^n w_{ji} x_{pi}$$

ณ. โหนดที่มีการรับข้อมูลเข้ามาโดยจะกระทำกับข้อมูลทุกค่าที่เข้ามายังเซลล์ประสาทเทียม และหลังจากนั้นเอาค่า sum_j ที่ได้มาแทนค่าในสูตร

$$f(sum_j) = (1 + e^{-sum_j})^{-1}$$

โดยจะทำได้จากชั้น อินพุตเข้าไปเรื่อยๆ จนทำครั้งสุดท้ายที่โหนดสุดท้ายซึ่งมีอยู่ โหนดเดียวของชั้นเอาต์พุต ซึ่งจะทำให้เราได้รับค่าเอาต์พุต และนำมาเทียบกับผลลัพธ์ที่เกิดขึ้นจริง

พาทิกัล	Input Pattern 1 ที่คำนวณได้	Input Pattern 2 ที่คำนวณได้
4	0.500187	0.499815
ข้อมูลจริง	0.2	0.8

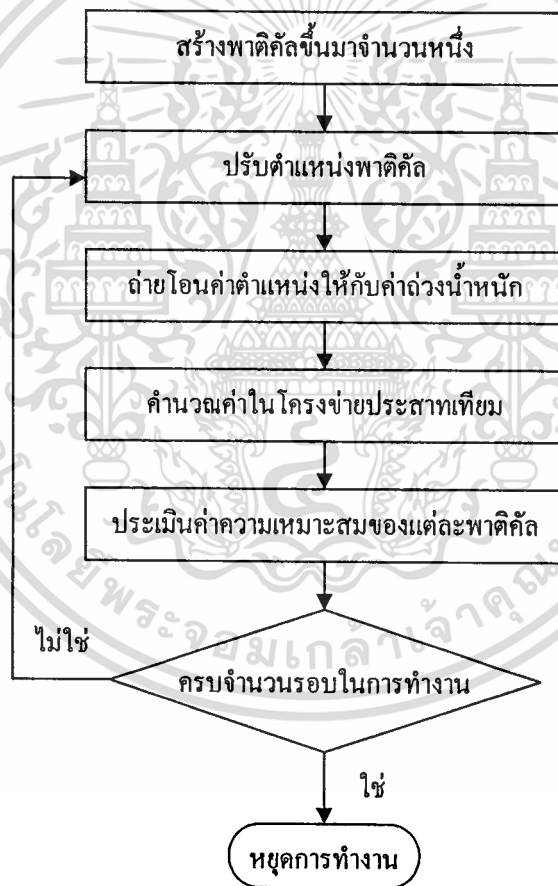
ตารางที่ 3.13 ผลลัพธ์ที่ออกมาจาก โครงข่ายประสาทเทียมเมื่อแทนค่าถ่วงน้ำหนักใน
ขั้นตอนทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่าค่าที่ได้จะค่อนข้างห่างจากค่าที่เราต้องการอยู่มากทั้งนี้เนื่องจากขั้นตอนการกำหนดค่าเบื้องต้นให้กับตัวแปรนั้นเราทำการกำหนดค่าง่ายๆ และทำแสดงการฝึกสอนให้ดูแค่ 1 รอบเพื่อจุดประสงค์ในการแสดงเป็นตัวอย่างให้ดูเท่านั้นแต่ถ้าหากเราต้องการทำเพื่อผลลัพธ์ที่มีประสิทธิภาพนั้นเราต้องปรับเปลี่ยนค่าตัวแปรให้มีความซับซ้อนมากกว่านี้

3.3 แผนภาพสรุปขั้นตอนการทำงานของระบบ

จากการทำงานของระบบการทำนายโดยใช้โครงข่ายประสาทเทียมร่วมกับ PSO เราสามารถสรุปขั้นตอนการทำงานได้ดังนี้



รูปที่ 3.7 แสดงแผนภาพสรุปขั้นตอนการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลอง และผลการทดลอง

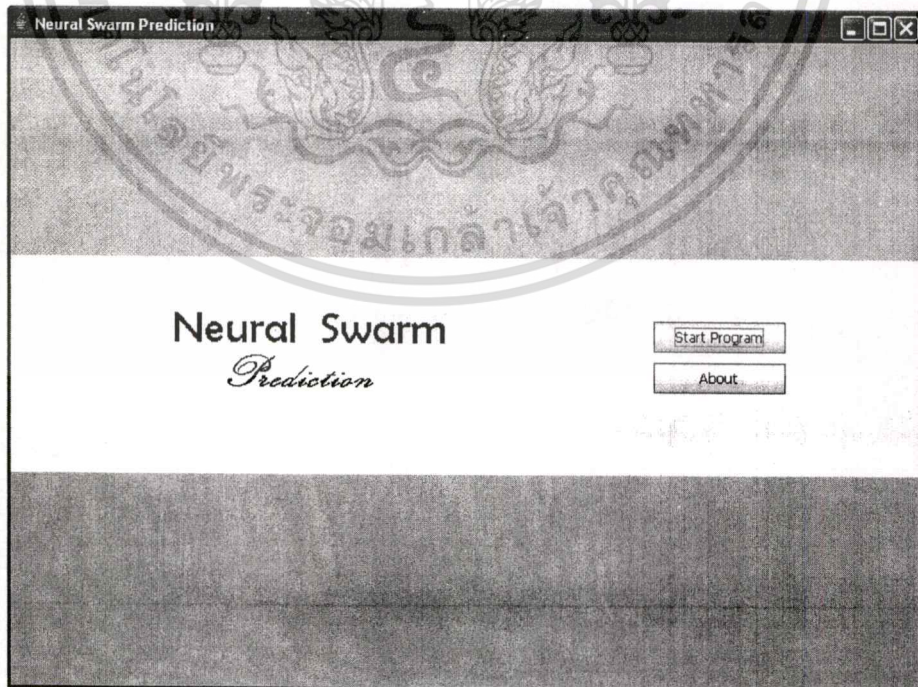
จากการทดลองใช้งานสามารถสรุปผลการทำงานจากการพัฒนาโปรแกรมการพยากรณ์ โดยประยุกต์ทฤษฎีโครงข่ายประสาทเทียมให้ทำงานร่วมกับ PSO อัลกอริทึม ซึ่งสามารถแบ่งออกได้เป็น 2 ส่วนหลักดังต่อไปนี้

1. การทำงานของโปรแกรม
2. การทดลองใช้งานและปรับค่าโปรแกรม

4.1 การทำงานของโปรแกรม

ลักษณะการทำงานของโปรแกรมนี้จะเป็นไปในลักษณะลำดับตามเมนูเพื่อให้ผู้ใช้งานสามารถเข้าใจ และสามารถใช้งานได้อย่างถูกต้อง และง่ายตามขั้นตอนโดยจะมีขั้นตอนการทำงานดังต่อไปนี้

4.1.1 หน้าที่ 1 : Neural Swarm Prediction



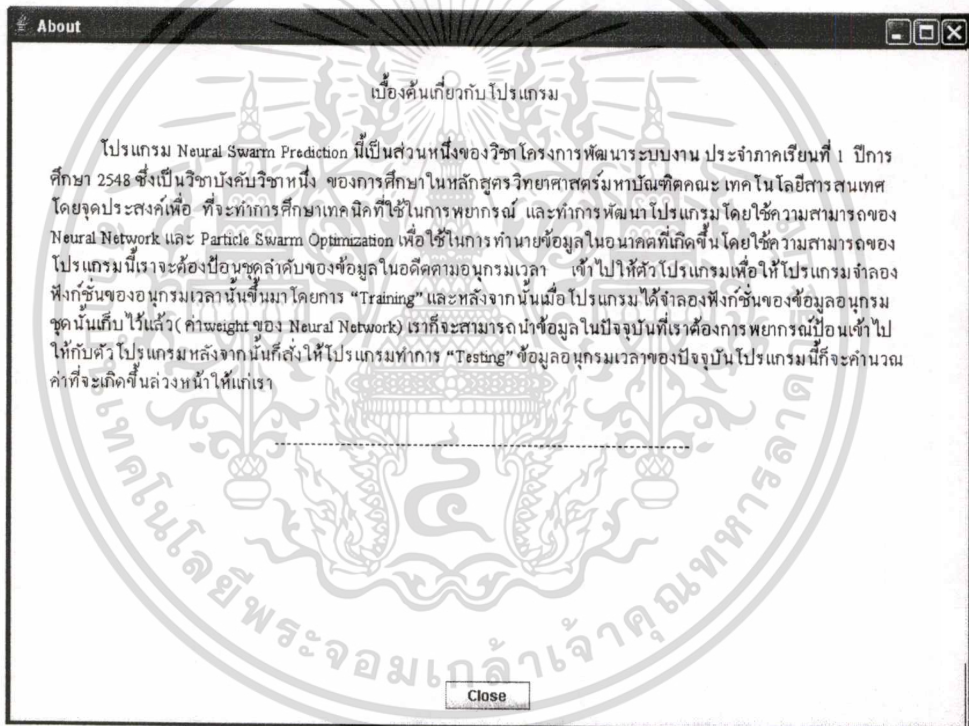
รูปที่ 4.1 แสดงหน้าที่ 1 : Neural Swarm Prediction

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนักเรียนได้เห็นว่าไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเริ่มต้นการทำงานในส่วนนี้จะปรากฏหน้าจอดังรูปที่ 4.1 จะทำหน้าที่บอกให้ผู้ใช้งานทราบว่าขณะนี้โปรแกรมนี้ได้เปิดพร้อมที่จะใช้งานเรียบร้อยแล้ว นอกจากนี้ยังมี About เอาไว้ให้ผู้ที่ยังไม่มีความเข้าใจในตัวโปรแกรมได้อ่านเพื่อทำความเข้าใจก่อนใช้งาน โดยมีสิ่งที่ผู้ใช้งานต้องพิจารณาคือ

- ปุ่ม Start Program : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการเริ่มใช้งาน Program
- ปุ่ม About : ใช้สำหรับอธิบายภาพรวม และลักษณะที่สำคัญของ Program

4.1.2 หน้าที 1.1 : หน้าจอ About



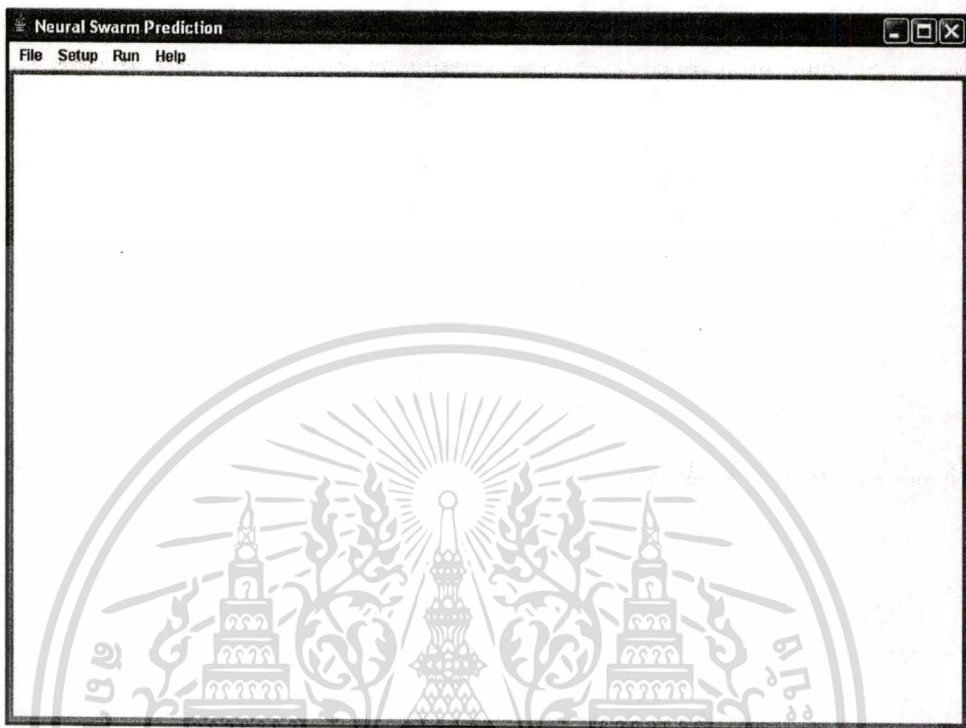
รูปที่ 4.2 แสดงหน้าที่ 1.1 : About

เป็นการแสดงข้อมูลเบื้องต้นเกี่ยวกับที่มา ความสามารถ และภาพรวมในการทำงานของโปรแกรม เพื่อใช้สร้าง และเพิ่มความเข้าใจให้กับผู้ใช้งานที่ไม่เคยเห็นโปรแกรมมาก่อน

- ปุ่ม Close : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการปิดหน้าต่าง About เมื่อทำการอ่านเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 หน้าที 2 : หน้าจอหลักของโปรแกรม



รูปที่ 4.3 แสดงหน้าที 2 : หน้าจอหลักของโปรแกรม

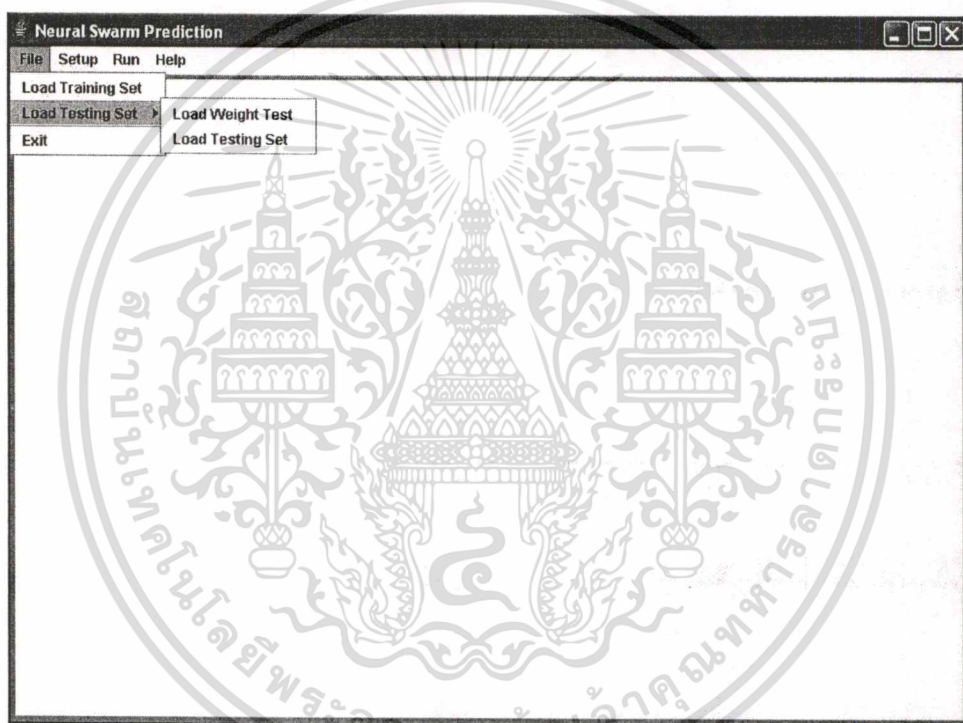
เมื่อเริ่มการกดปุ่ม Start ที่หน้าจอแรกผู้ใช้งานก็จะพบกับหน้าจอหลักของโปรแกรม ผู้ใช้งานจะต้องทำหน้าที่วางแผนที่จะเลือกลักษณะการใช้งานโปรแกรมโดยทั่วไปแล้วโปรแกรมนี้จะแบ่งการใช้งานออกเป็น 2 ส่วนหลัก ได้แก่ ส่วนการ Training และ ส่วนการ Testing โดยเมนูจะต้องทำการเลือกเป็นลำดับกันไป File > Setup > Run และ Help โดยมีสิ่งที่ผู้ใช้งานต้องพิจารณา คือ

- ปุ่ม File : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการจัดการเกี่ยวกับการโหลดไฟล์ เพื่อใช้งานโปรแกรมซึ่งจะมี 2 ส่วนหลักได้แก่ Load Training Set และ Load Testing Set
- ปุ่ม Setup : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการปรับแต่งค่าเกี่ยวกับอัลกอริทึม PSO และ โครงข่ายประสาทเทียมให้เหมาะสมกับข้อมูลนำเข้า และจุดประสงค์ของการใช้งานโปรแกรม โดยจะมี 2 โหมดได้แก่ Setup Normal Mode และ Setup Advance Mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ปุ่ม Run : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการให้โปรแกรมคำนวณและทำนายค่าจากข้อมูลที่ผู้ใช้ได้ป้อนเข้ามา โดยจะประกอบด้วย 2 ส่วน ได้แก่ Training และ Testing
- ปุ่ม Help : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการเปิดหน้าต่าง Help เพื่อใช้สำหรับอ่านในการช่วยเหลือการใช้งานโปรแกรม

4.1.4 หน้าที 2.1 : หน้าจอหลักของโปรแกรม (เมนู File)



รูปที่ 4.4 แสดงหน้าที่ 2.1 : หน้าจอหลักของโปรแกรม (เมนู File)

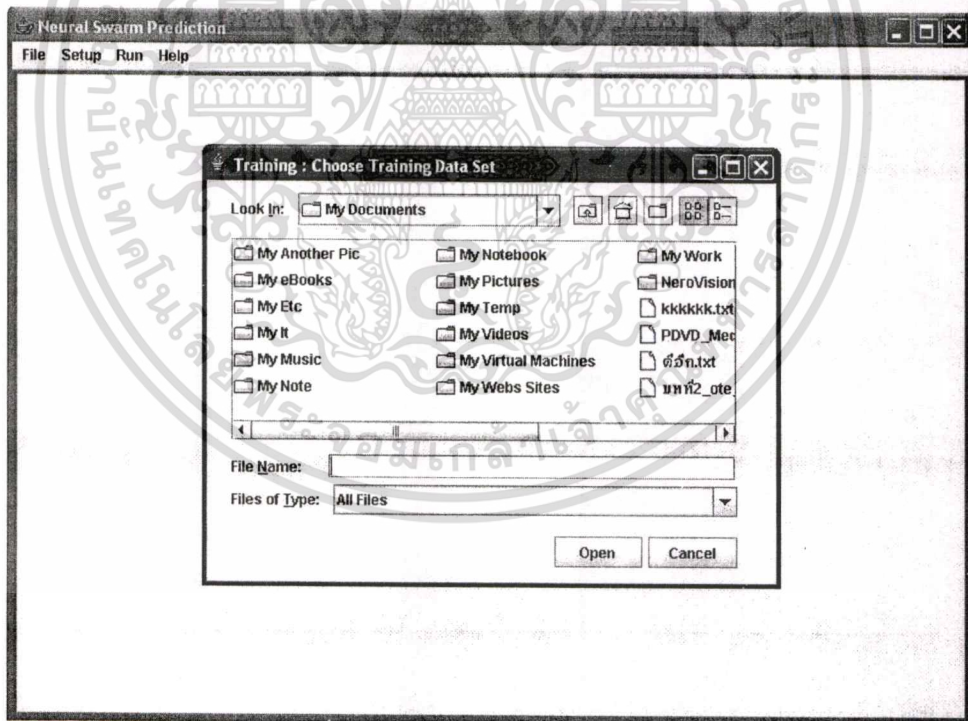
เมื่อเริ่มการกดปุ่ม File ที่หน้าจอหลักผู้ใช้งานก็จะพบกับเมนูย่อยต่างๆที่เอาไว้ใช้ในการเลือกไฟล์ที่ต้องการนำเข้ามาให้กับโปรแกรมสำหรับใช้ในขั้นตอนการ Training และ Testing เป็นหลัก โดยจะประกอบไปด้วย Load Training Set และ Load Testing Set > Load Weight Test และ Load Testing Set > Load Testing Set โดยมีสิ่งที่ผู้ใช้งานต้องพิจารณาคือ

- ปุ่ม Load Training Set : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการที่จะให้โปรแกรมโหลดนำค่าของข้อมูลนำเข้าเข้ามาซึ่งจะเป็นข้อมูลที่จะนำมาใช้ในการฝึกหัดโครงข่ายประสาทเทียมซึ่งจะประกอบไปด้วย Input และ Output ตามลักษณะที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาติให้เนาไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ปุ่ม Load Testing Set > Load Weight Test : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการที่จะให้โปรแกรมโหลดนำค่าของข้อมูลค่าถ่วงน้ำหนักเก่าที่เราเคยฝึกหัดโครงข่ายประสาทเทียม และได้บันทึกไว้มาเข้าเข้ามาซึ่งจะเป็นข้อมูลที่จะนำมาใช้ในการประหยัดเวลาในการฝึกหัดโครงข่ายประสาทเทียม โดยจะทำให้เราสามารถข้ามขั้นตอนการ Training ไป
- ปุ่ม Load Testing Set > Load Testing Set : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการที่จะให้โปรแกรม โหลดนำค่าของข้อมูลนำเข้ามาซึ่งจะเป็นข้อมูลที่จะนำมาใช้ในการทดสอบ โครงข่ายประสาทเทียมซึ่งจะประกอบไปด้วย Input และ Output ตามลักษณะที่ต้องการ
- ปุ่ม Exit : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการออกจากโปรแกรม

4.1.5 หน้า 2.1.1 : หน้าจอหน้าเข้าค่า



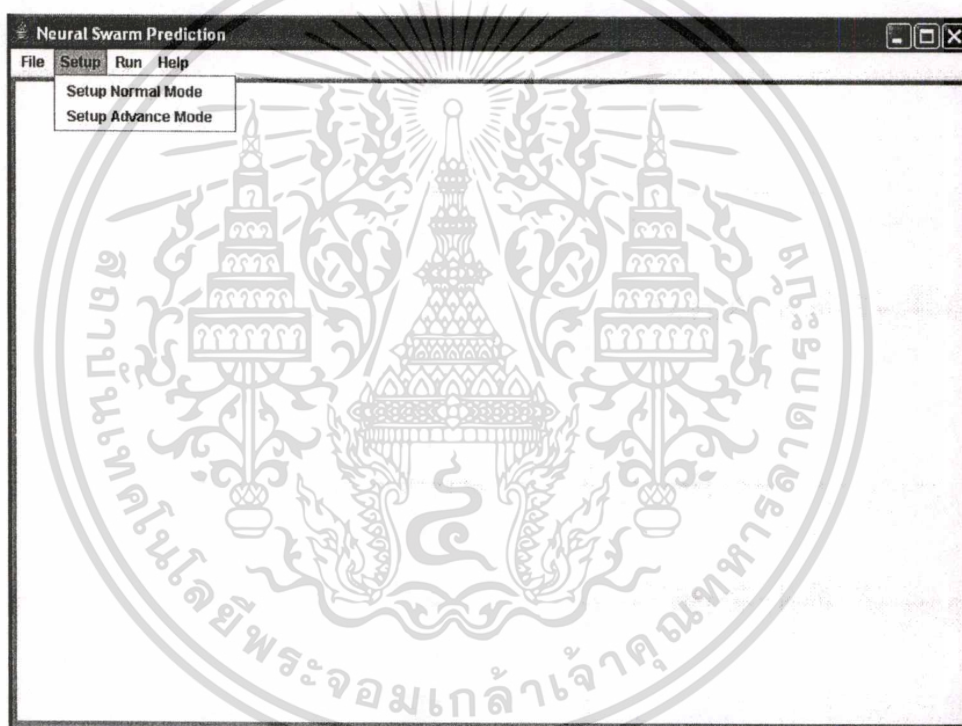
รูปที่ 4.5 แสดงหน้า 2.1.1 : หน้าจอหน้าเข้าค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าจอนี้จะเกิดขึ้นเมื่อเรากดปุ่มดังต่อไปนี้ได้แก่ ปุ่ม Load Training Set ปุ่ม Load Testing Set > Load Weight Test และปุ่ม Load Testing Set > Load Testing Set โดยมีสิ่งที่คุณใช้งานต้องพิจารณาคือ

- ปุ่ม Open : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการเลือกไฟล์ที่ใช้สำหรับสอน ฝึกหัด หรือนำเข้าค่าถ่วงน้ำหนักต่างๆของโครงข่ายประสาทเทียม
- ปุ่ม Cancel : ใช้สำหรับกดเมื่อผู้ใช้งานไม่ต้องการเลือกไฟล์

4.1.6 หน้า ที่ 2.2 : หน้าจอหลักของโปรแกรม (เมนู Setup)



รูปที่ 4.6 แสดงหน้า ที่ 2.2 : หน้าจอหลักของโปรแกรม (เมนู Setup)

เมื่อเริ่มการกดปุ่ม Setup ที่หน้าจอหลักผู้ใช้งานก็จะพบกับเมนูย่อยต่างๆที่เขาไว้ใช้ในการปรับตั้งค่าการใช้งานโปรแกรมให้เหมาะสมกับข้อมูลนำเข้าและวัตถุประสงค์ของผู้ใช้งานซึ่งประกอบไปด้วย 2 ส่วนคือ Setup Normal Mode และ Setup Advance Mode โดยจะประกอบไปด้วย Setup > Setup Normal Mode และ Setup > Setup Advance Mode โดยมีสิ่งที่คุณใช้งานต้องพิจารณาคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ปุ่ม Setup > Setup Normal Mode : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการที่จะปรับตั้งค่าโปรแกรมเป็นแบบธรรมดา
- ปุ่ม Setup > Setup Advance Mode : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการที่จะปรับตั้งค่าโปรแกรมเป็นแบบมากกว่าปกติ

4.1.7 หน้า 2.2.1 : หน้าจอ Setup Normal Mode

Normal Mode : Setting the effective variable

Particle Swarm	Neural Network
Number of Particle : 25	Number of Input Node : 4
Number of Iteration : 300	Number of Hidden Node : 25
Max Velocity : 0.022	Number of Output Node : 1

Submit

รูปที่ 4.7 แสดงหน้าที่ 2.2.1 : หน้าจอ Setup Normal Mode

เมื่อเริ่มการทำงานที่หน้าจอ Setup Normal Mode ผู้ใช้งานจะต้องทำหน้าที่ในการเลือกป้อนค่าที่เหมาะสม เพื่อใช้ในการฝึกสอนโครงข่ายประสาทเทียม โดยมีสิ่งที่ผู้ใช้งานต้องพิจารณาคือ

- แถบ Number of Particle : ใช้สำหรับให้เราเติมค่าจำนวนของพาคัลที่ต้องการให้ระบบนี้สร้างขึ้น ซึ่งโดยปกติแล้วเราจะสร้างพาคัลประมาณ 20-40 ตัว
- แถบ Number of Iteration : ใช้สำหรับให้เราเติมค่าจำนวนครั้งที่ต้องการให้พาคัลทำการปรับค่าตำแหน่งของมัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แถบ Max Velocity : ใช้สำหรับให้เราเติมค่าความเร็วที่มากที่สุดที่จะอนุญาตให้พาดิคล้มได้
- แถบ Number of Input Node : ใช้สำหรับให้เราเติมค่าจำนวนอินพุทโหนดสำหรับโครงข่ายประสาทเทียม ค่านี้ค่อนข้างสำคัญมากเราควรใส่ค่านี้ให้เท่ากับปริมาณอินพุทจากข้อมูลที่รับเข้ามา
- แถบ Number of Hidden Node : ใช้สำหรับให้เราเติมค่าจำนวนโหนดของชั้นซ่อนสำหรับโครงข่ายประสาทเทียม ตามปกติถ้าค่านี้ยิ่งมาจะทำให้โครงข่ายสามารถเลียนแบบแทน ฟังก์ชันยากๆ ได้ แต่ก็จะทำให้เสียเวลาในการคำนวณนานขึ้นค่อนข้างมาก
- ปุ่ม Submit : ใช้สำหรับกดเมื่อผู้ใช้ได้กรอกข้อมูลเบื้องต้นเสร็จหมดแล้ว และยอมรับว่าข้อมูลที่ได้ออกไปนั้นถูกต้อง

4.1.8 หน้า 2.2.2 : หน้าจอ Setup Advance Mode

Neural Swarm Prediction

File Setup Run Help

Advance Mode : Setting the effective variable

Particle Swarm		Neural Network	
Number of Particle :	<input type="text" value="25"/>	Number of Input Node :	<input type="text" value="4"/>
Number of Iteration :	<input type="text" value="250"/>	Number of Hidden Node :	<input type="text" value="25"/>
Max Velocity :	<input type="text" value="0.030"/>	Number of Output Node :	<input type="text" value="1"/>
c1 (pbest) :	<input type="text" value="2.0"/>		
c2 (gbest) :	<input type="text" value="2.10"/>		

Submit

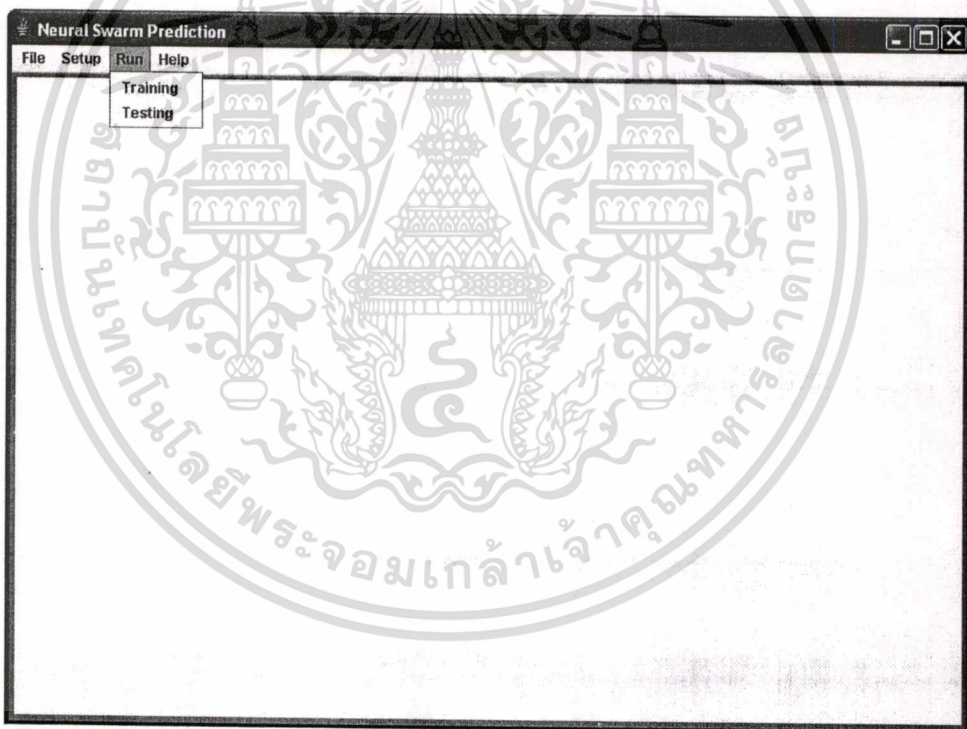
รูปที่ 4.8 แสดงหน้าที่ 2.2.2 : หน้าจอ Setup Advance Mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเริ่มการทำงานที่หน้าจอ Setup Advance Mode ผู้ใช้งานจะต้องทำหน้าที่ในการเลือกป้อนค่าที่เหมาะสมเหมือนกับหน้าจอ Setup Advance Mode เพื่อใช้ในการฝึกสอนโครงข่ายประสาทเทียมเพียงแต่ว่าจะมีค่า 2 ค่าเพิ่มขึ้นมาเพื่อให้ปรับตั้งค่าด้วย คือค่า C1 (pbest) และ C2 (gbest) โดยมีสิ่งที่ผู้ใช้งานต้องพิจารณาเพิ่มคือ

- แถบ C1 (pbest) : ใช้สำหรับให้เราเติมค่าคงที่สำหรับใช้คูณเพิ่มความสำเร็จให้กับตำแหน่งที่ดีที่สุดของแต่ละพาดิคัลเอง โดยโหมปกติค่านี้อาจมีค่าเท่ากับ 2.00
- แถบ C2 (lbest) : ใช้สำหรับให้เราเติมค่าคงที่สำหรับใช้คูณเพิ่มความสำเร็จให้กับตำแหน่งที่ดีที่สุดของทุกๆพาดิคัล โดยโหมปกติค่านี้อาจมีค่าเท่ากับ 2.00

4.1.9 หน้าที 2.3 : หน้าจอหลักของโปรแกรม (เมนู Run)



รูปที่ 4.9 แสดงหน้าที 2.3 : หน้าจอหลักของ โปรแกรม (เมนู Run)

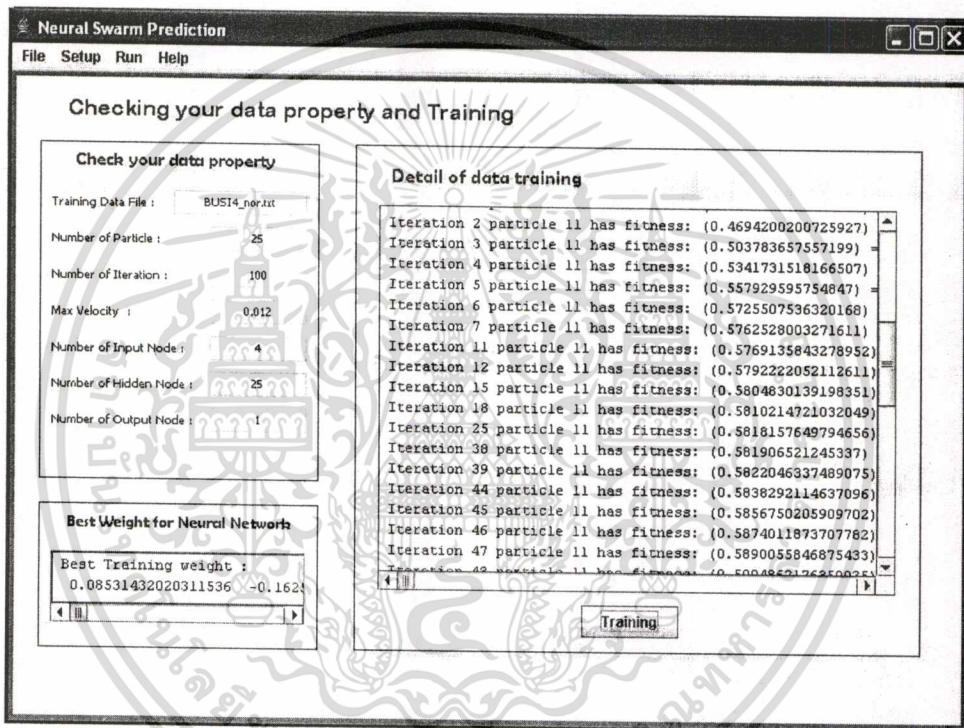
เมื่อเริ่มการกดปุ่ม Run ที่หน้าจอหลักผู้ใช้งานก็จะพบกับเมนูย่อยต่างๆที่เขาไว้ใช้ในการสั่งให้โปรแกรมทำการฝึกหัด และทดสอบโครงข่ายประสาทเทียม และแสดงค่ารายละเอียดในการทำงาน โดยจะประกอบไปด้วย Run > Training และ Run > Testing โดยมีสิ่งที่ผู้ใช้งานต้องพิจารณา

คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ปุ่ม Run > Training : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการที่จะให้โปรแกรมดำเนินขั้นตอนการฝึกหัดโครงข่ายประสาทเทียม
- ปุ่ม Run > Testing : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการที่จะให้โปรแกรมดำเนินขั้นตอนการทดสอบโครงข่ายประสาทเทียม

4.1.10 หน้า ที่ 2.3.1 : หน้าจอ Training



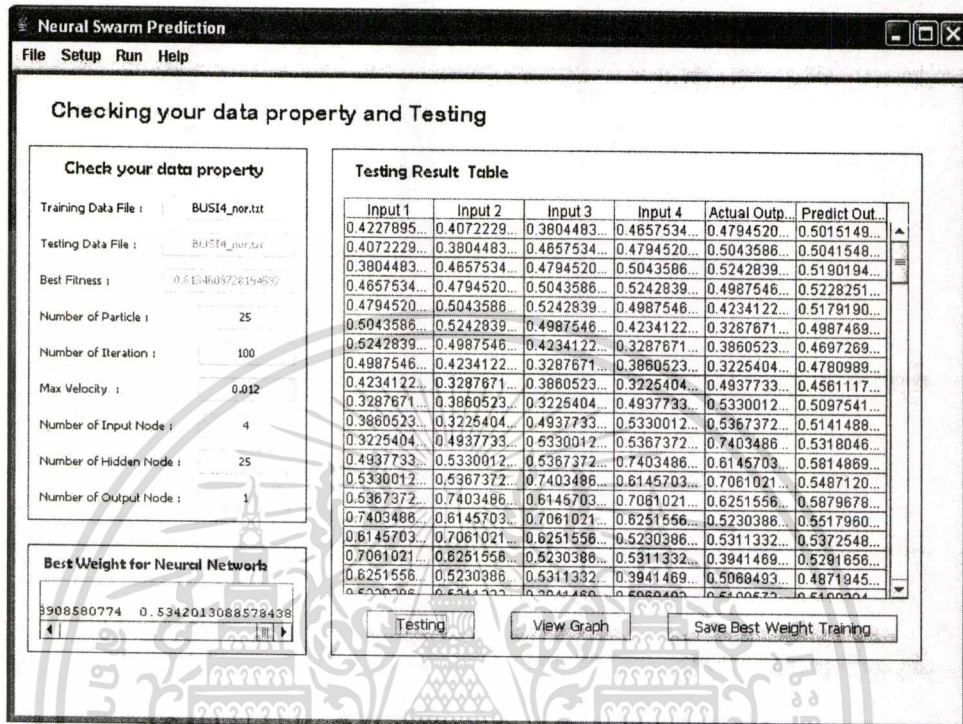
รูปที่ 4.10 แสดงหน้า ที่ 2.3.1 : หน้าจอ Training

เมื่อเริ่มการทำงานที่หน้าจอ Training ผู้ใช้งานจะต้องทำหน้าที่ในการตรวจเช็คค่าที่ป้อนเข้าไปว่าได้ป้อนค่าที่ถูกต้องตามความต้องการแล้วหรือยังถ้ายังจะได้กลับไปแก้ตอนกรอกข้อมูลใหม่เพื่อที่จะได้มีเสี้ยวเวลาโดยมีสิ่งๆที่ผู้ใช้งานต้องพิจารณาคือ

- ปุ่ม Training : ใช้สำหรับกดเมื่อผู้ใช้ได้เช็คข้อมูลแล้วว่าที่ได้กรอกข้อมูลเบื้องต้นไปนั้นถูกต้องตามความต้องการแล้ว ผู้ใช้ยืนยันว่าจะทำการฝึกสอนโครงข่ายตามคุณลักษณะนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.11 หน้า ที่ 2.3.2 : หน้าจอ Testing



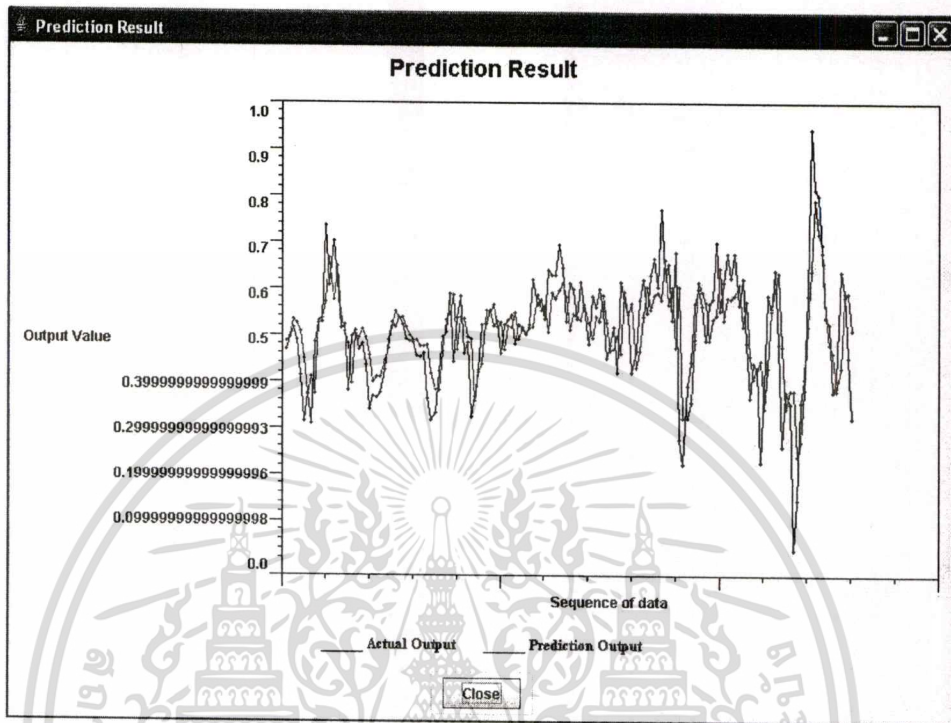
รูปที่ 4.11 แสดงหน้า ที่ 2.3.2 : หน้าจอ Testing

เมื่อเริ่มการทำงานที่หน้าจอ Testing ผู้ใช้งานจะต้องทำหน้าที่ในการตรวจเช็คค่าที่ป้อนเข้าไปว่าได้ป้อนค่าที่ถูกต้องตามความต้องการแล้วหรือยังถ้ายังจะได้กลับไปแก้ตอนกรอกข้อมูลใหม่เพื่อที่จะได้มีเสียเวลาโดยมีสิ่งๆที่ผู้ใช้งานต้องพิจารณาคือ

- ปุ่ม Testing : ใช้สำหรับกดเมื่อผู้ใช้ได้เช็คข้อมูลแล้วว่าที่ได้กรอกข้อมูลเบื้องต้นไปนั้นถูกต้องตามความต้องการแล้ว ผู้ใช้ยืนยันว่าจะทำการทดสอบโครงข่ายตามคุณลักษณะนี้
- ปุ่ม View Graph : ใช้สำหรับกดเมื่อผู้ใช้ต้องการที่จะดูผลเปรียบเทียบระหว่างเอาท์พุทของจริงกับ เอาท์พุทที่เราทำนายได้ให้มากยิ่งขึ้น
- ปุ่ม Save Best Weight Training : ใช้สำหรับกดเมื่อผู้ใช้ต้องการที่จะบันทึกค่านี้เก็บเอาไว้ใช้งานในครั้งต่อไปโดยไม่ต้องไปเสียเวลาฝึกสอนโครงข่ายประสาทเทียมใหม่อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.12 หน้า 2.3.2.1 : Graph

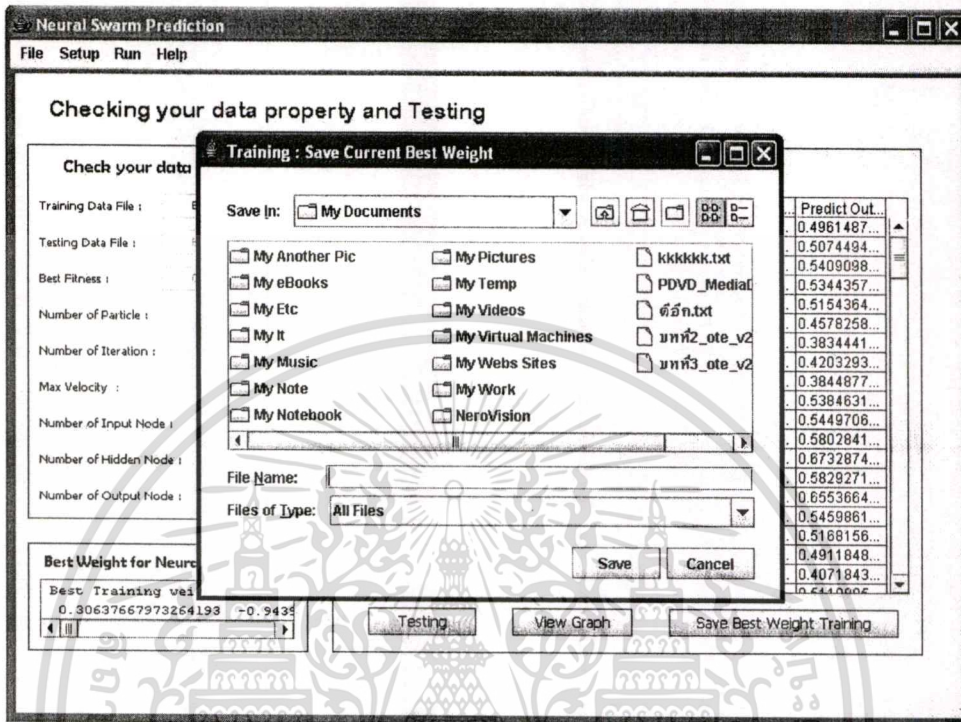


รูปที่ 4.12 แสดงหน้าที่ 2.3.2.1 : Graph

เมื่อเริ่มการทำงานที่หน้า Graph ผู้ใช้งานจะพบกับค่าการแสดงผลละเอียดจากการทดสอบโครงข่ายจากชุดข้อมูลทดสอบที่ใส่ให้ไปกับระบบในหน้าจอ Testing ที่มีลักษณะเป็นกราฟทั้งนี้เพื่อความง่ายแก่การตรวจสอบระหว่างค่าที่เกิดขึ้นจริงกับค่าที่ระบบสามารถทำนายได้มากยิ่งขึ้น โดยมีสิ่งที่ผู้ใช้งานต้องพิจารณาคือ

- ปุ่ม Close : ใช้สำหรับกดเมื่อผู้ใช้ต้องการที่จะปิดหน้าต่างการแสดงผลการทดสอบโครงข่ายนี้

4.1.13 หน้า 2.3.2.2 : Save Best Weight Training



รูปที่ 4.13 แสดงหน้าที่ 2.3.2.2 : Save Best Weight Training

หน้า Save Best Weight Training จะปรากฏขึ้นเมื่อผู้ใช้งานได้กดปุ่ม Save Best Weight Training เมื่อผู้ต้องการจะเซฟไฟล์ค่าถ่วงน้ำหนักเก็บเอาไว้

- ปุ่ม Save : ใช้สำหรับกดเมื่อผู้ใช้งานต้องการเลือกไฟล์ที่ใช้เก็บค่าถ่วงน้ำหนักสำหรับโครงข่ายประสาทเทียม
- ปุ่ม Cancel : ใช้สำหรับกดเมื่อผู้ใช้งานไม่ต้องการเลือกไฟล์และย้อนกลับไปหน้าที่แล้ว

4.2 การทดลองใช้งาน และปรับค่าโปรแกรม

ในหัวข้อนี้เป็นทดลองเพื่อทดสอบการทำงานร่วมกันระหว่าง PSO อัลกอริทึม และโครงข่ายประสาทเทียมว่าเมื่อเรามีการเปลี่ยนแปลงค่าตัวแปรต่างๆแล้วส่งผลต่อการทำนายผลอย่างไรบ้างซึ่งในที่นี้เราจะเน้นการทดสอบจากตัวแปร โดยจะแบ่งการปรับค่าตัวแปรออกได้เป็น 2 ส่วนคือ ส่วนของอัลกอริทึม PSO และส่วนของโครงข่ายประสาทเทียม โดยจะมีสิ่งที่เราจะพิจารณา ดังต่อไปนี้

1. ค่าที่ปรับตั้งให้กับตัวโปรแกรม ได้แก่ จำนวนพาดิคัล, จำนวนรอบในการทำซ้ำ, ค่าความเร็วสูงสุด และจำนวนโหนดของชั้นซ่อน

2. ผลที่ได้ที่เราต้องพิจารณาตัดสินจากความคลาดเคลื่อนและ เวลาที่ใช้ในการทำงาน

3. เราจะทำการไม่เปลี่ยนแปลงค่าที่เราจะตั้งให้กับโครงข่ายประสาทเทียมโดยเราจะปรับตั้งค่าไว้ดังต่อไปนี้

3.1) จำนวนของอินพุทโหนด = 4 โหนด

3.2) จำนวนของเอาต์พุทโหนด = 1 โหนด

4. เราจะนำข้อมูลที่ใช้ในการทดสอบมาจากเว็บไซต์ <http://forecasters.org>. โดยมีชื่อขององค์กรว่า International Institute of Forecasters ซึ่งเป็นเว็บไซต์ที่มีวัตถุประสงค์เพื่อที่จะกระตุ้นการพัฒนา และกระจาย วิธีการใช้งานความรู้เกี่ยวกับการพยากรณ์ ที่ได้รับการรับรองมาแล้วในส่วนของข้อมูล Time Series ซึ่งมีปริมาณข้อมูลจำนวนประมาณ 160 ชุดข้อมูล มาทำการทดสอบ 5 ลักษณะชนิดข้อมูล ได้แก่ BUSINV, COMPIND, FIXINV, IPDIMP และ REALGNP

5. เราจะแสดงรายละเอียดของข้อมูลที่จะนำมาทดสอบได้ดังต่อไปนี้

5.1) BUSINV เป็นข้อมูล การเปลี่ยนแปลงของปริมาณสินค้าคงคลังทุกๆ 3 เดือน ตั้งแต่ปี 1947-1987 และมีลักษณะของตัวอย่างข้อมูลต้นแบบดังต่อไปนี้

BUSINV 4 163

99.600 97.100 92.800 106.500 108.700 112.700

115.900 111.800 99.700 84.500 93.700 83.500

111.000 117.300 117.900 150.600 130.400 145.100

132.100 115.700 117.000 95.000 113.100 114.900

5.2) COMPIND เป็นข้อมูล การเปลี่ยนแปลงของค่าดัชนีค่าตอบแทนของธุรกิจที่ไม่เกี่ยวกับภาคเกษตรกรรมทุกๆ 3 เดือน ตั้งแต่ปี 1947-1987 และมีลักษณะของตัวอย่างข้อมูลต้นแบบดังต่อไปนี้

COMPIND 4 163

17.400	17.800	18.200	18.600	19.100	19.400
19.800	20.000	20.100	20.100	20.200	20.200
20.800	21.100	21.500	22.100	22.400	23.100
23.600	24.000	24.100	24.300	24.600	25.200

- 5.3) FIXINV เป็นข้อมูล การเปลี่ยนแปลงผลรวมเฉพาะส่วนคงที่ที่ไม่เกี่ยวกับการลงทุนประเภทอสังหาริมทรัพย์ทุกๆ 3 เดือน ตั้งแต่ปี 1947-1987 และมีลักษณะของตัวอย่างข้อมูลต้นแบบดังต่อไปนี้

FIXINV 4 163

123.200	121.100	119.900	122.200	128.300	125.600
125.900	129.800	123.800	116.900	110.700	108.000
111.100	121.400	131.500	132.100	128.000	131.900
134.400	132.500	133.200	134.000	123.300	132.000

- 5.4) IPDIMP เป็นข้อมูล การเปลี่ยนแปลงราคาที่แน่นอนที่ต่ำลงของสินค้านำเข้าทุกๆ 3 เดือน ตั้งแต่ปี 1947-1987 และมีลักษณะของตัวอย่างข้อมูลต้นแบบดังต่อไปนี้

IPDIMP 4 163

19.900	21.200	21.000	21.600	22.000	22.400
22.700	22.600	21.700	21.300	21.000	20.900
21.000	21.400	22.700	24.300	25.900	27.300
27.200	26.300	26.000	25.600	25.000	24.700

- 5.5) REALGNP เป็นข้อมูลการเปลี่ยนแปลงของผลิตภัณฑ์มวลรวมประชาชาติทุกๆ 3 เดือน ตั้งแต่ปี 1947-1987 และมีลักษณะของตัวอย่างข้อมูลต้นแบบดังต่อไปนี้

REALGNP 4 163

1056.500	1063.200	1067.100	1080.000	1086.800	1106.100
1116.300	1125.500	1112.400	1105.900	1114.300	1103.300
1148.200	1181.000	1225.300	1260.200	1286.600	1320.400
1349.800	1356.000	1369.200	1365.900	1378.200	1406.800

6. ลักษณะของการแสดงผลการทดลองจะทำการแสดงตารางสรุปผลการทดลองเป็นตารางของทุกๆชุด และจะทำการแสดงกราฟส่วนของข้อมูลการฝึกหัด โครงข่าย และกราฟส่วนของ การทดสอบ โครงข่ายของชุดค่าถ่วงน้ำหนักที่ดีที่สุดของแต่ละการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เราจะทำการแบ่งข้อมูลทั้งหมดของชุดข้อมูลที่จะใช้ในการทดลองออกเป็น 2 ส่วนด้วยกัน ได้แก่ ชุดฝึกหัดโครงข่ายประสาทเทียม 70% และชุดทดสอบโครงข่ายประสาทเทียม 30%

8. เราจะทำการปรับเปลี่ยนข้อมูล (Normalization) ให้มีค่าอยู่ในช่วงระหว่าง 0.0 ถึง 1.0 เพื่อให้ข้อมูลมีความเหมาะสมในการทำงานกับอัลกอริทึมได้และเพื่อให้มีความสอดคล้องกับฟังก์ชันโยกย้ายที่เลือกใช้ในการพัฒนาระบบ ซึ่งในที่นี้คือ ซิกมอยด์ฟังก์ชัน (Sigmoid Function) โดยให้ค่าสูงสุดของชุดข้อมูลมีค่าบวกเพิ่มอีก 5% ของค่าสูงสุด และให้ค่าต่ำสุดของชุดข้อมูลมีค่าลดลง 5% ของค่าต่ำสุดทั้งนี้เพื่อความยืดหยุ่นในการแสดงผลการทดลองโดยสูตรที่ใช้ในการนอร์มอลไลเซชันคือ

สูตรการนอร์มอลไลเซชัน

$$\text{newmax} = \text{max} + 5\%$$

$$\text{newmin} = \text{min} - 5\%$$

$$\text{newdata} = (\text{data} - \text{newmin}) / (\text{newmax} - \text{newmin})$$

โดย data คือ ข้อมูลที่ต้องการนอร์มอลไลเซชัน

max คือ ข้อมูลที่มีค่าสูงสุด

min คือ ข้อมูลที่มีค่าต่ำสุด

newmax คือ ข้อมูลที่มีค่าสูงสุดที่ถูกปรับค่าแล้ว

newmin คือ ข้อมูลที่มีค่าต่ำสุดที่ถูกปรับค่าแล้ว

newdata คือ ข้อมูลที่ได้ถูกทำการนอร์มอลไลเซชันแล้ว

ซึ่งหลังจากนอร์มอลไลเซชันแล้วจะทำให้ข้อมูลภายในชุดนั้นมีค่าอยู่ระหว่าง 0.0 ถึง 1.0 เราจึงสามารถนำมาใช้งานในการทำการใช้สำหรับฝึกสอน และฝึกหัดโครงข่ายประสาทเทียมของระบบได้โดยที่หลังจากเราได้ผลการทำนายแล้ว ถ้าเราต้องการแปลงค่าที่ทำนายได้จาก 0.0 ถึง 1.0 ให้เป็นค่าจริงเราก็ต้องทำการแปลงกลับจากสูตร

สูตรการแปลงกลับจากการนอร์มอลไลเซชัน

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

4.2.1 การทดลองที่ 1 ข้อมูล BUSINV

4.2.1.1 ตารางสรุปผลการทดลองที่ 1

ตารางที่ 4.1 เป็นการแสดงตารางสรุปรวมผลการทดลองซึ่งใน 1 การทดลองจะทำการทดลองทั้งหมด 3 ครั้งดังปรากฏค่าของ ค่า Fitness 1, ค่า Fitness 2 และค่า Fitness 3 ดังตารางที่ 4.1 โดยจะประกอบไปด้วยทดลองปรับตั้งค่าโปรแกรมทั้งหมด 11 ชุดด้วยกัน จะเห็นว่าในการทดลองแต่ละครั้งแม้เรามีการปรับตั้งค่าตัวแปรที่เหมือนกันแต่ผลการทำนายที่ได้อาจมีค่าความถูกต้องต่างกันอย่างมากก็ได้

การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	ค่า Fitness 1	ค่า Fitness 2	ค่า Fitness 3
1	25	150	0.04	10	2.1796	2.2588	2.1892
2	25	300	0.04	10	2.2817	2.2859	2.1990
3	50	300	0.04	10	2.2731	2.2341	2.2823
4	25	300	0.08	10	2.2829	2.2850	2.2793
5	25	300	0.12	10	2.2569	2.2671	2.2441
6	25	300	0.25	10	2.2339	2.2337	2.1987
7	25	300	0.04	5	1.7336	1.8799	1.8664
8	25	300	0.04	15	2.2825	2.2925	2.2826
9	25	300	0.04	20	2.2990	2.3022	2.3055
10	25	300	0.04	25	2.2954	2.3013	2.2961
11	25	300	0.04	35	2.2989	2.2989	2.2990

ตารางที่ 4.1 แสดงผลการทดลองที่ 1 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์

4.2.1.2 ตารางสรุปผลการทดลองที่ 1 แบบเฉลี่ยค่าความเหมาะสม

ตารางที่ 4.2 เป็นการแสดงตารางสรุปรวมผลการทดลองแบบการรวมค่า Fitness ทั้ง 3 ค่าเข้าด้วยกัน และนำมาหาค่าเฉลี่ย เพื่อที่จะสามารถพิจารณาวิเคราะห์ข้อมูล ได้ง่ายมากยิ่งขึ้น

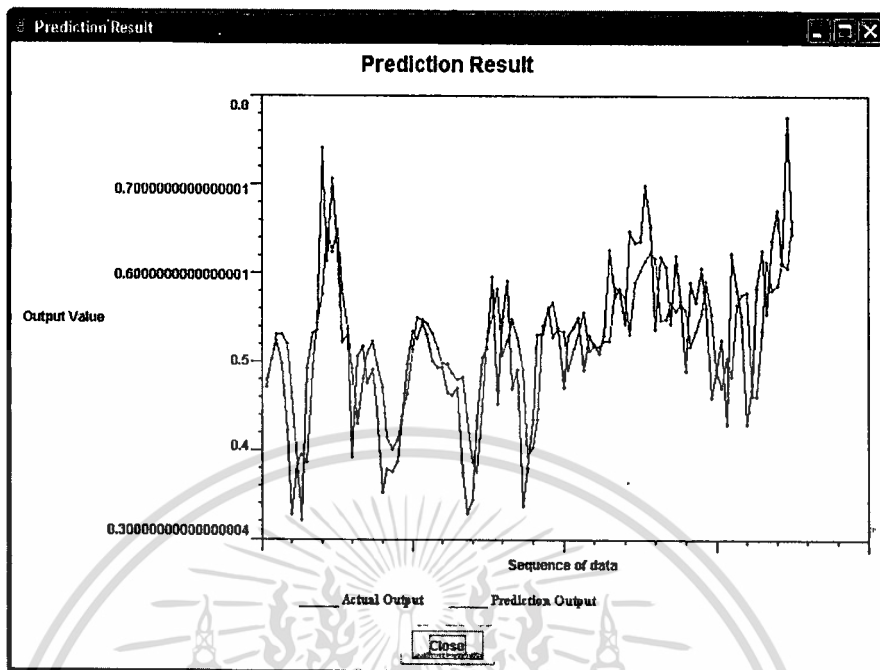
การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	เวลาที่ใช้ในการฝึกสอน (วินาที)	ค่าเฉลี่ย Fitness
1	25	150	0.04	10	29	2.2092
2	25	300	0.04	10	61	2.2555
3	50	300	0.04	10	115	2.2631
4	25	300	0.08	10	60	2.2824
5	25	300	0.12	10	62	2.2560
6	25	300	0.25	10	59	2.2221
7	25	300	0.04	5	41	1.8266
8	25	300	0.04	15	78	2.2858
9	25	300	0.04	20	102	2.3022
10	25	300	0.04	25	119	2.2976
11	25	300	0.04	35	145	2.2989

ตารางที่ 4.2 แสดงผลการทดลองที่ 1 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า

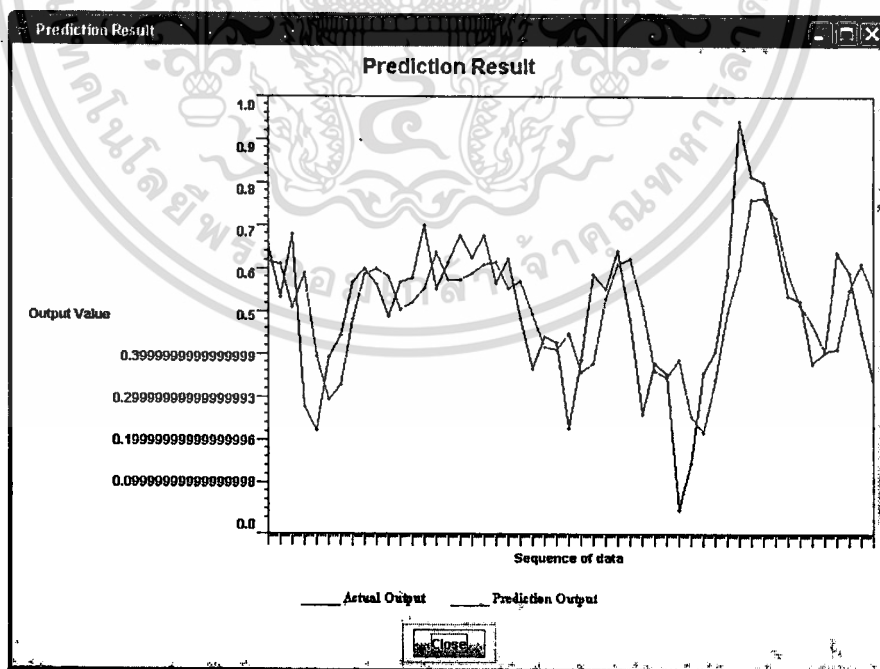
4.2.1.3 กราฟแสดงผลการทดสอบโครงข่ายประสาทเทียม

กราฟแสดงผลการใช้ข้อมูลทดสอบโครงข่ายประสาทเทียมซึ่งเราจะใช้ค่าถ่วงน้ำหนักที่ดีที่สุดที่บันทึกได้จากผลการทดลองที่ 1 มาทำการทดสอบโดยในหัวข้อนี้เราจะสามารถแบ่งกราฟออกเป็น 2 ส่วนด้วยกันได้แก่

1. กราฟที่เกิดจากการใช้ข้อมูลการฝึกสอนโครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.14
2. กราฟที่เกิดจากการใช้ข้อมูลการทดสอบโครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.15



รูปที่ 4.14 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ
ของการทดลองที่ 1



รูปที่ 4.15 แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ
ของการทดลองที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 การทดลองที่ 2 ข้อมูล COMPIND

4.2.2.1 ตารางสรุปผลการทดลองที่ 2

ตารางที่ 4.3 เป็นการแสดงตารางสรุปรวมผลการทดลองซึ่งใน 1 การทดลองจะทำการทดลองทั้งหมด 3 ครั้งดังปรากฏค่าของ ค่า Fitness 1, ค่า Fitness 2 และค่า Fitness 3 ดังตารางที่ 4.3 โดยจะประกอบไปด้วยทดลองปรับตั้งค่าโปรแกรมทั้งหมด 11 ชุดด้วยกัน จะเห็นว่าในการทดลองแต่ละครั้งแม้เรามีการปรับตั้งค่าตัวแปรที่เหมือนกันแต่ผลการทำนายที่ได้อาจมีค่าความถูกต้องต่างกันอย่างมากก็ได้

การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	ค่า Fitness 1	ค่า Fitness 2	ค่า Fitness 3
1	25	150	0.04	10	47.1667	21.8534	86.3705
2	25	300	0.04	10	97.3189	96.1938	91.7997
3	50	300	0.04	10	75.0745	96.8050	104.801
4	25	300	0.08	10	97.1413	97.4688	95.7831
5	25	300	0.12	10	28.1287	27.1141	100.2298
6	25	300	0.25	10	75.1820	15.0577	78.8684
7	25	300	0.04	5	4.2041	2.8581	4.4194
8	25	300	0.04	15	81.8340	82.2847	79.7559
9	25	300	0.04	20	70.7056	65.0939	65.8598
10	25	300	0.04	25	74.3631	65.9873	73.3811
11	25	300	0.04	35	55.6620	68.1697	62.5981

ตารางที่ 4.3 แสดงผลการทดลองที่ 2 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์

4.2.2.2 ตารางสรุปผลการทดลองที่ 2 แบบเฉลี่ยค่าความเหมาะสม

ตารางที่ 4.4 เป็นการแสดงตารางสรุปรวมผลการทดลองแบบการรวมค่า Fitness ทั้ง 3 ค่าเข้าด้วยกัน และนำมาหาค่าเฉลี่ย เพื่อที่จะสามารถพิจารณาวิเคราะห์ข้อมูลได้ง่ายมากยิ่งขึ้น

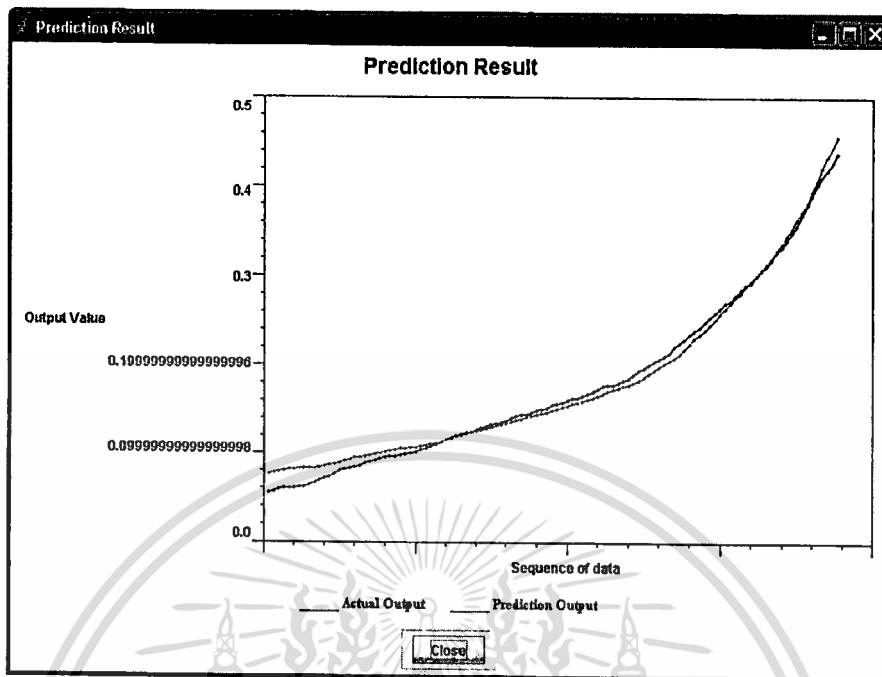
การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	เวลาที่ใช้ในการฝึกสอน (วินาที)	ค่าเฉลี่ย Fitness
1	25	150	0.04	10	29	51.7968
2	25	300	0.04	10	61	95.1041
3	50	300	0.04	10	115	92.2268
4	25	300	0.08	10	60	96.7977
5	25	300	0.12	10	62	51.8242
6	25	300	0.25	10	59	56.3693
7	25	300	0.04	5	41	3.8272
8	25	300	0.04	15	78	81.2915
9	25	300	0.04	20	102	67.2197
10	25	300	0.04	25	119	71.2438
11	25	300	0.04	35	145	62.1432

ตารางที่ 4.4 แสดงผลการทดลองที่ 2 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า

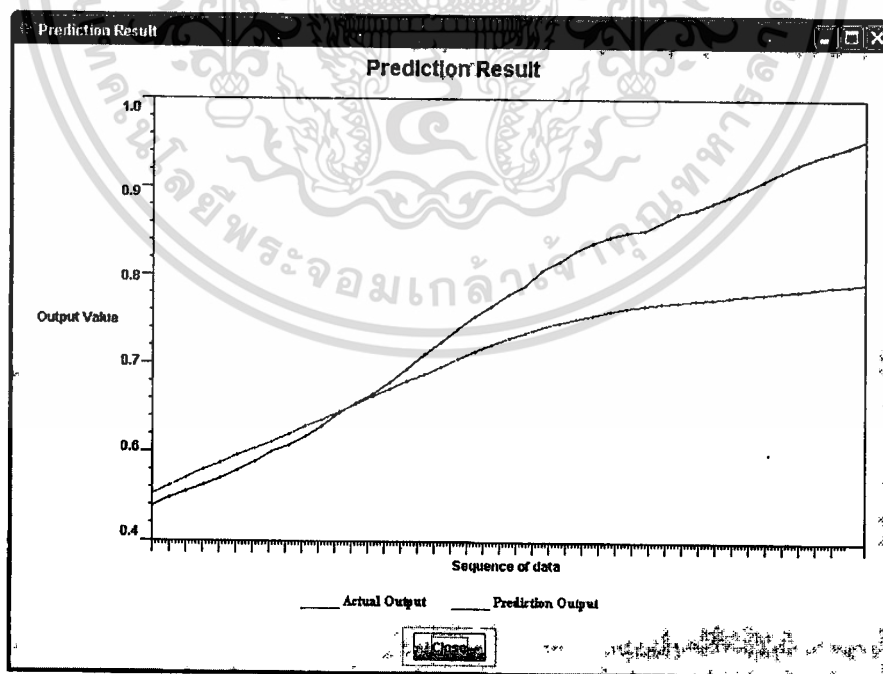
4.2.2.3 กราฟแสดงผลการทดสอบโครงข่ายประสาทเทียม

กราฟแสดงผลการใช้ข้อมูลทดสอบ โครงข่ายประสาทเทียมซึ่งเราจะใช้ค่าถ่วงน้ำหนักที่ดีที่สุดที่บันทึกได้จากผลการทดลองที่ 2 มาทำการทดสอบโดยในหัวข้อนี้เราจะสามารถแบ่งกราฟออกเป็น 2 ส่วนด้วยกันได้แก่

1. กราฟที่เกิดจากการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบ โครงข่ายประสาทเทียม ดังรูปที่ 4.16
2. กราฟที่เกิดจากการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบ โครงข่ายประสาทเทียม ดังรูปที่ 4.17



รูปที่ 4.16 แสดงการใช้ข้อมูลการฝึกสอนโครงข่ายประสาทเทียมในการทดสอบของการทดลองที่ 2



รูปที่ 4.17 แสดงการใช้ข้อมูลการทดสอบโครงข่ายประสาทเทียมในการทดสอบ

ของการทดลองที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 การทดลองที่ 3 ข้อมูล FIXINV

4.2.3.1 ตารางสรุปผลการทดลองที่ 3

ตารางที่ 4.5 เป็นการแสดงตารางสรุปรวมผลการทดลองซึ่งใน 1 การทดลองจะทำการทดลองทั้งหมด 3 ครั้งดังปรากฏค่าของ ค่า Fitness 1, ค่า Fitness 2 และค่า Fitness 3 ดังตารางที่ 4.5 โดยจะประกอบไปด้วยทดลองปรับตั้งค่าโปรแกรมทั้งหมด 11 ชุดด้วยกัน จะเห็นว่าในการทดลองแต่ละครั้งแม้เรามีการปรับตั้งค่าตัวแปรที่เหมือนกันแต่ผลการทำนายที่ได้อาจมีค่าความถูกต้องต่างกันอย่างมากก็ได้

การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	ค่า Fitness 1	ค่า Fitness 2	ค่า Fitness 3
1	25	150	0.04	10	14.2147	12.8615	12.9073
2	25	300	0.04	10	15.1358	11.7904	14.6477
3	50	300	0.04	10	13.8729	15.1283	15.2417
4	25	300	0.08	10	15.6571	14.8801	13.2089
5	25	300	0.12	10	11.5658	13.4865	12.8241
6	25	300	0.25	10	12.7671	13.2213	12.6047
7	25	300	0.04	5	1.4190	1.4196	1.5576
8	25	300	0.04	15	15.1932	21.3282	17.7411
9	25	300	0.04	20	19.4195	18.7440	18.4321
10	25	300	0.04	25	24.4054	20.1528	20.0330
11	25	300	0.04	35	20.4040	19.6641	17.7227

ตารางที่ 4.5 แสดงผลการทดลองที่ 3 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์

4.2.3.2 ตารางสรุปผลการทดลองที่ 3 แบบเฉลี่ยค่าความเหมาะสม

ตารางที่ 4.6 เป็นการแสดงตารางสรุปรวมผลการทดลองแบบการรวมค่า Fitness ทั้ง 3 ค่าเข้าด้วยกัน และนำมาหาค่าเฉลี่ย เพื่อที่จะสามารถพิจารณาวิเคราะห์ข้อมูลได้ง่ายมากยิ่งขึ้น

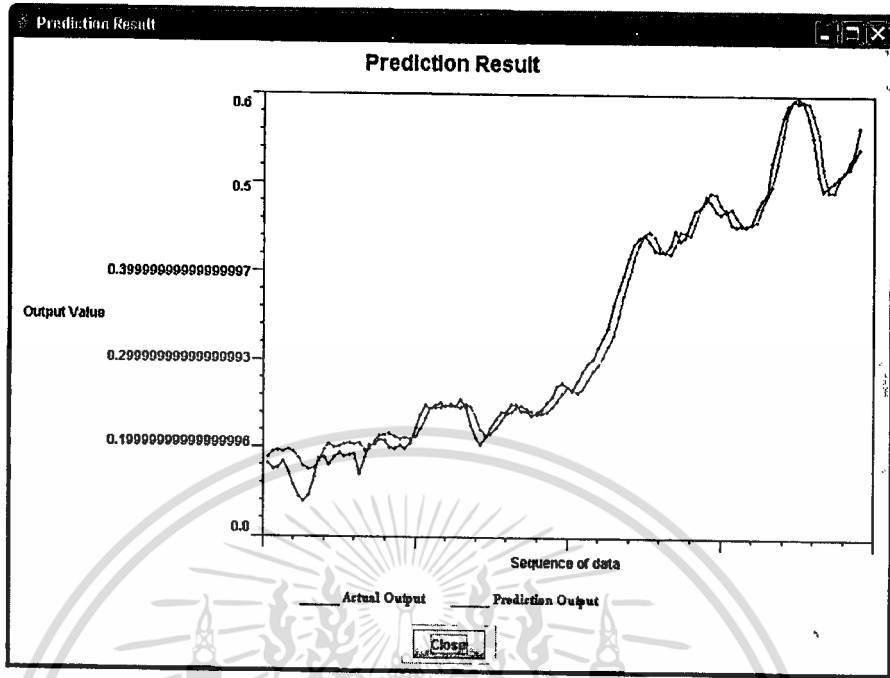
การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	เวลาที่ใช้ในการฝึกสอน (วินาที)	ค่าเฉลี่ย Fitness
1	25	150	0.04	10	29	13.3278
2	25	300	0.04	10	61	13.8579
3	50	300	0.04	10	115	14.7476
4	25	300	0.08	10	60	14.5820
5	25	300	0.12	10	62	12.6254
6	25	300	0.25	10	59	12.8643
7	25	300	0.04	5	41	1.4654
8	25	300	0.04	15	78	18.0875
9	25	300	0.04	20	102	18.8652
10	25	300	0.04	25	119	21.5304
11	25	300	0.04	35	145	19.2636

ตารางที่ 4.6 แสดงผลการทดลองที่ 3 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า

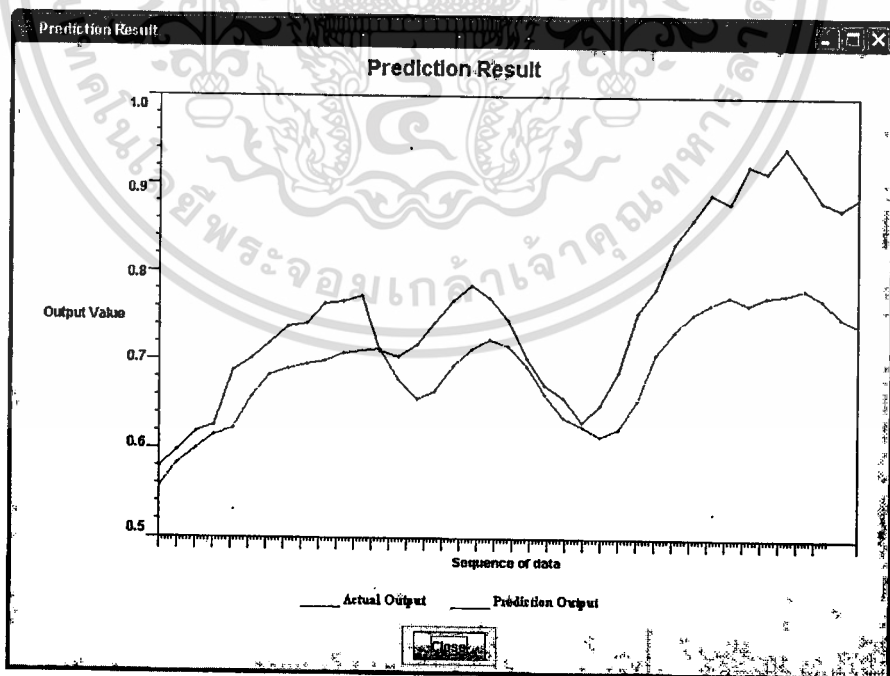
4.2.3.3 กราฟแสดงผลการทดสอบโครงข่ายประสาทเทียม

กราฟแสดงผลการใช้ข้อมูลทดสอบโครงข่ายประสาทเทียมซึ่งเราจะใช้ค่าถ่วงน้ำหนักที่ดีที่สุดที่บันทึกได้จากผลการทดลองที่ 3 มาทำการทดสอบโดยในหัวข้อนี้เราจะสามารถแบ่งกราฟออกเป็น 2 ส่วนด้วยกันได้แก่

1. กราฟที่เกิดจากการใช้ข้อมูลการฝึกสอนโครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.18
2. กราฟที่เกิดจากการใช้ข้อมูลการทดสอบโครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.19



รูปที่ 4.18 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ
ของการทดลองที่ 3



รูปที่ 4.19 แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ
ของการทดลองที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 การทดลองที่ 4 ข้อมูล IPDIMP

4.2.4.1 ตารางสรุปผลการทดลองที่ 4

ตารางที่ 4.7 เป็นการแสดงตารางสรุปรวมผลการทดลองซึ่งใน 1 การทดลองจะทำการทดลองทั้งหมด 3 ครั้งดังปรากฏค่าของ ค่า Fitness 1, ค่า Fitness 2 และค่า Fitness 3 ดังตารางที่ 4.7 โดยจะประกอบไปด้วยทดลองปรับตั้งค่าโปรแกรมทั้งหมด 11 ชุดด้วยกัน จะเห็นว่าในการทดลองแต่ละครั้งแม้เรามีการปรับตั้งค่าตัวแปรที่เหมือนกันแต่ผลการทำนายที่ได้อาจมีค่าความถูกต้องต่างกันอย่างมากก็ได้

การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	ค่า Fitness 1	ค่า Fitness 2	ค่า Fitness 3
1	25	150	0.04	10	25.9592	24.0597	22.1461
2	25	300	0.04	10	26.1779	26.2575	26.5846
3	50	300	0.04	10	26.7510	26.3994	26.0146
4	25	300	0.08	10	24.3126	21.7448	25.1587
5	25	300	0.12	10	23.3703	22.6491	22.7080
6	25	300	0.25	10	18.2938	15.7674	8.7039
7	25	300	0.04	5	5.1503	5.1358	5.0090
8	25	300	0.04	15	28.9092	27.6480	27.6198
9	25	300	0.04	20	30.0371	31.6456	28.7956
10	25	300	0.04	25	25.5896	29.9801	31.8053
11	25	300	0.04	35	29.8405	30.6900	29.3771

ตารางที่ 4.7 แสดงผลการทดลองที่ 4 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์

4.2.4.2 ตารางสรุปผลการทดลองที่ 4 แบบเฉลี่ยค่าความเหมาะสม

ตารางที่ 4.8 เป็นการแสดงตารางสรุปรวมผลการทดลองแบบการรวมค่า Fitness ทั้ง 3 ค่าเข้าด้วยกัน และนำมาหาค่าเฉลี่ย เพื่อที่จะสามารถพิจารณาวิเคราะห์ข้อมูล ได้ง่ายมากยิ่งขึ้น

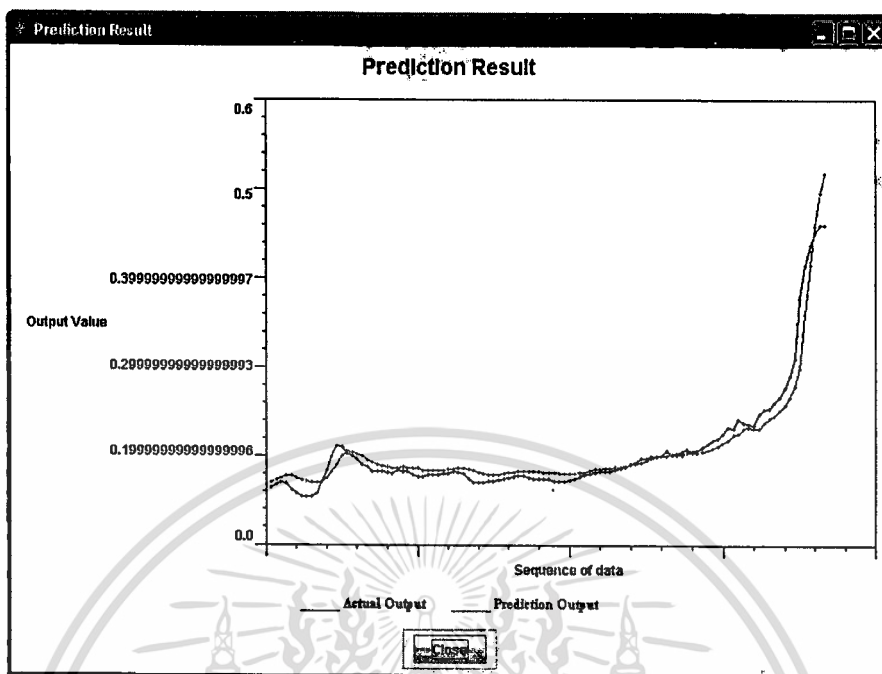
การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	เวลาที่ใช้ในการฝึกสอน (วินาที)	ค่าเฉลี่ย Fitness
1	25	150	0.04	10	29	24.055
2	25	300	0.04	10	61	26.3400
3	50	300	0.04	10	115	26.3883
4	25	300	0.08	10	60	23.7387
5	25	300	0.12	10	62	22.9091
6	25	300	0.25	10	59	14.2550
7	25	300	0.04	5	41	5.0983
8	25	300	0.04	15	78	28.0590
9	25	300	0.04	20	102	30.1594
10	25	300	0.04	25	119	29.1250
11	25	300	0.04	35	145	29.9692

ตารางที่ 4.8 แสดงผลการทดลองที่ 4 ในการทดสอบการปรับค่าที่ตั้งให้กับโปรแกรม เพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า

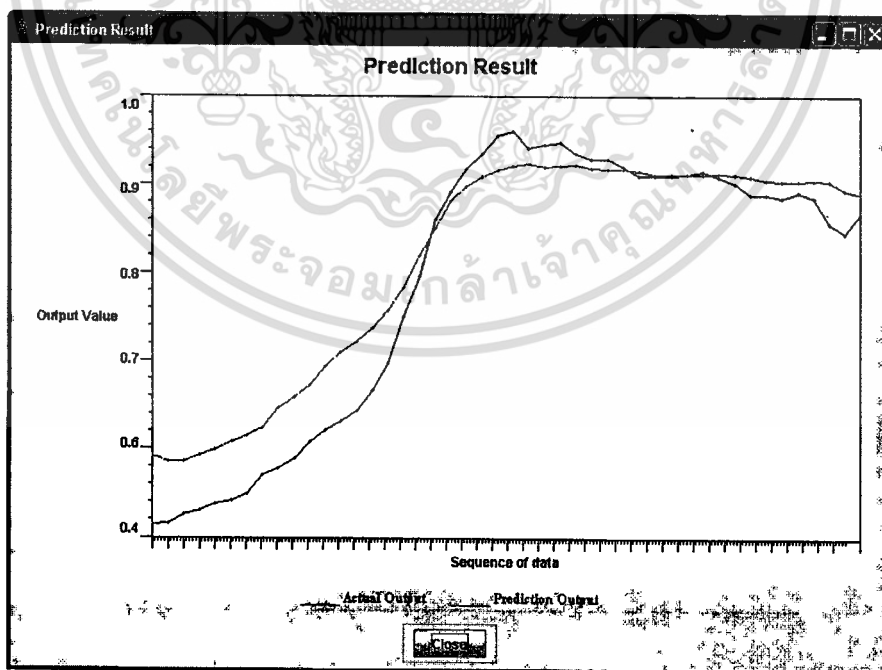
4.2.4.3 กราฟแสดงผลการทดสอบโครงข่ายประสาทเทียม

กราฟแสดงผลการใช้ข้อมูลทดสอบโครงข่ายประสาทเทียมซึ่งเราจะใช้ค่าถ่วงน้ำหนักที่ดีที่สุดที่บันทึกได้จากผลการทดลองที่ 4 มาทำการทดสอบโดยในหัวข้อนี้เราจะสามารถแบ่งกราฟออกเป็น 2 ส่วนด้วยกันได้แก่

1. กราฟที่เกิดจากการใช้ข้อมูลการฝึกสอนโครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.20
2. กราฟที่เกิดจากการใช้ข้อมูลการทดสอบโครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.21



รูปที่ 4.20 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ
ของการทดลองที่ 4



รูปที่ 4.21 แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ
ของการทดลองที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.5 การทดลองที่ 5 ข้อมูล REALGNP

4.2.5.1 ตารางสรุปผลการทดลองที่ 5

ตารางที่ 4.9 เป็นการแสดงตารางสรุปรวมผลการทดลองซึ่งใน 1 การทดลองจะทำการทดลองทั้งหมด 3 ครั้งดังปรากฏค่าของ ค่า Fitness 1, ค่า Fitness 2 และค่า Fitness 3 ดังตารางที่ 4.9 โดยจะประกอบไปด้วยทดลองปรับตั้งค่าโปรแกรมทั้งหมด 11 ชุดด้วยกัน จะเห็นว่าในการทดลองแต่ละครั้งแม้เรามีการปรับตั้งค่าตัวแปรที่เหมือนกันแต่ผลการทำนายที่ได้อาจมีค่าความถูกต้องต่างกันอย่างมากก็ได้

การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	ค่า Fitness 1	ค่า Fitness 2	ค่า Fitness 3
1	25	150	0.04	10	13.7823	50.9981	19.3233
2	25	300	0.04	10	11.11467	33.9401	54.4408
3	50	300	0.04	10	48.1370	49.3221	50.0936
4	25	300	0.08	10	48.6465	32.9705	26.6939
5	25	300	0.12	10	52.1699	50.2671	21.5009
6	25	300	0.25	10	48.6331	13.3009	17.0691
7	25	300	0.04	5	1.6154	1.6360	0.8342
8	25	300	0.04	15	45.0906	52.0724	45.3950
9	25	300	0.04	20	49.3919	48.0613	45.7793
10	25	300	0.04	25	42.8935	43.8770	42.3278
11	25	300	0.04	35	48.6241	43.6058	45.7562

ตารางที่ 4.9 แสดงผลการทดลองที่ 5 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์

4.2.5.2 ตารางสรุปผลการทดลองที่ 5 แบบเฉลี่ยค่าความเหมาะสม

ตารางที่ 4.10 เป็นการแสดงตารางสรุปรวมผลการทดลองแบบการรวมค่า Fitness ทั้ง 3 ค่า เข้าด้วยกัน และนำมาหาค่าเฉลี่ย เพื่อที่จะสามารถพิจารณาวิเคราะห์ข้อมูลได้ง่ายมากยิ่งขึ้น

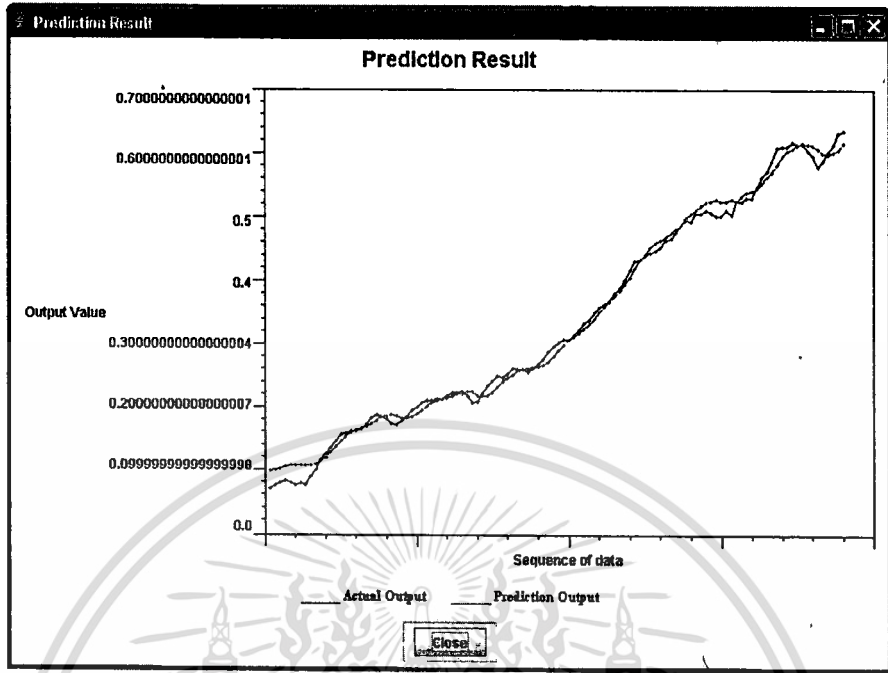
การทดลองที่	Number Particle	Number Iteration	Max. Velocity	Number Hidden Node	เวลาที่ใช้ในการฝึกสอน (วินาที)	ค่าเฉลี่ย Fitness
1	25	150	0.04	10	29	28.0345
2	25	300	0.04	10	61	33.1651
3	50	300	0.04	10	115	49.1842
4	25	300	0.08	10	60	36.1036
5	25	300	0.12	10	62	41.3126
6	25	300	0.25	10	59	26.3343
7	25	300	0.04	5	41	1.3618
8	25	300	0.04	15	78	47.5193
9	25	300	0.04	20	102	47.7441
10	25	300	0.04	25	119	43.0327
11	25	300	0.04	35	145	45.9953

ตารางที่ 4.10 แสดงผลการทดลองที่ 5 ในการทดสอบการปรับค่าที่ตั้งให้กับ โปรแกรม เพื่อใช้ในการวิเคราะห์แบบเฉลี่ยค่า

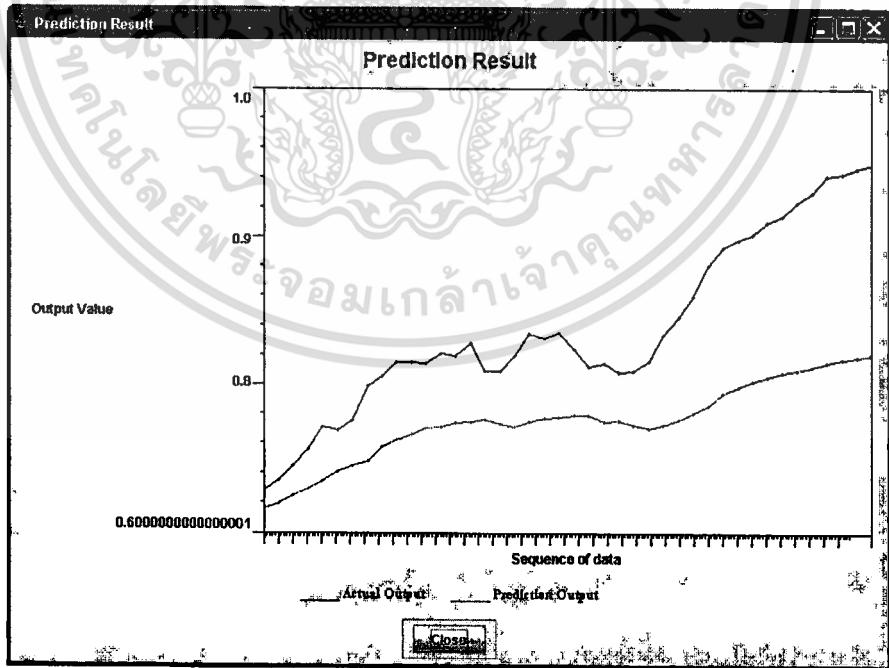
4.2.5.3 กราฟแสดงผลการทดสอบโครงข่ายประสาทเทียม

กราฟแสดงผลการใช้ข้อมูลทดสอบ โครงข่ายประสาทเทียมซึ่งเราจะใช้ค่าถ่วงน้ำหนักที่ดีที่สุดที่บันทึกได้จากผลการทดลองที่ 5 มาทำการทดสอบโดยในหัวข้อนี้เราจะสามารถแบ่งกราฟออกเป็น 2 ส่วนด้วยกันได้แก่

1. กราฟที่เกิดจากการใช้ข้อมูลการฝึกสอนโครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.22
2. กราฟที่เกิดจากการใช้ข้อมูลการทดสอบโครงข่ายประสาทเทียมเป็นข้อมูลที่ใช้ในการทดสอบโครงข่ายประสาทเทียม ดังรูปที่ 4.23



รูปที่ 4.22 แสดงการใช้ข้อมูลการฝึกสอน โครงข่ายประสาทเทียมในการทดสอบ
ของการทดลองที่ 5



รูปที่ 4.23 แสดงการใช้ข้อมูลการทดสอบ โครงข่ายประสาทเทียมในการทดสอบ
ของการทดลองที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผล และข้อเสนอแนะ

จากการศึกษาการพยากรณ์โดยใช้ทฤษฎีโครงข่ายประสาทเทียมร่วมกับหลักการของ PSO อัลกอริทึมเพื่อที่จะสร้างโครงงานพัฒนาระบบนี้ สามารถสรุปผลการดำเนินงาน และสรุปผลการทดลองรวมทั้งข้อเสนอแนะ ดังต่อไปนี้

6.1 สรุปผลการดำเนินงาน

จากการศึกษาในโครงการนี้สามารถสรุปผลการดำเนินงานในการออกแบบการประยุกต์โครงข่ายประสาทเทียมให้ใช้งานในการพยากรณ์ร่วมกับ PSO อัลกอริทึมได้ดังนี้

1. โครงข่ายประสาทเทียมเป็นลักษณะกระบวนการ และ โมเดลที่มีความสามารถในการใช้ตัวโครงข่ายประสาทเทียมเองแทนฟังก์ชันใดก็ได้ที่เราต้องการแต่เราไม่สามารถจะมองเห็นว่าลักษณะของฟังก์ชันเป็นอย่างไร
2. PSO อัลกอริทึมเป็นอัลกอริทึมสำหรับค้นหาค่าที่เหมาะสมที่สุด โดยจะทำการเลียนแบบพฤติกรรมทางธรรมชาติ ซึ่งมีแนวคิดคล้ายๆกับ จีเนติกอัลกอริทึม
3. ภาษา JAVA เป็นภาษาที่ใช้หลักการเชิงออฟเจ็คมีข้อดีคือมันเป็นจีนสามารถนำไป Reuse ได้ง่าย
4. ข้อมูลดิบที่เราจะนำมาใช้ทำเป็นอินพุท ในกรณีที่จะใช้งานโครงข่ายประสาทเทียมจะต้องทำการนอร์มอลไลซ์ค่าก่อนเสมอ
5. ช่วงขั้นตอนไหนของการพัฒนาระบบ ถ้าหากเรายังไม่เคยมีประสบการณ์มาก่อนเราจะต้องให้ความสำคัญกับขั้นตอนนั้นมากๆ เนื่องจากอาจทำให้งานเสร็จไม่ทันตามเวลาก็ได้

6.2 สรุปผลการทดลอง

จากการศึกษาในโครงการพัฒนาระบบงานนี้ สามารถสรุปผลการทดลองในการออกแบบการประยุกต์โครงข่ายประสาทเทียมให้ใช้งานในการพยากรณ์ร่วมกับ PSO อัลกอริทึมได้ดังนี้

1. ข้อมูลที่นำมาใช้สำหรับการฝึกสอนโครงข่ายมีความสำคัญมากหากยิ่งเราใช้ข้อมูลปริมาณยิ่งมากเท่าไรประสิทธิภาพของผลที่ได้จากการทำนายก็จะเพิ่มมากขึ้นตามไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ค่าปริมาณจำนวนของพาคัลที่ใช้ในการทดลองส่งผลบ้าง ต่อค่าความเหมาะสมที่ได้แต่จะไม่ส่งผลมากนักต่อการค้นหาค่าที่เหมาะสมที่สุด
3. ค่าจำนวนรอบในการปรับตำแหน่งของพาคัลยิ่งมากก่อนข้างจะส่งผลดีโดยตรง ต่อการค้นหาค่าที่เหมาะสมที่สุด
4. ค่าความเร็วมากที่สุดของพาคัลเป็นค่าที่ค่อนข้างมีความสำคัญโดยตรงต่อการหาค่า Fitness โดยถ้าเรากำหนดค่านี้มากไปจะส่งผลให้เราได้ผลลัพธ์ที่ไม่ดี แต่ถ้าเรากำหนดค่านี้น้อยก็ จะส่งผลให้ค่าอาจไม่ดีอีกเช่นกัน ดังนั้นเราจึงควรกำหนดให้พอดีกับ Problem Space เรา
5. เวลาที่ใช้ในการฝึกหัด โครงข่ายจะแปรผัน โดยตรงกับค่า จำนวนพาคัลที่เราสร้างขึ้น และค่าจำนวนรอบที่เราทำการฝึกหัด
6. ในการฝึกหัด โครงข่ายในแต่ละครั้ง ถึงแม้เราจะกำหนดตัวแปรนำเข้าทุกอย่างให้เหมือนกันหมด แต่ประสิทธิภาพในการทำงานของแต่ละครั้งจะได้ค่าไม่เท่ากัน ทั้งนี้ผลน่าจะมาจากการสุ่มค่าของ โปรแกรม
7. ค่าจำนวน โหนดของชั้นซ่อนนั้น จะส่งผลต่อการค้นหาค่าถ่วงน้ำหนักที่ถูกต้องที่สุดของ โปรแกรม โดยยิ่งจำนวน โหนดมากขึ้นเท่าไร ก็จะส่งผลให้ผลลัพธ์ที่ได้มีโอกาสที่จะได้ค่าถ่วงน้ำหนักที่มีค่า ความคลาดเคลื่อนน้อยมากเท่านั้น (ทั้งนี้จะต้องมีการปรับ ค่าจำนวนรอบของการทำงาน ให้มีค่ามากถึงระดับหนึ่งด้วย)
8. ค่าที่เหมาะสมสำหรับการปรับตั้งค่าให้กับ โปรแกรมทั้งส่วนของ PSO และรวมทั้งส่วนของ โครงข่ายประสาทเทียมนั้นจะมีประสิทธิภาพมากแค่ไหนจะขึ้นอยู่กับลักษณะของฟังก์ชันที่จะให้โครงข่ายประสาทเทียมเรียนรู้ด้วย

6.3 ข้อเสนอแนะ

จากการที่ได้ทดลองใช้งาน โปรแกรม และศึกษาการทำงานของระบบการทำงานอนุกรมเวลาโดยใช้โครงข่ายประสาทเทียมร่วมกับ PSO พบว่า

1. เราน่าจะสามารถปรับปรุง PSO อัลกอริทึมให้มีความสามารถมากขึ้นกว่านี้ได้โดยการ
 - 1.1 ให้มันสามารถลดและเพิ่มความเร็วของการเคลื่อนที่ได้
 - 1.2 ให้มีการกระจายของพาคัลให้ทั่ว โดยเราไม่ต้องสุ่มค่าให้กับพาคัล
 - 1.3 กำหนดให้พาคัลบินในทิศทางตามวิธีการคิดอื่นเพื่อค้นหาค่าที่เหมาะสม
2. การปรับปรุงค่าต่างๆของ โปรแกรม เพื่อสามารถที่จะหาค่าที่เหมาะสมที่ควรป้อนให้กับ โปรแกรมได้เพื่อที่จะให้โปรแกรมทำงานได้อย่างมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เราน่าจะสร้างโปรแกรมสำหรับทำนายไว้อีกชั้นหนึ่ง เพื่อทำนายหาจุดเวลาที่将会เกิดความผิดพลาดของฟังก์ชัน หรือการกระโดดของข้อมูล เพื่อที่จะใช้ทำนายสถานการณ์บางอย่างได้อย่างแม่นยำ
4. เราอาจพัฒนาอัลกอริทึม โดยทำการใช้พหาคัลหลายๆกลุ่มในการค้นหาค่าที่เหมาะสมครั้งเดียวกัน แต่ต้องมีข้อกำหนดคือพหาคัลแต่ละกลุ่มจะต้องไม่ไปอยู่บน Local Maximum เดียวกัน หรืออาจทำโดยการจำบริเวณแต่ละ Local Maximum เอาไว้แล้วห้ามให้พหาคัลชุดเดิมจะไม่ย้อนมาบริเวณ Local Maximum เดิมอีก
5. กำหนดเงื่อนไขเพิ่มในการค้นหา โดยให้พหาคัลตัวใหม่ที่พบตำแหน่งที่ดีที่สุดนั้นให้หยุดรอเพื่อนก่อนเพื่อความยุติธรรมก่อนเสมอเป็นจำนวนประมาณ 5 รอบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- กิตติ ภัคดีวัฒนกุล. 2546. **คัมภีร์ Java เล่ม 1,2.** กรุงเทพฯ : เททีพี คอมพ์ แอนด์คอนซัลท์ จำกัด
- วีระศักดิ์ ชิงถาวร. 2545. **Java Programming Volumn 1,2.** กรุงเทพฯ : ซีเอ็ดยูเคชั่น
- วิจิต หล่อจิระขุณหกุล. 2539. **เทคนิคการพยากรณ์.** กรุงเทพฯ : โรงพิมพ์จุฬาลงกรณ์มหาวิทยาลัย
- Andries P. Engelbrecht. 2002. **Computational Intelligence An Introuction.** England : Wiley
- Chunkai Zhang, Huihe Shao, Yu Li. "Particle swarm optimisation for evolving artificial neural network." 2487 – 2490. **Systems, Man, and Cybernetics, 2000 IEEE International Conference on Volume 4.**
- C. K. Mohan, or E. Ozcan. 2002. **Particle Swarm Optimization.** [Online]. Available: <http://www.cis.syr.edu/~mohan/pso/>
- David, M. Skapura. 1996. **Building Neural Networks.** New York : Addison-Wesley
- DEITEL. 2005. **Java How To Program.** New Jersey : Pearson Education.
- Eberhart R., Kennedy J.. "A new optimizer using particle swarm theory." 39 – 43. **Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium.**
- Gudise V.G., Venayagamoorthy G.K.. "Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks." 110 – 117. **Swarm Intelligence Symposium, 2003. SIS '03. Proceedings of the 2003 IEEE.**
- Linmin Fu. 1994. **Neural Network in Computer Intelligence.** Singapore : McGraw-Hill
- Mark C Sinclair. 2001. **Particle Swarm Optimization.** [Online]. Available: <http://uk.geocities.com/markcsinclair/pso.html>
- Xiaohui Hu. 2003. **Particle Swarm Optimization.** [Online]. Available: <http://www.swarmintelligence.org/index.php>
- Zhang Chunkai, Li Yu, Shao Huihe. "A new evolved artificial neural network and its application." 1065 – 1068. **Intelligent Control and Automation, 2000. Proceedings of the 3rd World Congress on Volume 2.**

ประวัติผู้เขียนโครงการงาน

ชื่อผู้จัดทำโครงการงาน

พิชักษณ์ สิริมาจันทร์

วันเดือนปีเกิด

23 พฤศจิกายน 2522

สถานที่เกิด

จ.ชลบุรี

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนพนัสศึกษาลัย

มัธยมศึกษา

โรงเรียนพนัสพิทยาคาร

อุดมศึกษา

วิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมโยธา

มหาวิทยาลัยเกษตรศาสตร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้