บทที่ 2

แนวคิดและทฤษฎี

ประวัติและวิวัฒนาการของนิวรอลเนตเวิร์ก

นิวรอลเนตเวิร์กเป็นศาสตร์แขนงหนึ่งทางด้านคอมพิวเตอร์ที่ได้รับแรงบันดาลใจจากต้นแบบทางด้านชีววิทยา โดยพยายามเลียนแบบความสามารถของสมองมนุษย์ การศึกษาเริ่มกันอย่างจริงจังเมื่อปี ค.ศ.1950 โดยกลุ่มนักวิทยาศาสตร์แขนงต่าง ๆ เช่น นักฟิสิกส์ วิศวกร และนัก ชีววิทยา กลุ่มนักวิทยาศาสตร์ดังกล่าวข้างต้นได้เริ่มประยุกต์ใช้นิวรอลเนตเวิร์กเพื่อจะแก้ไขปัญหาการรู้จำต่าง ๆ อาทิเช่น การรู้จำอักขระ

ในปลายปี ค.ศ.1950 แฟรงค์ โรเซนเบรท (Frank Rosenblatt) แห่งมหาวิทยาลัยคอร์เนล (Cornell) ได้พัฒนาเพอร์เซบทรอน (Perceptron) ซึ่งเป็นระบบสมองประดิษฐ์ภายใต้ต้นแบบนิวรอลเนตเวิร์ก ระบบของเขามีขีดความสามารถที่จะเรียนรู้วิธีการรู้จำอักขระต่าง ๆ ซึ่งนับเป็นการพัฒนาอันสำคัญที่เป็นแรงผลักดันการค้นคว้าในด้านนี้เป็นอันมาก

ในปี ค.ศ.1969 มาร์วิน มินสกี (Marvin Minsky) ซึ่งเป็นผู้ก่อตั้งแขนงวิชาปัญญาประดิษฐ์ และเป็นผู้แต่งหนังสือชื่อ เพอร์เซบทรอน ในหนังสือเล่มดังกล่าวได้โต้แย้งแนวความคิดเกี่ยวกับ การใช้ซอฟต์แวร์เลียนแบบการทำงานของสมองมนุษย์ ซึ่งการโต้แย้งนี้มีอิทธิพลอย่างมาก ทำให้การพัฒนานิวรอลเนตเวิร์กต้องหยุดชะงักมากกว่าทศวรรษ

ถึงแม้ว่าศาสตร์แขนงนี้เริ่มต้นมาตั้งแต่กลางปี ค.ศ.1950 แต่กลับไม่ได้รับการยอมรับในเชิงพาณิชย์มากนัก จนกระทั่งในกลางปี 1980 เทคโนโลยีทางด้านนิวรอลเนตเวิร์กได้ถูกพัฒนามาใช้ในเชิงพาณิชย์ ตัวอย่างเช่น การวิเคราะห์ความเสี่ยงในการลงทุนของสถาบันการเงินเอวีซีดี (AVCD Financial Services) การควบคุมขบวนการผลิตหลอดไฟฟ้าของจีทีอี (GTE) และตรวจค้นวัตถุระเบิดที่สนามบินนานาชาติจอร์นเอฟเคเนดี้ (John F. Kennedy) ในกรุงนิวยอร์ค เป็นต้น

ปัจจัยที่มีส่วนผลักดันการพัฒนานิวรอลเนตเวิร์ก ในเชิงพาณิชย์ ได้แก่ เครื่องคอมพิวเตอร์ส่วนบุคคลที่มีประสิทธิภาพสูงขึ้น การพัฒนาคอมพิวเตอร์ที่สามารถประมวลผลแบบขนาน และการพัฒนาทฤษฎีชื่อ เจนเนอรอลไรซ์เดลต้ารูล (Generalized Delta Rule) ซึ่งเป็นทฤษฎีที่กล่าวถึงขั้นตอนการเรียนรู้ ทฤษฎีดังกล่าวมีส่วนช่วยให้นิวรอลเนตเวิร์กสามารถแก้ปัญหาที่ซับซ้อนได้มากขึ้น

ปัจจัยหลักอีกอย่างหนึ่งคือ การเข้ามามีส่วนเกี่ยวข้องของหน่วยงานของรัฐบาลโดยเฉพาะอย่างยิ่งกระทรวงกลาโหมของสหรัฐ โดยหน่วยงานที่ติดต่อกับกระทรวงกลาโหมกลุ่มหนึ่งได้ผลักดันแนวความคิดนิวรอลเนตเวิร์กในเชิงพาณิชย์ขึ้น หน่วยงานดังกล่าวได้สร้างนิวรอลเนตเวิร์กที่เป็นฮาร์ดแวร์สำหรับกระทรวงกลาโหม ในโครงการดาร์ปา DARPA (Defense Advanced Research Project Agency) และให้การสนับสนุนทางด้านการเงินและผลักดันการพัฒนานิวรอลเนตเวิร์ก

ไม่เพียงแต่รัฐบาลสหรัฐเท่านั้นที่ให้ความสนใจในศาสตร์แขนงนี้ รัฐบาลประเทศต่าง ๆ ทั่วโลกก็เล็งเห็นถึงความสำคัญ ไม่ว่าจะเป็น ญี่ปุ่น หรือประเทศต่าง ๆ ในยุโรป

การนำนิวรอลเนตเวิร์กมาประยุกต์ใช้งาน

นิวรอลเนตเวิร์กถูกนำไปใช้ในงานหลายประเภท เช่น การประมวลผลสัญญาณ (Signal Processing) การแพทย์ การวิเคราะห์เสียงพูด การจำแนกแบบอย่าง (Pattern Recognition) และการแปลภาษา (Language Translation) เป็นต้น

การประมวลผลสัญญาณ

ตัวอย่างงานประเภทนี้คือ ลดคลื่นสัญญาณรบกวน บนสายโทรศัพท์ โดยใช้ขั้นตอนวิธีของอาดาลีน (Adaline) ในการสอนให้แยกเสียงหรือสัญญาณรบกวนต่างๆออก เช่นเสียงสะท้อนออกเนื่องจาก Repeater Amplifiers ทำให้เกิดการสะท้อนของเสียง

การแพทย์

ตัวอย่างงานประเภทนี้ถูกพัฒนาโดย แอนเดอร์สัน (Anderson) ในกลางปี 1980 โดยใช้แนวคิดนิวรอลเนตเวิร์กในการสอนให้ระบบเรียนรู้ จากการเก็บข้อมูลยาจำนวนมากและข้อมูลอื่นซึ่งประกอบด้วย อาการ (Symtoms) การวินิจฉัยโรค (Diagnosis) และการรักษา (Treatment) พบว่าหลังการสอนให้ระบบเรียนรู้แล้ว สามารถให้ระบบวินิจฉัยโรคและวิธีการรักษาที่เหมาะสมตามลักษณะอาการต่างๆได้

การวิเคราะห์เสียง

การสอนให้อ่านออกเสียง (Phonetic Pronunciation) ในภาษาอังกฤษเป็นเรื่องยาก เพราะการออกเสียงขึ้นกับตัวอักษรอื่นที่อยู่รอบๆ วิธีการเก่าที่ไม่ใช้นิวรอลเนตเวิร์กในการแก้ปัญหาจะทำ

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

การสร้างชุดของกฎการออกเสียงเก็บไว้เพื่อใช้ค้นหาตามรูปแบบที่เก็บไว้ ทำให้ระบบไม่สามารถอ่านคำใหม่ๆได้

ตัวอย่างระบบที่สร้างขึ้นโดยใช้นิวรอลเนตเวิร์กในการแก้ปัญหาคือ NetTalk มีโครงสร้างเป็นข่ายงานหลายชั้น (Multi Layer Neural Network) โดยมีข้อมูลนำเข้าเป็นชุดของภาษาเขียนที่ประกอบด้วยตัวอักษรที่ถูกอ่านและตัวอักษรก่อนและหลัง และการออกเสียงที่ถูกต้อง หลังการสอนพบว่าระบบสามารถอ่านคำใหม่ได้โดยมีข้อผิดพลาดน้อยมาก

การจดจำคำพูด (Speech Recognition) เป็นงานอีกรูปแบบหนึ่งที่ซับซ้อนและมักถูกจำกัดเรื่อง ไวยากรณ์ (Grammar) และการประมวลศัพท์ (Vocabulary) ระบบส่วนใหญ่ที่นำนิวรอลเนตเวิร์กมาแก้ปัญหาประเภทนี้ จะมีโครงสร้างเป็นข่ายงานหลายชั้น และบางระบบใช้การติดต่อแบบย้อนกลับ (Recurrent Network)

ระบบหนึ่งถูกพัฒนาโดย โคโฮเนน (Kohonen) เขาสร้าง Phonetic Typewriter โดยให้ข้อมูลนำเข้าเป็นคำพูดในลักษณะคลื่นสัญญาณ ผลลัพท์ที่ได้จะถูกส่งไปยังการแปลและสะกดคำได้ถูกต้อง

การแปลภาษา (Language Translation)

เนชันเนล เซคเคียวริตี้ เอเจนซี (National Security Agency) ได้พัฒนาระบบทิปสเตอร์ (Tipster) ซึ่งสามารถจะอ่านเอกสารภาษาญี่ปุ่นแล้วแปลเป็นภาษาอังกฤษ และจัดส่งเอกสารที่แปลแล้วไปยังเจ้าหน้าที่ที่เกี่ยวข้องเพื่อวิเคราะห์ต่อไป

การรู้จำแบบอย่าง

การรู้จำลายมือ (Handwriten Recognition) เป็นตัวอย่างงานประเภทหนึ่ง ที่จะทำการจดจำและสามารถหาความแตกต่างของขนาดตัวอักษร ตำแหน่ง ช่องว่าง และรูปแบบการเขียน ทำให้การใช้เทคนิคเดิมใช้ไม่ได้เนื่องจากลายมือของแต่ละบุคคลจะแตกต่างกันไป

ตัวอย่างหนึ่งในการวิเคราะห์งานประเภทนี้จะใช้แบ็กพรอเพเกชัน (Back Propagation) ซึ่งมีโครงสร้างแบบหลายชั้น แต่ละชั้นมีการติดต่อถึงกันแบบใดนั้นขึ้นกับการทดลองปรับปรุงรูปแบบเพื่อให้การเรียนรู้รวดเร็วขึ้น

บริษัทซอฟต์แวร์จำนวนมากได้ผลิตซอฟต์แวร์ที่ประยุกต์ใช้นิวรอลเนตเวิร์ก ในการจำแนกอักขระจากลายมือของผู้ใช้

งานอีกประเภทหนึ่งที่ใช้นิวรอลเนตเวิร์ก คือ การจับคู่รูปแบบ (Pattern Matching) โดยรูปแบบนี้ในการประมวลผลข้อมูลนำเข้าจะประกอบด้วยค่าเลขฐานสอง (Binary Digits) หรือค่าจำนวนจริง (Real Number) ที่ใช้แทน ค่าสัญญานที่เป็นอานาลอก (Analog Signals)

ลักษณะการจับคู่รูปแบบจะมี 2 แบบ คือ ถ้าให้รูปแบบ A เป็นข้อมูลนำเข้า แล้วต้องการได้รูปแบบ A ออกมาเช่นกัน เราเรียกว่า Auto Association เช่น เมื่อเราพยายามระลึกถึงใบหน้าคนที่ไม่ได้พบมานานและให้ข้อมูลนี้ซึ่งอาจจะไม่ชัดเจนเข้าในระบบการเรียนรู้ และผลลัพธ์ที่ต้องการเป็นใบหน้าที่ชัดเจนขึ้น แต่ถ้าให้รูปแบบ A แต่ต้องการผลลัพธ์เป็นรูปแบบ B เราเรียกว่า Hetero Association เช่น ต้องการรู้ชื่อของคนที่มีหน้าตาแบบนี้ ก็จะให้ข้อมูลใบหน้าหน้าแก่ระบบ เพื่อต้องการผลลัพธ์เป็นชื่อบุคคลนั้น

<u>ระบบขับเคลื่อนรถยนต์อัตโนมัติ</u>

ระบบ ALVINN (Autonomous Land Vehicle in Neural Network) เป็นต้นแบบระบบขับเคลื่อนรถยนต์ด้วยตนเอง ซึ่งพัฒนาโดยนักศึกษาและศาสตราจารย์ของมหาวิทยาลัยเคเนกี้เมลลอน (Carnegie-Mellon) ระบบ ALVINN จะได้รับข้อมูลป้อนเข้ามาเป็นสัญญาณจากกล้องถ่ายวิดีโอ สัญญาณดังกล่าวจะถูกส่งเข้าสู่ระบบคอมพิวเตอร์ ซึ่งจะเปรียบเทียบภาพวิดีโอเป็นพัน ๆ ภาพที่ระบบเรียนรู้ ไม่ว่าจะเป็น ต้นไม้, รถที่จอดอยู่ และทางเท้า ระบบ ALVINN จะเรียนรู้การขับเคลื่อนโดยการจับการขับเคลื่อนของมนุษย์เป็นเวลาประมาณ 5 นาที ALVINN จะเรียนรู้การหลีกเลี่ยงวัตถุต่างๆ

<u>การประยุกต์นิวรอลเนตเวิร์กในงานของสถาบันการเงิน</u>

สถาบันการเงินต่างๆมีการประยุกต์ใช้นิวรอลเนตเวิร์กหลายด้านในด้านช่วยการตัดสินใจ ลดความเสี่ยงในการทุจริต และหนี้สูญ เช่น

การสอนระบบให้รู้จักรูปแบบการใช้บัตรเครดิตปลอม 1,000 รูปแบบ จากนั้นได้ทำการทดสอบพบว่า ระบบสามารถที่จะแยกแยะรายการที่ใช้บัตรเครดิตปลอมได้อย่างถูกต้อง แม้ว่ารูปแบบการใช้บัตรเครดิตปลอมจะเปลี่ยนแปลงไป

การพัฒนาระบบ ADAM ซึ่งเป็นการประยุกต์เทคนิคนิวรอลเนตเวิร์ก ผสมผสานกับระบบผู้เชี่ยวชาญ (Expert System) ใช้ในการประเมินความเสี่ยงในแง่คุณสมบัติของผู้กู้เบื้องต้น

การพัฒนาระบบประเมินเงินกู้รายบุคคล โดยประยุกต์ใช้นิวรอลเนตเวิร์กทำการตรวจสอบปัจจัย 27 อย่าง เพื่อพิจารณาว่าจะอนุมัติเงินกู้หรือไม่ และประเมินอสังหาริมทรัพย์

การประเมินความเป็นไปได้ที่บริษัทที่เป็นลูกค้าจะล้มละลาย จากผลการทำงานพบว่าระบบดังกล่าวสามารถพยากรณ์บริษัทที่มีโอกาสล้มละลายได้ถูกต้องสูงถึง 99 เปอร์เซนต์

การพัฒนาระบบที่ปรึกษาในการซื้อขายหลักทรัพย์ หุ้นกู้ และอนุพันธ์ต่าง ๆ

การประยุกต์ใช้ในหน่วยงานของรัฐ

การพัฒนาระบบฮาร์ดแวร์ สำหรับการประยุกต์ใช้นิวรอลเนตเวิร์ก โครงการต่าง ๆ เท่าที่เปิดเผยในปัจจุบัน ได้แก่ ระบบนำทางของอาวุธ ระบบจำแนกเป้าหมายสำหรับเครื่องบินรบและเรือดำน้ำ

นอกจากนี้รัฐบาลสหรัฐยังประยุกต์ใช้นิวรอลเนตเวิร์ก ในแขนงอื่น ๆ เช่น การเข้ารหัสข่าวสารและข้อมูลต่าง ๆ, การสำรวจอากาศ และการตรวจจับทุจริต และการหลีกเลี่ยงการเสียภาษี

การประยุกต์นิวรอลเนตเวิร์กในด้านอื่น ๆ

การประยุกต์นิวรอลเนตเวิร์ก ได้ถูกพัฒนาให้สามารถใช้งานในด้านอื่น ๆ อาทิเช่น การแต่งเพลง, การพยากรณ์ผลการแข่งขันกีฬา, การพยากรณ์อากาศ และการตรวจสอบบัตรโทรศัพท์ปลอม และวิทยาการด้านหุ่นยนต์

นิวรอลเนตเวิร์กทางชีววิทยา

ในทางชีววิทยา นิวรอนมี 3 ประเภท คือ เดนไดรท์ (Dendrites), โซมา (Soma), แอคซอน (Axon) หลายๆ เดนไดรท์ จะได้รับสัญญานจากนิวรอนอื่น สัญญานนี้เป็นการกระตุ้นทางไฟฟ้าที่ถูกส่งข้ามช่องว่างระหว่างไซแนบ (Synapses) ด้วยค่าเฉลี่ยตามขั้นตอนทางเคมี การทำงานทางเคมีจะปรับปรุงสัญญาณที่รับเข้ามาด้วยการวัดความถี่ของสัญญาณที่ได้รับ ซึ่งคล้ายกับการทำงานของค่าน้ำหนักในนิวรอลเนตเวิร์กเทียม

โซมา หรือ ตัวเซล (Cell Body) จะรวบรวมสัญญาณที่ส่งเข้ามา เมื่อรับสัญญาณเข้ามามากพอก็จะส่งผ่านแอคซอนไปยังเซลอื่น ความถี่ของการส่งสัญญาณจะแตกต่างกันไป และอาจจะมองในลักษณะของสัญญาณที่มากกว่าหรือน้อยกว่าค่าค่าหนึ่ง

ความหมายของนิวรอลเนตเวิร์ก

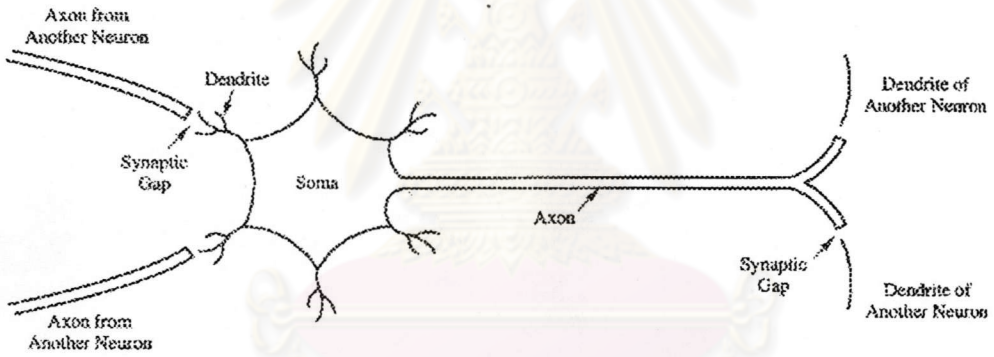
นิวรอลเนตเวิร์ก หรือ ที่เรียกว่า นิวรอลเนตเวิร์กเทียม (Artificial Neural Network) เป็นระบบประมวลผลที่เลียนแบบลักษณะทั่วไปทางชีววิทยาของระบบสมอง นิวรอลเนตเวิร์กเทียมถูกพัฒนาขึ้นตามรูปแบบทางคณิตศาสตร์ของการจดจำของของมนุษย์โดยมีสมมุติฐานว่า

1. การประมวลผลข้อมูลต่างๆเกิดทีส่วนประกอบเล็กๆจำนวนมากที่เรียกว่านิวรอน (Neuron)

2. การส่งสัญญาณ (Signal) ต่างๆระหว่างนิวรอนใช้ส่วนที่เชื่อมต่อระหว่างนิวรอนที่เรียกว่า คอนเนกชัน ลิงค์ (Connection Link)

3. แต่ละการเชื่อมต่อดังกล่าวนั้นจะมีค่าน้ำหนักที่ต่างกัน เพื่อบอกว่า นิวรอนได้รับอิทธิพลลักษณะใดจากนิวรอนอื่น

4. แต่ละนิวรอนจะได้รับข้อมูลจากนิวรอนอื่น โดยการผ่านแอคติเวชันฟังก์ชัน (Activation Function)



รูปที่ 2.1 แสดงนิวรอนในทางชีววิทยา

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

<u>การแบ่งประเภทของนิวรอลเนตเวิร์ก</u>

นิวรอลเนตเวิร์กเทียมได้ถูกพัฒนารูปแบบต่างๆมากมาย ซึ่งจะสามารถแบ่งรูปแบบนิวรอลเนตเวิร์ก ตามลักษณะสำคัญ 3 ประการคือ

1. รูปแบบของสถาปัตยกรรมการเชื่อมต่อระหว่างนิวรอน

2. วิธีกำหนดอัลกอริธึมของวิธีคำนวณหาค่าน้ำหนักของการเชื่อมต่อกัน

3. แอคติเวชันฟังก์ชันที่ใช้

นิวรอลเนตเวิร์กประกอบด้วยกลุ่มของหน่วยประมวลผล (Processing Element) ที่เรียกกันหลายอย่างว่า นิวรอน หรือหน่วย (Unit) หรือเซล (Cell) หรือโหนด (Node) แต่ละนิวรอนเชื่อมต่อด้วยค่าน้ำหนัก และค่าน้ำหนักชุดนั้นเป็นตัวแสดงถึงความรู้ที่จะนำไปใช้แก้ปัญหาในหลายๆรูปแบบซึ่งจะกล่าวต่อไป

ลักษณะของหน่วยประมวลผลย่อยของนิวรอลเนตเวิร์กเทียม อาจจะอธิบายได้ตามคุณสมบัติทางชีววิทยาของนิวรอนได้ดังนี้คือ หน่วยประมวลผลย่อยจะได้รับสัญญาณจากภายนอกมากมาย แต่ละสัญญานจะถูกปรับปรุงด้วยค่าน้ำหนักที่ได้รับจากหน่วยประมวลผลย่อยอื่นๆ และทำการรวบรวมสัญญาณ เมื่อสภาพแวดล้อมเหมาะสมก็จะส่งสัญญาณไปยังหน่วยประมวลผลย่อยอื่นๆ ถัดไป

ชั้น (Layer) ในนิวรอลเนตเวิร์กเทียม คือกลุ่มของหน่วยประมวลผลย่อยกลุ่มหนึ่ง ที่ทำการคำนวณแล้วส่งผลลัพธ์ไปยังชั้นถัดไป จนกระทั่งถึงชั้นสุดท้ายซึ่งอาจมีเพียงหนึ่งหน่วยประมวลผลหรือมากกว่าจะเป็นผลลัพธ์ของเนตเวิร์ก เราเรียกชั้นแรกนี้เป็นชั้นนำข้อมูลเข้า (Input Layer) และชั้นสุดท้ายเป็นชั้นผลลัพธ์ (Output Layer) และชั้นระหว่างชั้นแรกและชั้นสุดท้าย คือ ชั้นแอบแฝง (Hidden Layer) ซึ่งชั้นแอบแฝงนี้อาจมีมากกว่า 1 ชั้นขึ้นไปก็ได้

ลักษณะที่สำคัญและเหมือนกันอย่างหนึ่งระหว่างนิวรอลเนตเวิร์กทางชีววิทยาและนิวรอลเนตเวิร์กเทียมคือ (Fault Tolerance) ซึ่งนิวรอลเนตเวิร์กทางชีววิทยาจะมีลักษณะ 2 แบบดังนี้คือ

1. สามารถจดจำ สัญญาณนำเข้า (Input Signal) ต่างๆ ที่แตกต่างจากที่เคยเจอมาก่อน เช่น จดจำบุคคลที่ไม่ได้พบมานาน

2. สามารถเรียนรู้ต่อไปในขณะที่มีบางเซลตายไป เพราะสมองประกอบด้วยนิวรอนมากกว่าพันล้านเซล แม้บางเซลตายไปก็จะมีเซลอื่นที่จะถูกสอนขึ้นมาทำหน้าที่แทน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สถาปัตยกรรม (Architecture)

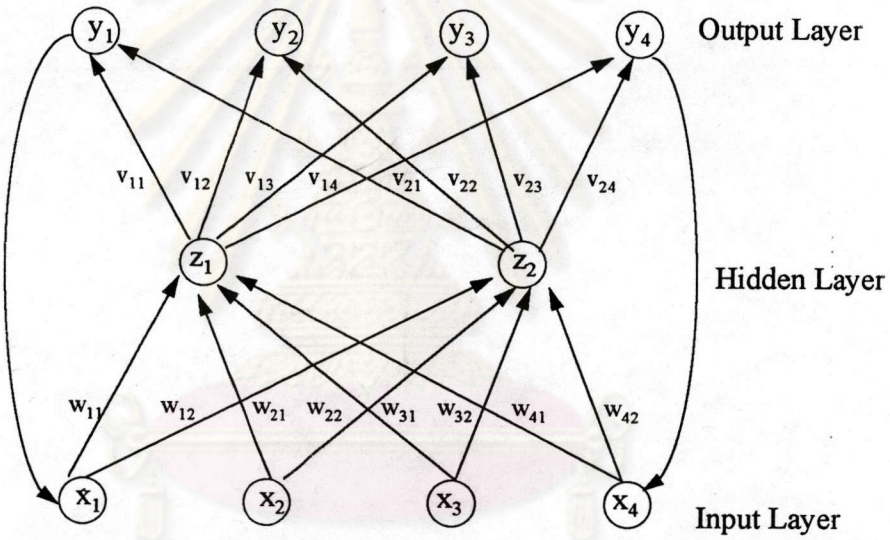
สามารถแบ่งประเภทนิวรอลเนตเวิร์กตามจำนวนชั้นของข่ายงานแบบกว้างๆได้ 2 แบบคือ ข่ายงานชั้นเดียว (Single Layer) และ ข่ายงานหลายชั้น (Multi-Layer Network)

ในบางเอกสารกล่าวว่าข่ายงาน 2 ชั้น เป็นประเภทที่ไม่มีชั้นแอบแฝง คือจะมีชั้นนำเข้าข้อมูลและชั้นผลลัพธ์เท่านั้น แต่ในบางเอกสารไม่นับชั้นนำเข้าข้อมูลเนื่องจากที่ชั้นนี้จะไม่มีการประมวลผล จะนับเฉพาะชั้นที่มีการประมวลผลเท่านั้น ดังนั้นถ้าเป็นข่ายงานสองชั้นจะมี 1 ชั้นแอบแฝง

หรือการนับจำนวนชั้นจะนับจำนวนชั้นของค่าน้ำหนักที่เชื่อมต่อระหว่างนิวรอนเท่านั้น ในบทนี้จะกล่าวถึงชั้นของข่ายงานแบบไม่นับชั้นนำเข้าข้อมูล

รูปแบบส่วนใหญ่ของงานที่พัฒนาขึ้นจากนิวรอลเนตเวิร์กจะเป็นข่ายงาน 2 ชั้น และมีเพียงเล็กน้อยเท่านั้นที่เป็นข่ายงาน 1 ชั้น และยังพอมีบ้างกรณีที่เป็นข่ายงาน 3 ชั้นหรือมากกว่า

ทุก ๆ ข่ายงานจะมีชั้นของข้อมูลนำเข้า และส่วนใหญ่จะส่งข้อมูลนำเข้านี้ไปยังชั้นถัดไปโดยไม่มีการคำนวณ แต่มีข่ายงานบางแบบจะมีกระบวนการติดต่อแบบย้อนกลับ (Recurrent Connections) เช่น ข่ายงานติดต่อย้อนกลับ (Recurrent Network) ตามรูป 2.2 ทำให้ชั้นนำข้อมูลเข้าจะต้องทำการประมวลผลด้วย



รูปที่ 2.2 แสดงการติดต่อย้อนกลับของข่ายงานติดต่อย้อนกลับ

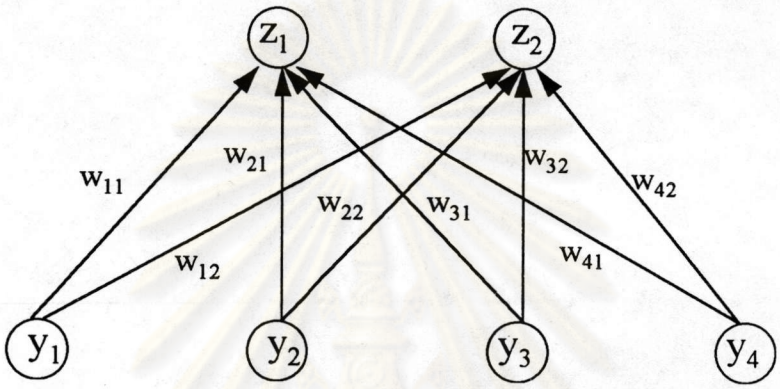
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ข่ายงานชั้นเดียว

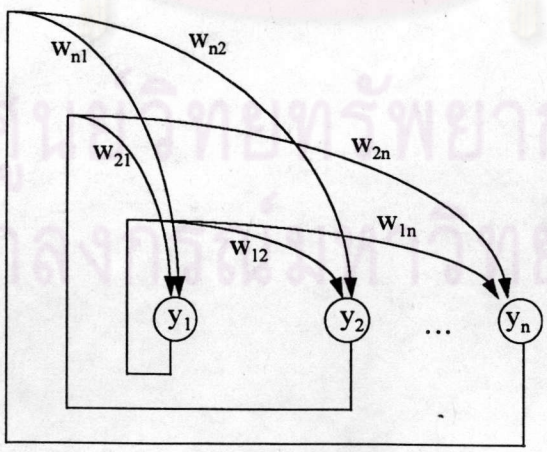
นิวรอลเนตเวิร์กที่เป็นแบบข่ายงานชั้นเดียว จะมีการรับสัญญาณเข้า และประมวลผลที่ชั้นนั้น รวมทั้งการหาผลลัพธ์ก็ทำที่ชั้นนั้นด้วย ข่ายงานรูปแบบนี้ จะมีความสามารถด้านการติดตั้งวงจรไอซี (Integrated Circuit Implement) หรือการควบคุมแนวทางประกอบชิ้นส่วน (Assembly Line Control และที่ใช้กันทั่วไป คือ การจดจำรูปแบบ (Pattern Recognition)

ตัวอย่างของข่ายงานชั้นเดียว คือ เพอเซบทรอนแบบง่าย (Simple Perceptron) และ ฮอปฟิวเนตเวิร์ก (Hopfield Network)

เพอเซปทรอนแบบง่าย จะมีติดต่อกันระหว่างนิวรอนคนละชั้นกัน ชั้นนำข้อมูลเข้าเก็บนิวรอนที่เป็นข้อมูลนำเข้า ชั้นถัดไปเก็บนิวรอนที่เป็นข้อมูลผลลัพธ์ จะไม่มีการเชื่อมติดกันระหว่างนิวรอนในชั้นเดียวกัน ตามรูป 2.3 แต่ฮอปฟิวเนตเวิร์กแล้ว ทุก ๆ นิวรอนในชั้นเดียวกันจะเชื่อมติดต่อซึ่งกันและกัน ตามรูป 2.4



รูปที่ 2.3 แสดงตัวอย่างเพอเซปทรอนอย่างง่าย

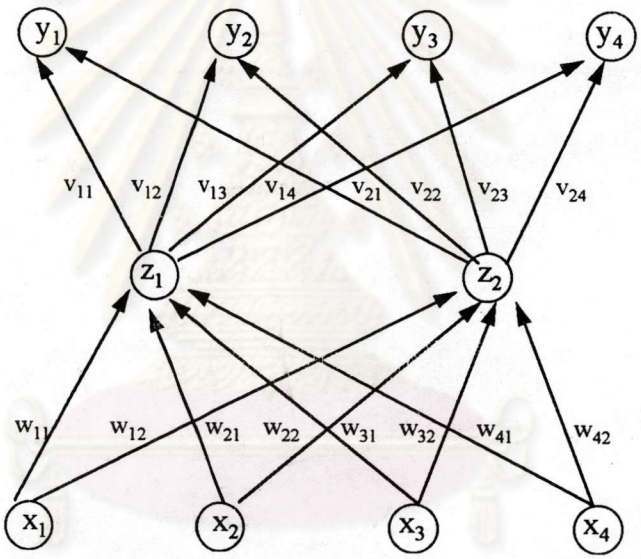


รูปที่ 2.4 แสดงตัวอย่างฮอปฟิวเนตเวิร์ก

ข่ายงานหลายชั้น

เป็นข่ายงานที่มีชั้นแอบแฝง ตั้งแต่ 1 ชั้นขึ้นไป ในกรณีของข่ายงานชั้นเดียวสามารถแก้ปัญหาง่ายๆได้ แต่กรณีปัญหาที่ซับซ้อนจะไม่สามารถแก้ปัญหาด้วยข่ายงานชั้นเดียวได้ การแก้ปัญหานี้ต้องใช้ข่ายงานสองชั้นเป็นอย่างน้อย เช่น งานประเภทจดจำลายมือ มีความจำเป็นต้องใช้ข่ายงานแบบหลายชั้น

ตัวอย่างของข่ายงานสองชั้น คือ แบ็กพรอเพเกชันอย่างง่าย (Simple Backpropagation) ตามรูป 2.5 และ บอลทซแมน แมชชีน (Boltzman Machine) ตามรูป 2.6 เป็นต้น



รูปที่ 2.5 แสดงตัวอย่างโครงสร้างของแบ็กพรอเพเกชัน

รูปแบบการเรียนรู้ (Learning) ของนิวรอลเนตเวิร์ก

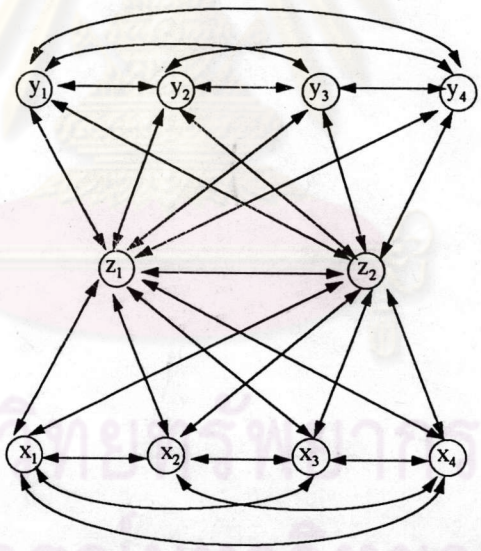
การเรียนรู้ของนิวรอลเนตเวิร์ก เป็นการวิธีการกำหนดหรือหาค่าน้ำหนัก ดังนั้นการวิธีการสอนให้ระบบเรียนรู้จะเป็นส่วนสำคัญในการแยกความแตกต่างระหว่างรูปแบบของนิวรอลเนตเวิร์กได้ การวิธีการสอนให้ระบบเรียนรู้อาจแบ่งได้ 2 รูปแบบ คือ การเรียนรู้แบบมีครู (Supervised Learning) และ การเรียนรู้แบบไม่มีครู (Unsupervised Learning)

การเรียนรู้แบบมีครู

การเรียนรู้จะต้องประกอบด้วย ชุดข้อมูลนำเข้าที่ใช้สอนกับผลลัพท์ที่ต้องการ และค่าน้ำหนักที่ต้องการจะถูกปรับปรุงตามขั้นตอนวิธีการเรียนรู้ (Learning Algorithm) โดยกำหนดว่าการเรียนรู้จะสำเร็จได้ต้องมีการเปรียบเทียบผลลัพธ์จากการคำนวณกับค่าคำตอบที่ให้มา แล้วนำค่าความผิดพลาด (error) มาปรับปรุงค่าน้ำหนัก แล้วเข้ามาคำนวณใหม่จนได้ค่าน้ำหนักที่ดีที่สุด

การเรียนรู้แบบไม่มีครู

เป็นการเรียนรู้แบบอิสระหรือเรียกว่าสามารถจัดการตัวเองได้ (Self Organizing) เพียงระบุรูปแบบข้อมูลนำเข้าโดยไม่ต้องระบุเป้าหมายผลลัพท์ที่ต้องการ ขอเพียงมีข้อมูลที่เหมาะสมที่จะหาคุณสมบัติของชุดข้อมูลนั้นได้ แล้วการเรียนรู้จะส่งคุณสมบัตินั้นไปยังผลลัพธ์ การหาคุณสมบัติเหล่านั้นข่ายงานจะต้องเรียนรู้ที่จะจดจำตามรูปแบบ และวิธีการเรียนรู้ (Learning Method) ต่างๆ เอง



รูปที่ 2.6 แสดงตัวอย่างโครงสร้างของบอทซแมนแมชชีน ในการแก้ปัญหา ENCODER

นอกจากการเรียนรู้ 2 แบบดังกล่าวแล้ว เพื่อให้เข้าใจถึงตัวอย่างต่างๆของนิวรอลเนตเวิร์กที่จะกล่าวถึงต่อไปแล้ว คงต้องกล่าวถึงลักษณะการติดต่อระหว่างนิวรอนในชั้นต่างๆ 2 ประการ คือ

ลักษณะการติดต่อย้อนกลับ (Feedback) และ ลักษณะการติดต่อไปข้างหน้า (Feed-Forward)

ลักษณะการติดต่อย้อนกลับ

โครงสร้างสมองมนุษย์มีลักษณะเป็นชั้น และส่วนใหญ่แต่ไม่ทั้งหมดจะมีลักษณะการติดต่อไปยังชั้นถัดไป ตัวอย่างหนึ่งของข่ายงานที่มีการติดต่อกลับมายังชั้นก่อนหน้าคือ ข่ายงานติดต่อย้อนกลับ เช่น บอลซแมน แมชชีน (Boltzmann Machine) ตามรูป 2.6

ลักษณะการติดต่อไปข้างหน้า

แต่ละนิวรอนจะติดต่อไปยังนิวรอนอื่น ๆ ในชั้นถัดไปเท่านั้น หรือเป็นลักษณะติดต่อกันในทิศทางเดียว เช่น ข่ายงานเพอเซบทรอน และข่ายงานแบ็กพรอเพเกชัน ตามรูป 2.3 และ 2.5

จากรูปแบบทั้งสองนี้ ปัจจุบันได้มีการพัฒนาทฤษฎีและวิธีการเรียนรู้แบบใหม่ขึ้นอีกมากมาย และยังพัฒนากันต่อไป จึงมักเกิดรูปแบบใหม่เสมอ แต่ละวิธีจะมีรูปแบบและวิธีการเรียนรู้ต่างกันออกไป ต่อไปนี้เป็นตัวอย่างกลุ่มหนึ่งเท่านั้น ของงานที่พัฒนาจากพื้นฐานของนิวรอลเนตเวิร์ก

1. การเรียนรู้แบบไม่มีครู (Unsupervised Learning)
   1.1. แบบมีการติดต่อย้อนกลับ (Feedback Nets)
      1.1.1. แอดดิทีฟ กรอสเบิร์ค Additive Grossberg (AG)
      1.1.2. ชันทิง กรอสเบิร์ค Shunting Grossberg (SG)
      1.1.3. ไบนารี อะแดปทีฟ เรโซแนนซ์ Binary Adaptive Resonance Theory (ART1)
      1.1.4. อานาลอก อะแดปทีฟ เรโซแนนซ์ Analog Adaptive Resonance Theory (ART2)
      1.1.5. ดิสครีท ฮอบฟิว Discrete Hopfield (DH)
      1.1.6. คอนตินิวอัส ฮอบฟิว Continuous Hopfield (CH)
      1.1.7. ดิสครีท ไบไดเรคชัน แอสโซซิเอทีฟ เมมโมรี Discrete Bidirectional Associative Memory (BAM)
      1.1.8. เทมโพรอล แอสโซซิเอทีฟ เมมโมรี Temporal Associative Memory (TAM)
      1.1.9. อะแดปทีฟ ไบไดเรคชัน แอสโซซิเอทีฟ เมมโมรี Adaptive Bidirectional Associative Memory (ABAM)

1.1.10. โคโฮเนน เซลออแกนไนซ์ซิง แมป โทโพโลยี พรีเซิฟ แมป Kohonen Self-organizing Map/Topology-preserving map (SOM/TPM)

1.1.11. คอมเพททิทีฟ เลินนิง Competitive learning

1.2. แบบการติดต่อไปข้างหน้า (Feedforward-only Nets)

1.2.1. เลินนิง แมททริค Learning Matrix (LM)

1.2.2. ไดรฟเวอร์ รีอินโฟสเมนต์ เลินนิง Driver-Reinforcement Learning (DR)

1.2.3. ลิเนียร์ แอสโซซิเอทีฟ เมมโมรี Linear Associative Memory (LAM)

1.2.4. ออปติไมซ์ ลิเนียร์ แอสโซซิเอทีฟ เมมโมรี Optimal Linear Associative Memory (OLAM)

1.2.5. สปาส ดิสติบิว แอสโซซิเอทีฟ เมมโมรี Sparse Distributed Associative Memory (SDM)

1.2.6. ฟัสซี แอสโซซิเอทีฟ เมมโมรี Fuzzy Associative Memory (FAM)

1.2.7. เคาเตอร์โพรเกชัน Counterprogation (CPN)

2. การเรียนรู้แบบมีครู (Supervised Learning)

2.1. แบบมีการติดต่อย้อนกลับ (Feedback Nets)

2.1.1. เบรน สเตท อิน อะ บอกซ์ Brain-State-in-a-Box (BSB)

2.1.2. ฟัสซี คอนจิทีฟ แมป Fuzzy Congitive Map (FCM)

2.1.3. บอลทซแมน แมชชีน Boltzmann Machine (BM)

2.1.4. มีน ฟิลด์ แอนนีลลิง Mean Field Annealing (MFT)

2.1.5. รีเคอร์เรนท์ แคสเคด คอร์รีเลชัน Recurrent Cascade Correlation (RCC)

2.1.6. แบ็กพรอเพเกชัน ทรู ทาม Backpropagation through time (BPTT)

2.1.7. เรียลทาม รีเคอร์เรนท์ เลินนิง Real-time recurrent learning (RTRL)

2.1.8. รีเคอร์เรนท์ เอกซเทนท์ คาลแมน ฟิลเทอร์ Recurrent Extended Kalman Filter (EKF)

2.2. แบบการติดต่อไปข้างหน้า (Feedforward-only Nets)

2.2.1. เพอร์เซปทรอน Perceptron

2.2.2. อาดาลีน แมดาลีน Adaline, Madaline

2.2.3. แบ็กพรอเพเกชัน Backpropagation (BP)

2.2.4. คอร์ชี แมชชีน Cauchy Machine (CM)

2.2.5. อะแดปทีฟ ฮิวลิสติค คริติค Adaptive Heuristic Critic (AHC)

2.2.6. ทาม ดีเลย์ นิวรอล เนตเวิร์ก Time Delay Neural Network (TDNN)

2.2.7. แอสโซซิเอทีฟ รีวาด พีนอลตี Associative Reward Penalty (ARP)

2.2.8. อวาแลนชี แมชท์ ฟิวเตอร์ Avalanche Matched Filter (AMF)

2.2.9. แบคเพอร์โคเลชัน Backpercolation (Perc)

2.2.10. อาทแมป Artmap

2.2.11. อะแดปทีฟ โลจิก เนตเวิร์ก Adaptive Logic Network (ALN)

2.2.12. แคสเคด คอร์รีเลชัน Cascade Correlation (CasCor)

2.2.13. เอกซเทนท์ คาลแมน ฟิลเตอร์ Extended Kalman Filter(EKF)

2.2.14. เลินนิง เวกเตอร์ ควอนติเซชัน Learning Vector Quantization (LVQ)

2.2.15. พรอบเบบิลิสติค นิวรอล เนตเวิร์ก Probabilistic Neural Network (PNN)

2.2.16. เจนเนอร์รอล รีเกรสชัน นิวรอล เนตเวิร์ก General Regression Neural Network (GRNN)

<u>ขั้นตอนวิธีของการเรียนรู้ (Learning Algorithm)</u>

การกำหนดวิธีการเรียนรู้ อาจกล่าวได้เป็น 2 ส่วนคือ

1. การกำหนดแอคติเวชันฟังก์ชัน (Activation Function) เพื่อจะจัดขอบเขตข้อมูลผลลัพท์ก่อนส่งไปให้นิวรอนอื่น

2. การกำหนดวิธีการปรับปรุงค่าน้ำหนัก

<u>1. แอคติเวชันฟังก์ชัน</u>

ลักษณะสำคัญที่ใช้กำหนดวิธีการประมวลผลคือ แอคติเวชันฟังก์ชัน และรูปแบบการติดต่อระหว่างนิวรอนที่ทำการรับส่งสัญญาณต่างๆ ปกติในแต่ละชั้น (Layer) มักจะใช้แอคติเวชันฟังก์ชันเดียวกัน และรูปแบบการติดต่อระหว่างนิวรอนก็เป็นรูปแบบเดียวกัน เช่น ในหลายๆ ประเภทของนิวรอลเนตเวิร์กจะมีลักษณะการติดต่อระหว่างนิวรอนแบบถึงกันหมด (Fully Connected)

แอคติเวชันฟังก์ชัน เป็นหลักการทำงานพื้นฐานของนิวรอลเนตเวิร์ก ที่เกี่ยวกับการรวบรวมค่าน้ำหนักและข้อมูลนำเข้าต่างๆ แล้วทำการปรับปรุงเพื่อส่งออกไปให้กับนิวรอนอื่น

ผลลัพธ์ของทุกนิวรอนก่อนที่จะส่งไปยังนิวรอนถัดไป จะผ่านฟังก์ชั่นนี้และจะทำหน้าที่ลดขนาดของผลลัพท์ เพื่อทำให้ผลลัพธ์มีความหมายสำหรับปัญหา

ฟังก์ชันที่ใช้กันบ่อย คือ ฟังก์ชันซิกมอยด์ (Sigmoid Function) ฟังก์ชันขั้นบันได (Step Function) , ฟังก์ชันลาดเอียง (Ramp Function) และ ฟังก์ชันเส้นตรง (Linear Function)

1.1. ฟังก์ชันซิกมอยด์

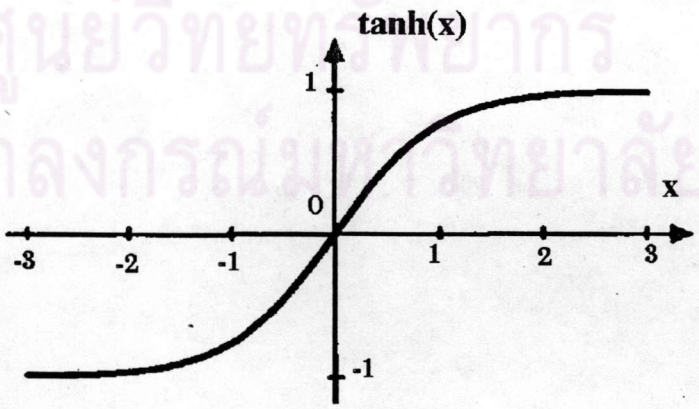
จะทำให้ผลลัพท์จากแอคติเวชั่นมีค่าระหว่าง [0,1] จะเห็นว่าจะมีมากกว่า 1 ฟังก์ชั่นที่ใช้ชื่อซิกมอยด์เช่นกัน ฟังก์ชั่นเหล่านั้นจะแตกต่างกันที่สูตรและขนาดของช่วง และทุกฟังก์ชั่นจะสามารถสร้างกราฟที่มีหน้าตาคล้ายตัว S เช่น ฟังก์ชั่นไฮเปอร์บอลิคแทนเจนท์ (Hyperbolic Tangent Function) และ ฟังก์ชั่นโลจิสติค (Logistic Function)

กรณี ฟังก์ชั่นไฮเปอร์บอลิคแทนเจนท์ จะมีสูตรตาม (1) จะทำให้ค่าผลลัพธ์มีค่าอยู่ในช่วง -1 และ 1 สามารถสร้างกราฟได้ตามรูป 2.7

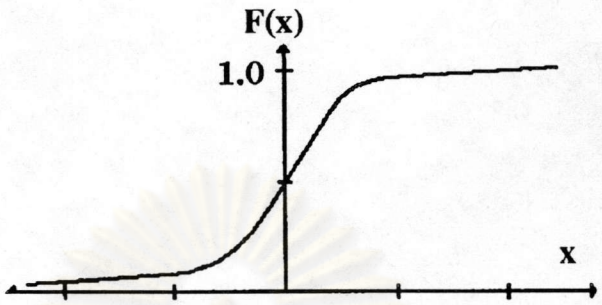
$$F(x) = \tanh(x) = (e^{x} - e^{-x}) / (e^{x} + e^{-x}) \qquad ...(1)$$

กรณี ฟังก์ชั่นโลจิสติค จะมีสูตรตาม (2) จะทำให้ค่าผลลัพธ์มีค่าอยู่ในช่วง 0 และ 1 โดยจะสามารถสร้างกราฟได้ตามรูป 2.8

$$F(x) = 1 / (1 + e^{-x}) \qquad ...(2)$$



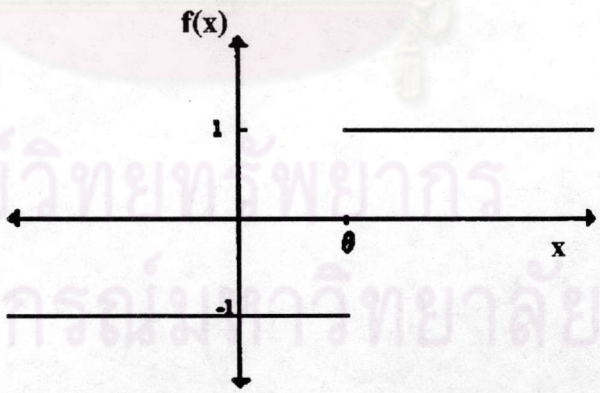
รูปที่ 2.7 แสดงฟังก์ชั่นไฮเปอร์บอลิคแทนเจนท์



รูปที่ 2.8 แสดงฟังก์ชั่นโลจิสติค

1.2. ฟังก์ชั่นขั้นบันได

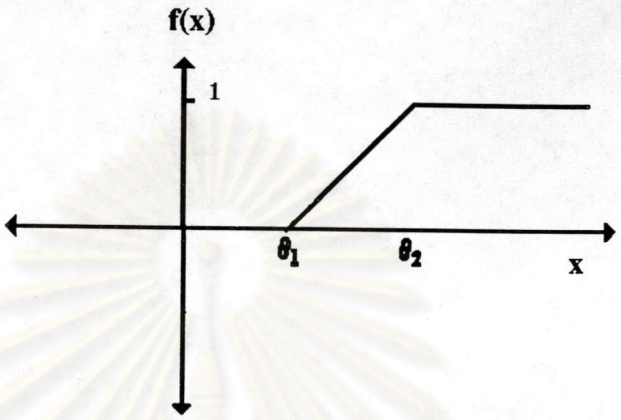
เป็นฟังก์ชั่นที่ใช้กันบ่อย มีลักษณะคล้ายขั้นบันได มีค่าเริ่มต้นที่ 0 และมีค่าเป็น 0 เมื่อฟังก์ชั่นนั้นอยู่ทางซ้ายของค่าขีดแบ่ง (θ) และจะกระโดดเป็น 1 เมื่ออยู่ทางขวาของค่าขีดแบ่งและมีค่าเป็น 1 ต่อไปถ้ายังอยู่ทางขวา นอกจาก 0 และ 1 แล้วยังสามารถใช้ค่าอื่นที่จะเป็นช่วงของฟังก์ชั่นได้ ตามรูป 2.9



รูปที่ 2.9 แสดงฟังก์ชั่นขั้นบันได

1.3. ฟังก์ชั่นลาดเอียง

จากฟังก์ชั่นขั้นบันไดที่ค่ากระโดดจาก 0 เป็น 1 ได้เลย ก็เปลี่ยนรูปแบบเป็นเส้นตรงคล้ายทางลาดแทน ดังนั้นค่าเปลี่ยนจาก 0 เป็น 1 จะได้กราฟดังรูป 2.10



รูปที่ 2.10 แสดงฟังก์ชั่นลาดเอียง

1.4. ฟังก์ชั่นเส้นตรง

จะมีรูปแบบดังนี้

$F(x) = \alpha x + \beta$

ถ้า α เป็น 1 แล้ว ฟังก์ชั่นขีดแบ่งนี้มีค่าเป็นการเพิ่มค่าความเบี่ยงเบน(β) เท่านั้น

2. การปรับปรุงค่าน้ำหนัก

เมื่อนิวรอลเนตเวิร์กถูกพัฒนาขึ้นในรูปแบบของการเรียนรู้แบบมีครู จะต้องกำหนดขั้นตอนการเรียนรู้ที่เหมาะสม ซึ่งขึ้นกับทฤษฎีของรูปแบบในการแก้ไขปัญหานั้นๆ สมการเริ่มต้นในการปรับปรุงค่าน้ำหนักจะเป็นการหาค่าอนุพันธ์ หลังจากปรับปรุงสมการอนุพันธ์ให้ง่ายขึ้นโดยการกำหนดค่าเริ่มต้นให้กับตัวแปรที่เหมาะสม จะทำให้สมการที่ซับซ้อนกลายเป็นเพียงสมการพีชคณิตเพื่อให้ง่ายและคำนวณได้รวดเร็ว

กฎการเรียนรู้ที่สำคัญมี 2 รูปแบบ คือ การเรียนรู้ของเฮบเบียน (Hebbian learning) และการเรียนรู้ของเดลต้า (Delta learning) นอกจาก 2 รูปแบบนี้แล้ว จะมีกฎการเรียนรู้อื่น ๆ ที่ปรับปรุงจากพื้นฐานของ 2 กฎนี้

2.1. กฎของเฮบ (Hebb's Rule)

โดยโดนัล เฮบ (Donald Hebb) เป็นกฎที่เกี่ยวกับการวัดค่าแอคติเวชั่น (ค่าผลรวมของการคูณระหว่างค่าข้อมูลนำเข้ากับค่าน้ำหนักที่เป็นคู่กัน) ของนิวรอนแรกที่จะมีผลต่อค่าแอคติเวชั่นของนิวรอนที่สอง จึงได้สูตรที่ใช้ปรับปรุงค่าน้ำหนัก ดังนี้

$$\Delta w_{ij} = \mu a_i a_j$$

μ เป็นอัตราการเรียนรู้ (learning rate)

a_i เป็นค่าแอคติเวชั่น ของนิวรอนที่ I

a_j เป็นค่าแอคติเวชั่น ของนิวรอนที่ j

2.2. กฎของเดลต้า Delta Rule

หรือเรียกอีกอย่างหนึ่งว่า Least Mean Squared Error Rule (LMS) โดยมีจุดหมายว่าจะหาชุดของค่าน้ำหนักที่ทำให้ค่ากำลังสองของค่าความผิดพลาด (mean squared error) มีค่าน้อยที่สุดซึ่งจำเป็นสำหรับการสอนรอบถัดไป โดย วิดรอพ และ ฮอฟ (Widrow and Hoff) โดยใช้วิธีการคำนวณเพื่อปรับปรุงค่าน้ำหนักดังนี้

$$\Delta w_{ij} = 2\mu x_i(\text{desired output value - computeed output value})_j$$

μ คือ อัตราการเรียนรู้ ซึ่งมีค่าน้อยกว่า 1 แต่ไม่เป็นค่าลบ

x_i คือ นิวรอนลำดับที่i

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย