บทที่ 2

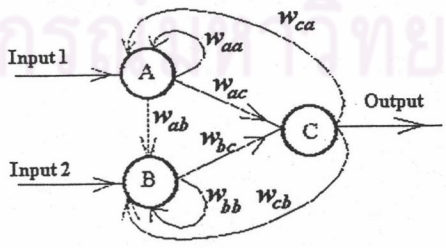
เครือข่ายนิวรอล (Neural Network)

ในบทนี้ได้กล่าวถึงความเป็นมาของเครือข่ายนิวรอล การแบ่งประเภทและรายละเอียดเกี่ยวกับเครือข่ายนิวรอลชนิดที่ใช้ในวิทยานิพนธ์ ได้แก่ แบบจำลองคณิต-ศาสตร์ การเชื่อมต่อและการเรียนรู้ของ NN ซึ่งจะเสนอตามลำดับดังนี้

ความเป็นมาของเครือข่ายนิวรอล

เครือข่ายนิวรอล คือ แบบจำลองทางคณิตศาสตร์ที่พัฒนาขึ้นมา เพื่อจำลองการทำงานของเครือข่ายเซลประสาทในสมองของมนุษย์ ด้วยจุดประสงค์ที่จะสร้างเครื่องมือ ซึ่งมีความสามารถในการเรียนรู้ การจำ การอนุมานความรู้(knowledge deduction) และการอุปมานความรู้(knowledge induction) เช่นเดียวกับความสามารถที่มีในสมองของมนุษย์ โดยทั่วไป NN ประกอบด้วยปมนิวรอลและแขนเชื่อมต่อระหว่างปม ปมนิวรอลมีหน้าที่จัดการสัญญาณที่ได้รับจากปมอื่นผ่านทางแขนของปมแล้วส่งออกให้แก่ปมถัดไป ส่วนวิธีการจัดการสัญญาณภายในปมจะแตกต่างกันไปตามชนิดของ NN แขนของปมนอกจากมีหน้าที่รับส่งสัญญาณผ่านระหว่างปมแล้ว ยังมีหน้าที่ขยายหรือลดทอนขนาดของสัญญาณ ซึ่งอัตราขยายหรือลดทอนถูกกำหนดด้วยค่าน้ำหนัก(weight) ของแขน ตัวอย่างของ NN แบบหนึ่งแสดงในรูปที่ 2-1

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 2-1 ตัวอย่างเครือข่ายนิวรอล

การพัฒนา NN ได้เริ่มต้นขึ้นในค.ศ. 1943 McCulloch และ Pitts ได้พัฒนาแบบจำลอง NN ขึ้นและได้ประยุกต์ใช้กับการคำนวณทางตรรกศาสตร์ ต่อมา D. Hebb ได้เสนออัลกอริทึม Hebb Learning Rule ซึ่งทำให้แบบจำลอง NN ของ McCulloch และ Pitts สามารถเรียนรู้ได้ ในค.ศ. 1958 F. Rosenblatt ได้ประดิษฐ์เครื่อง Perceptron ซึ่งมีความสามารถเรียนรู้และแยกแยะชนิดของรูปแบบได้(pattern classification) ในค.ศ. 1960 B. Widrow และ M. Hoff ได้เสนอแบบจำลอง ADALINE (ADAptive LINEar combiner)ซึ่งสามารถจำและแยกแยะรูปแบบชนิดต่างๆและเสนออัลกอริทึม Widrow & Hoff สำหรับการเรียนรู้ของ ADALINE ในค.ศ. 1972 S. Amari ได้พัฒนาทฤษฎีคณิต-ศาสตร์ของแบบจำลอง NN ขึ้น รวมถึงการวิเคราะห์แบบจำลองและอัลกอริทึมสำหรับการเรียนรู้ของ NN ในค.ศ. 1980 K. Fugushima และ Miyaka ได้พัฒนา Neocognitron ขึ้น Neocognitron เป็น NN ทำหน้าที่จำลองการทำงานของเรตินาในดวงตาของมนุษย์และประยุกต์ใช้ในการจำภาพ 2 มิติ ในค.ศ. 1982 T. Kohonen และ J. Anderson ได้พัฒนา Associative Memory และอัลกอริทึมสำหรับการเรียนรู้ขึ้น ในปีเดียวกันนี้ T. Kohonen ได้พัฒนา Self-organizing Map สำหรับจำรูปร่างของวัตถุ และ J. Hopfield ได้พัฒนา Recurrent Associative Memory ซึ่งมีความสามารถในการเรียนรู้แบบลำดับ(sequencial learning) ในค.ศ. 1986 J. McClelland และ D. Rumelhart ได้เสนอ Backpropagation Neural Network (BNN) และอัลกอริทึม Backpropagation สำหรับการเรียนรู้ BNN เป็น NN ชนิดหลายชั้นและมีความสามารถในการประมาณค่าฟังก์ชัน นอกจากนี้ยังมีการพัฒนาแบบจำลอง NN แต่ละชนิดที่ได้กล่าวมาให้มีประสิทธิภาพดีขึ้น ได้แก่

Counterpropagation Network โดย Hecht-Neilson(1987)

Hamming Net และ Maxnet โดย Lippman(1987)

Adaptive Resonance Network โดย G. carpenter และ S. Grossberg(1987)

Adaptive BAM โดย B. Kosko(1987)

Bidirectional Associative Memory(BAM) โดย B. Kosko(1988)

Wavelet Network โดย Zhang และ Benveniste(1992)

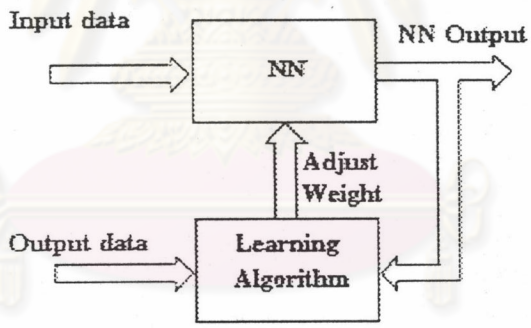
ประเภทของเครือข่ายนิวรอล

จากการพัฒนาเครือข่ายนิวรอลที่ได้กล่าวมาแล้ว NN แต่ละชนิดมีคุณสมบัติที่แตกต่างกันไปตามวิธีการเรียนรู้ โครงสร้างการเชื่อมต่อภายใน และลักษณะการทำงาน การแบ่งประเภทของ NN ได้แบ่งตามคุณสมบัติที่สำคัญ ซึ่งมีดังนี้ (Jacek,1993)

1) การเรียนรู้ของ NN มี 2 แบบ คือ

1.1) การเรียนรู้โดยใช้การแนะนำ(supervised learning)

ความรู้ที่ใช้ในการเรียนรู้วิธีนี้ ประกอบด้วย 2 ส่วน คือ ข้อมูลอินพุต คือ ข้อมูลที่แสดงถึงรูปแบบ(pattern)หรือสถานะของความรู้ และข้อมูลเอาท์พุต คือ ข้อมูลที่แสดงถึงระดับหรือผลตอบสนองต่อข้อมูลอินพุต การเรียนรู้ของ NN ทำโดยป้อนข้อมูลอินพุตให้แก่ NN แล้วนำข้อมูลเอาท์พุตมาเปรียบเทียบกับค่าเอาท์พุตของ NN จากนั้นนำความแตกต่างระหว่างค่าทั้งสองไปปรับค่าพารามิเตอร์ของ NN เพื่อให้ NN สามารถจำข้อมูลอินพุตและส่งเอาท์พุตออกมาเท่ากับข้อมูลเอาท์พุตที่ให้เรียนรู้ การเรียนรู้ด้วยวิธีนี้แสดงในรูปที่ 2-2

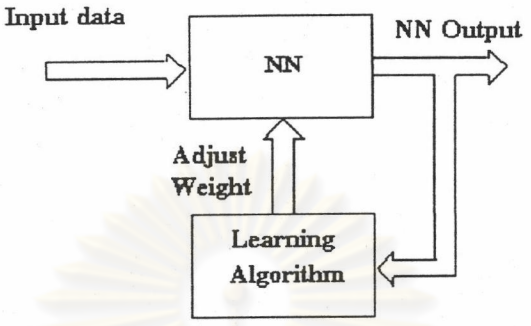


รูปที่ 2-2 การเรียนรู้ของเครือข่ายนิวรอลโดยใช้การแนะนำ

1.2) การเรียนรู้โดยไม่ใช้การแนะนำ (unsupervised learning)

การเรียนรู้ของวิธีนี้ ใช้ในกรณีที่ความรู้ประกอบด้วยข้อมูลอินพุตเพียงอย่างเดียว ไม่มีความรู้ในรูปของข้อมูลเอาท์พุตสำหรับเปรียบเทียบเพื่อปรับค่าพารามิเตอร์ของ NN การปรับค่าพารามิเตอร์จึงใช้เฉพาะค่าเอาท์พุตของ NN การปรับสามารถทำได้หลายวิธี โดยแต่ละวิธีมีจุดประสงค์เพื่อให้ NN สามารถจำ

และแยกแยะรูปแบบ(pattern classification)ของข้อมูลอินพุตได้ การเรียนรู้ของวิธีนี้แสดงในรูปที่ 2-3

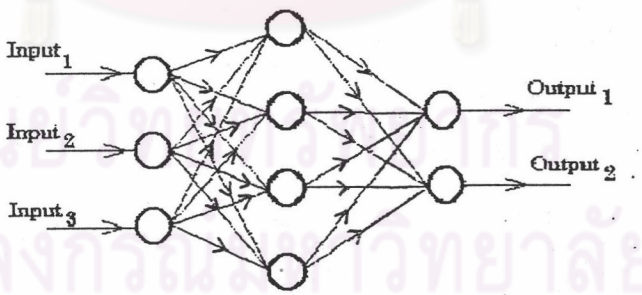


รูปที่ 2-3 การเรียนรู้ของเครือข่ายนิวรอลโดยไม่ใช้การแนะนำ

2) การเชื่อมต่อภายในของเครือข่ายนิวรอล

2.1) การเชื่อมต่อแบบป้อนไปข้างหน้า (feedforward connection)

เป็นการเชื่อมต่อ ซึ่งมีการส่งผ่านสัญญาณไปในทิศทางเดียว โดยมีทิศทางจากส่วนอินพุตของ NN ไปสู่ส่วนเอาท์พุตของ NN ดังแสดงในรูปที่ 2-4



รูปที่ 2-4 การเชื่อมต่อของเครือข่ายนิวรอลแบบป้อนไปข้างหน้า

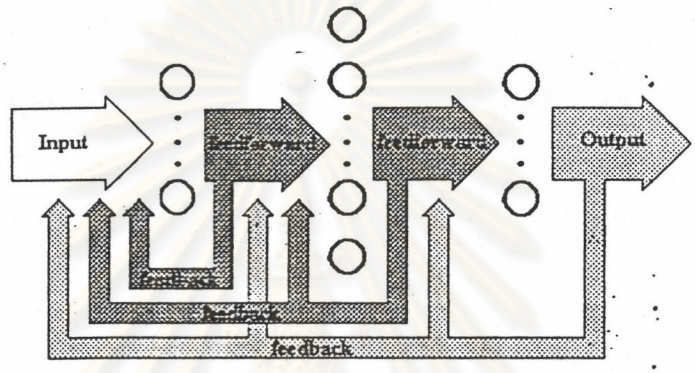
2.2) การเชื่อมต่อแบบหมุนเวียน (recurrent connection)

เป็นการเชื่อมต่อ ซึ่งมีการส่งผ่านสัญญาณ 2 ทิศทาง คือ

ก) ทิศทางจากส่วนอินพุตของ NN ไปสู่ส่วนเอาท์พุตของ NN หรือป้อนไปข้างหน้า (feedforward)

ข) ทิศทางจากส่วนเอาท์พุตไปสู่ส่วนอินพุต หรือ ป้อนกลับ (feedback)

รูปแสดงการเชื่อมต่อชนิดนี้ แสดงดังในรูป 2-5



รูปที่ 2-5 การเชื่อมต่อของเครือข่ายนิวรอลแบบหมุนเวียน

3) ภาวะการทำงานของเครือข่ายนิวรอล แบ่งออกเป็น 2 ภาวะ คือ

3.1) ภาวะการทำงานในโดเมนเวลาที่ต่อเนื่อง (continuous-time operation mode)

3.2) ภาวะการทำงานในโดเมนเวลาที่ไม่ต่อเนื่อง (discrete-time operation mode)

การแบ่งประเภทของเครือข่ายนิวรอลชนิดต่างๆ ได้แสดงไว้ในตารางที่ 2-11

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 2-1 การแบ่งประเภทของเครือข่ายนิวรอล

ชนิดเครือข่ายนิวรอล	วิธีการเรียนรู้ * Sup , Unsup	การเชื่อมต่อ * FF , REC	ภาวะการทำงาน * CT , DT
Single-layer Perceptrons	Sup	FF	CT , DT
Backpropagation Neural Network	Sup	FF	CT , DT
Linear Associative Memory	Sup	FF	CT , DT
Autoassociative Memory	Sup	REC	CT , DT
BAM และ Adaptive BAM	Sup	REC	CT , DT
Selforganized Map	Unsup	FF	CT , DT
Temporal Associative Memory	Sup	REC	CT , DT
Counterpropagation Network	Sup , Unsup	FF	CT , DT
Hamming Net	Unsup	FF	CT , DT
Maxnet	Unsup	REC	CT , DT
Adaptive Resonance Network	Unsup	REC	CT , DT
Wavelet Network	Sup	FF	CT , DT

* หมายเหตุ

Sup หมายถึง Supervised learning และ Unsup หมายถึง Unsupervised learning

FF หมายถึง Feed-Forward connection และ REC หมายถึง REcurrent Connection

CT หมายถึง Continuous-Time mode และ DT หมายถึง Discrete-Time mode

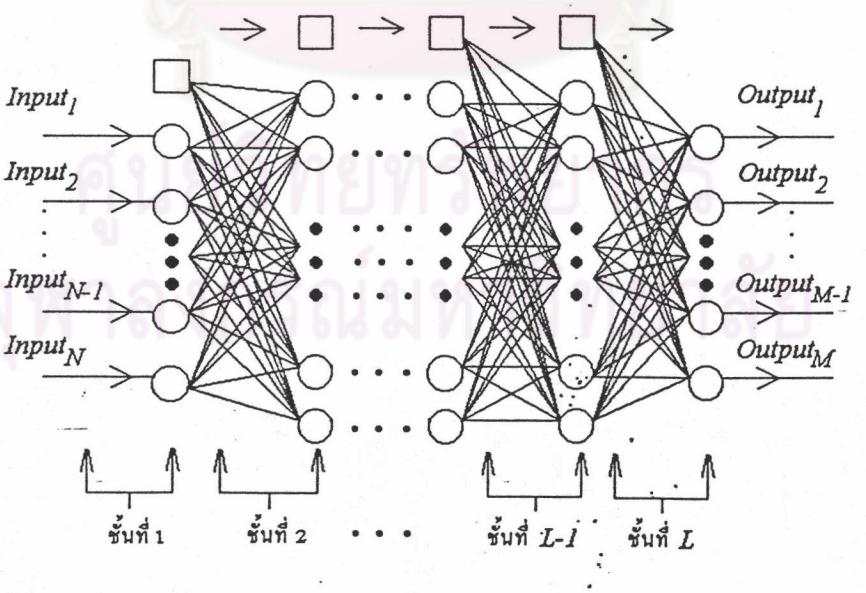
สำหรับเครือข่ายนิวรอลที่ใช้ในวิทยานิพนธ์ ได้เลือกเครือข่ายนิวรอลชนิด Backpropagation สำหรับการใช้งาน ด้วยเหตุผลดังนี้

- เป็น NN ซึ่งมีการเรียนรู้แบบใช้การแนะนำ(supervised learning)
- มีความสามารถในการประมาณค่าฟังก์ชันและสามารถสร้างความสัมพันธ์ในรูปฟังก์ชันขึ้นได้จากข้อมูลที่ให้เรียนรู้ คุณสมบัตินี้ได้พิสูจน์โดย Cybenko (1989)
- เป็น NN ชนิดหลายชั้น ซึ่งการเพิ่มจำนวนชั้นทำให้สามารถจำรูปแบบที่ซับซ้อนและประมาณค่าที่ซับซ้อนได้มากขึ้น

รายละเอียดเกี่ยวกับ NN ชนิดนี้ได้แสดงไว้ในหัวข้อดังต่อไปนี้

เครือข่ายนิวรอลชนิด Backpropagation

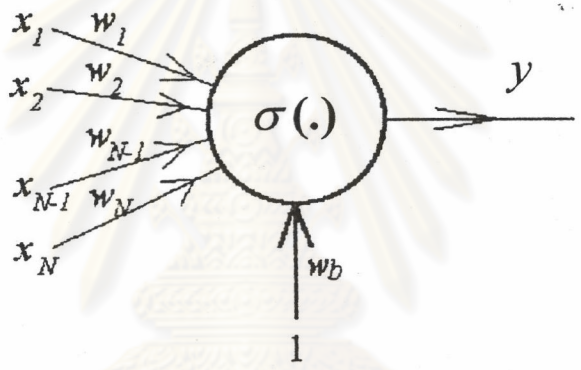
เครือข่ายนิวรอลชนิด Backpropagation (BNN) เป็น NN ชนิดหลายชั้น มีการเรียนรู้แบบใช้การแนะนำ (supervised learning) และมีความสามารถในการประมาณค่าฟังก์ชัน ภายใน BNN ประกอบด้วย ปมนิวรอล ปมไบแอสซึ่งเป็นปมที่มีค่าเอาท์พุตเท่ากับ 1 ตลอดเวลาและแขนเชื่อมต่อระหว่างปม การเชื่อมต่อภายในของ BNN ใช้การเชื่อมต่อแบบป้อนไปข้างหน้า(feedforward connection) ดังแสดงในรูปที่ 2-6 วงกลมแทนปมนิวรอล สี่เหลี่ยมแทนปมไบแอสและเส้นตรงแทนแขนเชื่อมต่อ



รูปที่ 2-6 เครือข่ายนิวรอลชนิด Backpropagation

การจัดเรียงปมภายใน BNN แบ่งปมออกเป็นชั้นๆ ได้แก่ ชั้นนอกสุดทางซ้ายหรือชั้นอินพุต(input layer) ถัดมาทางขวาคือชั้นซ่อนภายใน(hidden layer)และชั้นนอกสุดทางขวาหรือชั้นเอาท์พุต(output layer) ปมนิวรอลในชั้นอินพุตทำหน้าที่รับสัญญาณอินพุตหรือข้อมูลอินพุตเข้าแล้วส่งออกให้แก่ปมนิวรอลในชั้นภายใน ปมนิวรอลในชั้นภายในทำหน้าที่เข้ารหัสความรู้(knowledge encoding)จากสัญญาณที่ได้รับ แล้วส่งสัญญาณรหัสออกไปให้แก่ปมนิวรอลในชั้นเอาท์พุต ปมนิวรอลในชั้นเอาท์พุตทำหน้าที่ตัดสินใจแล้วส่งสัญญาณออกเป็นค่าเอาท์พุตของ BNN

จากภาพรวมกว้างๆเกี่ยวกับการทำงานของ BNN ต่อไปจะกล่าวถึงสมการคณิตศาสตร์สำหรับปมนิวรอลของ BNN ในรูปที่ 2-7



รูปที่ 2-7 ปมนิวรอลของเครือข่ายนิวรอลชนิด Backpropagation

สมการปมนิวรอลของ BNN คือ $$y = \sigma\left(\sum_{i=1}^{N} w_i \cdot x_i + wb\right) \qquad (2\text{-}1)$$

โดยที่ y คือ ค่าเอาต์พุตของปม

x_i คือ ค่าอินพุตของแขนที่ i ของปม

w_i คือ ค่าน้ำหนักของแขนที่ i ของปม

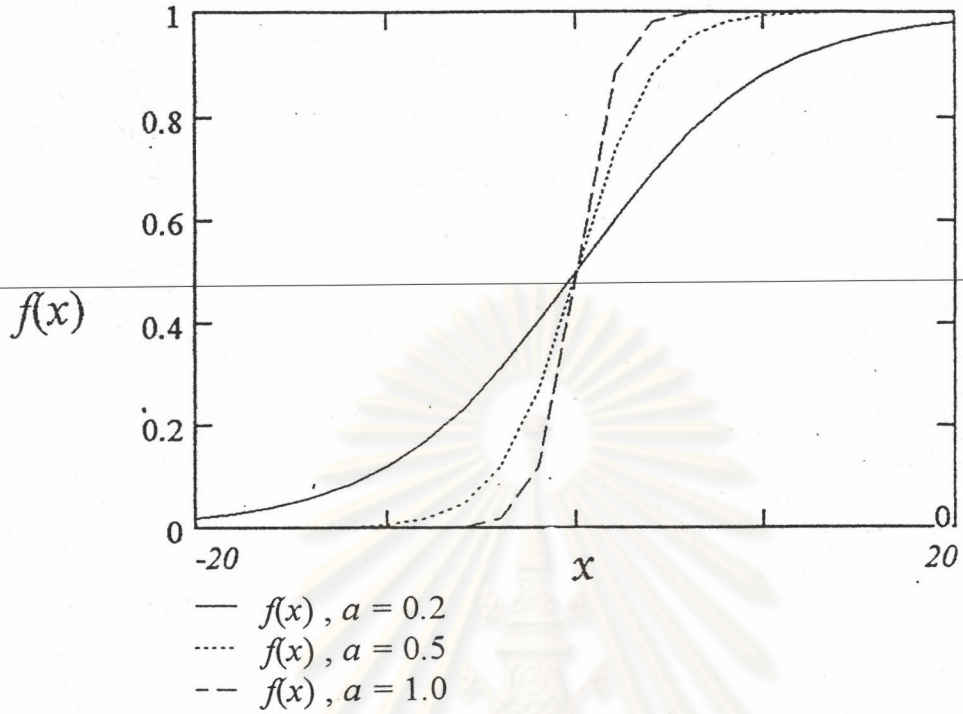
wb คือ ค่าน้ำหนักไบแอสของปม

N คือ จำนวนแขนของปม

$\sigma(x) = 1/(1+\exp(-a.x))$ เรียก $\sigma(.)$ ว่าฟังก์ชันซิกมอยด์

และ a คือค่าสัมประสิทธิ์ของฟังก์ชันซิกมอยด์

กราฟของฟังก์ชันซิกมอยด์ที่ค่า a ต่างๆกัน แสดงในรูป



รูปที่ 2-8 กราฟของฟังก์ชันซิกมอยด์ ที่ค่า a ต่างๆ

ฟังก์ชันซิกมอยด์เป็นฟังก์ชันต่อเนื่องและมีค่าไม่ลดลง(monotonic and non-decreasing) โดยมีค่าเพิ่มจาก 0 ไปยัง 1 เมื่อค่า x เปลี่ยนจากค่า $-\infty$ ไปยังค่า $+\infty$ ค่าสัมประสิทธิ์ของฟังก์ชันซิกมอยด์เป็นค่าที่กำหนดระยะการเปลี่ยนขนาดจาก 0 ไปยัง 1 ของฟังก์ชันซิกมอยด์ ถ้ามีค่าน้อย ระยะการเปลี่ยนมีค่ากว้าง ในทางกลับกันถ้ามีค่ามาก ระยะการเปลี่ยนแปลงจะแคบ ข้อดีของฟังก์ชันซิกมอยด์ คือ ทำให้ BNN มีคุณสมบัติของฟังก์ชันต่อเนื่องและมีค่าอนุพันธ์ ซึ่งสามารถใช้เทคนิคการออพติไมเซชันแบบเกรเดียนท์ สร้างอัลกอริทึมการเรียนรู้ของ BNN ได้

ต่อไปเป็นการแปลง BNN ให้อยู่ในรูปสมการคณิตศาสตร์ เพื่อให้ทราบถึงนิยามของสัญลักษณ์ต่างๆที่ใช้ใน BNN อันเป็นการช่วยเสริมความเข้าใจเพื่อความสะดวกในการคำนวณหาอัลกอริทึมการเรียนรู้ของ BNN การนำเสนอ BNN ได้จัดเรียงตามลำดับดังนี้

สำหรับ BNN ที่มีจำนวนชั้นเท่ากับ L และจำนวนปมในชั้นที่ l เท่ากับ N_l โดย $l = 1, 2, \ldots, L$ สามารถเขียนสมการปมนิวรอลที่ชั้นต่างๆ ได้ดังนี้

ปมนิวรอลในชั้นอินพุตมีหน้าที่แตกต่างจากปมในชั้นอื่น คือทำหน้าที่รับสัญญาณอินพุตเข้าแล้วส่งออกให้ปมในชั้นถัดไป สมการปมอินพุตปมที่ n ในชั้นอินพุตหรือชั้นที่ 1 คือ

$$y_n^1(k) = x_n(k) \qquad (2\text{-}2)$$

โดยที่ $n = 1, 2, \ldots, N_1$

$x_n(k)$ คือค่าอินพุตที่ป้อนให้แก่ปม n ที่เวลา k ในชั้นอินพุตของ BNN

และ $y_n^1(k)$ คือค่าเอาต์พุตของปม n ที่เวลา k ในชั้นอินพุตของ BNN

ส่วนปมนิวรอลในชั้นซ่อนภายใน(hidden layer)และชั้นเอาท์พุต ซึ่งทำหน้าที่เข้ารหัสความรู้และตัดสินใจ มีสมการปมเช่นเดียวกับสมการ(2-1) สมการปมนิวรอลปมที่ i ในชั้นที่ l สามารถเขียนได้ดังนี้คือ

$$y_i^l(k) = \sigma_i^l\left(\sum_{j=1}^{N_{l-1}} w_{ji}^l \cdot y_j^{l-1}(k) + wb_i^l\right) \qquad (2\text{-}3)$$

โดยที่ $i = 1, 2, \ldots, N_l$

N_l คือจำนวนปมนิวรอลในชั้นที่ l

w_{ji}^l คือค่าน้ำหนักของแขนซึ่งส่งผ่านสัญญาณจากปมนิวรอล j ในชั้น l -1 ไปยังปมนิวรอล i ในชั้น l

wb_i^l คือค่าน้ำหนักไบแอสของปมนิวรอล i ในชั้น l

$\sigma_i^l(.)$ คือค่าฟังก์ชันซิกมอยด์ของปมนิวรอล i ในชั้น l และมีค่าสัมประสิทธิ์ของฟังก์ชันซิกมอยด์ประจำปม คือ a_i^l

และ $y_i^l(k)$ คือค่าเอาต์พุตของปมนิวรอล i ที่เวลา k ในชั้น l ของ BNN

สมการปมเอาท์พุตปมที่ m ในชั้นเอาท์พุตหรือชั้นที่ L ของ BNN สามารถเขียนได้ดังนี้คือ

$$y_m^L(k) = \sigma_m^L\left(\sum_{j=1}^{N_{L-1}} w_{jm}^L \cdot y_j^{L-1}(k) + wb_m^L\right) \qquad (2\text{-}4)$$

โดยที่ $m = 1, 2, \ldots, N_L$

N_L คือจำนวนปมนิวรอลในชั้นที่ L

การเรียนรู้ของเครือข่ายนิวรอลชนิด Backpropagation

การเรียนรู้ของเครือข่ายนิวรอลเป็นการแปลงความรู้ ซึ่งประกอบด้วยข้อมูล 2 ส่วนคือ ข้อมูลอินพุต(x)และข้อมูลเอาท์พุต(r)ให้อยู่ในรูปของพารามิเตอร์ของ BNN ได้แก่ ค่าน้ำหนัก ค่าน้ำหนักไบแอส และค่าสัมประสิทธิ์ของฟังก์ชันซิกมอยด์ ในกระบวนการเรียนรู้ BNN สร้างความสัมพันธ์ระหว่างข้อมูลอินพุตและข้อมูลเอาท์พุตด้วยการปรับค่าพารามิเตอร์ภายในตัวมัน การปรับค่าน้ำหนักที่แต่ละแขนเป็นการปรับค่าเพื่อให้ BNN สามารถเข้ารหัสความรู้(knowledge encoding)ได้ การปรับค่าน้ำหนักไบแอสที่แต่ละปมมีผลทำให้จุดศูนย์กลางของฟังก์ชันซิกมอยด์เลื่อนที่ไป การปรับค่าสัมประสิทธิ์ของฟังก์ชันซิกมอยด์ที่แต่ละปมมีผลต่อรูปร่างของฟังก์ชันซิกมอยด์ ผลกระทบของค่าต่างๆของสัมประสิทธิ์ต่อฟังก์ชันซิกมอยด์แสดงในรูปที่ 2-8 วิธีการปรับค่าพารามิเตอร์ทั้ง 3 แบบได้ใช้อัลกอริทึม Backpropagation ซึ่งพัฒนาโดย D. Rumelhart (1986) อัลกอริทึม Backpropagation เป็นอัลกอริทึมที่ใช้หลักการออพติไมเซชันปรับค่าพารามิเตอร์ในทิศทางลบของค่าอนุพันธ์ของดรรชนีสมรรถนะ(performance index)เทียบกับค่าพารามิเตอร์ ดรรชนีสมรรถนะที่ใช้เป็นผลรวมของกำลังสองของค่าความแตกต่างระหว่างข้อมูลเอาท์พุตที่ให้เรียนรู้กับค่าเอาท์พุตของ BNN ดรรชนีสมรรถนะ(J)เขียนได้ดังนี้ คือ

$$J = \sum_{k=0}^{M} \sum_{m=1}^{N_L} (LE_m(k))^2 \qquad (2\text{-}5)$$

โดยที่ $LE_m(k)$ คือค่าผิดพลาดของการเรียนรู้ของปมเอาท์พุตปมที่ m ที่เวลา k มีค่าเท่ากับ $LE_m(k) = r_m(k) - y_m^L(k)$

L คือชั้นนอกสุดมีค่าเท่ากับจำนวนชั้นของ BNN

M คือจำนวนขั้นเวลาที่เรียนรู้มีค่าเท่ากับจำนวนชุดข้อมูลที่ป้อนให้เรียนรู้

N_L คือจำนวนปมนิวรอลในชั้นที่ L

$r_m(k)$ คือค่าข้อมูลเอาต์พุตที่เวลา k ที่ต้องการ ซึ่งป้อนให้ปมเอาท์พุตปม m ในชั้น L ของ BNN เรียนรู้

และ $y_m^L(k)$ คือค่าเอาต์พุตของปม m ที่เวลา k ในชั้นที่ L ของ BNN

การคำนวณหาสมการการปรับค่าน้ำหนักของ BNN คำนวณจากค่าอนุพันธ์ของดรรชนีสมรรถนะเทียบกับน้ำหนัก โดยใช้สมการ(2-2)ถึง(2-5) ได้ดังนี้คือ

$$\frac{\partial J}{\partial w_{ij}^{l}(k)} = \sum_{m=1}^{N_L} \frac{\partial J}{\partial y_m^L(k)} \cdot \frac{\partial y_m^L(k)}{\partial w_{ij}^{l}(k)} \qquad (2\text{-}6)$$

โดยที่ $$\frac{\partial y_m^L(k)}{\partial w_{ij}^{l}(k)} = \sum_{s=1}^{N_{L-1}} \frac{\partial y_m^L(k)}{\partial y_s^{L-1}(k)} \cdot \frac{\partial y_s^{L-1}(k)}{\partial w_{ij}^{l}(k)}$$

$$\frac{\partial y_m^L(k)}{\partial y_s^{L-1}(k)} = \sigma'^{L}_{m}\left(\sum_{n=1}^{N_{L-1}} w_{nm}^{L} \cdot y_n^{L-1}(k) + wb_m^L\right) \cdot w_{sm}^{L}$$

การคำนวณค่า $\frac{\partial y_m^L(k)}{\partial w_{ij}^{l}(k)}$ เป็นการคำนวณแบบสืบทอด(recursive) ซึ่งคำนวณจนกระทั่งชั้นของปมนิวรอลเท่ากับชั้นของค่าน้ำหนักที่กำลังพิจารณา การคำนวณแบบสืบทอดจึงยุติลงที่สมการต่อไปนี้

$$\frac{\partial y_p^l(k)}{\partial w_{ij}^{l}(k)} = \sigma'^{l}_{p}\left(\sum_{n=1}^{N_{l-1}} w_{np}^{l} \cdot y_n^{l-1}(k) + wb_p^l\right) \cdot y_i^{l-1}(k)$$

$$= 1 \qquad \text{เมื่อ } p = j$$

$$= 0 \qquad \text{เมื่อ } p \neq j$$

โดยที่ $\sigma'(x) = \frac{\partial \sigma}{\partial x}(x)$

การปรับค่าน้ำหนักได้ใช้ค่าอนุพันธ์ของดรรชนีสมรรถนะเทียบกับน้ำหนัก และค่าโมเมนตัมของน้ำหนัก ซึ่งคำนวณได้จากค่าผลต่างสืบเนื่องย้อนหลังของน้ำหนัก การใช้วิธีโมเมนตัมร่วมในการปรับค่าจะช่วยเร่งให้ดรรชนีสมรรถนะลู่เข้าสู่ศูนย์เร็วขึ้น อัตราการปรับค่าน้ำหนัก คำนวณได้ดังนี้คือ

$$\Delta w_{ij}^{l}(k) = -\eta_w \cdot \frac{\partial J}{\partial w_{ij}^{l}(k)} + \mu_w \cdot (w_{ij}^{l}(k) - w_{ij}^{l}(k-1)) \qquad (2\text{-}7)$$

โดยที่ η_w คือ อัตราการเรียนรู้ของน้ำหนัก และ μ_w คือ อัตราโมเมนตัมของน้ำหนัก η_w และ μ_w มีความหมายเช่นเดียวกับค่าความยาวก้าว(step size)ในหลักการออพติไมเซชันแบบเกรเดียนท์

คำนวณค่าน้ำหนักที่เวลา $k+1$ ได้คือ

$$w_{ij}^{l}(k+1) = w_{ij}^{l}(k) + \Delta w_{ij}^{l}(k) \qquad (2\text{-}8)$$

การคำนวณหาสมการการปรับค่าน้ำหนักไบแอสของ BNN คำนวณจากค่าอนุพันธ์ของดรรชนีสมรรถนะเทียบกับค่าน้ำหนักไบแอส โดยใช้สมการ(2-2)ถึง(2-5)ได้คือ

$$\frac{\partial J}{\partial wb_j^l(k)} = \sum_{m=1}^{N_L} \frac{\partial J}{\partial y_m^L(k)} \cdot \frac{\partial y_m^L(k)}{\partial wb_j^l(k)} \qquad (2\text{-}9)$$

โดยที่ $$\frac{\partial y_m^L(k)}{\partial wb_j^l(k)} = \sum_{s=1}^{N_{L-1}} \frac{\partial y_m^L(k)}{\partial y_s^{L-1}(k)} \cdot \frac{\partial y_s^{L-1}(k)}{\partial wb_j^l(k)}$$

$$\frac{\partial y_m^L(k)}{\partial y_s^{L-1}(k)} = \sigma'^{L}_{m}\left(\sum_{n=1}^{N_{L-1}} w_{nm}^{L} \cdot y_n^{L-1}(k) + wb_m^L\right) \cdot w_{sm}^{L}$$

$$\frac{\partial y_p^l(k)}{\partial wb_j^l(k)} = \sigma'^{l}_{p}\left(\sum_{n=1}^{N_{l-1}} w_{np}^{l} \cdot y_n^{l-1}(k) + wb_p^l\right) \quad \text{เมื่อ } p=j$$

$$= 0 \quad \text{เมื่อ } p \neq j$$

การปรับค่าน้ำหนักไบแอสได้ใช้ค่าอนุพันธ์ของดรรชนีสมรรถนะเทียบกับน้ำหนักไบแอสและค่าโมเมนตัมของน้ำหนักไบแอส อัตราการปรับค่าน้ำหนักไบแอสคำนวณได้ดังนี้คือ

$$\Delta wb_j^l(k) = -\eta_b \cdot \frac{\partial J}{\partial wb_j^l(k)} + \mu_b \cdot (wb_j^l(k) - wb_j^l(k-1)) \qquad (2\text{-}10)$$

โดยที่ η_b คือ อัตราการเรียนรู้ของน้ำหนักไบแอส

μ_b คือ อัตราโมเมนตัมของน้ำหนักไบแอส

คำนวณค่าน้ำหนักไบแอสที่เวลา $k+1$ ได้คือ

$$wb_j^l(k+1) = wb_j^l(k) + \Delta wb_j^l(k) \qquad (2\text{-}11)$$

การคำนวณสมการการปรับค่าสัมประสิทธิ์ของฟังก์ชันซิกมอยด์ของ BNN คำนวณจากค่าอนุพันธ์ของดรรชนีสมรรถนะเทียบกับค่าสัมประสิทธิ์ โดยใช้สมการ(2-2)ถึง(2-5)ได้คือ

$$\frac{\partial J}{\partial a_j^l(k)} = \sum_{m=1}^{N_L} \frac{\partial J}{\partial y_m^L(k)} \cdot \frac{\partial y_m^L(k)}{\partial a_j^l(k)} \qquad (2\text{-}12)$$

โดยที่ $$\frac{\partial y_m^L(k)}{\partial a_j^l(k)} = \sum_{s=1}^{N_{L-1}} \frac{\partial y_m^L(k)}{\partial y_s^{L-1}(k)} \cdot \frac{\partial y_s^{L-1}(k)}{\partial a_j^l(k)}$$

$$\frac{\partial y_m^L(k)}{\partial y_s^{L-1}(k)} = \sigma'^L_m\left(\sum_{n=1}^{N_{L-1}} w_{nm}^L \cdot y_n^{L-1}(k) + wb_m^L\right) \cdot w_{sm}^L$$

$$\frac{\partial y_p^l(k)}{\partial a_j^l(k)} = \frac{\partial \sigma_p^l}{\partial a_j^l}\left(\sum_{n=1}^{N_{l-1}} w_{np}^l \cdot y_n^{l-1}(k) + wb_p^l\right) \text{ เมื่อ } p=j$$

$$= 0 \qquad \text{เมื่อ } p \neq j$$

และ $$\frac{\partial \sigma}{\partial a}(x) = -a \cdot \exp(-a \cdot x) \cdot (\sigma(x))^2$$

การปรับค่าสัมประสิทธิ์ได้ใช้ค่าอนุพันธ์ของดรรชนีสมรรถนะเทียบกับน้ำหนักร่วมกับค่าโมเมนตัมของค่าสัมประสิทธิ์ อัตราการปรับค่าน้ำหนักไบแอสคำนวณได้ดังนี้คือ

$$\Delta a_j^l(k) = -\eta_a \cdot \frac{\partial J}{\partial a_j^l(k)} + \mu_a \cdot (a_j^l(k) - a_j^l(k-1)) \qquad (2\text{-}13)$$

โดยที่ η_a คือ อัตราการเรียนรู้ของค่าสัมประสิทธิ์

μ_a คือ อัตราโมเมนตัมของค่าสัมประสิทธิ์

คำนวณค่าสัมประสิทธิ์ที่เวลา $k+1$ ได้คือ

$$a_j^l(k+1) = a_j^l(k) + \Delta a_j^l(k) \qquad (2\text{-}14)$$

<u>ขั้นตอนการเรียนรู้ของ BNN</u> สามารถสรุปได้ดังนี้ คือ

1) สุ่มค่าน้ำหนักของ BNN ในแต่ละชั้น จากค่าจำนวนจริงในช่วง -10 ถึง +10

2) กำหนดค่าน้ำหนักไบแอสของปมในแต่ละชั้น โดยแบ่งช่วงของจำนวนจริงอย่างเท่าๆกันตั้งแต่ -10 ถึง +10 เพื่อกระจายค่าน้ำหนักไบแอสในแต่ละชั้นให้อยู่ในช่วงของจำนวนจริงตั้งแต่ -10 ถึง +10 อย่างเท่าๆกัน

3) กำหนดค่าสัมประสิทธิ์(a)ของฟังก์ชันซิกมอยด์แต่ละปม สำหรับการทดลองได้กำหนดให้มีค่าเท่ากับ 0.2

4) เมื่อเก็บข้อมูลอินพุตและข้อมูลเอาท์พุตแล้ว แปลงพิสัย(range)ของข้อมูลเอาท์พุตให้อยู่ในช่วง [0.1, 0.9] เพื่อให้ BNN เรียนรู้ข้อมูลความรู้แล้วได้ค่าน้ำหนักที่มีค่าจำกัด (finite value) ซึ่งทำให้ฟังก์ชันของปมแมพ(map)จากอินพุตซึ่งมีค่าจำกัดไปสู่เอาท์พุต

ซึ่งมีค่าจำกัด ในทางกลับกันถ้าแปลงพิสัยให้อยู่ในช่วง [0, 1] ต้องใช้ค่าน้ำหนักเท่ากับค่า $-\infty$ หรือ $+\infty$ เพื่อทำให้ฟังก์ชันของปมแมพจากอินพุตซึ่งมีค่าจำกัดไปสู่ค่าเอาท์-พุตซึ่งมีค่า 0 หรือค่า 1 ตามลำดับ เพราะฟังก์ชันซิกมอยด์มีค่าเข้าใกล้ 0 หรือ 1 ต่อเมื่อตัวแปรอินพุตของฟังก์ชันมีค่าเข้าใกล้ค่า $-\infty$ หรือค่า $+\infty$ ตามลำดับ

5) ป้อนข้อมูลอินพุตที่ใช้เรียนรู้เข้าทางชั้นอินพุต คำนวณค่าเอาท์พุตของ BNN จากนั้นนำข้อมูลเอาท์พุตที่แปลงพิสัยแล้ว มาเปรียบเทียบกับค่าเอาท์พุตของ BNN แล้วใช้อัลกอริทึม Backpropagation ตามสมการ(2-6)ถึง(2-14) ในตอนเริ่มต้นปรับค่าพารา-มิเตอร์ของ BNN โดยใช้ค่า $\eta_w = 0.2$ $\eta_b = 0.1$ และ $\eta_a = 0.002$

6) เมื่อป้อนข้อมูลอินพุตจนครบหมดทุกค่าแล้ว คำนวณค่าดรรชนีสมรรถนะของข้อมูลความรู้ทั้งหมด จากนั้นเริ่มป้อนข้อมูลความรู้ใหม่และคำนวณค่าดรรชนีสมรรถนะในแต่ละรอบของการเรียนรู้ เมื่อค่าของดรรชนีสมรรถนะมีค่าต่ำสุด ปรับค่า η_w, η_b และ η_a ให้ลดค่าลง 1/2 เท่า และลดค่าลงเช่นนี้ทุกครั้งที่ค่าของดรรชนีสมรรถนะมีค่าต่ำสุด จนกระทั่งค่าของดรรชนีสมรรถนะไม่สามารถลดลงได้อีก

โปรแกรมจำลองการเรียนรู้และการทำงานของ BNN บนคอมพิวเตอร์ ได้แสดงไว้ในภาคผนวกที่ 2

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย