บทที่ 2

ทฤษฎีที่เกี่ยวข้องกับการสกัดความรู้

ทฤษฎีต่าง ๆ ที่นำมาใช้ในการสกัดความรู้เพื่อให้สามารถสกัดความรู้ที่อยู่ในรูปของกฎ “ถ้า-แล้ว” ประกอบด้วย แผนที่การจัดกลุ่มเอง (Self-Organizing Map) หลักเอ็นโทรพีค่าต่ำสุด (Minimization Entropy Principle) ฟัซซี่เซต (Fuzzy Set) และราฟเซต (Rough Set) รายละเอียดดังต่อไปนี้

2.1 แผนที่การจัดกลุ่มเอง (Self-Organizing Map)

แผนที่การจัดกลุ่มเองเป็นโครงข่ายประสาทเทียมประเภทหนึ่งที่มีการเรียนรู้แบบไม่มีผู้สอน ซึ่งเป็นเทคนิคหนึ่งของการทำเหมืองข้อมูล โดยทำการจัดกลุ่มข้อมูลที่มีลักษณะเหมือนกันให้อยู่ในกลุ่มเดียวกัน และทำการลดมิติข้อมูลให้อยู่ที่ 1 มิติ หรือ 2 มิติ มีประโยชน์คือทำให้สามารถเข้าใจลักษณะของข้อมูลในภาพรวมได้ ซึ่งในส่วนถัดไปจะกล่าวถึงการเรียนรู้ของแผนที่การจัดกลุ่มเองที่เป็นการเรียนรู้แบบแข่งขัน (Competitive Learning) การทำงานของแผนที่การจัดกลุ่มเอง ฟังก์ชันเพื่อนบ้าน (Neighborhood Function) และอัตราการเรียนรู้ (Learning Rate) โดยมีรายละเอียดดังนี้

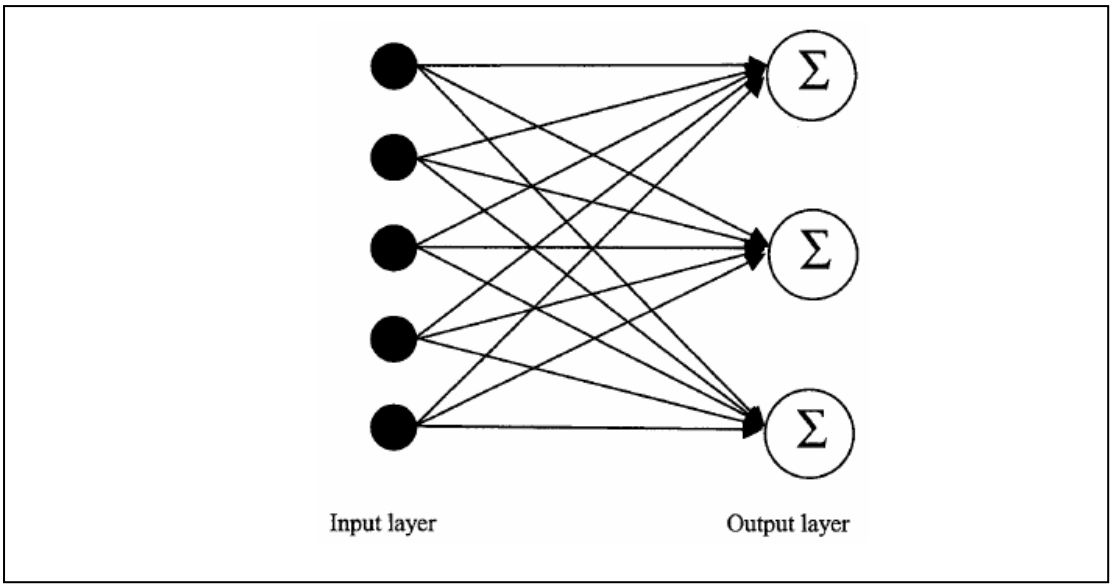
2.1.1 การเรียนรู้แบบแข่งขัน (Competitive Learning)

การเรียนรู้แบบแข่งขันเป็นการเรียนรู้ที่ไม่มีผู้สอน โดยที่ไม่จำเป็นต้องมีตัวอย่างในการสอน การเรียนรู้แบบแข่งขันประกอบด้วยโครงข่ายประสาทเทียมชั้นข้อมูลเข้า (Input Layer) 1 ชั้น และชั้นผลลัพธ์ (Output Layer) 1 ชั้น ดังตัวอย่างภาพประกอบ 2.1 [28]

เว็กเตอร์ข้อมูลเข้าในชั้นข้อมูลเข้าของโครงข่ายการเรียนรู้แบบแข่งขันถูกเชื่อมต่อกับชั้นผลลัพธ์ด้วยเว็กเตอร์น้ำหนัก (Weight Vector) ระหว่างการเรียนรู้นั้นเว็กเตอร์ข้อมูลเข้าจะถูกแทนเข้าไปในโครงข่ายและมีการคำนวณหานิวรอนผลลัพธ์ที่มีลักษณะใกล้เคียงกับเว็กเตอร์ข้อมูลเข้ามากที่สุด หรือเป็นนิวรอนที่ชนะในการแข่งขัน ซึ่งคำนวณจากระยะห่างที่น้อยที่สุดระหว่างเว็กเตอร์ข้อมูลเข้ากับเว็กเตอร์น้ำหนักโดยใช้หลักการของ Euclidean Distance หลังจากนั้นนิวรอนผลลัพธ์ที่เป็นผู้ชนะจะทำการปรับค่าเว็กเตอร์น้ำหนักเพื่อให้ใกล้เคียงกับเว็กเตอร์ข้อมูลเข้า ซึ่งการปรับค่าเว็กเตอร์น้ำหนักนี้จะเรียกว่า “Winner-Takes-All Learning”

กำหนดให้ X คือ เว็กเตอร์ข้อมูลเข้า, W คือ เว็กเตอร์น้ำหนักของแต่ละนิวรอนผลลัพธ์ ดังนั้นระยะห่างระหว่างเว็กเตอร์ข้อมูลเข้า X และ เว็กเตอร์น้ำหนักของนิวรอนผลลัพธ์ตัวที่ i โดยใช้หลักการของ Euclidean Distance ดังสมการ (2.1)

$$\left\| X(t) - W_i(t) \right\| \qquad (2.1)$$



ภาพประกอบ 2.1 โครงข่ายของการเรียนรู้แบบแข่งขัน (The competitive learning network)

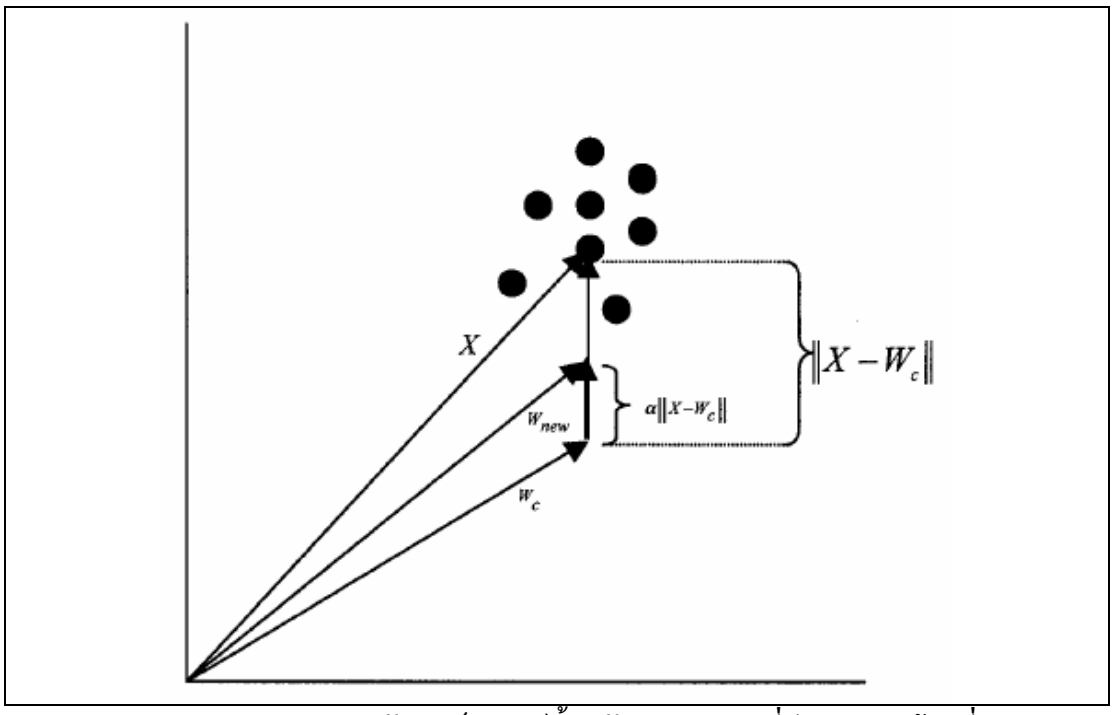
ดังนั้นนิวรอนผลลัพธ์ตัวที่ c ที่มีระยะห่างน้อยที่สุดจะเป็นผู้ชนะ ซึ่ง W_c คือ นิวรอนผลลัพธ์ที่มีระยะห่างน้อยที่สุด ดังสมการ (2.2)

$$\|X(t) - W_c(t)\| = \min_i \{\|X(t) - W_i(t)\|\} \tag{2.2}$$

หลังจากหานิวรอนผลลัพธ์ที่มีระยะห่างน้อยที่สุดได้แล้ว เว็กเตอร์น้ำหนักของนิวรอนผลลัพธ์นั้น จะทำการปรับค่าน้ำหนักใหม่เพื่อให้ใกล้เคียงกับเว็กเตอร์ข้อมูลเข้า X ซึ่ง W_{new} คือ เว็กเตอร์น้ำหนักใหม่, W_{old} คือ เว็กเตอร์น้ำหนักเก่า และ α คือ อัตราการเรียนรู้ ดังสมการ (2.3)

$$W_{new} = W_{old} + \alpha[X - W_{old}] \tag{2.3}$$

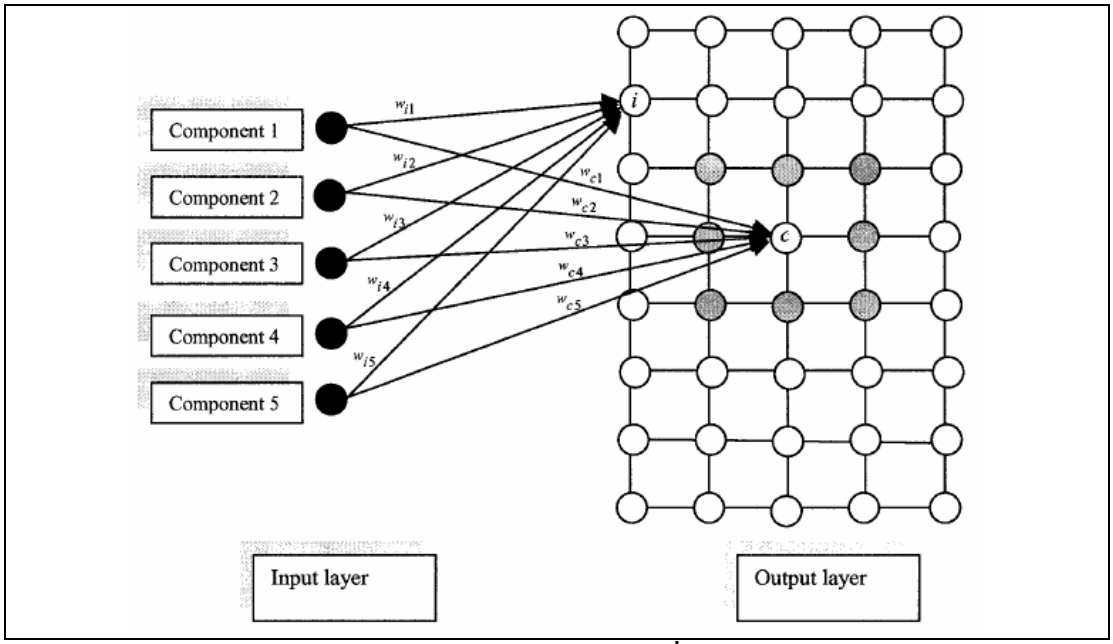
ตัวอย่างภาพประกอบ 2.2 แสดงให้เห็นเว็กเตอร์น้ำหนักใหม่ (W_{new}) ของนิวรอนผลลัพธ์ที่มีระยะห่างน้อยที่สุดเคลื่อนที่เข้าหาเว็กเตอร์ข้อมูลเข้า X



ภาพประกอบ 2.2 การปรับค่าเว็กเตอร์น้ำหนักของนิวรอนที่มีระยะห่างน้อยที่สุด

2.1.2 การทำงานของแผนที่การจัดกลุ่มเอง

แผนที่การจัดกลุ่มเอง ประกอบไปด้วยชั้นข้อมูลเข้าและชั้นผลลัพธ์ ดังตัวอย่างภาพประกอบ 2.3



ภาพประกอบ 2.3 ภาพจำลองแผนที่การจัดกลุ่มเอง

จากภาพประกอบ 2.3 มีเว็กเตอร์ข้อมูลเข้า 5 มิติ ซึ่งแทนด้วยวงกลมสีดำ โดยที่จำนวนนิวรอนข้อมูลเข้านั้นขึ้นอยู่กับมิติของเว็กเตอร์ข้อมูลเข้า วงกลมสีขาวที่อยู่ในชั้นผลลัพธ์แทนด้วยนิวรอนผลลัพธ์ที่อยู่บนตารางแผนที่ 2 มิติ สำหรับวงกลมสีเทาแสดงให้เห็นถึงเพื่อนบ้าน (Neighborhood) ของนิวรอน c บนตารางแผนที่

นิวรอนผลลัพธ์แต่ละตัวถูกเชื่อมกับเว็กเตอร์ข้อมูลเข้าด้วยเว็กเตอร์น้ำหนัก เช่น เว็กเตอร์น้ำหนักของเว็กเตอร์ข้อมูลเข้าตัวที่ 1 ที่เชื่อมกับนิวรอนผลลัพธ์ตัวที่ i ถูกแทนด้วย W_{i1} ดังนั้นมิติของเว็กเตอร์น้ำหนักจะเท่ากับมิติของเว็กเตอร์ข้อมูลเข้า จากภาพประกอบ 2.3 เว็กเตอร์น้ำหนัก คือ เส้นลูกศร

สำหรับแผนที่การจัดกลุ่มเองนั้น ในชั้นข้อมูลเข้าจะแทนด้วยเว็กเตอร์ข้อมูลเข้า ซึ่งกำหนดให้ X คือ เว็กเตอร์ข้อมูลเข้า, n คือ จำนวนมิติของข้อมูลเข้า และ t คือ รอบของการเรียนรู้ ดังสมการ (2.4)

$$X(t) = \left[x_1(t), x_2(t), ..., x_n(t)\right] \tag{2.4}$$

สำหรับชั้นผลลัพธ์จะเป็นตารางแผนที่ 1 มิติ หรือ 2 มิติ โดยที่นิวรอนผลลัพธ์ที่อยู่บนตารางแผนที่จะเชื่อมต่อกับเว็กเตอร์ข้อมูลเข้าด้วยเว็กเตอร์น้ำหนัก ซึ่งกำหนดให้ $W_i(t)$ คือ เว็กเตอร์น้ำหนักของนิวรอนผลลัพธ์ตัวที่ i ของรอบที่ t โดยที่ $1 \le i \le l$ ซึ่ง l คือ จำนวนนิวรอนผลลัพธ์ทั้งหมดบนตารางแผนที่ ดังสมการ (2.5)

$$W_i(t) = \left[w_{i1}(t), w_{i2}(t), ..., w_{in}(t)\right] \tag{2.5}$$

การเรียนรู้ของแผนที่การจัดกลุ่มเองนั้น มีการเรียนรู้แบบแข่งขัน (Competitive Learning) โดยแต่ละรอบของการเรียนรู้จะมีการเลือกนิวรอนที่ชนะ โดยใช้วิธีการ Euclidean Distance ในการคำนวณหาระยะห่างระหว่างเว็กเตอร์ข้อมูลเข้ากับเว็กเตอร์น้ำหนักของแต่ละนิวรอนผลลัพธ์ ดังสมการ (2.6)

$$\left\|X(t) - W_i(t)\right\| = \sqrt{\sum_{j=1}^{n} (x_j - w_{ij})^2} \tag{2.6}$$

สำหรับนิวรอนผลลัพธ์ตัวใดที่มีเว็กเตอร์น้ำหนักใกล้เคียงกับเว็กเตอร์ข้อมูลเข้าแล้วจะเรียกว่า Best Matching Unit (BMU) หรือเป็นนิวรอนผลลัพธ์ที่ชนะในการเรียนรู้แบบแข่งขันของแต่ละรอบเวลา t กำหนดให้ $W_c(t)$ คือ นิวรอนผลลัพธ์ตัวที่ c ที่มีเว็กเตอร์น้ำหนักใกล้เคียงกับเว็กเตอร์ข้อมูลเข้า หรือเป็น BMU ดังสมการ (2.7)

$$\|X(t) - W_c(t)\| = \min_i \{\|X(t) - W_i(t)\|\} \quad (2.7)$$

หลังจากหา BMU ได้แล้ว เว็กเตอร์น้ำหนักของ BMU และเว็กเตอร์น้ำหนักเพื่อนบ้านของ BMU จะมีการปรับค่าน้ำหนักใหม่เพื่อให้ใกล้เคียงกับเว็กเตอร์ข้อมูลเข้า ซึ่งกำหนดให้ $W_i(t+1)$ คือ ค่าเว็กเตอร์น้ำหนักใหม่, $\alpha(t)$ คือ อัตราการเรียนรู้ และ $h_{ci}(t)$ คือ ฟังก์ชันเพื่อนบ้าน (Neighborhood Function) ดังสมการ (2.8)

$$W_i(t+1) = W_i(t) + \alpha(t)h_{ci}(t)[X(t) - W_i(t)] \quad (2.8)$$

หลังจากที่ได้ปรับค่าน้ำหนักใหม่ของ BMU และเพื่อนบ้านของ BMU แล้ว เว็กเตอร์ข้อมูลเข้าตัวถัดไปจะถูกแทนเข้าไปในชั้นข้อมูลเข้าและทำการหา BMU ต่อไป สำหรับขั้นตอนการเรียนรู้ของแผนที่การจัดกลุ่มเอง จะหยุดจัดกลุ่มก็ต่อเมื่อจำนวนรอบการทำงานเกินกว่าที่กำหนดไว้แล้ว

สรุปขั้นตอนการเรียนรู้ของแผนที่การจัดกลุ่มเอง ดังนี้

ขั้นตอนที่ 1 : กำหนดค่าเริ่มต้นให้กับเว็กเตอร์น้ำหนักของนิวรอนผลลัพธ์ทุก ๆ ตัวที่อยู่บนตารางแผนที่, กำหนดฟังก์ชันเพื่อนบ้าน (Neighborhood Function) และกำหนดอัตราการเรียนรู้ (Learning Rate)

ขั้นตอนที่ 2 : พิจารณาเงื่อนไขของการหยุดจัดกลุ่ม ถ้ายังไม่หยุดจัดกลุ่ม ให้ทำขั้นตอนที่ 3 ถึง ขั้นตอนที่ 6

ขั้นตอนที่ 3 : แต่ละรอบของการเรียนรู้ t ทำการสุ่มเว็กเตอร์ข้อมูลเข้า $X(t)$

- คำนวณหาระยะห่างระหว่างเว็กเตอร์ข้อมูลเข้ากับเว็กเตอร์น้ำหนักของทุก ๆ นิวรอนผลลัพธ์

$$\|X(t) - W_i(t)\| = \sqrt{\sum_{j=1}^{n} (x_j - w_{ij})^2}$$

- หา Best Matching Unit (BMU)

$$\|X(t) - W_c(t)\| = \min_i \{\|X(t) - W_i(t)\|\}$$

- ปรับค่าน้ำหนักใหม่ของเว็กเตอร์น้ำหนักของ BMU และเว็กเตอร์น้ำหนักเพื่อนบ้านของ BMU เพื่อให้ใกล้เคียงกับเว็กเตอร์ข้อมูลเข้า

$$W_i(t+1) = W_i(t) + \alpha(t)h_{ci}(t)\left[X(t) - W_i(t)\right]$$

ขั้นตอนที่ 4 : ลดอัตราการเรียนรู้

ขั้นตอนที่ 5 : ลดขนาดของฟังก์ชันเพื่อนบ้าน

ขั้นตอนที่ 6 : พิจารณาเงื่อนไขของการหยุดจัดกลุ่ม ถ้ายังไม่เป็นตามเงื่อนไขที่ต้องการ ให้ไปทำซ้ำขั้นตอนที่ 3

2.1.3 ฟังก์ชันเพื่อนบ้าน (The Neighborhood Function)

ฟังก์ชันเพื่อนบ้าน $\left(h_{ci}(t)\right)$ ของแผนที่การจัดกลุ่มเองที่ใช้โดยทั่ว ๆ ไป มี 2 ชนิด คือ ฟังก์ชันเพื่อนบ้านแบบบับเบิ้ล (The Bubble-based Neighborhood) และฟังก์ชันเพื่อนบ้านแบบเกาส์เสี้ยน (Gaussian-based Neighborhood)

2.1.3.1 ฟังก์ชันเพื่อนบ้านแบบบับเบิ้ล (The Bubble-based Neighborhood)

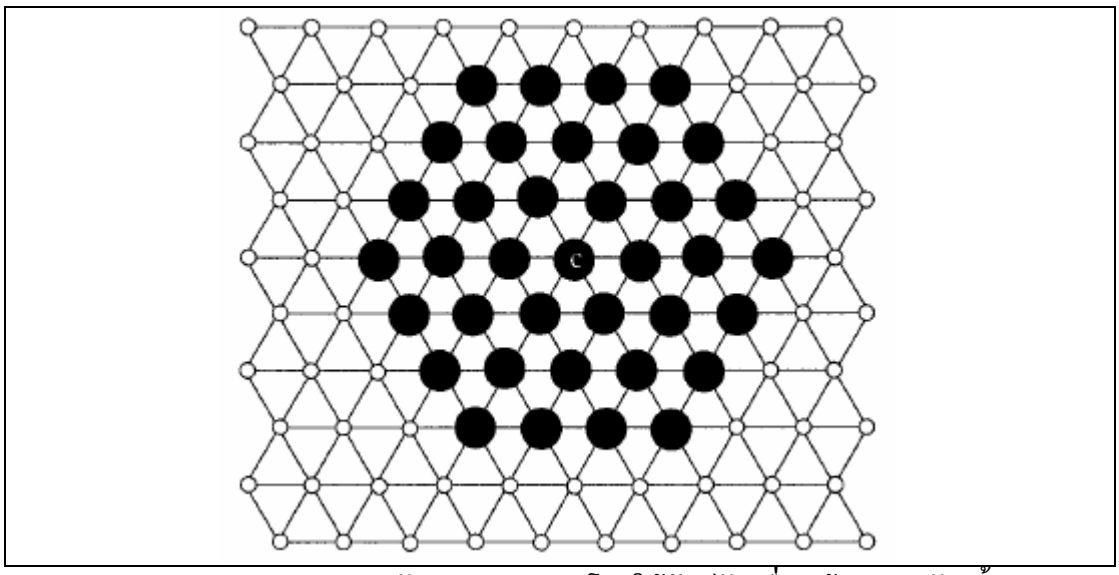
กำหนดให้ $N_c(t)$ คือ กลุ่มเพื่อนบ้านของนิวรอนที่เป็น BMU หรือนิวรอนตัวที่ c ในรอบเวลา t และ $N_c(t)$ จะลดลงตามอัตราการเรียนรู้ สำหรับฟังก์ชันเพื่อนบ้านแบบบับเบิ้ลแสดงได้ดัง สมการ (2.9)

$$h_{ci}(t) = \begin{cases} 1; i = N_c(t) \\ 0; i \neq N_c(t) \end{cases} \qquad (2.9)$$

เว็กเตอร์น้ำหนักของนิวรอนที่เป็นผู้ชนะหรือเรียกว่า BMU และนิวรอนเพื่อนบ้านของ BMU มีการปรับค่าโดยตรง ส่วนนิวรอนที่ไม่ใช่เพื่อนบ้านของ BMU ก็จะไม่มีการปรับค่าใหม่ ดังสมการ (2.10)

$$W_i(t+1) = \begin{cases} W_i(t) + \alpha(t)\left[X(t) - W_i(t)\right]; i = N_c(t) \\ W_i(t); i \neq N_c(t) \end{cases} \qquad (2.10)$$

สำหรับฟังก์ชันเพื่อนบ้านแบบนี้ ค่าเว็กเตอร์น้ำหนักของกลุ่มเพื่อนบ้านจะมีการปรับค่าน้ำหนักตามอัตราการเรียนรู้ ดังตัวอย่างภาพประกอบ 2.4



ภาพประกอบ 2.4 การปรับค่าของนิวรอนโดยใช้ฟังก์ชันเพื่อนบ้านแบบบับเบิ้ล

จากภาพประกอบ 2.4 เป็นการปรับค่านิวรอนผลลัพธ์เมื่อค่ารัศมีของ $N_c(t)$ มีค่าเท่ากับ 3 โดยที่จุดสีขาว คือ นิวรอนที่อยู่บนตารางแผนที่ และจุดสีดำ คือ นิวรอนที่ได้มีการปรับค่าแล้ว

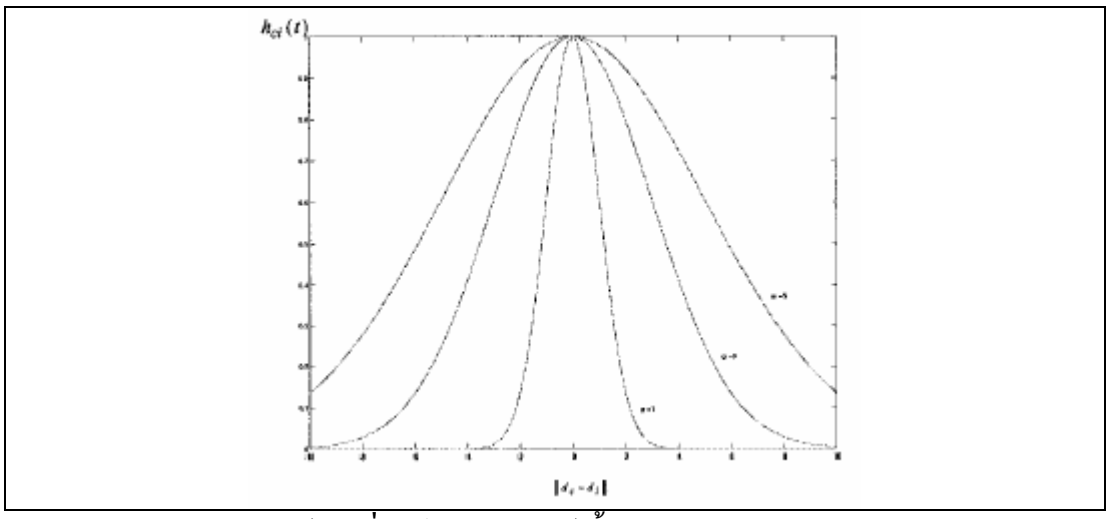
2.1.3.2 ฟังก์ชันเพื่อนบ้านแบบเกาส์เสี้ยน (The Gaussian-based Neighborhood)

สำหรับฟังก์ชันเพื่อนบ้านแบบเกาส์เสี้ยนนี้จะมีการปรับค่าเว็กเตอร์น้ำหนักของทุก ๆ นิวรอนผลลัพธ์ระหว่างการเรียนรู้ โดยที่นิวรอนเพื่อนบ้านที่ใกล้เคียงกับ BMU จะมีการปรับค่าเว็กเตอร์น้ำหนักมากกว่าหรือมีผลกระทบมากกว่านิวรอนตัวอื่น ๆ ที่ไกลจาก BMU ออกไป

กำหนดให้ d_i คือ ตำแหน่งของนิวรอนบนตารางแผนที่ และ d_c คือ ตำแหน่งของนิวรอนที่เป็น BMU บนตารางแผนที่ ดังนั้น $\|d_c - d_i\|$ คือระยะทางระหว่างนิวรอนตัวที่ c ที่เป็น BMU และนิวรอนเพื่อนบ้านตัวที่ i บนตารางแผนที่ และ $\sigma(t)$ คือ ค่าความแปรปรวนของนิวรอนเพื่อนบ้านในรอบเวลา t ซึ่งเป็นตัวควบคุมขนาดของนิวรอนเพื่อนบ้าน สำหรับฟังก์ชันเพื่อนบ้านแบบเกาส์เสี้ยนเป็นดังสมการ (2.11)

$$h_{ci} = e^{\frac{\|d_c - d_i\|^2}{2\sigma^2(t)}} \qquad (2.11)$$

ภาพประกอบ 2.5 แสดงให้เห็นฟังก์ชันเพื่อนบ้านแบบเกาส์เสี้ยนและระยะทางระหว่างนิวรอนตัวที่ c ที่เป็น BMU และนิวรอนเพื่อนบ้านตัวที่ i บนตารางแผนที่



ภาพประกอบ 2.5 ฟังก์ชันเพื่อนบ้านแบบเกาส์เสี้ยนและระดับค่าความแปรปรวน ($\sigma(t)$)

จะเห็นได้ว่าในตอนแรกขนาดเพื่อนบ้านของ BMU มีขนาดใหญ่และจะลดลงไปเรื่อย ๆ ตามระดับค่าความแปรปรวน และในตอนสุดท้ายเพื่อนบ้านที่อยู่ใกล้เคียงกับ BMU มากที่สุดเท่านั้นจะมีการปรับค่า

2.1.4 อัตราการเรียนรู้ (Learning Rate)

สำหรับการกำหนดอัตราการเรียนรู้ $(\alpha(t))$เริ่มต้น ควรกำหนดให้มีค่าใกล้ ๆ 1 หลังจากนั้นในแต่ละรอบการเรียนรู้ให้ลดค่าอัตราการเรียนรู้ไปเรื่อย ๆ จนถึงรอบการเรียนรู้สุดท้าย โดยส่วนมากอัตราการเรียนรู้จะอยู่ระหว่าง $0 < \alpha(t) \leq 1$

สำหรับการลดค่าอัตราการเรียนรู้นั้นอาจเป็นแบบเอ็กโพเนนเชียล กำหนดให้ α_0 คือ อัตราการเรียนรู้เริ่มต้น, t คือ รอบเวลาการเรียนรู้ และ N คือ จำนวนรอบที่กำหนดไว้สำหรับการเรียนรู้ ดังสมการ (2.12)

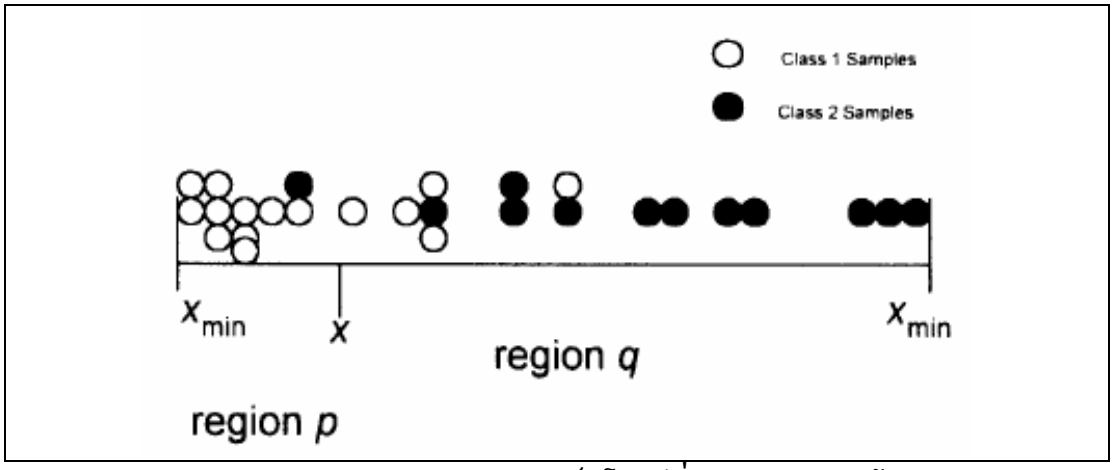
$$\alpha(t) = \alpha_0 e^{\left(-\frac{t}{N}\right)} \tag{2.12}$$

2.2 หลักเอ็นโทรพีค่าต่ำสุด (Minimization Entropy Principle)

สำหรับแนวคิดเบื้องต้นของหลักเอ็นโทรพีที่เกี่ยวข้องกับการจัดกลุ่ม คือ การหาสิ่งที่ทำให้กลุ่มที่เรากำลังสนใจอยู่มีความแตกต่างจากกลุ่มอื่น หรือหาขอบเขตที่ทำให้เกิดการแบ่งแยกกลุ่มได้ชัดเจนที่สุด [29]

หลักเอ็นโทรพีค่าต่ำสุด เป็นค่าที่บอกถึงความคาดหวังของข้อมูล ซึ่งถ้ามีข้อมูลหรือมีเนื้อหาเกี่ยวกับสิ่งนั้น ๆ อยู่แล้ว ก็ไม่จำเป็นต้องใช้ข้อมูลอื่น ๆ เพิ่มเติมเพื่อการตัดสินใจมาก เนื่องจากมีข้อมูลที่เพียงพออยู่แล้วสำหรับการตัดสินใจ

หลักเอ็นโทรพีค่าต่ำสุดเป็นหลักการที่นำมาใช้ในการสร้างฟังก์ชันความเป็นสมาชิกแบบอัตโนมัติ โดยทำการเลือกจุดแบ่งที่ดีที่สุดที่ใช้ในการแบ่งข้อมูลออกเป็น 2 กลุ่ม และจุดแบ่งที่ถูกเลือกมาต้องมีค่าเอ็นโทรพีต่ำที่สุด ซึ่งจุดแบ่งเหล่านี้จะเป็นตัวกำหนดช่วงค่าของฟังก์ชันความเป็นสมาชิก



ภาพประกอบ 2.6 การคำนวณหาเอ็นโทรพีต่ำสุดของจุดแบ่งข้อมูล

จากภาพประกอบ 2.6 ต้องการหาจุดแบ่งที่ดีที่สุดในการแบ่งข้อมูลออกเป็น 2 กลุ่ม กำหนดให้จุดแบ่งข้อมูล คือ x ถ้าต้องการหาจุดแบ่งข้อมูลที่ดีที่สุดซึ่งอยู่ระหว่างข้อมูลตัวอย่างที่อยู่ในช่วงค่า x_{min} กับ x_{max} โดยการคำนวณหาค่าเอ็นโทรพีของแต่ละจุดแบ่งข้อมูล (x) ระหว่างขอบเขต $p[x_{min}, x]$ และขอบเขต $q[x, x_{max}]$ ดังสมการ (2.13)

$$S(x) = p(x)S_p(x) + q(x)S_q(x) \tag{2.13}$$

จากสมการ (2.13) $p(x)$ คือ สัดส่วนของข้อมูลทั้งหมดในขอบเขต p, $q(x)$ คือ สัดส่วนของข้อมูลทั้งหมดในขอบเขต q โดยที่ $p(x)+q(x) = 1$ เสมอ

โดยแต่ละขอบเขตสามารถกำหนดค่าเอ็นโทรพีของแต่ละขอบเขตได้ กำหนดให้ $S_p(x)$ คือ ค่าเอ็นโทรพีของขอบเขต p และ $S_q(x)$ คือ ค่าเอ็นโทรพีของขอบเขต q ซึ่งสามารถคำนวณได้จากสมการ (2.14) และ (2.15) ตามลำดับ

$$S_p(x) = -(p_1(x)\ln p_1(x) + p_2(x)\ln p_2(x)) \tag{2.14}$$

$$S_q(x) = -(q_1(x)\ln q_1(x) + q_2(x)\ln q_2(x)) \tag{2.15}$$

โดยที่ค่า $p_k(x)$ คือ ความเป็นไปได้ที่ข้อมูลกลุ่ม k ที่อยู่ในขอบเขต p และ $q_k(x)$ คือ ความเป็นไปได้ที่ข้อมูลกลุ่ม k อยู่ในขอบเขต q คำนวณได้ดังสมการ (2.16) ถึง (2.19) ตามลำดับ

$$p_k(x) = \frac{n_k(x)+1}{n(x)+1} \quad (2.16)$$

$$p(x) = \frac{n(x)+1}{n+1} \quad (2.17)$$

$$q_k(x) = \frac{m_k(x)+1}{m(x)+1} \quad (2.18)$$

$$q(x) = \frac{m(x)+1}{n+1} \quad (2.19)$$

โดยที่ $n_k(x)$ คือ จำนวนข้อมูลกลุ่ม k ที่อยู่ในขอบเขต p, $m_k(x)$ คือ จำนวนข้อมูลกลุ่ม k ที่อยู่ในขอบเขต q, $n(x)$ คือ จำนวนข้อมูลทั้งหมดที่อยู่ในขอบเขต p, $m(x)$ คือ จำนวนข้อมูลทั้งหมดที่อยู่ในขอบเขต q, และ n คือ จำนวนข้อมูลทั้งหมดที่อยู่ในขอบเขต p และขอบเขต q รวมกัน

2.3 ฟัซซี่เซต (Fuzzy Set)

ฟัซซี่เซตเป็นเซตที่แสดงถึงความสัมพันธ์ของสมาชิกภายในกลุ่มแต่ละตัวกับคำจำกัดความของเซตนั้น ๆ โดยความสัมพันธ์นี้จะถูกแสดงอยู่ในลักษณะของระดับความเป็นสมาชิกที่มีค่าอยู่ในช่วง [0, 1] แทนที่จะแสดงว่าสมาชิกตัวใดเป็นสมาชิกหรือไม่เป็นสมาชิกของเซตนั้นอย่างเซตธรรมดา ที่มีค่าเป็น {0, 1} [22]

ตัวแปรฟัซซี่เซต (Fuzzy Variable) หรือเรียกว่าตัวแปร Linguistic ซึ่งเป็นฟัซซี่เซตใด ๆ ของระบบที่เราสนใจ เช่น ถ้าพิจารณา “ความกว้างของกลีบดอกไม้” ซึ่งเป็นตัวแปรฟัซซี่ และถ้าค่าของตัวแปรของเซตนี้ คือ {เล็ก, กลาง, ใหญ่} ซึ่งจะเรียกค่าของตัวแปรนี้ว่า เทอมเซต (Term Set) ซึ่งในส่วนถัดไปจะกล่าวถึงการแทนข้อมูลด้วยฟัซซี่เซต และฟังก์ชันความเป็นสมาชิก

2.3.1 การแทนข้อมูลด้วยฟัซซี่เซต

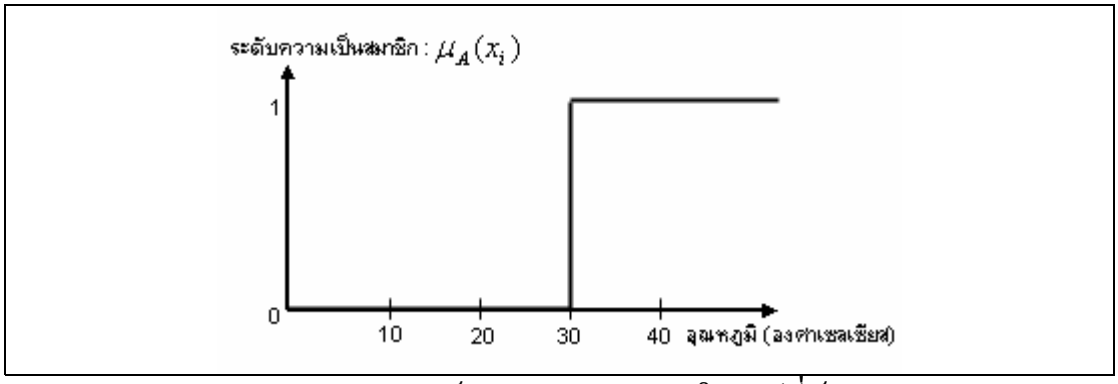
เพื่อให้สามารถเข้าใจความหมายของฟัซซี่เซตได้ชัดเจน จะทำการเปรียบเทียบการแทนข้อมูลของฟัซซี่เซตกับเซตธรรมดา สำหรับเซตธรรมดา การแปลงข้อมูลเพื่อคำนวณหาค่าความเป็นสมาชิกนั้นจะต้องมีการกำหนดค่าเทรชโฮลด์ (Threshold) เพื่อตัดสินว่าสมาชิกตัวใดอยู่ในเซตที่เรากำลังพิจารณาอยู่บ้าง

ตัวอย่างเช่น กำหนดให้เซต A คือ เซตอุณหภูมิของประเทศหนึ่งอยู่ในช่วง 2 – 40 องศาเซลเซียส และ x_i คือ อุณหภูมิของบริเวณที่ i

ในกรณีของเซตธรรมดาจะต้องมีการกำหนดค่าเทรชโฮลด์ (Threshold) ซึ่งในที่นี้กำหนดให้เท่ากับ 30 องศาเซลเซียส คือ ถ้าอุณหภูมิมีค่ามากกว่าหรือเท่ากับ 30 องศาเซลเซียส แสดงว่าอุณหภูมิในบริเวณนั้นร้อน (เป็นสมาชิกของเซต A) ซึ่งสามารถเขียนเป็นฟังก์ชันคุณลักษณะหรือฟังก์ชันความเป็นสมาชิก ได้ดังสมการ (2.20)

$$\mu_A(x_i) = \{x_i \geq 30\} \tag{2.20}$$

จากสมการข้างต้นสามารถนำมาเขียนเป็นกราฟของความเป็นสมาชิกได้ดังภาพประกอบ 2.7



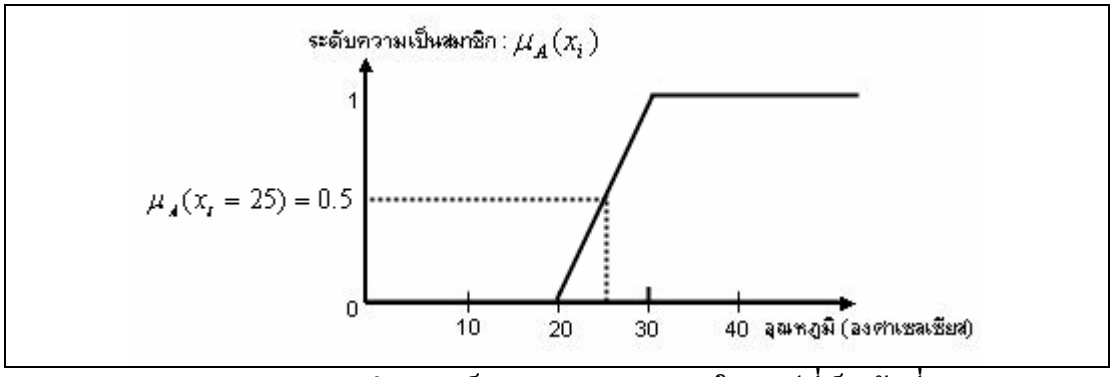
ภาพประกอบ 2.7 ค่าความเป็นสมาชิกของเซต A ในกรณีที่เป็นเซตธรรมดา

จากภาพประกอบ 2.7 แกน x คือ อุณหภูมิ (องศาเซลเซียส) และแกน y คือ ค่าระดับความเป็นสมาชิกที่มีค่าเป็น 0 (ไม่เป็นสมาชิกของเซต A) และ 1 (เป็นสมาชิกของเซต A) จะเห็นได้ว่าอุณหภูมิที่มากกว่าหรือเท่ากับ 30 องศาเซลเซียส ถือว่าเป็นอุณหภูมิที่สูง

ในกรณีของฟัซซี่เซต ฟังก์ชันความเป็นสมาชิกจะมีความยืดหยุ่นมากกว่าและสอดคล้องกับความเป็นจริงมากกว่าเซตธรรมดา ดังสมการ (2.21)

$$\mu_A(x_i) = \begin{cases} \dfrac{1}{(30-20)}(x_i - 20); 20 \leq x_i \leq 30 \\ 1 \ ; \ x_i > 30 \\ 0 \ ; \ \text{กรณีอื่นๆ} \end{cases} \tag{2.21}$$

จากสมการ (2.21) เป็นการกำหนดค่าความเป็นสมาชิกให้กับเทอมเซตของอุณหภูมิสูง และภาพประกอบ 2.8 แสดงกราฟระดับความเป็นสมาชิกของสมการ (2.21) ดังนี้



ภาพประกอบ 2.8 ค่าความเป็นสมาชิกของเซต A ในกรณีที่เป็นฟัซซี่เซต

เมื่อเปรียบเทียบฟังก์ชันความเป็นสมาชิกจากภาพประกอบ 2.7 กับภาพประกอบ 2.8 จะเห็นว่ากราฟที่แสดงความเป็นสมาชิกของฟัซซี่เซตจะมีความต่อเนื่องในค่าระดับความเป็นสมาชิกในช่วง [0, 1] เช่น อุณหภูมิที่บริเวณหนึ่งเป็น 25 องศาเซลเซียส แสดงว่าเป็นอุณหภูมิที่สูงเช่นกัน แต่จะมีระดับความเป็นสมาชิกของเซตอุณหภูมิสูง คือ 0.5 แต่ถ้าเป็นเซตธรรมดาแล้ว อุณหภูมิ 25 องศาเซลเซียส จะไม่เป็นสมาชิกของเซต A หรือแม้แต่อุณหภูมิ 29 องศาเซลเซียส ก็จะถือว่าเป็นอุณหภูมิสูงเช่นกัน ซึ่งไม่ตรงกับความเป็นจริง แต่สำหรับการแทนข้อมูลด้วยฟัซซี่เซตจะถือว่าอุณหภูมิ 29 องศาเซลเซียสเป็นเซตของอุณหภูมิสูงด้วยค่าความเป็นสมาชิกเท่ากับ 0.9

2.3.2 ฟังก์ชันความเป็นสมาชิก (Membership Function)

ฟังก์ชันความเป็นสมาชิกแบบต่าง ๆ ที่ใช้ในการประมาณค่าระดับความเป็นสมาชิกของแต่ละเทอมเซตแบบประมาณค่าเชิงเส้น เช่น รูปสามเหลี่ยม, รูปสี่เหลี่ยมคางหมู ซึ่งมีรายละเอียด ดังนี้

2.3.2.1 รูปสามเหลี่ยม

การกำหนดฟังก์ชันความเป็นสมาชิกโดยใช้รูปสามเหลี่ยม เทอมเซตที่กำหนดโดยฟังก์ชันนี้จะต้องมีค่าที่เหมาะสมที่สุดอยู่เพียงค่าเดียวที่ทำให้ค่าความเป็นสมาชิกเท่ากับ 1 ส่วนค่าอื่น ๆ จะมีค่าความเป็นสมาชิกลดลงเรื่อย ๆ ถ้ากำหนดให้ $a \le b \le c$ เมื่อ a, b, และ c เป็นเลขจำนวนจริงใด ๆ ดังนั้นฟังก์ชันความเป็นสมาชิกของรูปสามเหลี่ยมสามารถกำหนดได้ดังสมการ (2.22)

$$\mu_A(x_i) = \begin{cases} \frac{1}{b-a}(x_i - a); a \le x_i \le b, a \ne b \\ \frac{1}{c-b}(c - x_i); b < x_i \le c, b \ne c \\ 0 \ ; \ \text{กรณีอื่นๆ} \end{cases} \quad (2.22)$$

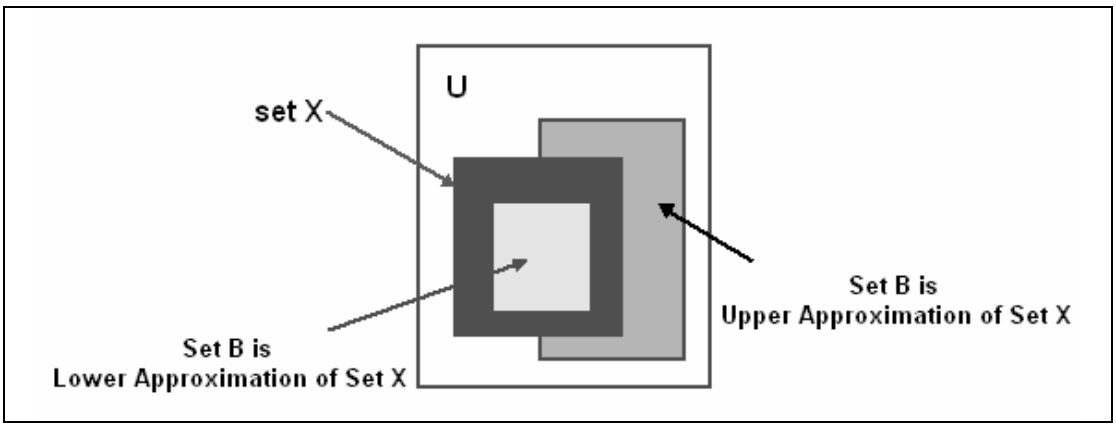
2.3.2.2 รูปสี่เหลี่ยมคางหมู

การกำหนดฟังก์ชันความเป็นสมาชิกโดยใช้รูปสี่เหลี่ยมคางหมู เทอมเซตที่กำหนดโดยฟังก์ชันนี้จะต้องมีช่วงค่าที่เหมาะสมมากที่สุดอยู่กลุ่มหนึ่งที่ทำให้ค่าความเป็นสมาชิกของเทอมเซตนั้นๆ มีค่าเท่ากับ 1 ส่วนค่าอื่นๆ จะมีค่าความเป็นสมาชิกลดลงเรื่อยๆ ถ้ากำหนดให้ $a \le b \le c \le d$ เมื่อ a, b, c และ d เป็นเลขจำนวนจริงใดๆ ดังนั้นฟังก์ชันความเป็นสมาชิกของรูปสี่เหลี่ยมคางหมูสามารถกำหนดได้ดังสมการ (2.23)

$$\mu_A(x_i) = \begin{cases} \frac{1}{b-a}(v_r - a); a \le v_r \le b, a \ne b \\ 1 \ ; \ b < x_i \le c \\ \frac{1}{d-c}(d - v_r); c < v_r \le d, c \ne d \\ 0 \ ; \ \text{กรณีอื่นๆ} \end{cases} \quad (2.23)$$

2.4 ราฟเซต (Rough Set)

ราฟเซตเป็นวิธีการที่ใช้จัดการเรื่องความคลุมเครือและความไม่แน่นอนของข้อมูล โดยมีการนำเรื่องการประมาณค่าขอบเขตล่าง (Lower Approximation) และการประมาณค่าขอบเขตบน (Upper Approximation) มาประยุกต์ใช้ในการจัดการเรื่องความไม่แน่นอน [25] แนวคิดของราฟเซตอธิบายได้ดังภาพประกอบ 2.9



ภาพประกอบ 2.9 แนวคิดของราฟเซต

กำหนดให้ U เป็นเซตอนันต์ X เป็นสับเซตของเซตอนันต์ การประมาณค่าขอบเขตล่าง (Lower Approximation) หมายถึง สมาชิกทุกตัวของเซต B อยู่ในเซต X ดังสมการ (2.24)

$$\underline{B}X = \{x \in U : B(x) \subseteq X\} \tag{2.24}$$

การประมาณค่าขอบเขตบน (Upper Approximation) หมายถึง สมาชิกบางตัวของเซต B อยู่ในเซต X ดังสมการ (2.25)

$$\overline{B}X = \{x \in U : B(x) \cap X \neq \phi\} \tag{2.25}$$