

บทที่ 2

โครงข่ายประสาทเทียม

โครงข่ายประสาทเทียม (Artificial Neural Networks:ANNs) มีการพัฒนามาเป็นเวลากว่า 50 ปีมาแล้ว นับเป็นสาขาหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence , AI) ซึ่งถูกพัฒนาขึ้นมาในยุคคอมพิวเตอร์ ANNs คือระบบการคำนวณที่สร้างเลียนแบบการทำงานของระบบสมองมนุษย์ เพื่อใช้ประโยชน์ในการคาดคะเนเหตุการณ์จากข้อมูลที่มีอยู่ เช่น การพยากรณ์อากาศ การพยากรณ์หุ้นในตลาดหลักทรัพย์ และเริ่มมีการประยุกต์ใช้ ANNs มาใช้ในการพยากรณ์ความต้องการไฟฟ้าทั้งในระยะสั้น และในระยะยาว ในพื้นที่ของประเทศต่างๆ [5-15] นอกจากนี้ได้มีการพัฒนา ANNs ไปใช้ประโยชน์ในสาขาอื่นๆอีกมากมาย เช่น Image Processing และ Pattern Recognition ANNs มีชื่อเรียกอื่นอีกเช่น Machine Learning Algorithms , Neural Intelligent System

โครงข่ายประสาทเทียมเป็นที่สนใจของนักวิจัยในหลายสาขา ในส่วนของวิศวกรไฟฟ้าได้มีการประยุกต์ใช้ในเรื่องของระบบควบคุมและการประมวลผลสัญญาณ วิศวกรคอมพิวเตอร์ได้มีการพัฒนาระบบฮาร์ดแวร์เพื่อที่จะใช้โครงข่ายประสาทเทียมได้อย่างมีประสิทธิภาพและได้มีการใช้กับหุ่นยนต์อีกด้วย [16]

2.1 ประวัติและความเป็นมาของการพัฒนาโครงข่ายประสาทเทียม

การพัฒนาโครงข่ายประสาทเทียมเกิดขึ้นเมื่อมีการนำเสนอแบบจำลองทางตรรกะของโครงข่ายประสาทเทียมครั้งแรกในปี ค.ศ.1943 โดย Warren McCulloch นักประสาทวิทยา และ Walter Pitts นักคณิตศาสตร์ซึ่งมีอายุเพียง 18 ปีในขณะนั้น โดยมีแนวคิดสำคัญคือการคำนวณค่าสัญญาณเอาต์พุตโดยพิจารณาจากระดับกระตุ้นกล่าวคือ นิวรอนจะให้สัญญาณเอาต์พุตก็ต่อเมื่อผลรวมของสัญญาณอินพุตมีค่ามากกว่าระดับกระตุ้นซึ่งแนวคิดนี้ยังคงเป็นส่วนสำคัญในการพัฒนาโครงข่ายประสาทเทียมมาจนทุกวันนี้

ในปี 1949 Donald Hebb นักจิตวิทยาของมหาวิทยาลัย McGill ได้ออกแบบกฎการเรียนรู้ชนิดแรกสำหรับโครงข่ายประสาทเทียม กฎการเรียนรู้นี้ก็คือ ถ้าสองนิวรอนถูกกระตุ้นพร้อมกันแล้ว ความแข็งแรงของการเชื่อมโยงระหว่างพวกมันควรจะเพิ่มขึ้น

ในช่วงปี ค.ศ.1950 ถึง ค.ศ.1960 ถือว่าเป็นช่วงที่มีการพัฒนาโครงข่ายประสาทเทียมกันอย่างกว้างขวาง เมื่อ John van Neumann บิดาแห่งวงการคอมพิวเตอร์ยุคใหม่ได้นำเอาแบบจำลองของสมองที่เสนอโดยนักวิจัยที่ทำงานเกี่ยวกับโครงข่ายประสาทเทียมไม่ว่าจะเป็น Warren McCulloch และคนอื่นๆ มาเป็นแนวทางในการพัฒนาเครื่องคอมพิวเตอร์ในยุคต่อมา จนในปี ค.ศ. 1956 ก็ได้มีการสร้างแบบจำลองโดยใช้คอมพิวเตอร์ขึ้นได้สำเร็จ

ในปี ค.ศ.1957 Frank Rosenblatt ได้เริ่มศึกษาและพัฒนาประเภทใหญ่ของโครงข่ายประสาทเทียมที่เรียกว่า Perceptron ซึ่งเป็นรูปแบบที่สำคัญรูปแบบหนึ่งของโครงข่ายประสาทเทียม โดยส่วนมากสถาปัตยกรรมของโครงข่ายประสาทเทียมชนิดนี้จะประกอบด้วยชั้นอินพุตเชื่อมโยงผ่านด้วยค่าถ่วงน้ำหนัก เพื่อเป็นตัวเชื่อมโยงกับนิวรอนที่ชั้นเอาต์พุต ค่าถ่วงน้ำหนักบนทางผ่านเชื่อมโยงถูกปรับเปลี่ยน กฎการเรียนรู้ Perceptron ปรับเปลี่ยนค่าถ่วงน้ำหนักซ้ำ ๆ กันซึ่งมีประสิทธิภาพดีกว่ากฎของ Hebb การเรียนรู้ของ Perceptron สามารถแสดงให้เห็นว่าผู้เข้าสู่ค่าถ่วงน้ำหนักที่ถูกต้อง ถ้าค่าถ่วงน้ำหนักเหล่านั้นจะแก้ปัญหาที่เตรียมพร้อม

ในปี ค.ศ.1959 Bernard Widrow และนักศึกษาของเขา Marcian Hoff ได้พัฒนาการเรียนรู้ซึ่งใกล้เคียงกับกฎการเรียนรู้ของ Perceptron โดยกฎการเรียนรู้ของ Perceptron ปรับค่าถ่วงน้ำหนักที่เชื่อมโยงที่โหนดเมื่อผลตอบสนองของโหนดไม่ถูกต้อง กฎ Delta ปรับค่าถ่วงน้ำหนักเพื่อลดความแตกต่างระหว่างโครงข่ายอินพุตถึงโหนดเอาต์พุตและค่าเอาต์พุตที่ต้องการ สิ่งเหล่านี้จะสรุปในรูปของค่าผิดพลาดยกกำลังสองเฉลี่ยที่น้อยที่สุด (Smallest Mean Square Error : SMSE) และนำสิ่งต่าง ๆ เหล่านี้ไปใช้สร้างแบบจำลองของ MADALINE (Multiple ADaptive LINear Element) ซึ่งพัฒนามาจาก ADALINE และเป็นการนำเอาโครงข่ายประสาทเทียมมาใช้งานในทางปฏิบัติเป็นครั้งแรก โดยนำมาลดเสียงก้องในสายโทรศัพท์

ในปี ค.ศ. 1969 Marvin Minsky และ Seymour Papert ได้ตีพิมพ์หนังสือ Perceptron ซึ่งกล่าวถึงข้อจำกัดของ Perceptron ว่าไม่สามารถนำไปใช้ในการแก้ปัญหาหลาย ๆ ปัญหาที่มีความน่าสนใจได้ ซึ่งจุดนี้เองทำให้นักวิจัยหลายคนกลับไปให้ความสนใจในด้านปัญญาประดิษฐ์ (AI) มากขึ้น แต่ยังมีนักวิจัยบางกลุ่มที่ไม่ย่อท้อและยังคงทำงานวิจัยด้านโครงข่ายประสาทเทียมต่อไป

ในปี ค.ศ.1968 John Anderson ของมหาวิทยาลัย Brown เริ่มวิจัยงานของเขาในโครงข่ายประสาทเทียมด้วยโครงข่ายรวมกลุ่มความสามารถในการจดจำ (Associative Memory Nets) เขาได้พัฒนาแบบจำลองของโครงข่ายประสาทเทียมขึ้นมา และให้ชื่อว่า Brain-State-in-a-Box (BSB) งานประยุกต์ที่ใช้ในโครงข่ายประสาทเทียมชนิดนี้คือการวินิจฉัยทางการแพทย์ (Medical Diagnosis)

ในปี ค.ศ.1972 Teuvo Kohonen วิศวกรไฟฟ้าชาวฟินแลนด์ของมหาวิทยาลัยเทคโนโลยี Helsinki ได้พัฒนาการเรียนรู้ของโครงข่ายประสาทเทียมด้วยการแก้ไขค่าถ่วงน้ำหนักแบบเมตริกซ์ โดยอาศัยพื้นฐานจากทฤษฎีของ Hebb และในปีเดียวกันเขาก็ได้เสนอทฤษฎีการเรียนรู้ของโครงข่ายประสาทเทียมแบบ Self - Organizing map โดยการจัดรวมนิวรอนเป็นกลุ่มเพื่อใช้ในการวิเคราะห์เสียงพูด

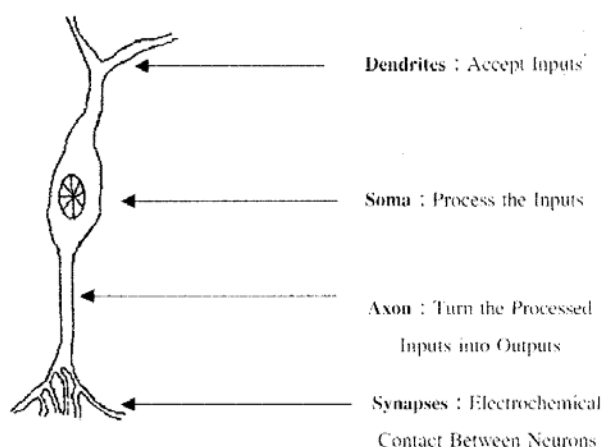
ในช่วงปี ค.ศ.1985 ถึง ค.ศ.1990 Gail Carpenter ได้ร่วมวิจัยกับ Stephen Grossberg ได้เสนอทฤษฎี Adaptive Resonance สำหรับวิเคราะห์ข้อมูลแบบไบนารี (ART1) และข้อมูลอินพุตแบบต่อเนื่อง (ART2) และในปี ค.ศ. 1985 David Parker ได้เสนอทฤษฎีการเรียนรู้แบบแพร่ค่า

ย้อนกลับ (Back-propagation) ทฤษฎีการเรียนรู้นี้ใช้การนำค่าผิดพลาดของสัญญาณเอาต์พุตที่คำนวณได้กับสัญญาณเอาต์พุตตัวอย่างกลับมาเป็นค่าแก้ไขของค่าถ่วงน้ำหนัก

ในปัจจุบันโครงข่ายประสาทเทียมได้ถูกพัฒนา ค้นคว้า วิจัยและนำไปใช้ในด้านต่าง ๆ มากขึ้น อาทิ วิทยาศาสตร์ด้านระบบประสาท, การเงินการธนาคาร, ด้านคณิตศาสตร์, การแพทย์, ระบบสื่อสาร รวมไปถึงงานทางด้านวิศวกรรมไฟฟ้าอีกด้วย จึงถือได้ว่าโครงข่ายประสาทเทียมเป็นศาสตร์แขนงหนึ่งที่มีความสนใจเป็นอย่างมากในช่วงเวลานี้

2.2 นิวรอน (Neurons) ในสมองมนุษย์ [17]

องค์ประกอบพื้นฐานของสมองมนุษย์คือ นิวรอน (Neurons) ซึ่งก็คือกลุ่มของเซลล์ประสาทที่ทำหน้าที่ในการจดจำ คิด นำความรู้ และประสบการณ์ที่บันทึกไว้ในสมองมาใช้ในการตัดสินใจในชีวิตประจำวัน แต่ละนิวรอนอาจเชื่อมต่อกับนิวรอนอื่น ๆ มากถึง 200,000 นิวรอนพลังสมองของมนุษย์เกิดจากนิวรอนจำนวนมากและระบบการเชื่อมต่อแบบซับซ้อนของนิวรอนจำนวนมากเหล่านี้ นิวรอนของมนุษย์ตามหลักวิชาประสาทวิทยา (Neuroscience) จะแบ่งออกเป็น 4 ส่วน ดังรูปที่ 2.1 แต่ละส่วนมีฟังก์ชันการทำงานดังนี้



รูปที่ 2.1 โครงสร้างของ Neurons ในสมองมนุษย์

Dendrites คือ ส่วนที่ทำหน้าที่รับข้อมูล (Accept Inputs)

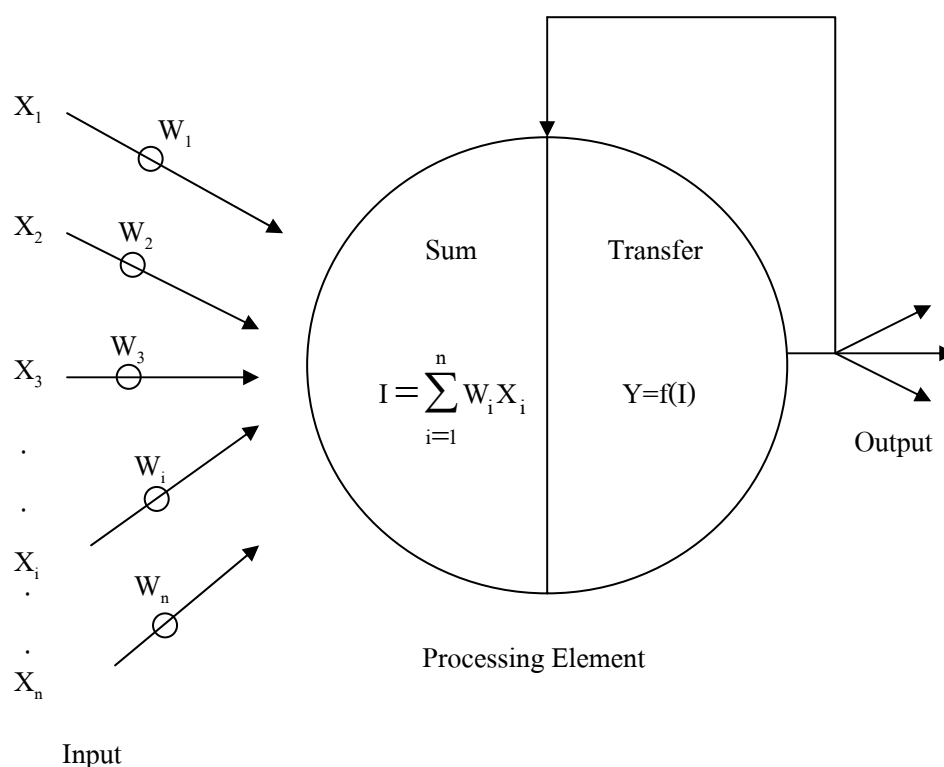
Soma คือ ส่วนที่ประมวลผลข้อมูลเบื้องต้น (Process Inputs)

Axon คือ ส่วนที่แปลงข้อมูลที่ได้ประมวลผลเบื้องต้นเป็นผลลัพธ์ที่ต้องการ
(Turn the Processed Inputs into Outputs)

Synapses คือ เส้นประสาทที่ทำหน้าที่เชื่อมต่อเพื่อการสื่อสารกับ Neurons อื่นในระบบ
สมอง (Electrochemical contact between neurons)

ในระบบประสาทของมนุษย์ Dendrites ทำหน้าที่รับข้อมูลจากแหล่งต่าง ๆ แล้วส่งให้ Soma ประมวลผลข้อมูลเบื้องต้น หลังจากนั้น Axon จะแปลงข้อมูลต่าง ๆ เป็นผลลัพธ์แล้ว Synapses จะส่งผลลัพธ์ให้นิวรอนอื่นเพื่อช่วยกันสร้างผลลัพธ์ขั้นสุดท้าย ความจริงแล้วการทำงานของระบบประสาทในสมองมนุษย์มีความซับซ้อนเกินกว่าที่มนุษย์จะเข้าใจได้ ซึ่งที่กล่าวมาข้างต้นนั้นเป็นเพียงแนวความคิดเกี่ยวกับการทำงานของนิวรอนอย่างง่าย ๆ ที่จะนำมาใช้ในการพัฒนา Artificial Neural Networks

จากที่กล่าวมานี้เป็นระบบการทำงานของนิวรอนในสมองของมนุษย์ ซึ่งนิวรอนในโครงข่ายประสาทเทียมก็จะจำลองการทำงานของนิวรอนในสมองมนุษย์มาเช่นเดียวกัน ซึ่งนิวรอนในโครงข่ายประสาทเทียมนั้นจะมีโครงสร้างการทำงานที่ง่ายกว่านิวรอนในสมองมนุษย์อยู่มาก โดยในรูปที่ 2.2 จะแสดงโครงสร้างการทำงานของนิวรอนในโครงข่ายประสาทเทียม



รูปที่ 2.2 โครงสร้างการทำงานของนิวรอนในโครงข่ายประสาทเทียม

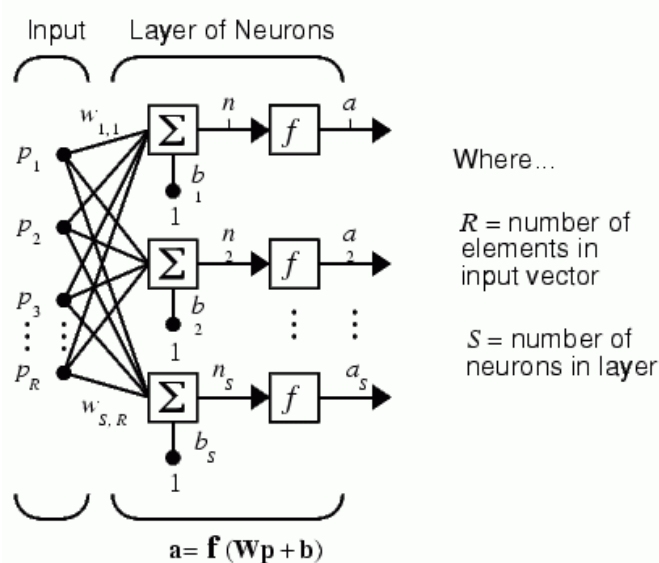
จากรูปที่ 2.2 ถ้ากำหนดให้ข้อมูลที่ป้อนเข้าสู่โครงข่ายประสาทเทียม คือ $X_1, X_2, \dots, X_n$ ข้อมูลแต่ละตัวจะถูกคูณด้วยค่าถ่วงน้ำหนัก (Weight) $W_1, W_2, \dots, W_n$ ตามลำดับ จากนั้นข้อมูลแต่ละตัวก็จะถูกรวมเข้าด้วยกันซึ่งกระบวนการนี้เรียกว่า Summation ก่อนที่จะแปลงข้อมูล (Transfer) เป็นผลลัพธ์แล้วส่งให้นิวรอนตัวอื่นนำไปประมวลผลในลักษณะเดียวกันเพื่อสร้างผลลัพธ์ขั้นสุดท้าย

2.3 โครงสร้างของโครงข่ายประสาทเทียม (Architectures of Neural Networks) [18]

โครงข่ายประสาทเทียมนั้นสามารถแบ่งตามลักษณะโครงสร้างได้ 2 แบบ คือ

2.3.1 โครงข่ายประสาทเทียมแบบชั้นเดียว (Single Layer Neural Networks)

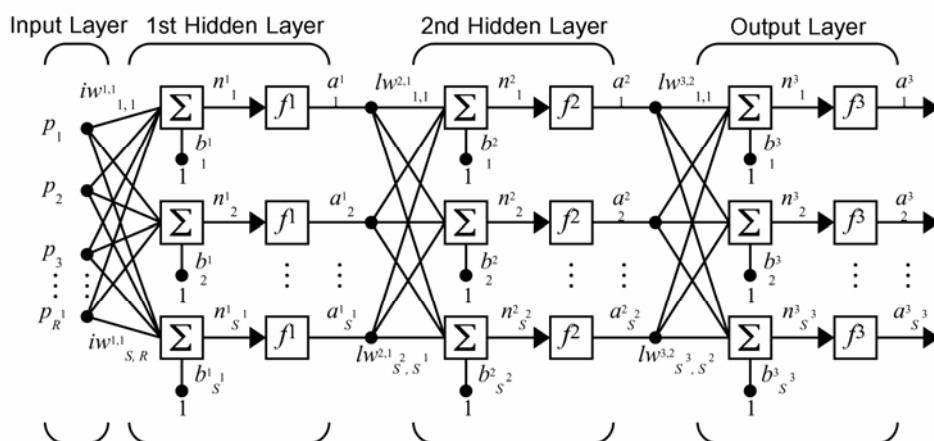
โครงข่ายประสาทเทียมแบบชั้นเดียวจะประกอบไปด้วยชั้นอินพุต (Input Layer) และชั้นเอาต์พุต (Output Layer) ในชั้นอินพุตนั้นเราจะไม่นับเนื่องจากเป็นชั้นที่ไม่มีการประมวลผลใด ๆ โครงข่ายประสาทเทียมแบบชั้นเดี่ยวนี้อาจมีความสามารถในการคำนวณที่ค่อนข้างน้อย แต่เวลาที่ใช้ในการคำนวณนั้นจะเร็ว รูปโครงข่ายประสาทเทียมแบบชั้นเดียวแสดงดังรูปที่ 2.3



รูปที่ 2.3 โครงข่ายประสาทเทียมแบบชั้นเดียว

2.3.2 โครงข่ายประสาทเทียมแบบหลายชั้น (Multi Layer Neural Networks)

โครงข่ายประสาทเทียมแบบหลายชั้นนี้จะมีชั้นที่เพิ่มขึ้นมาจากโครงข่ายประสาทเทียมแบบชั้นเดียวที่มีแค่ชั้นอินพุตชั้นเอาต์พุต นั่นคือจะมีชั้นซ่อน (Hidden Layer) เพิ่มขึ้นมา โดยในชั้นซ่อนนี้สามารถมีได้ตั้งแต่ 1 ชั้นขึ้นไป ขึ้นอยู่กับความซับซ้อนของปัญหาที่นำมาให้โครงข่ายประสาทเทียมทำการวิเคราะห์ แต่โดยทั่วไปแล้วก็มักจะมีชั้นซ่อนไม่เกิน 2 ชั้น เนื่องจากเวลาที่ใช้ในการคำนวณ เพราะถ้าจำนวนชั้นซ่อนมาก เวลาที่ใช้ในการคำนวณก็จะมากตามไปด้วย



รูปที่ 2.4 โครงข่ายประสาทเทียมแบบหลายชั้น

โครงข่ายประสาทเทียมแบบหลายชั้นนี้จะมีสามารถในการคำนวณที่มากกว่าโครงข่ายประสาทเทียมแบบชั้นเดียว แต่ว่าเวลาในการคำนวณนั้นก็จะใช้เวลานานกว่าโครงข่ายประสาทเทียมแบบชั้นเดียว

2.4 กระบวนการเรียนรู้ (Learning) ของโครงข่ายประสาทเทียม

การเรียนรู้ของโครงข่ายประสาทเทียมจะมีประสิทธิภาพมากขึ้นขึ้นอยู่กับค่าถ่วงน้ำหนักของโครงข่ายประสาทเทียม ซึ่งการฝึกสอน (Training) โครงข่ายประสาทเทียม ก็คือการหาค่าถ่วงน้ำหนักที่เหมาะสมให้แก่โครงข่ายประสาทเทียม เพื่อให้ผลลัพธ์สุดท้ายที่ได้จากโครงข่ายประสาทเทียมมีค่าผิดพลาด (Error) น้อยที่สุด กระบวนการเรียนรู้ของโครงข่ายประสาทเทียมแบ่งออกได้เป็น 2 วิธี คือ

2.4.1 การเรียนรู้แบบ Supervised (Supervised Learning)

การเรียนรู้แบบ Supervised นั้นจะมีข้อมูลที่เราเรียกว่า ชุดฝึกสอน (Training Pattern) โดยในชุดฝึกสอนนี้จะมีข้อมูลอินพุตและเอาต์พุต ซึ่งโครงข่ายประสาทเทียมจะถูกฝึกสอนจากข้อมูลในชุดฝึกสอนเพื่อหาค่าถ่วงน้ำหนักที่เหมาะสม โดยนำเอาต์พุตที่ได้จากโครงข่ายประสาทเทียมไปเปรียบเทียบกับเอาต์พุตของชุดฝึกสอน จากนั้นค่าผิดพลาดที่ได้จากความแตกต่างของเอาต์พุตจากโครงข่ายประสาทเทียมและเอาต์พุตจากชุดฝึกสอนก็จะถูกป้อนกลับไปใหม่ เพื่อให้คำนวณหาค่าถ่วงน้ำหนักใหม่ เพื่อให้ค่าความผิดพลาดระหว่างเอาต์พุตของโครงข่ายประสาทเทียมกับเอาต์พุตของชุดฝึกสอนมีค่าน้อยที่สุด

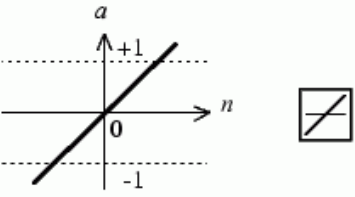
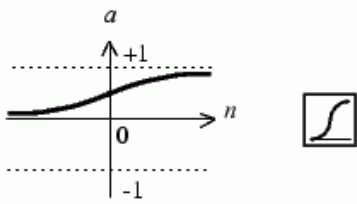
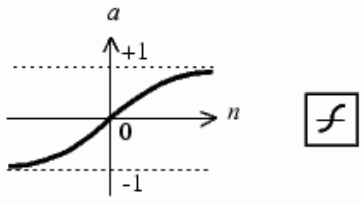
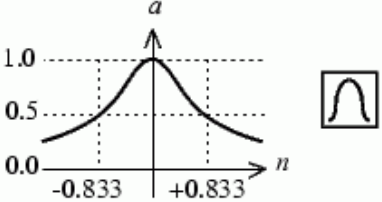
2.4.2 การเรียนรู้แบบ Unsupervised (Unsupervised Learning)

การเรียนรู้แบบ Unsupervised นี้จะเป็นการเรียนรู้ที่ไม่ต้องอาศัยชุดฝึกสอนมาทำการฝึกสอนให้กับโครงข่ายประสาทเทียม การปรับค่าถ่วงน้ำหนักนั้นนิวรอนจะทำการวิเคราะห์ด้วยตัวของมันเอง นั่นคือการปรับค่าถ่วงน้ำหนักจะไม่มีกำหนดค่าที่ถูกต้องว่าควรจะใช้ค่าถ่วงน้ำหนักเท่าไร แต่นิวรอนจะอาศัยการป้อนกลับไปของค่าผิดพลาดและวิเคราะห์หาค่าถ่วงน้ำหนักด้วยตัวของมันเอง

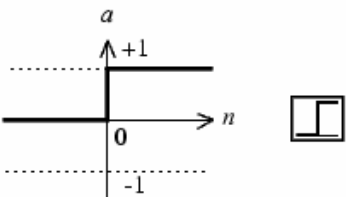
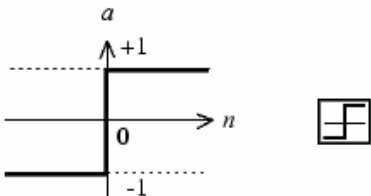
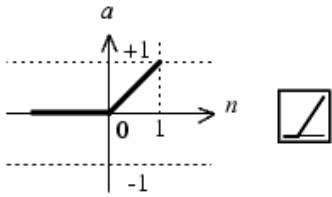
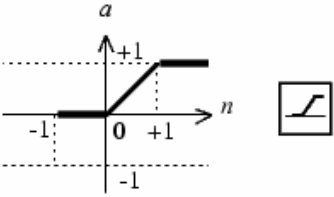
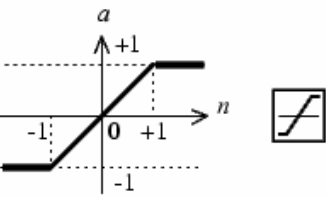
2.5 ฟังก์ชันการถ่ายโอน (Transfer Function)

จากรูปที่ 2.3 จะเห็นได้ว่าความสัมพันธ์ระหว่างอินพุตและเอาต์พุตของนิวรอนจะขึ้นอยู่กับฟังก์ชันการถ่ายโอน (Transfer Function) หรือบางครั้งเรียกว่าฟังก์ชันการกระตุ้น (Activation Function) โดยผลบวกของอินพุตกับค่าถ่วงน้ำหนักจะกลายเป็นอินพุตของฟังก์ชันการถ่ายโอน ซึ่งจะตัวบ่งชี้ค่าเอาต์พุตที่ออกจากนิวรอนแต่ละตัว ฟังก์ชันการถ่ายโอนจะทำหน้าเป็นตัวจำกัดขอบเขตของเอาต์พุต ถ้าหากไม่มีฟังก์ชันการถ่ายโอนแล้วก็จะเป็นการยากในการแสดงค่าเอาต์พุตออกมาจากนิวรอน โดยตารางที่ 2.1 จะแสดงถึงฟังก์ชันการถ่ายโอนที่เป็นที่นิยมใช้ในงานโครงข่ายประสาทเทียม

ตารางที่ 2.1 ฟังก์ชันการถ่ายโอนในโครงข่ายประสาทเทียม

ชนิดฟังก์ชันการถ่ายโอน	สมการฟังก์ชันการถ่ายโอน
 $a = \text{purelin}(n)$ Linear Transfer Function	$a(n) = n$
 $a = \text{logsig}(n)$ Log-Sigmoid Transfer Function	$a(n) = \frac{1}{1 + e^{-n}}$
 $a = \text{tansig}(n)$ Tan-Sigmoid Transfer Function	$a(n) = \frac{1 - e^{(-n)}}{1 + e^{(-n)}}$
 $a = \text{radbas}(n)$ Radial Basis Function	$a(n) = e^{-n^2}$
 $a = \text{softmax}(n)$ Compet Transfer Function	$a(n_i) = \begin{cases} 1; & i = i^* \\ 0; & i \neq i^* \end{cases}$

ตารางที่ 2.1 (ต่อ)

ชนิดฟังก์ชันการถ่ายโอน	สมการฟังก์ชันการถ่ายโอน
 $a = \text{hardlim}(n)$ Hard-Limit Transfer Function	$a(n) = \begin{cases} 1; & n \geq 0 \\ 0; & n < 0 \end{cases}$
 $a = \text{hardlims}(n)$ Symmetric Hard-Limit Trans. Funct.	$a(n) = \begin{cases} 1; & n \geq 0 \\ -1; & n < 0 \end{cases}$
 $a = \text{poslin}(n)$ Positive Linear Transfer Funct.	$a(n) = \begin{cases} n; & n \geq 0 \\ 0; & n < 0 \end{cases}$
 $a = \text{satlin}(n)$ Satlin Transfer Function	$a(n) = \begin{cases} 0; & n \leq 0 \\ n; & 0 \leq n \leq 1 \\ 1; & 1 \leq n \end{cases}$
 $a = \text{satlins}(n)$ Satlins Transfer Function	$a(n) = \begin{cases} -1; & n \leq -1 \\ n; & -1 \leq n \leq 1 \\ 1; & 1 \leq n \end{cases}$

2.6 การเลือกชนิดของโครงข่ายประสาทเทียม [19]

เนื่องจากปัจจุบันมีโครงข่ายประสาทเทียมหลายแบบมาก และปัจจุบันก็ยังไม่มียุทธวิธีในการเลือกชนิดของโครงข่ายประสาทเทียม ดังนั้นการเลือกใช้งานโครงข่ายประสาทเทียมให้เหมาะสมกับปัญหาก็เป็นไปได้ค่อนข้างยาก ตารางที่ 2.2 จะเป็นแนวทางในการใช้งานโครงข่ายประสาทเทียมสำหรับงานในแต่ละประเภท

ตารางที่ 2.2 แนวทางการเลือกโครงข่ายแต่ละประเภท

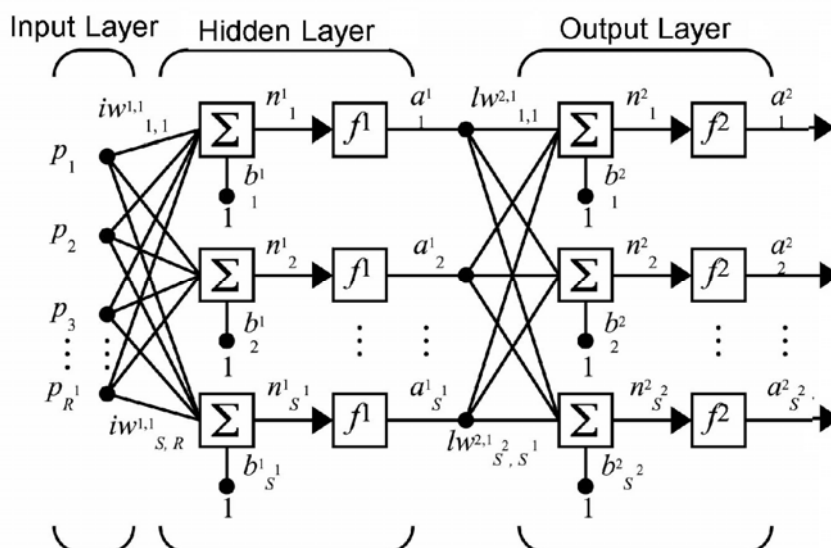
ประเภทของงาน	โครงข่าย
Prediction	- Feedforward Back-propagation - Delta Bar Delta - Extended Delta Bar Delta - Directed Random Search - Higher-order Neural Networks - Self Organizing Map into Back-propagation
Classification	- Learning Vector Quantization - Counter-propagation Network - Probabilistic Neural Network
Data association	- Hopfield Network - Boltzmann Machine - Hamming Network - Bidirectional Associative Memory - Spatio-Temporal Pattern Recognition
Data conceptualization	- Adaptive Resonance Network - Self Organizing Map
Data filtering	- Recirculation

ตารางที่ 2.2 จะเป็นเพียงแค่แนวแนะนำทางในการเลือกชนิดของโครงข่ายประสาทเทียมสำหรับงานประเภทต่างๆ เท่านั้น ซึ่งในการปฏิบัติงานจริงอาจจะไม่เป็นไปตามตารางดังกล่าวก็ได้ อย่างเช่นโครงข่ายประสาทเทียมแบบ Feed forward Back-propagation จะเป็นโครงข่ายประสาทเทียมที่เป็นที่นิยมและสามารถใช้ได้ดีในงานหลายประเภทนอกเหนือจากการทำนายค่า

2.7 โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับ (Back - Propagation Neural Networks) [18]

โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับนั้นมีโครงสร้างเป็นแบบโครงข่ายประสาทเทียมหลายชั้น (Multi Layer Neural Networks) ซึ่งประกอบไปด้วยชั้นอินพุต (Input Layer) , ชั้นซ่อน (Hidden Layer) และชั้นเอาต์พุต (Output Layer) ฟังก์ชันการถ่ายโอนที่ใช้ในโครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับนี้จะใช้ฟังก์ชันการถ่ายโอนแบบ Linear Transfer Function , Log – Sigmoid Transfer Function และ Tan – Sigmoid Transfer Function ในวิทยานิพนธ์ฉบับนี้ได้ใช้โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับ 2 แบบ คือ โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับที่มีชั้นซ่อน 1 ชั้นและโครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับที่มีชั้นซ่อน 2 ชั้น

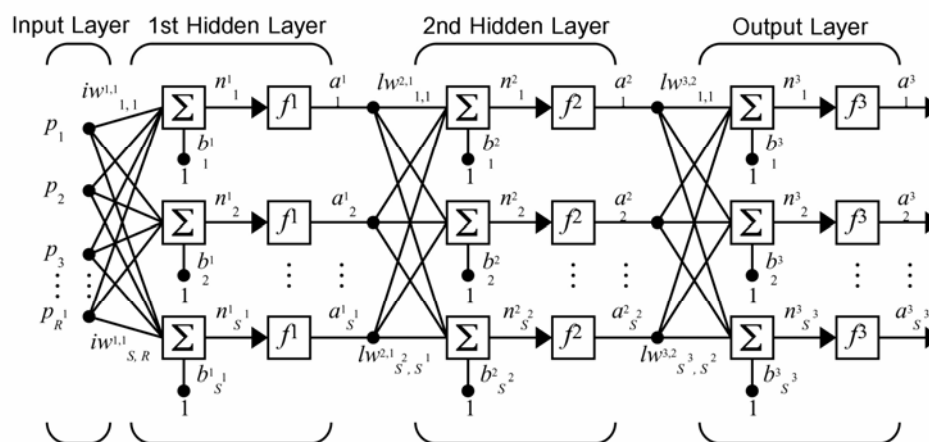
2.7.1 โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับที่มีชั้นซ่อน 1 ชั้น



รูปที่ 2.5 โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับที่มีชั้นซ่อน 1 ชั้น

จากรูปที่ 2.5 จะแสดงโครงข่ายประสาทเทียมที่มีชั้นซ่อน 1 ชั้น โดยในรูปที่ 2.5 ในชั้นอินพุตจะมีจำนวนอินพุตทั้งหมด R ชุด จำนวนนิวรอนในชั้นซ่อนเท่ากับ S1 ชุด และมีเอาต์พุต S2 ชุด สมาชิกของอินพุตเวกเตอร์ P แต่ละตัวจะถูกเชื่อมเข้ากับนิวรอนแต่ละนิวรอนในชั้นซ่อน ด้วยค่าถ่วงน้ำหนัก w1 (Input Weight : $iw_{1,1}$) จากนั้นก็จะทำการรวมกับค่าไบอัส (Bias : b_1) แล้วจึงนำไปผ่านฟังก์ชันการถ่ายโอนเพื่อแปลงข้อมูลให้เป็นผลลัพธ์ ซึ่งเวกเตอร์ a_1 เป็นผลลัพธ์ของชั้นซ่อน ก็จะไปเป็นอินพุตให้กับชั้นเอาต์พุต โดยเวกเตอร์ a_1 ก็จะเชื่อมต่อไปกับค่าถ่วงน้ำหนัก w2 (Layer Weight : $lw_{2,1}$) และนำไปรวมกับค่าไบอัส (Bias : b_2) แล้วนำไปผ่านฟังก์ชันการถ่ายโอนเพื่อคำนวณหาผลลัพธ์ของโครงข่ายประสาทเทียม ซึ่งจะได้เวกเตอร์ a_2 ออกมา

2.7.2 โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับที่มีชั้นซ่อน 2 ชั้น



รูปที่ 2.6 โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับที่มีชั้นซ่อน 2 ชั้น

จากรูปที่ 2.6 จะแสดงโครงข่ายประสาทเทียมที่มีชั้นซ่อน 2 ชั้น โดยในรูปที่ 2.6 ในชั้นอินพุตจะมีจำนวนอินพุตทั้งหมด R ชุด จำนวนนิวรอนในชั้นซ่อนที่ 1 และ 2 เท่ากับ $S1$ และ $S2$ ชุด ตามลำดับ และมีเอาต์พุต $S3$ ชุด สมาชิกของอินพุตเวกเตอร์ P แต่ละตัวจะถูกเชื่อมเข้ากับนิวรอนแต่ละนิวรอนในชั้นซ่อน ด้วยค่าถ่วงน้ำหนัก $w1$ (Input Weight : $iw1,1$) จากนั้นก็จะทำการรวมกับค่าไบอัส (Bias : $b1$) แล้วจึงนำไปผ่านฟังก์ชันการถ่ายโอนเพื่อแปลงข้อมูลให้เป็นผลลัพธ์ ซึ่งจะได้เวกเตอร์ $a1$ ออกมา โดยเวกเตอร์ $a1$ เป็นผลลัพธ์ของชั้นซ่อนที่ 1 ก็จะไปเป็นอินพุตให้กับชั้นซ่อนที่ 2 โดยเวกเตอร์ $a1$ ก็จะเชื่อมต่อไปกับค่าถ่วงน้ำหนัก $w2$ (Layer Weight : $lw2,1$) และนำไปรวมกับค่าไบอัส (Bias : $b2$) แล้วนำไปผ่านฟังก์ชันการถ่ายโอนเพื่อแปลงข้อมูลให้เป็นผลลัพธ์ ซึ่งจะได้เวกเตอร์ $a2$ ออกมา โดยเวกเตอร์ $a2$ ก็จะไปเป็นอินพุตให้กับชั้นเอาต์พุต ซึ่งจะเชื่อมต่อไปกับค่าถ่วงน้ำหนัก $w3$ (Layer Weight : $lw3,2$) จากนั้นก็จะทำการรวมกับค่าไบอัส (Bias : $b3$) แล้วนำไปผ่านฟังก์ชันการถ่ายโอนเพื่อหาผลลัพธ์ของโครงข่ายประสาทเทียม โดยผลลัพธ์สุดท้ายก็คือเวกเตอร์ a^3

2.7.3 ขั้นตอนการปรับค่าถ่วงน้ำหนักและไบอัส

ขั้นตอนการปรับค่าถ่วงน้ำหนักและไบอัสนั้นมีวิธีการอยู่หลายวิธีด้วยกัน เช่น Lavenberg-Marquardt Algorithm , Resilient Back Propagation , Conjugate Gradient เป็นต้น โดยในวิทยานิพนธ์ฉบับนี้ได้ใช้ขั้นตอนการปรับค่าถ่วงน้ำหนักและค่าไบอัสด้วยวิธีการของ Lavenberg-Marquardt Algorithm เนื่องจากเป็นขั้นตอนการปรับค่าถ่วงน้ำหนักและไบอัสที่มีประสิทธิภาพดีที่สุด[6] โดยมีขั้นตอนการปรับค่าถ่วงน้ำหนักและค่าไบอัสดังนี้

$$x_{(new)} = x_{(old)} + \Delta x = x_{(old)} - [J^T \cdot J + \mu I]^{-1} \cdot J^T \cdot e \quad (2.1)$$

$$g = 2J^T \cdot e \quad (2.2)$$

โดยที่ x คือ ค่าถ่วงน้ำหนัก (w) และค่าไบอัส (b)

J คือ จาโคเบียนเมตริกซ์ (Jacobian matrix) ของอนุพันธ์ของค่าผิดพลาด (แต่ละค่า) ต่อค่า x (แต่ละค่า)

J^T คือ Transpose matrix ของ J

g คือ Gradient Factor

e คือ เมตริกซ์ของค่าผิดพลาด (Error) ระหว่างคำตอบ (Target, t_3) กับค่าที่
โครงข่ายประสาทเทียมคำนวณได้ (Answer, a_3)

I คือ Identity matrix

μ คือ Mu factor หรือ Learning factor

$$\text{จาโคเบียนเมตริกซ์ของ } (w) \quad J(w) = \begin{bmatrix} \frac{\partial e_1}{\partial w_{1,1}} & \frac{\partial e_1}{\partial w_{1,2}} & \dots & \frac{\partial e_1}{\partial w_{1,s2}} \\ \frac{\partial e_2}{\partial w_{2,1}} & \frac{\partial e_2}{\partial w_{2,2}} & \dots & \frac{\partial e_2}{\partial w_{2,s2}} \\ \vdots & \vdots & \dots & \vdots \\ \frac{\partial e_{s3}}{\partial w_{s3,1}} & \frac{\partial e_{s3}}{\partial w_{s3,2}} & \dots & \frac{\partial e_{s3}}{\partial w_{s3,s2}} \end{bmatrix} \quad (2.3)$$

$$\text{จาโคเบียนเมตริกซ์ของค่า } (b) \quad J(b) = \begin{bmatrix} \frac{\partial e_1}{\partial b_1} \\ \frac{\partial e_2}{\partial b_2} \\ \vdots \\ \frac{\partial e_{s3}}{\partial b_{s3}} \end{bmatrix} \quad (2.4)$$

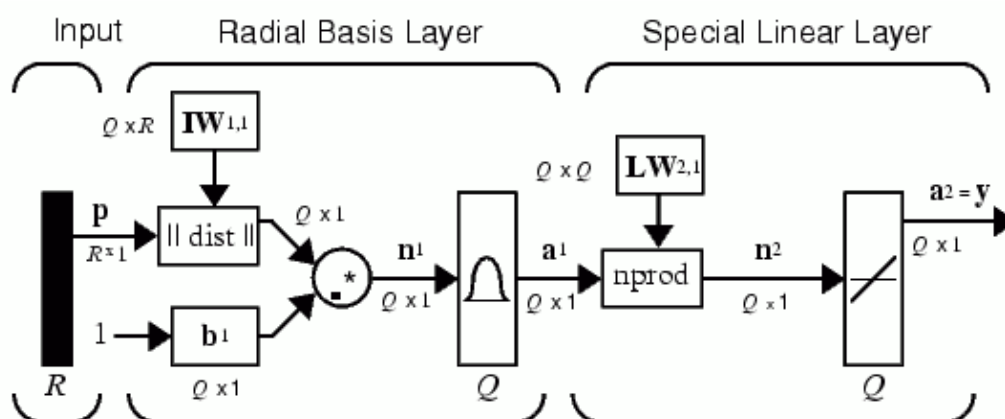
$$\text{เมตริกซ์ของค่าผิดพลาด} \quad \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_{s3} \end{bmatrix} = \begin{bmatrix} t_1^3 - a_1^3 \\ t_2^3 - a_2^3 \\ \vdots \\ t_{s3}^3 - a_{s3}^3 \end{bmatrix} \quad (2.5)$$

โครงข่ายประสาทเทียมแบบแพร่ค่าย้อนกลับนั้นจะมีจำนวนนิวรอนในชั้นอินพุตและชั้นเอาต์พุตคงที่ ซึ่งเท่ากับจำนวนอินพุตและเอาต์พุตของปัญหานั้นเอง ส่วนจำนวนนิวรอนในชั้นซ่อนนั้นสามารถปรับเปลี่ยนได้ตามความเหมาะสมและไม่มีจำนวนที่เป็นสูตรสำเร็จ ดังนั้นในการฝึกสอนของโครงข่ายประสาทเทียมจึงต้องมีการปรับเปลี่ยนจำนวนนิวรอนในชั้นซ่อนเพื่อหาโครงสร้างที่เหมาะสมเพื่อให้ค่าผิดพลาดระหว่างคำตอบที่แท้จริงกับผลลัพธ์ที่ได้จากโครงข่ายประสาทเทียมนั้นมีค่าน้อยที่สุด จำนวนนิวรอนในชั้นซ่อนนั้นมีผลต่อระยะเวลาในการฝึกสอนด้วย นั่นคือถ้ามีจำนวนนิวรอนในชั้นซ่อนมากเวลาที่ใช้ในการฝึกสอนก็จะมากกว่าโครงข่ายประสาทเทียมที่มีจำนวนนิวรอนในชั้นซ่อนน้อยกว่า เมื่อฝึกสอนด้วยเงื่อนไขเดียวกัน แต่อย่างไรก็ตามจำนวนนิวรอนในชั้นซ่อนที่มากก็ไม่ได้เป็นตัวบ่งบอกว่าโครงข่ายจะมีประสิทธิภาพมากกว่าโครงข่ายประสาทเทียมที่มีจำนวนนิวรอนในชั้นซ่อนที่น้อยกว่า ซึ่งต้องทำการลองปรับเปลี่ยนดูเพื่อหาโครงข่ายที่เหมาะสมที่สุด

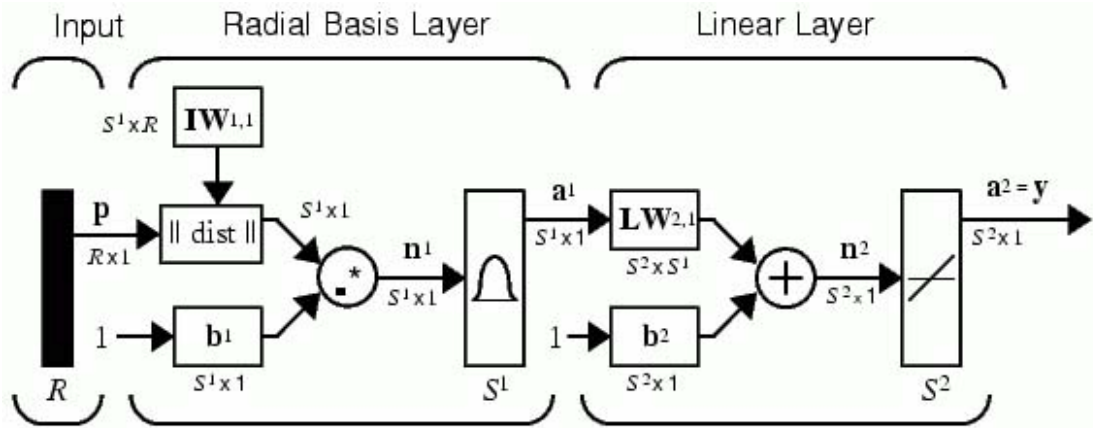
2.8 โครงข่ายประสาทเทียมแบบ Generalized Regression [18]

โครงสร้างพื้นฐานของโครงข่ายประสาทเทียมแบบ Generalized Regression (Generalized Regression Neural Networks : GRNN) นั้นจะประกอบไปด้วยชั้นของนิวรอนทั้งหมด 3 ชั้น คือ ชั้นอินพุต (Input Layer) , ชั้นซ่อนหรือบางครั้งเรียกว่าชั้นฐานรัศมี (Radial Basis Layer) และชั้นเอาต์พุตหรือบางครั้งเรียกว่าชั้นเชิงเส้นพิเศษ (Special Linear Layer)

โครงข่ายประสาทเทียมแบบ Generalized Regression นั้นเป็นโครงข่ายประสาทเทียมที่มีลักษณะคล้ายกับโครงข่ายประสาทเทียมแบบฟังก์ชันฐานรัศมี (Radial Basis Function Neural Networks : RBF) ซึ่งลักษณะโครงสร้างของโครงข่ายประสาทเทียมทั้งสองแบบนี้จะมีลักษณะโครงสร้างที่คล้ายกันมาก ดังรูปที่ 2.7 และรูปที่ 2.8 โดยในรูปที่ 2.7 จะเป็นโครงสร้างของโครงข่ายประสาทเทียมแบบ Generalized Regression และรูปที่ 2.8 เป็นโครงสร้างของโครงข่ายประสาทเทียมแบบฟังก์ชันฐานรัศมี



รูปที่ 2.7 โครงสร้างของโครงข่ายประสาทเทียมแบบ Generalized Regression



รูปที่ 2.8 โครงสร้างของโครงข่ายประสาทเทียมแบบฟังก์ชันฐานรัศมี

จากรูปที่ 2.7 และ 2.8 นั้นจะเห็นว่าในชั้นซ่อนหรือชั้น Radial Basis Layer นั้นจะมีลักษณะโครงสร้างที่เหมือนกัน นั่นคือมีทั้งค่าถ่วงน้ำหนักและค่าไบอัส แต่ว่าในชั้นเอาต์พุตนั้นจะมีจุดที่แตกต่างกันระหว่างโครงสร้างของโครงข่ายประสาทเทียมทั้ง 2 ชนิดนั่นคือ ในโครงข่ายประสาทเทียมแบบฟังก์ชันฐานรัศมีนั้นจะมีทั้งค่าถ่วงน้ำหนักและค่าไบอัส ในขณะที่โครงข่ายประสาทเทียมแบบ Generalized Regression นั้นจะมีเพียงค่าถ่วงน้ำหนักเพียงค่าเดียว และการหาค่า n_2 ของทั้งสองโครงข่ายประสาทเทียมนั้นก็ใช้สมการที่ใช้ในการหาค่า n_2 ที่ต่างกันด้วยเช่นกัน นั่นคือ

โครงข่ายประสาทเทียมแบบ Generalized Regression นั้นจะหาค่า n_2 จากสมการ

$$n_2 = (LW_{2,1} * a_1) / \text{sum}(a_1) \quad (2.6)$$

สำหรับโครงข่ายประสาทเทียมแบบฟังก์ชันฐานรัศมีจะหาค่า n_2 จากสมการ

$$n_2 = b_2 + (LW_{2,1} * a_1) \quad (2.7)$$

ฟังก์ชันการถ่ายโอนที่ใช้ในโครงสร้างของโครงข่ายประสาทเทียมแบบ Generalized Regression นั้นจะใช้ฟังก์ชันการถ่ายโอนชนิด Radial Basis Function (ดังแสดงในตารางที่ 2.1) ในชั้นฐานรัศมี และในชั้นเอาต์พุตนั้นจะใช้ฟังก์ชันการถ่ายโอนชนิด Linear (ดังแสดงในตารางที่ 2.1)

จากรูปที่ 2.7 โครงสร้างของโครงข่ายประสาทเทียมแบบ Generalized Regression จะมีอินพุต R ชุด, นิวรอนในชั้นฐานรัศมี Q ชุด และมีนิวรอนในชั้นเอาต์พุต Q ชุด สมาชิกของอินพุตเวกเตอร์ P แต่ละตัวจะถูกเชื่อมเข้ากับนิวรอนแต่ละนิวรอนในชั้นฐานรัศมีด้วยค่าถ่วงน้ำหนัก (Input weight, $IW_{1,1}$) ซึ่งเป็นเมตริกซ์ขนาด $Q \times R$ และเวกเตอร์ a_1 จะเป็นผลลัพธ์ของชั้นฐานรัศมีโดยมีขนาด Q ก็จะไปเป็นอินพุตของชั้นเอาต์พุต สมาชิกแต่ละตัวของเวกเตอร์ a_1 ก็จะถูกเชื่อมเข้ากับนิวรอนแต่ละนิวรอนในชั้นเอาต์พุตด้วยค่าถ่วงน้ำหนัก (Layer weight, $LW_{2,1}$) ซึ่งเป็นเมตริกซ์ขนาด $Q \times Q$ และเวกเตอร์ a_2 จะเป็นผลลัพธ์ของโครงข่ายประสาทเทียมจะมีขนาด Q

2.9 การเลือกโครงสร้างที่เหมาะสมของโครงข่ายประสาทเทียม [22]

โครงสร้างของโครงข่ายประสาทเทียมจะมีผลต่อประสิทธิภาพภายหลังการเรียนรู้ โดยทั่วไปจำนวนชั้นซ่อนหนึ่งชั้นก็สามรถใช้แก้ปัญหาที่ซับซ้อนทั่วไปในทางปฏิบัติได้ แต่ถ้าหากใช้จำนวนโหนดในชั้นซ่อนจนกระทั่งมีขนาดใหญ่มาแล้วแต่ยังไม่เพียงพอที่จะใช้แก้ปัญหาได้ ดังนั้นก็ต้องเพิ่มจำนวนชั้นซ่อนเป็นสองชั้น

แนวทางในการเลือกจำนวนโหนดในชั้นซ่อนอย่างคร่าวๆ สำหรับปัญหาโดยทั่วไป คือ กฎรูปพีระมิดเรขาคณิต โดยจำนวนเซลล์ประสาทในชั้นซ่อนจะมีรูปร่างเหมือนพีระมิด นั่นคือจำนวนจะค่อยๆ ลดลงจากชั้นอินพุตไปยังชั้นเอาต์พุต

สำหรับโครงข่ายประสาทเทียมที่มีจำนวนชั้นซ่อนหนึ่งชั้นจะสามารถหาจำนวนเซลล์ประสาทในชั้นซ่อนได้จากสมการที่ (2.8)

$$\text{จำนวนของเซลล์ประสาทในชั้นซ่อน} = \sqrt{c \cdot d} \quad (2.8)$$

โดยที่ c = จำนวนตัวแปรอินพุต

d = จำนวนตัวแปรเอาต์พุต

สำหรับโครงข่ายประสาทเทียมที่มีจำนวนชั้นซ่อนสองชั้น จะสามารถหาจำนวนโหนดในชั้นซ่อนแต่ละชั้นได้จากสมการที่ (2.9) และ (2.10) ซึ่งเป็นค่าประมาณอย่างคร่าวๆ เท่านั้น

$$\text{จำนวนของเซลล์ประสาทในชั้นซ่อนที่หนึ่ง} = c \cdot r \quad (2.9)$$

$$\text{จำนวนของเซลล์ประสาทในชั้นซ่อนที่สอง} = c \cdot r^2 \quad (2.10)$$

โดยที่ $r = \sqrt[3]{\frac{d}{c}}$

c = จำนวนตัวแปรอินพุต

d = จำนวนตัวแปรเอาต์พุต