การจัดกลุ่มโหนดใน Self-Organizing Map โดยใช้เจเนติกอัลกอริธึม

NODE CLUSTERING IN SELF-ORGANIZING MAP USING GENETIC ALGORITHM

กสานติ์ ศรีกุลนาถ

KASAN SRIKULNATH

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าธนบุรี

พ.ศ. 2549

ISBN 974-15-2643-1

การจัดกลุ่มโหนดใน Self-Organizing Map โดยใช้เจเนติกอัลกอริธึม

NODE CLUSTERING IN SELF-ORGANIZING MAP USING GENETIC ALGORITHM

กสานติ์ ศรีกุลนาถ

KASAN SRIKULNATH

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2549

ISBN 974-15-2643-1

NODE CLUSTERING IN SELF-ORGANIZING MAP USING GENETIC ALGORITHM

KASAN SRIKULNATH

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN COMPUTER ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

ISBN 974-15-2643-1

หัวข้อวิทยานิพนธ์	การจัดกลุ่มโหนดใน Self-Organizing Map โดยใช้เจเนติกอัลกอริธึม
นักศึกษา	นายกสานติ์ ศรีกุลนาถ
รหัสนักศึกษา	44061633
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
พ.ศ.	2549
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร.เอื้อน ปิ่นเงิน

บทคัดย่อ

การจัดกลุ่มข้อมูลโดยใช้ Self-Organizing Map (SOM) ข้อมูลจะถูกจัดให้อยู่ในรูปแบบของแผนภาพ 2 มิติ จุดเด่นคือกลุ่มของข้อมูลที่มีลักษณะคล้ายกันจะอยู่ในโหนดใกล้ที่เคียงกัน แต่ในกรณีที่แผนภาพมีขนาดใหญ่ข้อมูลที่อยู่ในกลุ่มเดียวกันอาจจะแตกออกเป็นกลุ่มย่อยอยู่ในโหนดที่ห่างออกไป ทำให้ไม่สามารถระบุกลุ่มได้อย่างชัดเจนและทำให้การสำรวจแผนภาพเป็นไปด้วยความยากลำบาก ดังนั้นจำเป็นต้องจัดกลุ่มของโหนดในแผนภาพ SOM หลังจากเสร็จสิ้นกระบวนการเรียนรู้ งานวิจัยนี้นำเสนอการจัดกลุ่มโดยใช้เจเนติกอัลกอริธึมกับแผนภาพ SOM เพื่อแก้ไขปัญหาการกระจายของข้อมูล โดยแบ่งการทำงานออกเป็น 2 ขั้นตอนหลักคือ ขั้นตอนแรกเป็นการจัดกลุ่มข้อมูลโดยใช้แผนภาพ SOM ขั้นตอนที่สองเป็นการจัดกลุ่มโหนดของแผนภาพ SOM โดยใช้เจเนติกอัลกอริธึม ในการทดลองกับข้อมูล KDD cup 1999 ซึ่งเป็นข้อมูลรูปแบบพฤติกรรมการบุกรุกเครือข่าย และเปรียบเทียบกับการจัดกลุ่มโหนดโดยใช้วิธี K-mean ผลปรากฏว่าการจัดกลุ่มโดยใช้เจเนติกอัลกอริธึมที่นำเสนอนั้นให้ประสิทธิภาพในการจัดกลุ่มที่ดีกว่า โดยวัดประสิทธิภาพจากค่าเอนโทรพี

Thesis Title	Node Clustering in Self-Organizing Map using Genetic Algorithm
Student	Mr. Kasan Srikulnath
Student ID.	44061633
Degree	Master of Engineering
Programme	Computer Engineering
Year	2006
Thesis Advisor	Assoc. Prof. Dr. Ouen Pinngern

ABSTRACT

Data clustering using self-organizing map (SOM) is represented as a two dimensional map. The advantage of this method is that similar feature of data are clustered into the neighbor node. In case of a large map, the SOM may separate those data into sub-groups that are in the other nodes. It is difficult to identify the appropriate groups. Therefore, after finished the training process it is necessary to cluster the node in SOM's map again. In this research we present a clustering method using genetic algorithm in SOM's map. Our proposed algorithm has two processes. First, cluster data using SOM and second, cluster nodes in SOM using genetic algorithm. In our experiments, we applied SOM to KDD cup 1999 dataset. Then genetic algorithm and K-Mean were separately applied to the results from SOM. The final results of genetic algorithm yields a better performance than that of K-mean based on Entropy value.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก รศ.ดร.เอื้อน ปิ่นเงิน ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ รศ.ดร.บุญธีร์ เครือตราชู เป็นที่ปรึกษาด้านอัลกอริธึม ข้าพเจ้ารู้สึกทราบซึ้งในความอนุเคราะห์จากท่านอาจารย์ทั้งสองท่าน และขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณองค์กร JICA และสำนักวิจัยการสื่อสารและเทคโนโลยีสารสนเทศ (ReCCIT) ที่ได้สนับสนุนเครื่องมือ ตลอดจนข้อมูล และหนังสือต่างๆ ที่ใช้ในการทำวิจัย

ขอขอบคุณ นายพรเทพ โรจนวสุ นายไพฑูรย์ ศรีนิล นักศึกษาปริญญาเอก นายณรงค์ชัย มุ่งแฝงกลาง นักศึกษาปริญญาโท ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ เพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัย คณะวิศวกรรมศาสตร์ที่ให้ความช่วยเหลือในเรื่องต่างๆ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

กสานติ์ ศรีกุลนาถ

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญรูป

สารบัญรูป (ต่อ)

สารบัญรูป (ต่อ)

รูปที่ หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

Self-Organizing Map (SOM) [1] นิยมนำมาใช้ในการจัดกลุ่มข้อมูลที่มีหลายมิติ และผลลัพธ์จาก SOM ที่ได้จะแสดงอยู่ในรูปของแผนภาพ 2 มิติ โดยทั่วไปแล้วข้อมูลที่มีลักษณะคล้ายคลึงกันจะอยู่ในโหนดที่ใกล้เคียงกัน ซึ่งเป็นคุณลักษณะเด่นที่ทำให้ SOM ถูกนำมาใช้กันอย่างกว้างขวางในงานหลาย ๆ ด้าน [2]

อย่างไรก็ตามในกรณีที่แผนภาพมีขนาดใหญ่นั้นทำให้การเลือกดูข้อมูลทำได้ค่อนข้างยาก เนื่องจากไม่ทราบขอบเขตของกลุ่มข้อมูลที่แน่นอน และในบางกรณีข้อมูลที่อยู่ในกลุ่มเดียวกันอาจจะแยกออกเป็นกลุ่มย่อยอยู่ในโหนดที่ห่างออกไป ซึ่งจะทำให้เกิดปัญหาในการเลือกดูข้อมูลที่ต้องการและไม่สามารถระบุกลุ่มของข้อมูลได้ จึงได้มีการนำเสนอวิธีการแก้ปัญหาการจัดกลุ่มโหนดของแผนภาพ SOM โดยใช้ การจัดกลุ่มแบบ K-mean [3] แต่วิธีการดังกล่าวยังไม่สามารถจัดกลุ่มข้อมูลที่กระจายห่างไกลออกไปได้ หลังจากนั้นได้มีการนำเสนอการจัดกลุ่มข้อมูลโดยใช้เจเนติกอัลกอริธึม [4] ที่สามารถช่วยจัดการกลุ่มข้อมูลที่ไม่เป็นรูปแบบและกระจายกันได้เป็นอย่างดี

ในวิทยานิพนธ์นี้นำเสนอการจัดกลุ่มโหนดของแผนภาพ SOM โดยใช้การจัดกลุ่มด้วยวิธี เจเนติกอัลกอริธึมเพื่อแก้ไขปัญหาการกระจายของกลุ่มข้อมูล จุดอ่อนของเจเนติกอัลกอริธึมคือใช้เวลาในการทำงานนานกว่าวิธี K-mean แต่การเข้ารหัสของโครโมโซมให้มีความเหมาะสมกับจำนวนข้อมูลของอินพุทจะช่วยลดเวลาการทำงานของเจเนติกอัลกอริธึมได้ นอกจากนี้การเกิดคำตอบที่ไม่เป็นคำตอบที่ดีที่สุด (Local Optimal) ทำให้ผลของการจัดกลุ่มโหนดผิดพลาดได้ ซึ่งปัญหาเหล่านี้จะสามารถแก้ได้โดยการเก็บผลลัพธ์ที่ดีที่สุดไว้และนำผลลัพธ์ที่เก็บไว้ไปเปรียบเทียบกับผลลัพธ์ในเจเนอเรชั่นส์ถัดไป

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วัตถุประสงค์ของวิทยานิพนธ์เพื่อศึกษาวิธีการทำงานของ SOM ก่อนและหลังกระบวนการเรียนรู้ ในกรณีข้อมูลที่ใช้ในการทดลองที่มีความซับซ้อน ซึ่งพบว่าผลที่ได้ในการจัดกลุ่มข้อมูลจากแผนภาพ SOM หลังการเรียนรู้กลับไม่สมบูรณ์ ดังนั้นเพื่อให้ได้คำตอบที่ถูกต้อง จึงมีแนวคิดในการแก้ปัญหาดังกล่าวด้วยเจเนติกอัลกอริธึม โดยการเพิ่มเลเยอร์ของผลลัพธ์จากแผนภาพ SOM อีกหนึ่งเลเยอร์ เลเยอร์ใหม่ที่เพิ่มขึ้นเกิดจากการจัดกลุ่มโหนดบนแผนภาพ SOM โดยใช้เจเนติกอัลกอริธึม และได้เพิ่มประสิทธิภาพในการค้นหาคำตอบและความเร็วในการหาคำตอบ ทำให้

ผลลัพธ์ที่ได้จากเจเนติกอัลกอริธึมนี้ได้จำนวนกลุ่มที่ถูกต้องมากกว่าเมื่อเปรียบเทียบกับผลลัพธ์ของการจัดกลุ่มด้วยวิธีอื่นๆ

1.3 สมมติฐานของการศึกษา

ปัจจุบันข้อมูลโดยทั่วไปมีความซับซ้อนมากขึ้น ดังนั้นจึงมีความพยายามในการแยกข้อมูลออกเป็นหมวดหมู่เพื่อให้สามารถวิเคราะห์ข้อมูลได้ง่ายขึ้น ข้อมูลส่วนใหญ่จะมีหลายมิติหรือหลายคุณลักษณะ (Attribute) ในบางปัญหาไม่สามารถตัดบางคุณลักษณะของข้อมูลออกไปได้ จึงส่งผลให้โอกาสข้อมูลชุดหนึ่งที่มีลักษณะใกล้เคียงหรือคล้ายคลึงกันกระจายห่างออกไป ใกล้เคียงกับกลุ่มอื่นๆ แทนที่จะอยู่บริเวณกลุ่มข้อมูลที่มีลักษณะใกล้เคียงกัน จึงทำให้การจัดกลุ่มข้อมูลโดยใช้ Self-Organizing Map มีแผนภาพหลังกระบวนการเรียนรู้ไม่สมบูรณ์ จึงจำเป็นต้องมีการจัดกลุ่มโหนดจากแผนภาพอีกครั้งหนึ่ง

1.4 แนวคิดที่ใช้ในการวิจัย

วิทยานิพนธ์นี้นำเสนอการจัดกลุ่มโหนดจากแผนภาพ SOM หลังกระบวนการเรียนรู้ด้วยการประยุกต์ใช้เจเนติกอัลกอริธึม โดยการจัดกลุ่มโหนดที่เป็นชนิดเดียวกันแต่กลับกระจายอยู่บนแผนภาพ SOM ซึ่งกำหนดวิธีองค์ประกอบของการจัดกลุ่มจาก Intra Class และ Inter Class ลงในฟังก์ชันความเหมาะสมในกระบวนการของเจเนติกอัลกอริธึม และทำการหาเซตของคำตอบที่เกิดขึ้นจากการสร้างเลเยอร์ของเจเนติก (Genetic Layer) หลังจากนั้นได้ทำการปรับความเร็วของเจเนติกอัลกอริธึมให้มีประสิทธิภาพมากยิ่งขึ้น

ในวิทยานิพนธ์นี้ได้นำเสนอแนวคิดการจัดกลุ่มโหนดบนแผนภาพ SOM ด้วยเจเนติกอัลกอริธึม ซึ่งศึกษาแนวคิดจากงานวิจัยดังนี้

- T. Kohonen [1] นำเสนอ Self-Organizing Map เป็นกระบวนการเรียนรู้ โดยไม่อาศัยผู้สอน เหมาะสำหรับการจัดกลุ่มข้อมูลที่มีจำนวนมากๆ เพื่อให้สามารถเรียกดูและวิเคราะห์กลุ่มข้อมูลในรูปแบบ 2 มิติ
- J.Vesanto และ E.Alhoniemi [3] เป็นงานวิจัยที่การจัดกลุ่มสองระดับ โดยให้จัดกลุ่มข้อมูลในขั้นแรกด้วย SOM และจัดกลุ่มข้อมูลอีกครั้งจากข้อมูลของ SOM ข้อดีที่สำคัญของการจัดกลุ่มด้วยการเพิ่มวิธีการจัดกลุ่มให้หลากหลายวิธี โดยเอาข้อดีของแต่ละวิธีการจัดกลุ่มมาปรับปรุงความเร็วและประสิทธิภาพให้ดีขึ้น และทำการเปรียบเทียบจากการจัดกลุ่มข้อมูลด้วยวิธีเดียวและการจัดกลุ่มหลังจากมีการจัดกลุ่มของ SOM ไว้แล้ว

- L. Y. Tseng และ S. B. Yang [4] ได้ศึกษาวิธีการจัดกลุ่มที่เกิดจากข้อมูลที่มีลักษณะไม่เป็นรูปแบบ ซึ่งจะสามารถจัดกลุ่มข้อมูลได้อย่างอัตโนมัติ

1.5 การเปรียบเทียบระหว่างวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน

ในขั้นตอนการทดลองเพื่อเปรียบเทียบวัดประสิทธิภาพได้ทำการแบ่งการทดลองออกเป็นสองส่วน ดังนี้

ส่วนแรกเป็นการทดลองจัดกลุ่มข้อมูลจากข้อมูล Clown มีขนาดสองมิติและชุดข้อมูลการบุกรุกเครือข่ายมีขนาดหลายมิติที่ผ่านการ Normalize แล้ว หลังจากนั้นทำการจัดกลุ่มข้อมูลด้วยวิธีต่างๆ ดังนี้

1. K-mean
2. Hierarchical

ส่วนที่สองเป็นการจัดกลุ่มโหนดหลังกระบวนการเรียนรู้จากวิธี SOM จากข้อมูล Clown มีขนาดสองมิติและชุดข้อมูลการบุกรุกเครือข่ายมีขนาดหลายมิติที่ผ่านการ Normalize แล้วด้วย

1. เจเนติกอัลกอริธึม
2. K-mean
3. Hierarchical

โดยทั้งสองส่วนจะทำการวัดประสิทธิภาพจากค่าเอนโทรพี (Entropy) ซึ่งเป็นการวัดว่ามีสมาชิกหรือข้อมูลอื่นมาปะปนอยู่ภายในกลุ่มมากน้อยเพียงใด และเปรียบเทียบความเร็วในการจัดกลุ่มข้อมูลระหว่างการทดลองทั้งสองส่วน

1.6 ขอบเขตการวิจัย

การวิจัยนี้เน้นศึกษาในเรื่องของการจัดกลุ่มข้อมูลที่มีขนาดใหญ่และจำนวนคุณลักษณะเฉพาะของข้อมูลหลายมิติ โดยมีขั้นตอนการศึกษาดังนี้

1. ศึกษาทฤษฎีตลอดจนขั้นตอนทำงานของ SOM
2. ศึกษาปัญหาที่เกิดภายหลังการเรียนรู้จากแผนภาพ SOM
3. จัดกลุ่มโหนดของแผนภาพ SOM เพื่อให้ได้คำตอบที่สมบูรณ์มากขึ้น
4. เปรียบเทียบวิธีพื้นฐานด้วยการวัดค่าเอนโทรพีและเวลาที่ใช้ในการจัดกลุ่มโหนด

1.7 ขั้นตอนของการศึกษา

1.7.1 ศึกษาทฤษฎีและแนวคิดจากบทความต่างๆ ที่เกี่ยวข้องกับงานวิจัย

1.7.2 ศึกษาการทำงาน SOM

1.7.3 ศึกษาปัญหาการจัดกลุ่มข้อมูลด้วย SOM

1.7.4 ทดลองการจัดกลุ่มจากชุดข้อมูล Clown ขนาดสองมิติและชุดข้อมูลรูปแบบการบุกรุกเครือข่ายขนาดหลายมิติ (Intrusion Detection System) [6] ด้วย SOM

1.7.5 สังเกตผลลัพธ์หลังกระบวนการเรียนรู้จาก SOM

1.7.6 จัดกลุ่มข้อมูลจากแผนภาพ SOM หลังกระบวนการเรียนรู้ด้วยเจเนติกอัลกอริธึม

1.7.7 เขียนโปรแกรมเพื่อทำการจัดกลุ่มข้อมูลจากแผนภาพ SOM

1.7.8 เปรียบเทียบการจัดกลุ่มข้อมูลด้วย SOM, K-Mean และ Hierarchical

1.7.9 เปรียบเทียบผลลัพธ์จากการจัดกลุ่มแผนภาพ SOM ด้วยเจเนติกอัลกอริธึมกับ K-Mean และ Hierarchical

1.7.10 วิเคราะห์และสรุปผลการดำเนินการ

1.7.11 จัดทำเอกสารประกอบวิทยานิพนธ์

1.8 รายละเอียดในแต่ละบท

วิทยานิพนธ์ฉบับนี้ แบ่งเนื้อหาออกเป็น 6 บทคือ

บทที่ 1 กล่าวถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของการวิจัย และขั้นตอนการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการวิจัย และแนวคิดการจัดกลุ่มข้อมูล

บทที่ 3 เป็นการศึกษาพื้นฐานการเรียนรู้ Self-Organizing Map

บทที่ 4 กล่าวถึงหลักการทำงานในงานวิจัยนี้ โดยอธิบายปัญหาหลังกระบวนการเรียนรู้ของ SOM และการแก้ปัญหาด้วยเจเนติกอัลกอริธึม

บทที่ 5 การทดลองจากชุดข้อมูลต่างๆ และผลการทดลองของการจัดกลุ่มโหนด

บทที่ 6 บทสรุปผลการวิจัยและข้อเสนอแนะ

เอกสารอ้างอิง

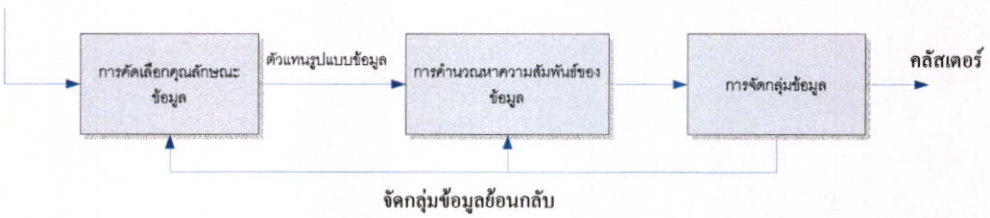
ภาคผนวกและงานวิจัยที่ได้รับการตีพิมพ์

บทที่ 2
ทฤษฎีพื้นฐานและแนวคิดการจัดกลุ่มข้อมูลที่เกี่ยวข้อง

2.1 บทนำ

การจัดกลุ่มข้อมูล เป็นวิธีการรวมกลุ่มของข้อมูลตามคุณลักษณะข้อมูลที่คล้ายกันหรือรูปแบบที่เหมือนกัน โดยข้อมูลมีคุณลักษณะที่คล้ายกันจะถูกจัดให้อยู่กลุ่มเดียวกัน การจัดกลุ่มข้อมูลสามารถแยกประเภทของการจัดกลุ่มได้เป็น การจัดหมวดหมู่ (Classification) และการรวมตัว (Clustering)

โดยการจัดหมวดหมู่เป็นการจำแนกประเภทของการจัดกลุ่มแบบมีผู้สอน (Supervised Classification) ซึ่งกลุ่มข้อมูลจะถูกกำหนดไว้ล่วงหน้าแล้ว จากนั้นข้อมูลจะถูกจัดอยู่ในกลุ่มที่กำหนดไว้ ส่วนการรวมตัว เป็นการจำแนกประเภทของการจัดกลุ่มแบบไม่มีผู้สอน (Unsupervised Classification) คือไม่มีข้อมูลใดๆ ถูกจัดกลุ่มไว้ก่อนแล้ว การจัดกลุ่มข้อมูล[1] มีขั้นตอนการทำงานที่สำคัญดังรูปที่ 2.1



รูปที่ 2.1 แสดงขั้นตอนการจัดกลุ่มข้อมูล

จากรูปที่ 2.1 ขั้นตอนการจัดกลุ่มข้อมูลเริ่มต้นเมื่อมีรูปแบบของข้อมูล (Pattern) จำนวนหนึ่ง แล้วนำข้อมูลเหล่านั้นมาเลือกคุณลักษณะ (Feature) ที่สามารถแยกข้อมูลออกจากกันได้เพื่อเป็นตัวแทนของแต่ละข้อมูล (Feature Selection/Extraction) ซึ่งในการแทนรูปแบบของข้อมูลนั้นมีความแตกต่างกันออกไปขึ้นอยู่กับการนำไปใช้ในอัลกอริธึม หลังจากที่ทราบคุณลักษณะของแต่ละข้อมูลแล้ว การดำเนินการขั้นตอนต่อไปของการจัดกลุ่มข้อมูล คือการคำนวณหาความสัมพันธ์ระหว่างข้อมูล (Interpattern Similarity) โดยทั่วไปแล้วใช้ฟังก์ชันการวัดระยะห่างระหว่างข้อมูลซึ่งมีอยู่หลายวิธี ตัวอย่างของฟังก์ชันการวัดระยะห่างข้อมูลคือ Euclidean distance เมื่อหาค่า

---

[1] คำว่าการจัดกลุ่มข้อมูลในวิทยานิพนธ์นี้ ใช้แทนการจัดกลุ่มแบบรวมตัวหรือคลัสเตอร์

ความสัมพันธ์ระหว่างข้อมูลได้แล้วก็เข้าสู่ขั้นตอนการจัดกลุ่มข้อมูล (Grouping) โดยที่ข้อมูลที่มีระยะความสัมพันธ์ห่างน้อยๆ กับข้อมูลที่ต้องการอ้างอิง จะอยู่กลุ่มเดียวกัน ส่วนข้อมูลที่มีค่าระยะความสัมพันธ์ห่างกันมากก็จะอยู่ต่างกลุ่มกัน จากนั้นสุดท้ายข้อมูลจะถูกแยกออกเป็นกลุ่มๆ หรือคลัสเตอร์ นอกจากนี้ในขั้นตอนของการจัดกลุ่มข้อมูลสามารถย้อนกลับ (Feedback Loop) ไปยังขั้นตอนของการเลือกคุณลักษณะและการหาค่าความสัมพันธ์ระหว่างข้อมูลได้ในลักษณะวนรอบอีกได้ด้วย ทั้งนี้ขึ้นอยู่กับอัลกอริธึมที่นำมาใช้งานในการจัดกลุ่มข้อมูล

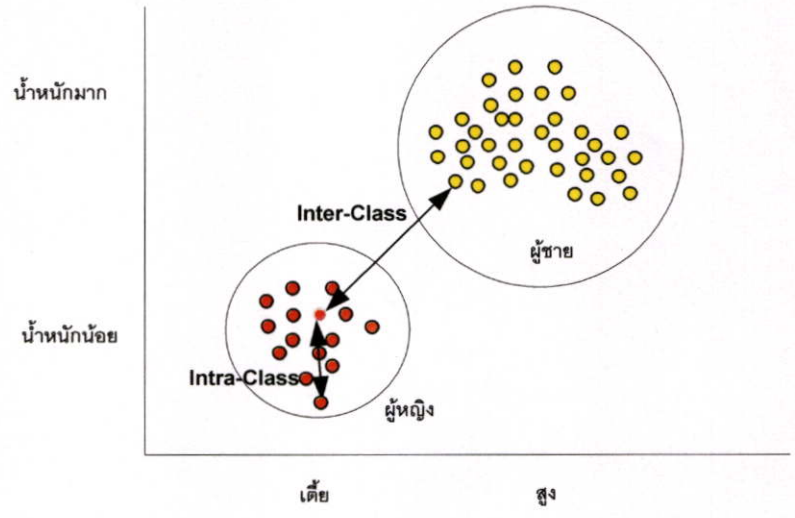
2.2 คุณลักษณะและองค์ประกอบของการจัดกลุ่ม

องค์ประกอบของการจัดกลุ่มข้อมูลที่ใช้ในวิทยานิพนธ์นี้ ได้กำหนดการแบ่งคลาสความสัมพันธ์ของแต่ละข้อมูลดังนี้

Intra-Class เป็นการหาความสัมพันธ์ระหว่างข้อมูลภายในกลุ่มเดียวกัน เช่น จากข้อมูลคุณลักษณะของชายและหญิง ข้อมูล A อยู่ในกลุ่มผู้หญิง การหาความสัมพันธ์ของ *Intra Class* สามารถหาได้จากความสัมพันธ์ของข้อมูล A กับทุกๆ ข้อมูลอื่นที่อยู่ภายในกลุ่มผู้หญิง

Inter-Class เป็นการหาความสัมพันธ์ระหว่างข้อมูลของกลุ่มหนึ่งกับข้อมูลกลุ่มอื่นๆ เช่น การหาความสัมพันธ์ระหว่างข้อมูล A ที่อยู่ในกลุ่มผู้หญิงกับทุกๆ ข้อมูลที่อยู่ในกลุ่มผู้ชาย การหาความสัมพันธ์นี้จะเป็นตัวกำหนดความชัดเจนว่าข้อมูลนั้นๆ ควรจัดให้อยู่กลุ่มใด

วิธีการจัดกลุ่มที่ดี ควรจะมี***ค่าของของ Intra class มากๆ และมีค่าของ Inter class น้อยๆ*** ดังรูปที่ 2.2 แสดงองค์ประกอบของการจัดกลุ่มในวิทยานิพนธ์



รูปที่ 2.2 แสดงองค์ประกอบของการจัดกลุ่ม

เทคนิคการหาความสัมพันธ์ของการจัดกลุ่มมักจะนิยมใช้ระยะยูคลิเดียน (Euclidean Distance) แต่การหาความสัมพันธ์ยังมีเทคนิคหลากหลาย ซึ่งการเลือกใช้เทคนิคการหาความสัมพันธ์ ควรพิจารณาจากลักษณะรูปร่างของข้อมูล เช่น การกระจายของข้อมูล บางครั้งเราอาจไม่ทราบการกระจายข้อมูล แต่จะใช้การทดลองเพื่อความเหมาะสมกับรูปแบบของข้อมูลซึ่งสามารถสรุปเทคนิคการหาความสัมพันธ์ [4]ได้ดังนี้

รูปแบบการหาระยะห่างความสัมพันธ์ภายในกลุ่ม $Intra(Q_k)$ แบบต่างๆ

Average Distance

$$Intra_{av} = \frac{\sum_{i,i'} \|x_i - x_{i'}\|}{N_k(N_k - 1)} \tag{2.1}$$

Nearest Neighbor Distance

$$Intra_{nn} = \frac{\sum_{i} \min_{i'} \{\|x_i - x_{i'}\|\}}{N_k} \tag{2.2}$$

Centroid Distance

$$Intra_{c} = \frac{\sum_{i} \|x_i - c_k\|}{N_k} \tag{2.3}$$

เมื่อ x_i แทนข้อมูลที่อยู่ภายในกลุ่ม

$x_{i'}$ แทนข้อมูลอื่นที่อยู่ภายในกลุ่ม

N_k แทนจำนวนข้อมูลทั้งหมดที่อยู่ในกลุ่ม

c_k แทนจุดศูนย์กลางจากสมาชิกทั้งหมดภายในกลุ่มสามารถหาได้จาก

$$c_k = \frac{1}{N_k} \sum_{x_i \in Q_k} x_i \tag{2.4}$$

Q_k แทนกลุ่มข้อมูล

k แทนลำดับกลุ่มข้อมูล

รูปแบบการหาระยะห่างความสัมพันธ์ระหว่างกลุ่มข้อมูล $Inter(Q_k, Q_l)$ ที่สำคัญมีดังนี้

Single Linkage

$$Inter_{sl} = \min_{i,j} \{\|x_i - x_j\|\} \tag{2.5}$$

Complete Linkage

$$Inter_{co} = \max_{i,j} \{\|x_i - x_j\|\} \tag{2.6}$$

Average Linkage

$$Inter_{av} = \frac{\sum_{i,j} \|x_i - x_j\|}{N_k N_l} \tag{2.7}$$

Centroid Linkage

$$Inter_{ce} = \|c_k - c_l\| \tag{2.8}$$

เมื่อ x_i แทนข้อมูลที่อยู่ภายในกลุ่ม

x_j แทนข้อมูลที่อยู่นอกกลุ่มข้อมูล

N_l แทนจำนวนข้อมูลทั้งหมดที่อยู่นอกกลุ่ม

c_k แทนจุดศูนย์กลางจากสมาชิกทั้งหมดภายในกลุ่ม

c_l แทนจุดศุนย์กลางจากสมาชิกทั้งหมดของอีกกลุ่มหนึ่ง

Q_l แทนกลุ่มข้อมูลอีกกลุ่มหนึ่ง

การเลือกใช้วิธีหาความสัมพันธ์จะขึ้นอยู่กับรูปแบบของข้อมูล เช่น $Intra_{nn}$ และ $Inter_{sl}$ เหมาะสำหรับหาความสัมพันธ์กับข้อมูลข้างเคียงที่ใกล้ที่สุดและมีข้อมูลรบกวน (Noise) น้อย [5] ในวิทยานิพนธ์นี้ทดลองหาความสัมพันธ์แบบ $Intra_{nn}$ และ $Inter_{sl}$ กับชุดข้อมูลรูปแบบการบุกรุกเครือข่าย

2.3 ทฤษฎีการจัดกลุ่มข้อมูลที่เกี่ยวข้อง

2.3.1 วิธีการจัดกลุ่มข้อมูลแบบลำดับชั้น (Hierarchical Clustering Algorithms)

วิธีการจัดกลุ่มข้อมูลแบบลำดับชั้น เป็นวิธีการวิเคราะห์ข้อมูลวิธีการหนึ่งที่ได้รับความเชื่อถือมานานและถูกนำไปประยุกต์ใช้ในงานวิจัยด้านต่าง ๆ อย่างกว้างขวาง ไม่ว่าจะเป็นงานวิจัยทางเศรษฐศาสตร์ ดาราศาสตร์ และชีววิทยา ฯลฯ [2] การทำงานของวิธีการจัดกลุ่มข้อมูล แบบลำดับชั้นแบ่งออกเป็น 3 ขั้นตอน ได้แก่ (1) ขั้นตอนการเตรียมข้อมูล (2) ขั้นตอนการทำงาน และ (3) ขั้นตอนการแสดงผล ซึ่งมีรายละเอียดดังต่อไปนี้

1) ขั้นตอนการเตรียมข้อมูล

การเตรียมข้อมูล หมายถึง การบันทึกรายละเอียดของข้อมูลที่ต้องการจัดกลุ่มลงในเมตริกซ์ (Matrix) เพื่อให้สามารถคำนวณตามกระบวนการของวิธีการจัดกลุ่มข้อมูล โดยทั่วไป ข้อมูลที่จะนำมาจัดกลุ่มได้ต้องมีคุณสมบัติ (Attribute) อย่างใดอย่างหนึ่งเสมอ ทั้งนี้เพื่อให้แยกได้ว่า ข้อมูลแต่ละชิ้นมีคุณลักษณะ (Characteristic) เหมือนหรือต่างกันอย่างไร

ในการเตรียมข้อมูล จะบันทึกชื่อ (Label) และคุณสมบัติของข้อมูลไว้ในเมตริกซ์ที่เรียกว่าเมตริกซ์ข้อมูล (Data Matrix) ซึ่งเป็นตารางที่แสดงชื่อในคอลัมน์แรก และแสดงค่าคุณสมบัติมีค่า 1 เมื่อมีคุณสมบัติใดๆ และมีค่าเป็น 0 เมื่อไม่มีคุณสมบัตินั้น ดังตัวอย่างในตารางที่ 2.1

ตารางที่ 2.1 ตัวอย่างเมตริกซ์ข้อมูล

	คุณสมบัติที่ 1	คุณสมบัติที่ 2	คุณสมบัติที่ 3	คุณสมบัติที่ 4
วัตถุ A	1.0	0.0	1.0	0.0
วัตถุ B	0.0	1.0	1.0	0.0
วัตถุ C	0.0	1.0	1.0	0.0
วัตถุ D	0.0	0.0	0.0	1.0

ขั้นตอนต่อไปของการเตรียมข้อมูลคือ การนำค่าคุณสมบัติที่แสดงไว้ในเมตริกซ์ข้อมูลมาคำนวณเพื่อหาค่าระยะทาง (Distance) ค่าระยะทางหมายถึง ตัวเลขที่บอกปริมาณความแตกต่างระหว่างข้อมูลคู่หนึ่งๆ การคำนวณค่าระยะทางสามารถทำได้สองแบบ ได้แก่ แบบที่หนึ่งการคำนวณโดยปฏิบัติ (Treat) ต่อคุณสมบัติทุกอย่างเท่าเทียมกัน (Unweighting) และแบบที่สองรายการคำนวณโดยปฏิบัติต่อคุณสมบัติแตกต่างกัน (Weighting) การพิจารณาว่าการคำนวณแบบใดมีความเหมาะสมมากกว่า ต้องคำนึงถึงลักษณะของคุณสมบัติและจุดมุ่งหมายของการจัดกลุ่มข้อมูล

การคำนวณค่าระยะทางโดยปฏิบัติต่อคุณสมบัติแตกต่างกัน ทำได้โดยเลือกใช้วิธีคำนวณค่าระยะทางที่อนุญาตให้ใส่ค่าน้ำหนัก วิทยานิพนธ์นี้เลือกใช้วิธีที่เรียกว่า วิธียูคลิเดียน แบบให้ค่าน้ำหนัก (Weighted Euclidean Distance Method) เพราะเป็นวิธีเดียวที่สามารถคำนวณได้ทั้งแบบให้ค่าน้ำหนัก (กำหนดให้น้ำหนักของคุณสมบัติแต่ละอย่างมีค่าแตกต่างกัน) และแบบไม่ให้ค่าน้ำหนัก (กำหนดให้น้ำหนักของคุณสมบัติทั้งหมดมีค่าเท่ากับ 1.0) เพื่อความสะดวกในการพัฒนาโปรแกรม รายละเอียดของวิธียูคลิเดียนแบบให้ค่าน้ำหนักเป็นตามสมการ 2.9 ดังนี้

$$d(i,j) = \left(\sum_{k=1}^{p} w_k \left| x_{ik} - x_{jk} \right|^2 \right)^{\frac{1}{2}} \qquad (2.9)$$

โดยที่ $d(i,j)$ คือ ค่าระยะทางระหว่างข้อมูลลำดับที่ i กับข้อมูลลำดับที่ j

x_{ik} คือ ค่าคุณสมบัติที่ k ของข้อมูลในลำดับที่ i

x_{jk} คือ ค่าคุณสมบัติที่ k ของข้อมูลในลำดับที่ j

w_k คือ ค่าน้ำหนักที่ให้แก่คุณสมบัติที่ k

p คือ จำนวนคุณสมบัติข้อมูล

ตัวอย่างเช่น เมื่อกำหนดให้ค่า w_k = 1.0 ทุกค่า k จะสามารถคำนวณค่าระยะทางระหว่างวัตถุ A กับ วัตถุ B ได้ดังนี้

$$d(1,2) = \left(\sum_{k=1}^{4} w_k |x_{1k} - x_{2k}|^2 \right)^{\frac{1}{2}}$$

$$= \left(1.0 \times (1.0-0.0)^2 + 1.0 \times (0.0-1.0)^2 + 1.0 \times (1.0-1.0)^2 + 1.0 \times (0.0-0.0)^2\right)^{\frac{1}{2}}$$

$$= \left(1.0+1.0+0.0+0.0\right)^{\frac{1}{2}}$$

$$= 1.414$$

ภายหลังการคำนวณจะบันทึกค่าระยะทางที่ได้ลงในเมตริกซ์ความแตกต่าง (Dissimilarity Matrix) ซึ่งเมตริกซ์ชนิดนี้มีลักษณะที่น่าสนใจคือ มีขนาดเท่ากับ n × n เสมอ (เมื่อ n คือจำนวนข้อมูล) และแสดงข้อมูลไว้เพียงครึ่งเดียว ดังตัวอย่างในตารางที่ 2.2

ตารางที่ 2.2 ตัวอย่างเมตริกซ์ความแตกต่าง

	วัตถุ A	**วัตถุ B**	**วัตถุ C**	**วัตถุ D**
วัตถุ A	X	1.414	1.414	1.732
วัตถุ B		X	0.000	1.732
วัตถุ C			X	1.732
วัตถุ D				X

2) ขั้นตอนการทำงาน

วิธีจัดการกลุ่มข้อมูล จัดเรียงข้อมูลโดยมีลำดับการทำงานดังนี้

2.1 เริ่มต้นโดยนำข้อมูลไปแยกบรรจุไว้ในคลัสเตอร์ว่าง หนึ่งคลัสเตอร์ต่อหนึ่งข้อมูล

2.2 ค้นหาคู่ข้อมูลที่มีความแตกต่างกันน้อยที่สุด โดยพิจารณาจากค่าระยะทางที่แสดงไว้ในเมตริกซ์ความแตกต่าง เช่น คู่ วัตถุ B และ วัตถุ C ในตารางที่ 2.2 เป็นคู่ที่มีความแตกต่างกันน้อยที่สุด เพราะมีค่าระยะทางน้อยที่สุด คือ 0.000

2.3 นำคลัสเตอร์ของข้อมูลคู่นั้นมารวมกัน จากนั้นสร้างเป็นคลัสเตอร์อันใหม่ เช่น นำคลัสเตอร์ที่บรรจุ วัตถุ B มารวมเข้ากับคลัสเตอร์ที่บรรจุ วัตถุ C สร้างเป็นคลัสเตอร์ วัตถุ B+C

2.4 ปรับค่าระยะทางระหว่างคลัสเตอร์ใหม่กับคลัสเตอร์อื่นๆ ที่มีอยู่เดิมโดยใช้กฎการปรับค่า (Updating Rule)

2.5 กลับไปทำขั้นตอนที่ 2.1 อีกครั้ง จนกระทั่งข้อมูลทั้งหมดถูกรวมกันไว้ในคลัสเตอร์อันเดียว จึงหยุดการทำงาน

กฎการปรับค่า คือ เกณฑ์สำหรับเลือกค่าระยะทางของสมาชิกในคลัสเตอร์เก่า (ที่ถูกรวมเข้าด้วยกัน) มาใช้เป็นค่าระยะทางของสมาชิกคลัสเตอร์อันใหม่ วิทยานิพนธ์นี้เลือกใช้ค่าที่เรียกว่า กฎเลือกค่ามากสุด (Maximum Distance Rule) เพราะต้องการให้ข้อมูลมีความแตกต่างอย่างเด่นชัด รายละเอียดของกฎการเลือกค่ามากที่สุดเป็นตามสมการ 2.10 ดังนี้

$$d_{\max}(C_i, C_j) = \max_{p \in C_i, p' \in C_j} |p - p'| \tag{2.10}$$

โดยที่ $d_{\max}(C_i, C_j)$ คือ ค่าระยะทางระหว่างคลัสเตอร์ C_i กับคลัสเตอร์ C_j

$|p - p'|$ คือ ค่าระยะทางระหว่าง p (สมาชิกของคลัสเตอร์ใหม่) กับ p' (คลัสเตอร์อื่นที่มีอยู่เดิม)

ตัวอย่างเช่น การปรับค่าระยะทางระหว่างคลัสเตอร์วัตถุ B+C (ใหม่) กับคลัสเตอร์วัตถุ A (เก่า) สามารถทำได้ดังนี้

$$\begin{aligned} d_{\max}(C_{2+3}, C_1) &= \max_{p \in C_{2+3}, p' \in C_1} |p - p'| \\ &= \max(1.41, 1.41) \\ &= 1.41 \end{aligned}$$

หลังการคำนวณจะบันทึกค่าระยะทางใหม่ไว้ในเมตริกซ์ความแตกต่างอันใหม่อีกเมตริกซ์ ดังตารางที่ 2.3

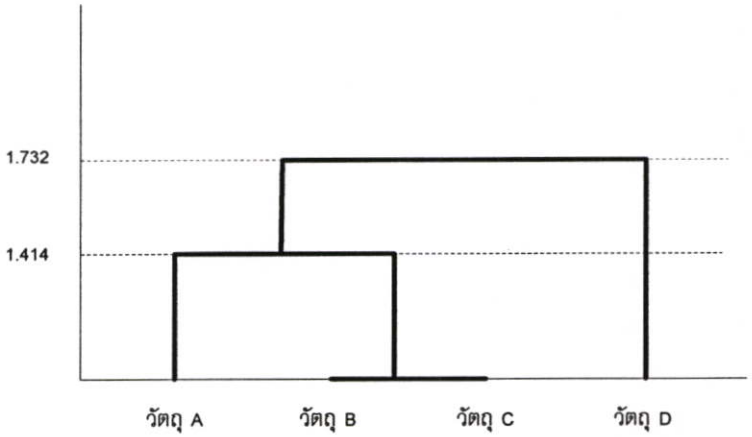
ตารางที่ 2.3 ตัวอย่างเมตริกซ์ความแตกต่างหลังการรวมกลุ่มครั้งที่หนึ่ง

	วัตถุ A	**วัตถุ B+C**	**วัตถุ D**
วัตถุ A	X	1.414	1.732
วัตถุ B+C		X	1.732
วัตถุ D			X

3) ขั้นตอนการแสดงผล

ผลลัพธ์ของการจัดกลุ่มข้อมูลจะแสดงในรูปแบบของแผนภาพต้นไม้ของคลัสเตอร์ (Tree of Cluster) ซึ่งเป็นรูปแบบที่ช่วยให้ทราบได้ง่ายว่า คลัสเตอร์ต่างๆ ถูกสร้างขึ้นและรวมเข้าด้วยกันอย่างไร

มีค่าระยะทางระหว่างกันเป็นเท่าไร แผนภาพดังกล่าวมีชื่อเรียกว่า เดนโดแกรม (Dendogram) ดังแสดงในรูปที่ 2.3



รูปที่ 2.3 ตัวอย่างเดนโดแกรม

2.3.2 K-mean Clustering

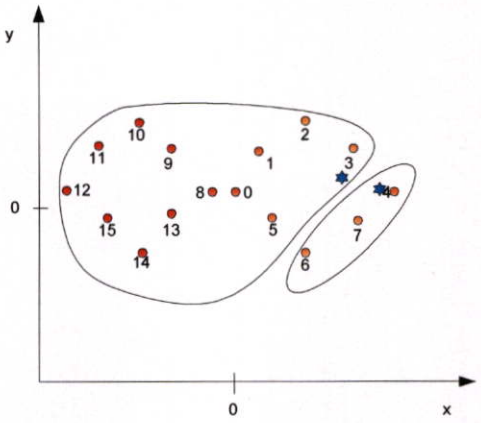
เทคนิคของ K-mean จะใช้วิธีจัดแบ่งกลุ่ม โดยการเรียนรู้จากปัญหาง่ายๆ ก่อน และพัฒนาไปสู่แก้ไขปัญหาใหม่ๆ เทคนิคนี้จะตัดสินใจว่ากลุ่มใดบ้างที่จะแทนเงื่อนไขใหม่ๆ ได้ โดยการตรวจสอบจำนวนบางจำนวนหรือค่า K ของกรณีหรือเงื่อนไขที่เหมือนกันหรือใกล้เคียงกันมากที่สุด โดยจะหาผลรวมของจำนวนเงื่อนไขของแต่ละกลุ่ม และกำหนดเงื่อนไขใหม่ๆ ให้กลุ่มที่เหมือนกันกับกลุ่มที่ใกล้เคียงกันมากที่สุด

ในขั้นตอนแรกของเทคนิคนี้จะหาวิธีการวัดระยะห่าง (Distance) ระหว่างแต่ละคุณสมบัติของข้อมูล และคำนวณค่าระยะห่าง ซึ่งข้อมูลที่ใช้ในการวัดจะเป็นตัวเลข หรือข้อมูลที่ทำการ Normalize มาแล้ว เมื่อทราบระยะห่างระหว่างเงื่อนไขต่างๆ แล้ว จากนั้นทำการเลือกชุดข้อมูลจากเงื่อนไขที่ใช้จัดกลุ่มมาเป็นฐานสำหรับการจัดกลุ่มในเงื่อนไขใหม่ๆ แล้วทำการตัดสินใจว่าขอบเขตของจุดข้างเคียงที่ควรจัดกลุ่มนั้นควรมีขนาดใหญ่เท่าใด และอาจตัดสินใจได้ว่าจะนับจากจำนวนจุดข้างเคียงตัวมันได้อย่างไร โดยอาจจะใช้น้ำหนักเพื่อเลือกจุดข้างเคียงที่ใกล้ตัวมันมากที่สุด โดยทั่วไปจะมีความน่าจะเป็นมากกว่าที่จะเลือกจุดที่ห่างไกลออกไป

ขั้นตอนการทำงานของ K-mean

1. กำหนดจำนวนกลุ่มข้อมูล
2. กำหนดค่าศูนย์กลางข้อมูลเริ่มต้นด้วยวิธีการสุ่ม

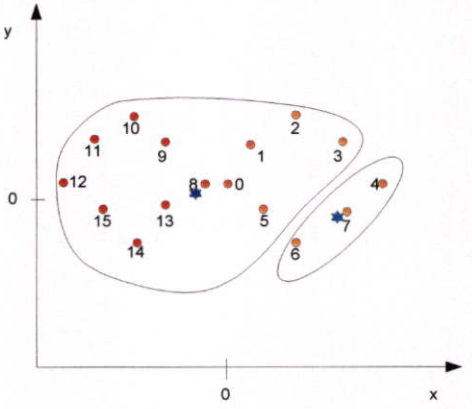
<u>ตัวอย่าง</u> ให้จำนวนกลุ่ม K=2 โดยสุ่มการจัดกลุ่มข้อมูลเป็น (4 6 7), (0 1 2 3 5 8 9 10 11 12 13 14 15) สุ่มจุดศูนย์กลางของกลุ่ม (7.0 -2.0), (-1.61538 0.46153) เมื่อหาระยะห่างค่าเฉลี่ย 4.35887 ดังรูปที่ 2.4



รูปที่ 2.4 ขั้นตอนการจัดกลุ่มด้วยวิธี K-mean โดยการสุ่มจัดกลุ่มข้อมูลและจุดศูนย์กลางข้อมูลเริ่มต้น

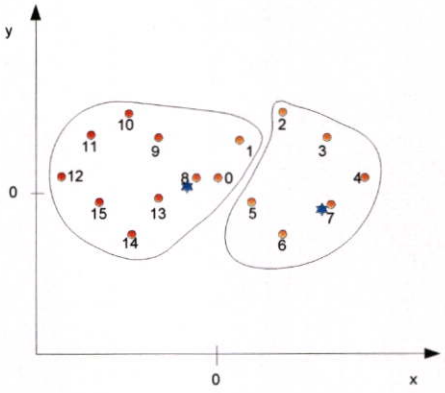
3. คำนวณหาระยะเฉลี่ยของข้อมูลจากสมการ 2.1 และจุดศูนย์กลางใหม่จากสมการ 2.4 ได้ศูนย์กลางข้อมูลใหม่และระยะเฉลี่ยใหม่ดังนี้

ศูนย์กลางกลุ่มข้อมูลใหม่ (6.0 -0.33334) , (-3.6 0.2) และระยะเฉลี่ยเท่ากับ 3.6928 ดังรูป 2.5



รูปที่ 2.5 ระยะเฉลี่ยและจุดศูนย์กลางข้อมูลใหม่ที่คำนวณได้

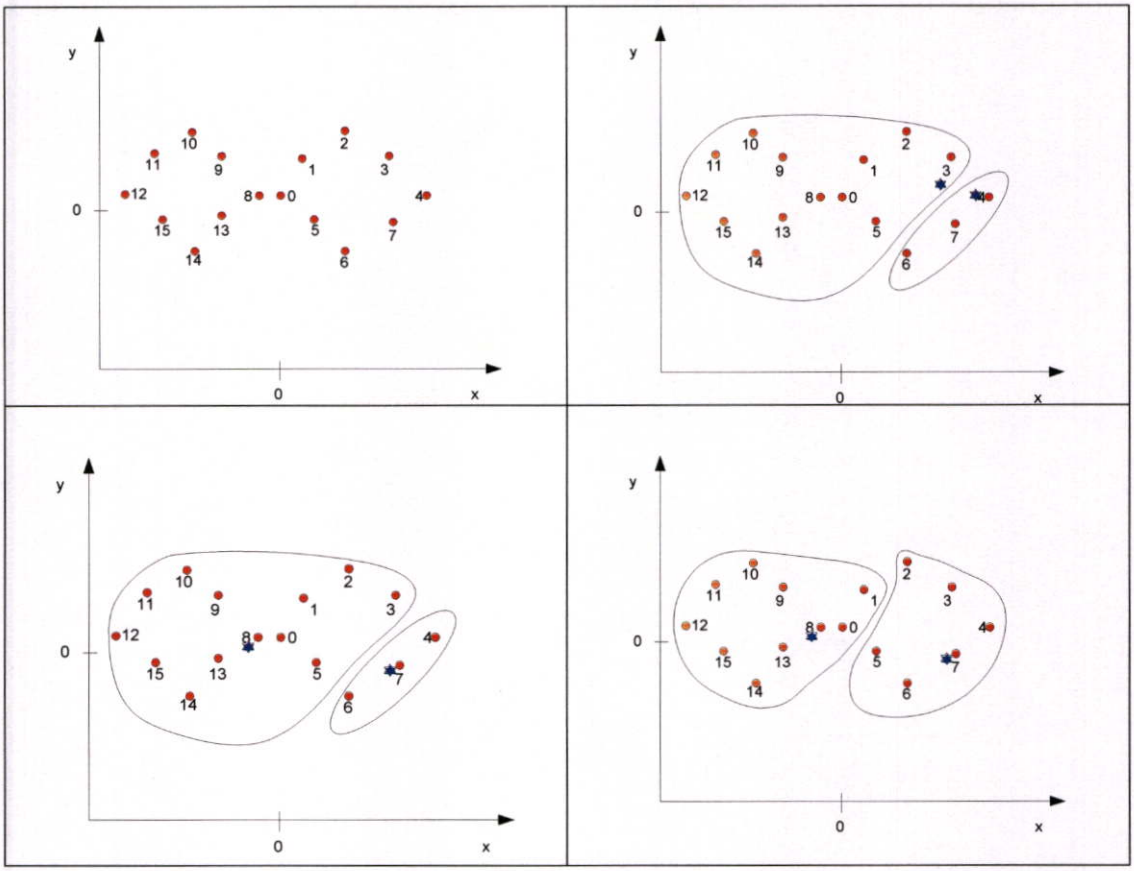
4. เปรียบเทียบข้อมูลจากวัตถุกับศูนย์กลางข้อมูลในกลุ่มจะได้กลุ่มข้อมูลใหม่ดังนี้ Clustering (2 3 4 5 6 7) , (0 1 8 9 10 11 12 13 14 15) แสดงดังรูปที่ 2.6



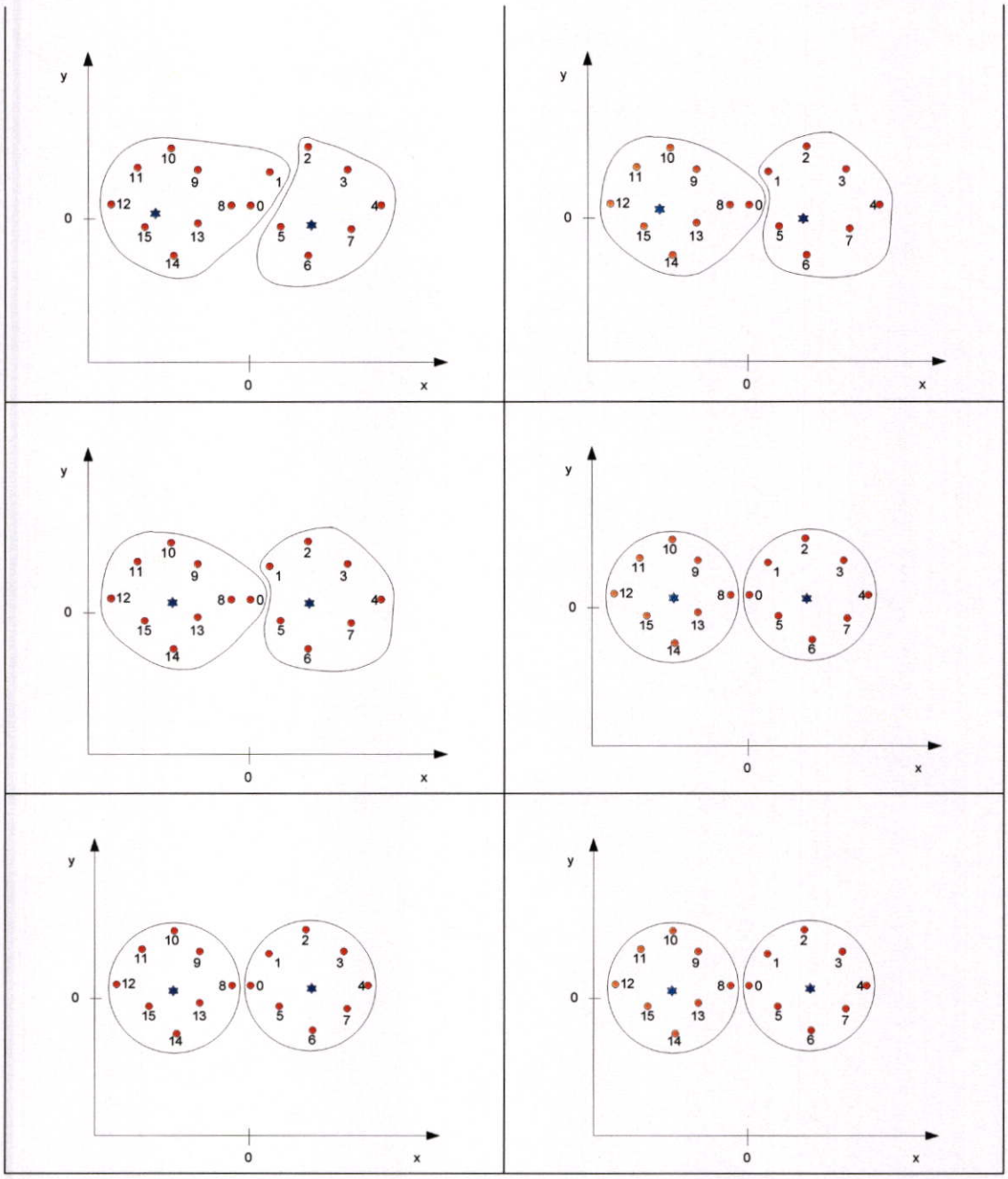
รูปที่ 2.6 กลุ่มข้อมูลใหม่ที่ได้จากการเปรียบเทียบข้อมูลกับจุดศูนย์กลางข้อมูล

5. หาจุดศูนย์กลางใหม่เช่นเดียวกับขั้นตอนที่ 3
6. คำนวณจนกระทั่งจุดศูนย์กลางและค่าเฉลี่ยไม่มีการเปลี่ยนแปลง

ซึ่งจะพบการเปลี่ยนแปลงของการจัดกลุ่มด้วยวิธี K-mean ได้ดังรูปที่ 2.7



รูปที่ 2.7 แสดงการเปลี่ยนแปลงกลุ่มข้อมูลหลังการคำนวณด้วยวิธี K-mean



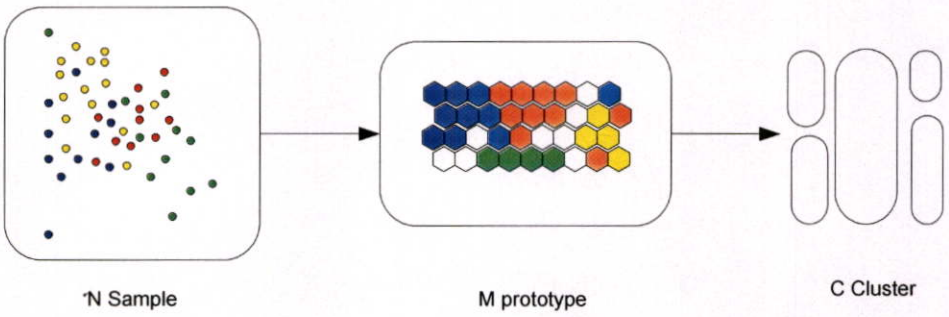
รูปที่ 2.7(ต่อ)

การทำงานของวิธี K-mean จะใช้ปริมาณงานในการคำนวณมากเพราะเวลาที่ใช้สำหรับการคำนวณจะเพิ่มขึ้นเป็นแฟคเทอเรียลตามจำนวนจุดทั้งหมด และต้องเกิดการคำนวณทุกครั้งเมื่อมีกรณีใหม่ๆ เกิดขึ้น

งานวิจัย J. Vesanto and E. Alhoniemi, [3] ได้กล่าวถึงวัตถุประสงค์ของการสร้างเหมืองข้อมูลที่จะต้องกำหนดข้อมูลให้เพียงพอและเลือกคุณสมบัติของข้อมูลก่อนเข้ากระบวนการจัดกลุ่ม หลังจากนั้นทำการเลือกใช้เครื่องมือเพื่อให้การจัดกลุ่มนั้นสามารถเข้าใจและวิเคราะห์โครงสร้างได้

ง่าย ดังนั้นการจัดกลุ่มที่มีคุณภาพควรจะใช้เวลาและความถูกต้องในการจัดกลุ่มข้อมูลรวดเร็วและเรียกดูได้อย่างมีประสิทธิภาพ SOM จึงเป็นเครื่องมือที่เหมาะสมในการสำรวจข้อมูลเพราะสามารถเรียกดูข้อมูลและเข้าใจโครงสร้างของข้อมูลได้ง่าย

SOM เป็นนิวรอลเน็ตเวิร์คแบบไม่อาศัยผู้สอนโดยแปลงข้อมูลที่มีขนาดหลายมิติและไม่เป็นเชิงเส้นไปยังแผนภาพกริดขนาดสองมิติ ข้อมูลหรือเอกสารที่มีความคล้ายๆ กันจะพบอยู่บริเวณใกล้เคียงกัน แต่อย่างไรก็ตาม SOM มีข้อเสียในการกำหนดจำนวนโหนดหรือนิวรอลให้เรียบร้อยเสียก่อนและใช้เวลาการคำนวณนาน ในการจัดกลุ่มข้อมูลที่มีขนาดใหญ่มากๆ บางครั้งการจัดกลุ่มด้วย SOM ไม่สามารถบอกข้อมูลนั้นเป็นกลุ่มใดได้เนื่องจากการกระจายของข้อมูลในแผนภาพนั้นไม่อยู่บริเวณเดียวกัน และได้พยายามแบ่งขั้นตอนการจัดกลุ่มออกเป็นสองขั้นตอน เพื่อแก้ปัญหาข้อเสียของ SOM ในขั้นแรกข้อมูลที่มีลักษณะกระจัดกระจายจะถูกจัดกลุ่มด้วย SOM เพื่อกำหนดเป็นต้นแบบ (prototype) ขั้นที่สองเป็นการจัดกลุ่มจากข้อมูลเอาท์พุทของ SOM โดยใช้ Partitive Clustering หรือ Agglomerative Clustering จัดกลุ่มข้อมูลอีกครั้ง ดังรูปที่ 2.8 ข้อดีของการจัดกลุ่มสองระดับจะทำให้ลดเวลาการคำนวณลงได้



รูปที่ 2.8 แสดงแนวความคิดการจัดกลุ่มข้อมูลเป็นออกเป็นสองขั้นตอน

ในงานวิจัยดังกล่าว เมื่อจัดกลุ่มข้อมูลที่มีมิติมากๆ ด้วย SOM เช่น ข้อมูลการบุกรุกเครือข่ายเพื่อสร้างต้นแบบ หลังจากนั้นทำการจัดกลุ่มข้อมูลด้วย Partitive Clustering และ Agglomerative Clustering แล้วผลลัพธ์ที่ได้มีการจัดข้อมูลผิดกลุ่มเกิดขึ้น ดังนั้นในวิทยานิพนธ์นี้ได้พยายามปรับปรุงการจัดกลุ่มข้อมูลให้ถูกต้องมากขึ้นโดยอาศัยแนวคิดการจัดกลุ่ม 2 ระดับ

บทที่ 3
พื้นฐานการเรียนรู้ Self-Organizing Map

3.1 บทนำ

Self-Organizing Map (SOM) ได้พัฒนาจากศาสตราจารย์โคโฮเนน [1] จากมหาวิทยาลัย Helsinki ประเทศฟินแลนด์ ซึ่งได้รับความความนิยมอย่างแพร่หลาย และได้มีการนำไปประยุกต์ใช้งานอย่างต่อเนื่อง เช่น การแบ่งส่วนย่อยของภาพ (Image Segmentation) การจัดกลุ่มข้อมูล (Data Clustering) การรู้จำรูปแบบ (Pattern Recognition) งานด้าน Machine Learning และอื่นๆ เป็นต้น

3.2 ประเภทของนิวรอลเน็ตเวิร์ค

สถาปัตยกรรมและกระบวนการทำงานของนิวรอลเนตเวิร์คจะแบ่งออกเป็น 3 ประเภทหลักๆ ตามลักษณะทางกายภาพได้ดังนี้

1. นิวรอลแบบส่งค่าต่อ (Feed Forward Network) เป็นการส่งถ่ายกลุ่มข้อมูลของอินพุทไปยังกลุ่มข้อมูลเอาต์พุท ในการส่งถ่ายข้อมูลจะทำการแปลงค่าจากฟังก์ชันตามความต้องการในแต่ละปัญหา โดยส่วนมากจะเป็นนิวรอลที่ต้องอาศัยผู้สอน (Supervise Learning)

2. นิวรอลแบบส่งข้อมูลย้อนกลับ (Feedback Network) กลุ่มข้อมูลอินพุตจะเป็นสถานะเริ่มต้นและหลังจากนั้นข้อมูลจะส่งผ่านไปยังสถานะถัดไปจนถึงสถานะสุดท้าย แล้วนำข้อมูลย้อนกลับไปคำนวณใหม่อีกครั้ง

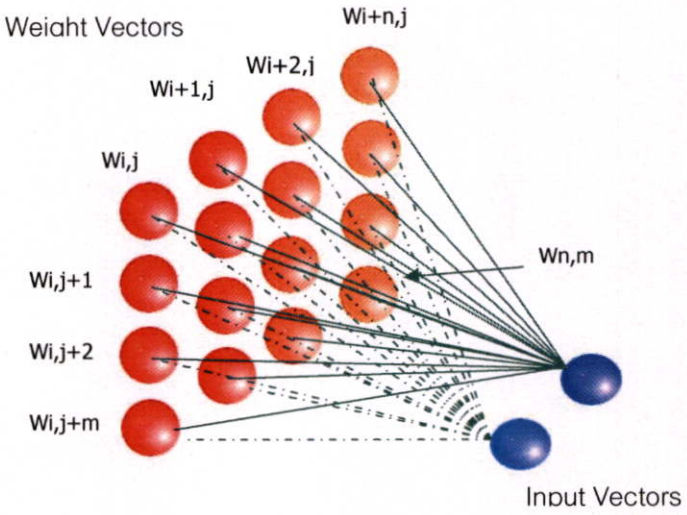
3. นิวรอลแบบแข่งขัน (Competitive Network) ซึ่งจะอาศัยข้อมูลข้างเคียง (Neighboring Cell) เพื่อเปรียบเทียบเซลล์ที่แข็งแรงที่สุดหรือเซลล์ที่ใกล้เคียงกับข้อมูลอินพุทมากที่สุด ซึ่งวิธีนี้เป็นการเรียนรู้แบบไม่ต้องอาศัยผู้สอน (Unsupervised Learning) โดยที่ SOM เป็นนิวรอลลักษณะที่ไม่ต้องอาศัยผู้สอน

3.3 วิธีการทำงาน Self-Organizing Map

SOM คือนิวรอนเน็ตเวิร์กแบบไม่มีผู้สอน ซึ่งประกอบด้วยอินพุทเวคเตอร์ $\mathbf{x}$ ที่มีขนาด n มิติ และมีโหนดของนิวรอนขนาด 2 มิติ ซึ่งแต่ละโหนดของนิวรอนจะประกอบไปด้วยเวคเตอร์น้ำหนักแทนด้วย w_i โดยที่มิติของเวคเตอร์น้ำหนักจะต้องมีมิติเท่ากับอินพุตเวคเตอร์

$$x(t) = \{x_1, x_2, x_3, ..., x_n\}$$

$$w_i(t) = \{w_1, w_2, w_3, ..., w_n\}$$

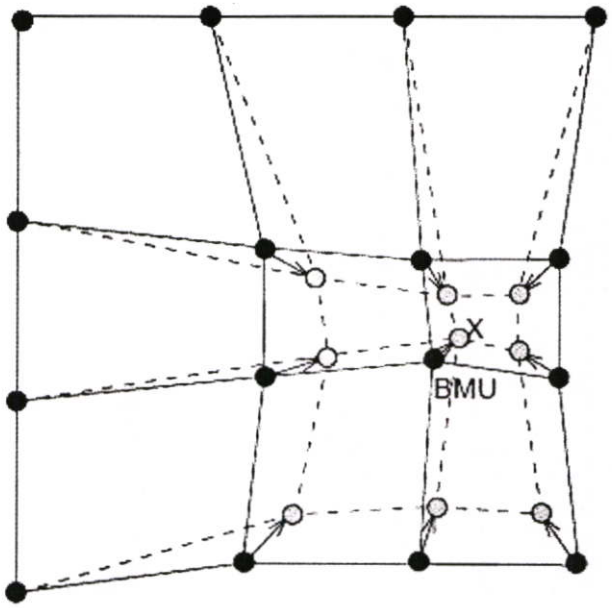


รูปที่ 3.1 แบบแผนภาพจำลองโคโฮเนน

กระบวนการเรียนรู้ของ SOM เกิดขึ้นจากการปรับค่าของเวคเตอร์น้ำหนักที่มีต่ออินพุตเวคเตอร์ ในแต่ละรอบ t ของการเรียนรู้เราจะทำการสุ่มเลือกอินพุตเวคเตอร์ $x(t)$ จากนั้นทำการเปรียบเทียบกับโหนดทุกโหนดเพื่อที่จะหาโหนดชนะ สำหรับอินพุตเวคเตอร์นั้น ฟังก์ชั่นที่มักใช้ในการเปรียบเทียบหาความห่างของข้อมูลคือฟังก์ชั่นวัดระยะห่างแบบยูคลิดเดียน (Euclidean Distance) ระยะห่างระหว่างอินพุตเวคเตอร์กับโหนดน้อยที่สุดจะเป็นโหนดชนะ (Best Match Unit: BMU) ดังสมการที่ 3.1

$$\|x(t) - w_c(t)\| = \min_i \|x(t) - w_i(t)\| \tag{3.1}$$

จากนั้นเวคเตอร์น้ำหนักของโหนดชนะจะถูกปรับค่าดังรูปที่ 3.2 โดยการปรับจะพิจารณาจากผลต่างของอินพุตเวคเตอร์และเวคเตอร์น้ำหนัก โดยค่าเวคเตอร์น้ำหนักของโหนดชนะจะพยายามปรับให้เข้าใกล้ค่าอินพุตเวคเตอร์ และอัตราการเรียนรู้แต่ละรอบจะค่อย ๆ ลดลง นอกจากการเรียนรู้ที่เกิดขึ้นที่โหนดชนะแล้ว โหนดใกล้เคียง (Neighborhood Node) จะเกิดการเรียนรู้ด้วย โดยเวคเตอร์น้ำหนักของโหนดใกล้เคียงจะปรับค่าให้เข้าใกล้กับอินพุตเวคเตอร์เดียวกัน ดังรูปที่ 3.2 เพื่อเพิ่มโอกาสให้อินพุตใหม่ที่ใกล้เคียงกับอินพุตเดิมสามารถที่จะมีโหนดชนะใหม่ใกล้กับโหนดชนะเดิมได้ สมการในการปรับค่าสามารถแสดงได้ดังสมการที่ 3.2



รูปที่ 3.2 แสดงการปรับค่าเวคเตอร์น้ำหนักของโหนดชนะให้ใกล้เคียงกับค่าอินพุทแทนสัญลักษณ์ 'x' จุดสีดำและจุดสีเทาแทนค่าน้ำหนักประจำโหนดก่อนและหลังการเรียนรู้ตามลำดับ

$$w_i(t+1) = w_i(t) + \alpha(t)h_{ci}(t)[x(t) - w_i(t)] \tag{3.2}$$

โดย $x(t)$ คืออินพุตเวคเตอร์

$w_i(t)$ คือเวคเตอร์น้ำหนักของโหนด

$\alpha(t)$ คืออัตราการเรียนรู้ในแต่ละรอบ

t คือรอบในการเรียนรู้ ซึ่งแสดงเป็นสมการเชิงเส้นได้ดังสมการที่ 3.3

$$\alpha(t+1) = \alpha(0)e^{-\frac{t}{T}} \tag{3.3}$$

เมื่อ T คือจำนวนรอบทั้งหมด

t คือจำนวนรอบที่ปัจจุบันและ

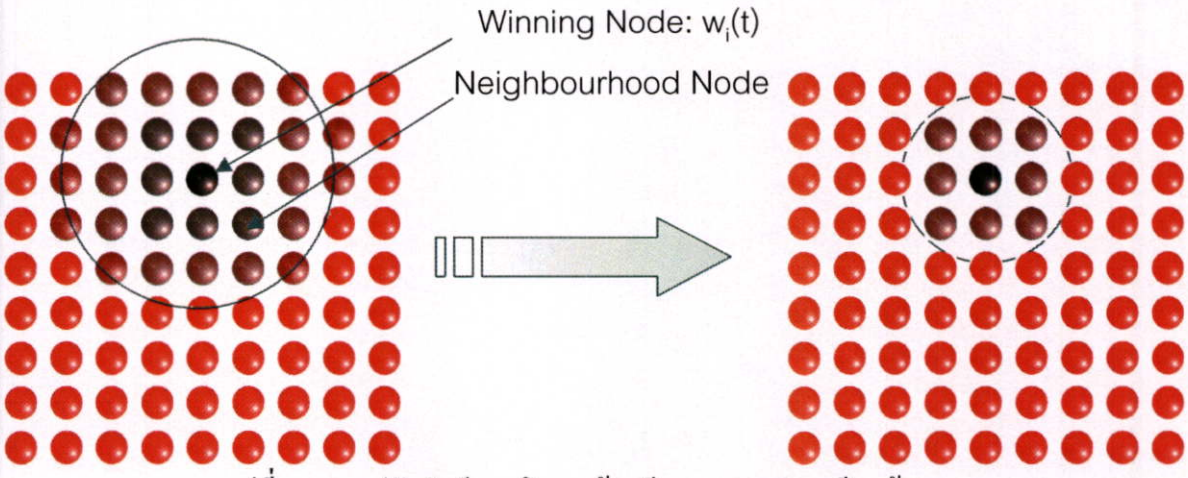
$h_{ci}(t)$ คือฟังก์ชันที่ใช้ในการกำหนดขนาดของโหนดใกล้เคียงโดยทั่วไปแล้วจะใช้ฟังก์ชันเกาส์เซียน (Gaussian Function) ดังแสดงตามสมการที่ 3.4

$$h_{ci}(t) = e^{\frac{\|r_i - r_c\|^2}{2\sigma^2(t)}} \tag{3.4}$$

เมื่อ $\|r_c - r_i\|$ คือระยะห่างระหว่างโหนด i กับโหนดชนะ c

σ (*t*) คือรัศมีของบริเวณโหนดใกล้เคียง โดยปกติรัศมีจะค่อยๆ ลดลงตามจำนวนรอบในการเรียนรู้ ดังแสดงตามสมการที่ 3.5 และแสดงการปรับรัศมี σ ตามรอบการเรียนรู้ดังรูปที่ 3.3

$$\sigma(t+1) = 1+(\sigma(t)-1)\times\frac{T-t}{T} \tag{3.5}$$



รูปที่ 3.3 การปรับรัศมีของโหนดข้างเคียงตามรอบการเรียนรู้

ความสามารถของการกระจายและจัดเรียงตัวของนิวรอลหรือโหนดจะอาศัยความสัมพันธ์ระหว่างโหนดชนะกับโหนดข้างเคียงที่มีผลทำให้โหนดในแผนภาพมีลักษณะคล้ายคลึงกับข้อมูลอินพุทนั้นได้แก่ ฟังก์ชันโหนดข้างเคียง (Neighborhood Function) $h_{ci}(t)$

3.4 Neighborhood Function $h_{ci}(t)$

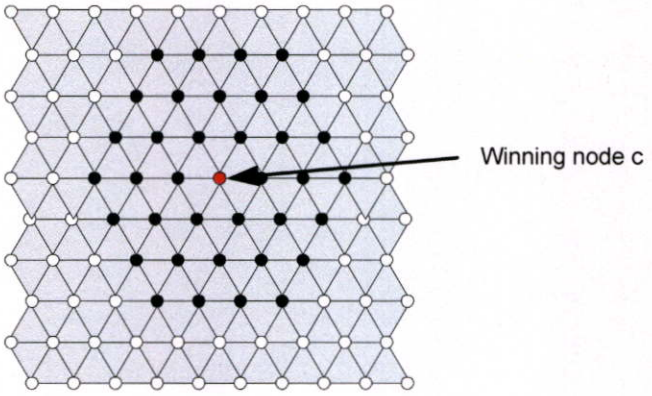
การหาฟังก์ชันโหนดข้างเคียงที่ใช้บ่อยมีอยู่ 2 ประเภทได้แก่ 1) การวัดแบบฟองสบู่ (Bubble) 2) การวัดแบบเกาส์เซี่ยน (Gaussian) สำหรับการวัดแบบฟองสบู่ โดยให้กลุ่มโหนดข้างเคียงของโหนดชนะ *c* ที่เวลา *t* สามารถเขียนได้โดย $N_c(t)$ และ $N_c(t)$ จะลดลงระหว่างการเรียนรู้จากสมการที่ 3.6

$$h_{ci}(t) = \begin{cases} 1 & \text{if } i = N_c(t) \\ 0 & \text{if } i \neq N_c(t) \end{cases} \tag{3.6}$$

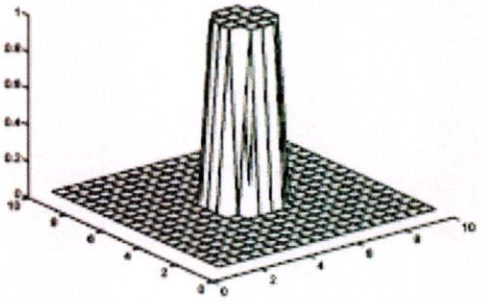
เวคเตอร์น้ำหนักของโหนดชนะและโหนดข้างเคียงจะถูกปรับโดยจำนวนที่แน่นอน โหนดใดที่อยู่ด้านนอกของกลุ่มโหนดข้างเคียงจะไม่ถูกปรับเวคเตอร์น้ำหนัก ดังนั้นการปรับเวคเตอร์น้ำหนักสามารถปรับได้ดังสมการที่ 3.7

$$w_i(t+1) = \begin{cases} w_i(t) + \alpha(t)[x(t) - w_i(t)] & if \quad i = N_c(t) \\ w_i(t) & if \quad i \neq N_c(t) \end{cases} \tag{3.7}$$

ในฟังก์ชันฟองสบู่ เวคเตอร์น้ำหนักของกลุ่มโหนดข้างเคียงจะถูกปรับตามอัตราการเรียนรู้ที่เวลา t รูปที่ 3.4(ก) แสดงโหนดบนแผนภาพที่มีการปรับค่าน้ำหนักเมื่อรัศมีของ $N_c(t)$ เท่ากับ 3 จุดสีขาวแทนโหนดบนแผนภาพ และจุดสีดำแทนโหนดที่มีการปรับค่าเวคเตอร์น้ำหนัก



(ก)



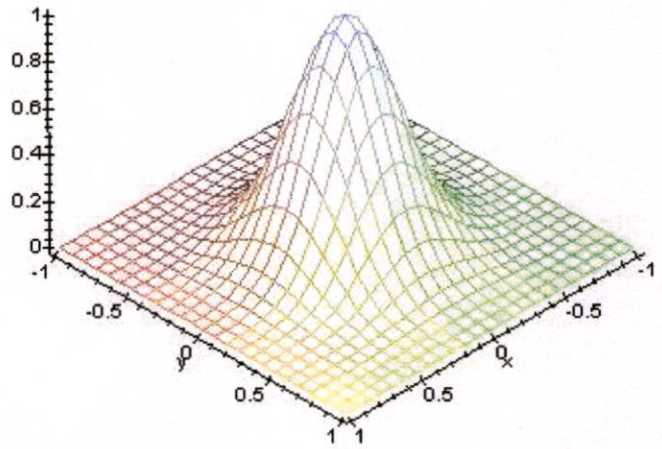
(ข)

รูปที่ 3.4 (ก) โหนดที่มีการปรับเวคเตอร์น้ำหนักโดยใช้ฟังก์ชันฟองสบู่ (ข) ฟังก์ชันฟองสบู่

ในฟังก์ชันโหนดข้างเคียงอีกแบบเป็นการปรับโดยใช้ฟังก์ชันเกาส์เชี่ยน โหนดใดๆ ใน SOM อาจเป็นโหนดชนะได้หลายตัว นั้นหมายความว่ามีโหนดจำนวนหนึ่งที่ได้มีการเรียนรู้ เหตุการณ์อาจเกิดจากการกำหนดค่าเวคเตอร์น้ำหนักเริ่มต้นให้กับโหนดเอาท์พุทที่มีลักษณะแตกต่างกันกับข้อมูลอินพุทมากเกินไป ในการแก้ปัญหานี้จะใช้ฟังก์ชันข้างเคียงแบบเกาส์เชี่ยน เพราะฟังก์ชันนี้จะปรับเวคเตอร์น้ำหนักทุกตัวของโหนดระหว่างการเรียนรู้ โดยที่โหนดที่ใกล้โหนดชนะมากที่สุดจะถูกปรับค่าน้ำหนักมากกว่าโหนดที่อยู่ห่างจากโหนดชนะ

ฟังก์ชันข้างเคียงแบบเกาส์เชี่ยนจะกำหนดให้ r_i เป็นตำแหน่งของโหนดที่อยู่บนแผนภาพ และ r_c แทนตำแหน่งของโหนดชนะ ดังนั้น $\|r_c - r_i\|$ เป็นระยะห่างความเหมือนของโหนดชนะ c กับโหนดข้างเคียง i ที่เกิดขึ้นบนแผนภาพ ฟังก์ชันข้างเคียงแบบเกาส์เชี่ยนจะนิยมใช้มากในการ

ปรับค่าน้ำหนัก หาได้จากสมการที่ 4 เมื่อ $\sigma(t)$ เป็นตัวแปรของโหนดข้างเคียงที่เวลา t $\sigma(t)$ จะลดลงตามเวลาเพื่อควบคุมขนาดของโหนดข้างเคียงที่เวลา t นั้นคือค่าสูงสุดของฟังก์ชันข้างเคียงจะเกิดขึ้นที่โหนดชนะ c และ $h_{ci}(t)$ จะค่อยๆ ลดลงซึ่งขึ้นอยู่กับระยะห่างของโหนดชนะกับโหนดข้างเคียง รูปที่ 3.5 แสดงฟังก์ชันข้างเคียงแบบเกาส์เชี่ยนและระยะห่างระหว่างโหนดชนะกับโหนดข้างเคียงบนแผนภาพ



รูปที่ 3.5 ฟังก์ชันเกาส์เชี่ยนที่มีระดับ $\sigma(t)$ แตกต่างกันไป

3.5 อัตราการเรียนรู้ $\alpha(t)$

อัตราการเรียนรู้ควรจะเริ่มต้นมีค่าใกล้เคียง 1 หลังจากนั้นค่านั้นจะลดลงจนกระทั่งจบกระบวนการเรียนรู้แต่จะมีค่ามากกว่า 0 ดังนั้น $0 \le \alpha(t) \le 1$

ค่าอัตราการเรียนรู้จะลดลงแบบเชิงเส้นดังนี้

$$\alpha(t) = 0.9(1 - \frac{t}{N}) \tag{3.8}$$

หรือลงลงแบบเอ็กซ์โพเนนเชี่ยน

$$\alpha(t) = \alpha(0)e^{(-\frac{t}{N})} \tag{3.9}$$

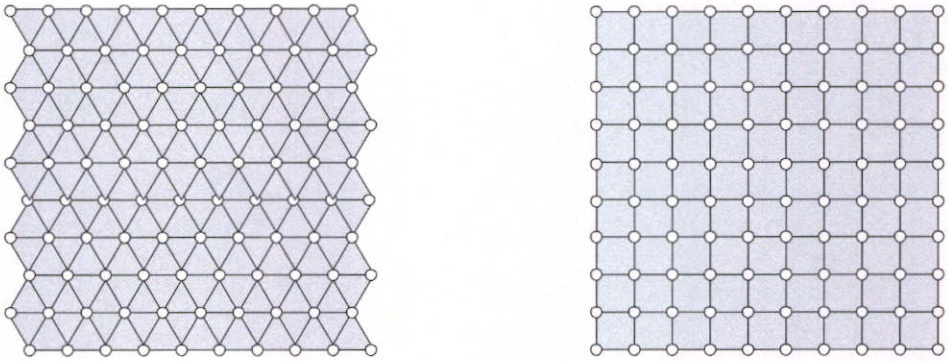
เมื่อ $\alpha(0)$ เป็นอัตราการเรียนรู้เริ่มต้น

และ N แทนจำนวนรอบทั้งหมดของกระบวนการเรียนรู้ โดยทั่วไปแล้ว อัตราการเรียนรู้ควรจะมีค่าน้อยกว่า 0.02 [1]

3.6 ตาข่ายอาร์เรย์ของแผนภาพ SOM

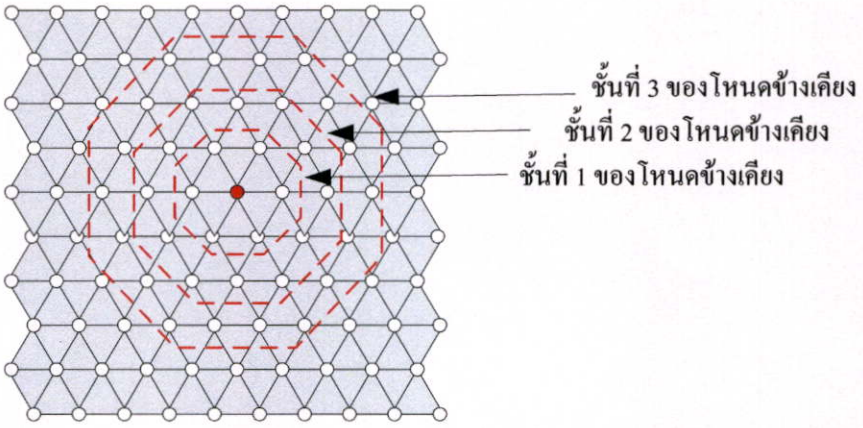
(Lattice Array of the Neuron Map)

โหนดบนแผนภาพจะเรียงตัวเป็นรูปทรงแบบสี่เหลี่ยม หกเหลี่ยมหรือไม่เป็นรูปทรง ส่วนแผนภาพเรียงตัวแบบหกเหลี่ยมจะง่ายต่อการเลือกดู ดังรูป 3.6 แสดงแผนภาพตาข่ายรูปทรงสี่เหลี่ยมและหกเหลี่ยมที่มีขนาด 10×10

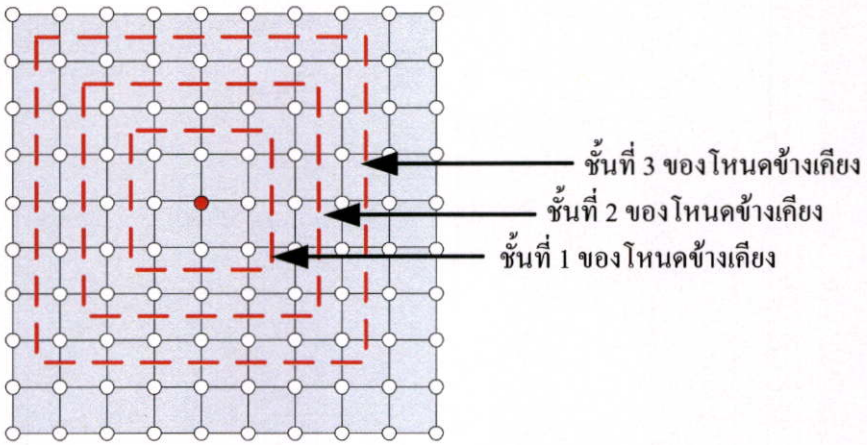


รูปที่ 3.6 แผนภาพตาข่ายรูปทรงหกเหลี่ยมและสี่เหลี่ยม

โหนดข้างเคียงของโหนดชนะ c ใน SOM ที่เวลา t แทนโดย $N_c(t)$ กลุ่มของโหนดข้างเคียงจะแสดงในรูปที่ 3.7 จากตัวอย่างเป็นลำดับชั้นของโหนดข้างเคียงบนตาข่ายรูปทรงหกเหลี่ยม และรูปที่ 3.8 แสดงลำดับชั้นของโหนดข้างเคียงบนตาข่ายรูปทรงสี่เหลี่ยม



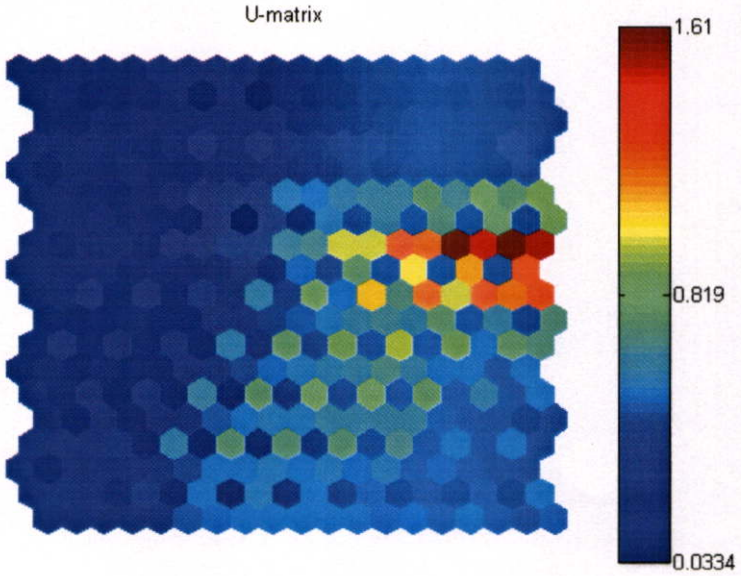
รูปที่ 3.7 แสดงลำดับชั้นของโหนดข้างเคียงในตาข่ายอาร์เรย์รูปทรงหกเหลี่ยม



รูปที่ 3.8 แสดงลำดับชั้นของโหนดข้างเคียงในตาข่ายอาร์เรย์รูปทรงสี่เหลี่ยม

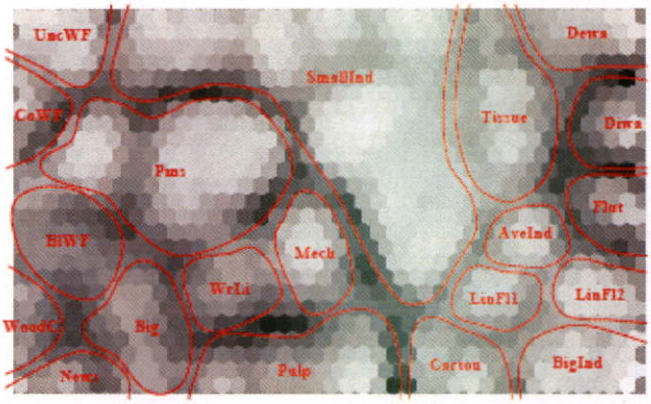
3.7 Unified Distance Matrix (U-Matrix)

U-Matrix เป็นแผนภาพในการเรียกดูกลุ่มข้อมูลในรูปแบบ 2 มิติที่แทนกลุ่มโหนดด้วยระดับสี



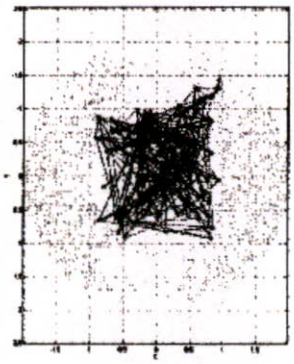
รูปที่ 3.9 แสดง U-matrix แผนภาพขนาด 10×10 ที่แสดงระดับสีแสดงกลุ่มข้อมูล และขอบเขตของข้อมูล

โดยปกติสีเข้มจะแทนระยะห่างน้อย และสีสว่างแทนระยะห่างมาก ดังรูป 3.9 แสดง U-Matrix ที่อธิบายด้วยสี ซึ่งสามารถวิเคราะห์จากกลุ่มสีน้ำเงินเข้มแทนกลุ่มข้อมูลหนาแน่นที่มีความเหมือนกับข้อมูลอินพุทมากและระดับสีแดงแสดงกลุ่มข้อมูลที่ไม่เหมือนกับข้อมูลอินพุท และรูปที่ 3.10 แสดง U-Matrix อธิบายด้วยโทนสีเทา

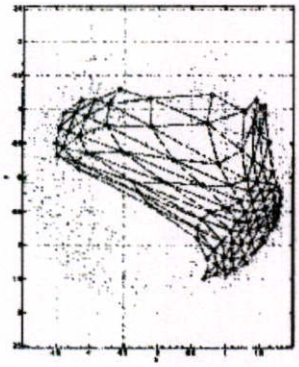


รูปที่ 3.10 U-matrix โดยแสดงด้วยภาพโทนสีเทาและแสดงกลุ่มข้อมูลที่ถูกจัดไว้แล้ว

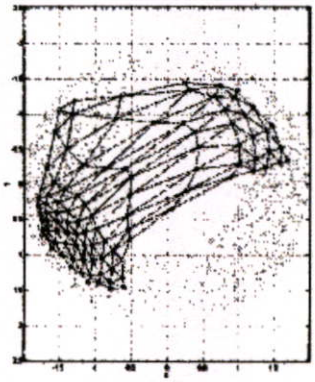
<u>ตัวอย่างที่ 3.1</u> เป็นลำดับการทำงานในระหว่างกระบวนการเรียนรู้ของ SOM ดังรูปที่ 3.11 เมื่อข้อมูลอินพุทเวคเตอร์ขนาด 2 มิติมีลักษณะการกระจายแบบวงแหวน โดยแทนสัญลักษณ์ '.' โหนดเอาท์พุทแทนด้วยสัญลักษณ์จุด และแต่ละโหนดจะเชื่อมต่อกันกับโหนดข้างเคียงเป็นรูปทรงหกเหลี่ยม โดยเส้นจะแทนการเชื่อมต่อแต่ละโหนด โหนดเอาท์พุตขนาด 10×10 จะเขียนออกมาเป็นแผนภาพที่มีค่าน้ำหนักประจำอยู่แต่ละโหนดและมีอัตราการเรียนรู้ (α) ที่ 0.9 สามารถแสดงโดยแผนภาพที่รอบเวลาแตกต่างกัน



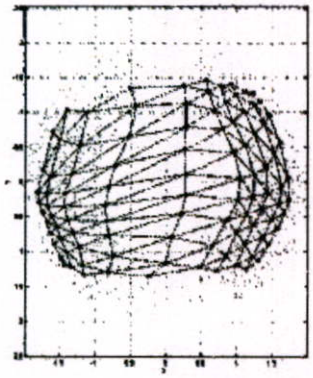
ก) รอบที่ 1



ข) รอบที่ 5,000

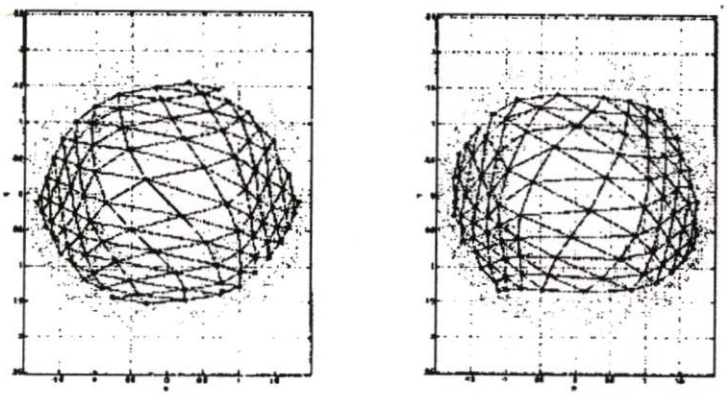


.ค) รอบที่ 10,000



ง) รอบที่ 20,000

รูปที่ 3.11 แสดงกระบวนการเรียนรู้ของโหนดเอาท์พุทขนาด 10×10 ในแต่ละช่วงรอบการทำงาน

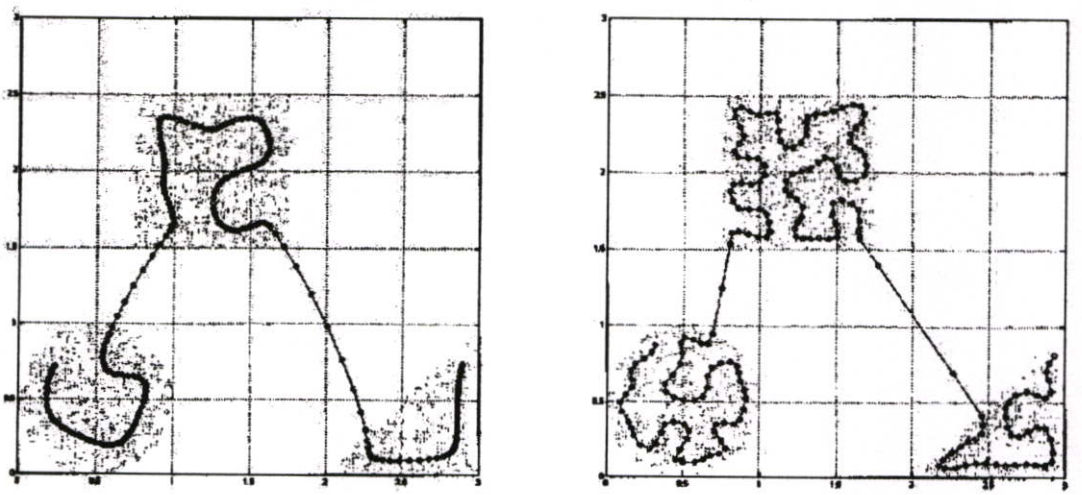


จ) รอบที่ 50,000 ฉ) รอบที่ 100,000

รูปที่ 3.11 (ต่อ) แสดงกระบวนการเรียนรู้ของโหนดเอาท์พุทขนาด 10×10 ในแต่ละช่วงรอบการทำงาน

ในรอบแรก ค่าเวคเตอร์น้ำหนักจะถูกสุ่มขึ้นมาประจำโหนดเอาท์พุทดังรูป 3.11(ก) ในระหว่าง SOM เรียนรู้ โหนดเอาท์พุทจะเริ่มจัดเรียงตัวตามการกระจายของข้อมูลอินพุทดังรูป 3.11 (ข) แผนภาพจะมีการหดและขยายตัว รูปที่ 3.11(ค-ฉ) แสดงโหนดเอาท์พุทที่มีการเปลี่ยนแปลงในระหว่างการเรียนรู้จนกระทั่งเวคเตอร์น้ำหนักมีลักษณะเป็นรูปร่างและรูปร่างนั้นจะมีการเปลี่ยนแปลงน้อยมากที่รอบที่ 100,000 เลเยอร์ของเอาท์พุทก็จะมีรูปร่างเป็นวงแหวนเช่นเดียวกันกับการกระจายข้อมูลของอินพุท

ตัวอย่างที่ 3.2 เป็นการแสดงการเรียนรู้ของ SOM ที่มีข้อมูลอินพุทขนาดหลายมิติไปยังแผนภาพเอาท์พุทขนาดมิติเดียว ข้อมูลตัวอย่างจำนวน 3,000 ชุดขนาด 2 มิติที่แสดงโดยสัญลักษณ์ '.' ในรูปที่ 3.12 แสดงการจัดกลุ่ม 3 กลุ่ม กลุ่มแรกจะเป็นข้อมูลรูปทรงแบบสามเหลี่ยม กลุ่มที่สองเป็นข้อมูลรูปทรงแบบสี่เหลี่ยม และกลุ่มสุดท้ายเป็นรูปทรงแบวงกลม



ก. แผนภาพของโหนดรอบที่ 10,000 ข. แผนภาพของโหนดที่รอบ 100,000

รูปที่ 3.12 แสดงกระบวนการเรียนรู้รอบที่ 10,000 และ 100,000

จากตัวอย่างที่ 3.2 SOM จะมีโหนดขนาด 200 โหนด และมีขนาดแผนภาพ 1 มิติ ในรอบที่ 10,000 จะมีอัตราการเรียนรู้ (α) ที่ 0.9 ผลลัพธ์ของแผนภาพจะแสดงในรูปที่ 3.12(ก) ค่าเวคเตอร์น้ำหนักของโหนดจะถูกปรับและจัดเรียงให้มีลักษณะคล้ายกับข้อมูลอินพุท หลังจาก SOM เรียนรู้รอบที่ 100,000 ค่าอัตราการเรียนจะลดลงมากเพียง 0.02 ผลลัพธ์ที่ได้จะแสดงในรูปที่ 3.12(ข) และสังเกตแผนภาพจะถูกแบ่งออกเป็นสามกลุ่ม

บทที่ 4

การจัดกลุ่มข้อมูลจากแผนภาพ Self-Organizing Map ด้วยเจเนติกอัลกอริธึม

4.1 ปัญหาหลังการเรียนรู้ด้วยวิธี Self-Organizing Map

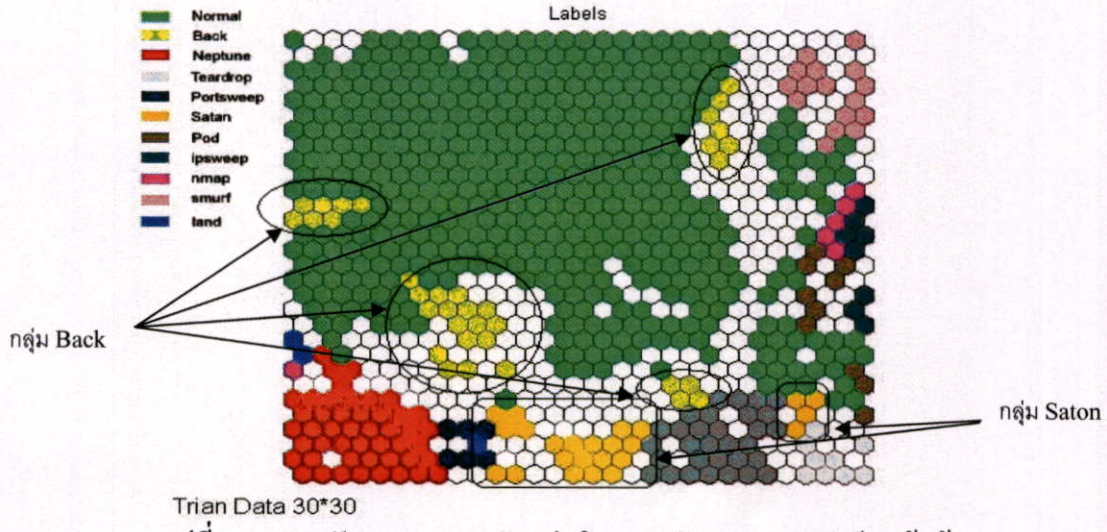
เมื่อได้แผนภาพ SOM หลังการเรียนรู้แล้วในความเป็นจริงข้อมูลที่เหมือนหรือคล้ายกันจะอยู่ในโหนดเดียวกันหรือบริเวณโหนดใกล้เคียง แต่ในทางปฏิบัติไม่เป็นเช่นนั้น เนื่องจากองค์ประกอบการการเรียนรู้ที่เกิดขึ้นมีผลจากคุณลักษณะหลายประการดังนี้

1. คุณสมบัติเฉพาะของข้อมูล เช่น ข้อมูลสองชุดซึ่งต่างชนิดกัน แต่มีลักษณะข้อมูลใกล้เคียงกันมากจนไม่สามารถแยกแยะออกได้

2. ระยะเวลาในการเรียนรู้ของแผนภาพที่จะต้องใช้เวลานานมากจึงจะได้แผนภาพที่สมบูรณ์หรืออาจจะใช้เวลานานจนไม่สิ้นสุดก็ได้

3. ขนาดของแผนภาพไม่เหมาะสมกับข้อมูลที่นำมาจัดกลุ่ม ซึ่งมักจะเกิดกับข้อมูลที่มีมิติมากๆ ซึ่งจะพบปัญหานี้ได้บ่อยครั้ง

ดังนั้นแผนภาพ SOM ที่ผ่านกระบวนการเรียนรู้แล้วจะยังไม่สามารถบอกกลุ่มของข้อมูลได้อย่างชัดเจนดังรูป 4.1 ซึ่งกลุ่มของข้อมูลกระจายไปตามโหนดต่างๆ ในกรณีที่แผนภาพมีขนาดใหญ่ข้อมูลที่ควรจะอยู่ในกลุ่มเดียวกันอาจจะแยกออกเป็นกลุ่มย่อยอยู่ในโหนดที่ห่างออกไปได้ ทำให้เราไม่สามารถระบุกลุ่มได้อย่างชัดเจน จึงส่งผลให้การเลือกดูข้อมูลจากแผนภาพทำได้ยาก ดังนั้นเราจำเป็นต้องจัดกลุ่มของโหนดในแผนภาพใน SOM หลังจากเสร็จสิ้นกระบวนการเรียนรู้เพื่อให้การสำรวจและการวิเคราะห์ข้อมูลมีประสิทธิภาพมากยิ่งขึ้น



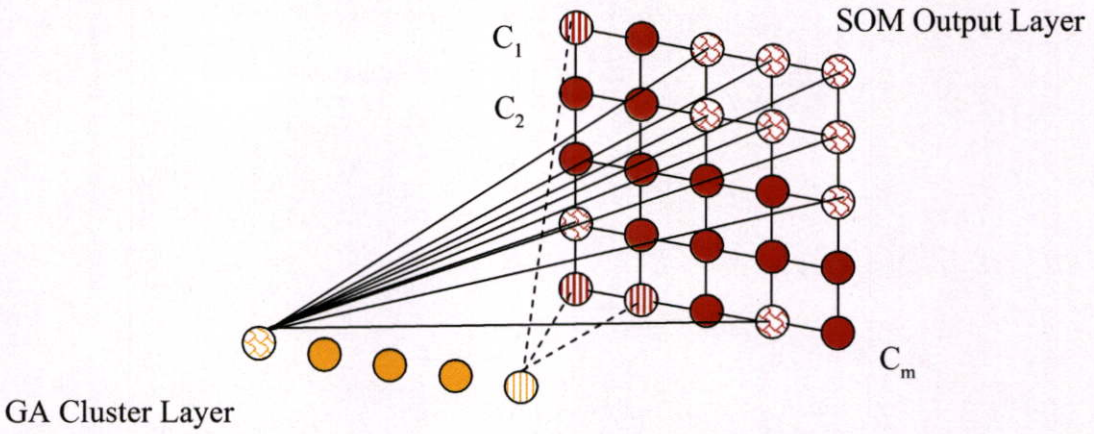
รูปที่ 4.1 แสดงปัญหาของการจัดกลุ่มโหนดหลังกระบวนการเรียนรู้แล้ว

4.2 ขั้นตอนการจัดกลุ่มโหนดด้วยวิธีเจเนติกอัลกอริธึม

การจัดกลุ่มข้อมูลโดยอาศัยหลักการทางการถ่ายทอดพันธุกรรมหรือเจเนติกอัลกอริธึม (Genetic Algorithms: GAs) เป็นวิธีการค้นหาคำตอบโดยมีพื้นฐานมาจากกระบวนการคัดเลือกทางธรรมชาติ (Natural Selection) และกระบวนการคัดเลือกทางพันธุศาสตร์ (Natural Genetic Selection) โดย John Holland เมื่อปี ค.ศ. 1975 โดยนำมาจากแนวคิดที่ว่า สิ่งมีชีวิตทั้งหลายมีทั้งส่วนดีและส่วนที่ด้อย ซึ่งลักษณะที่ดีย่อมมีโอกาสอยู่รอดได้มากกว่าและจะได้รับการสนับสนุนให้มีการถ่ายทอดพันธุกรรมไปยังรุ่นลูกหลานได้มากกว่าลักษณะด้อย สำหรับวิธีเจเนติกอัลกอริธึมนี้จะพิจารณาและดำเนินการจากกลุ่มคำตอบของปัญหาที่ถูกสร้างขึ้นโดยการเข้ารหัส (Coding) คือการแปลงค่าตัวแปรหรือพารามิเตอร์ต่างๆของปัญหาให้อยู่ในรูปโครงสร้างของโครโมโซม (Chromosomes) หรือสายอักขระ (String) ตามที่กำหนดโดยจะทำการคัดเลือกโครโมโซมคำตอบที่มีความเหมาะสมจากกลุ่มของโครโมโซมทั้งหมดด้วยวิธีการสุ่ม และนำโครโมโซมเหล่านี้ไปผ่านกระบวนการคัดเลือกที่เลียนแบบกระบวนการคัดเลือกทางธรรมชาติเพื่อหาโครโมโซมที่มีความเหมาะสมในการอยู่รอด โดยใช้ค่าฟังก์ชันความเหมาะสม (Fitness Function) ที่สอดคล้องกับฟังก์ชันวัตถุประสงค์ (Objective Function) ซึ่งโครโมโซมที่มีความเหมาะสมนี้คือคำตอบที่ดีที่สุดหรือใกล้เคียงคำตอบที่ดีที่สุดของปัญหา (Optimal Solution)

รูปแบบโครโมโซม ในการแก้ปัญหาโดยใช้เจเนติกอัลกอริธึมนั้นจะกำหนดปัญหาเท่ากับโครโมโซมหนึ่งโครโมโซม ซึ่งประกอบไปด้วยยีนส์ลักษณะต่างๆ เปรียบเหมือนกับตัวแสดงค่าคำตอบของปัญหาที่แปรผันไปตามการประยุกต์ใช้งานซึ่งได้แก่ ตัวแปร พารามิเตอร์ เงื่อนไขหรือข้อกำหนดต่างๆ ที่เป็นองค์ประกอบของปัญหา ดังนั้นการกำหนดรูปแบบโครโมโซมของแต่ละปัญหาโดยการแปลงตัวแปร พารามิเตอร์ เงื่อนไขหรือข้อกำหนดต่างๆ ให้อยู่ในรูปแบบของยีนส์บนโครโมโซม ซึ่งก็คือสายอักขระหรือสตริง โดยประกอบไปด้วยบิท (Bit) หรืออักขระ (Character) ซึ่งลักษณะต่างๆ ที่เป็นไปได้ของแต่ละยีนส์คือค่าของบิท (Bit Value) หรือค่าตัวแปรพารามิเตอร์ต่างๆ ที่เป็นไปได้

ในวิทยานิพนธ์นี้ได้นำเสนอการใช้เจเนติกอัลกอริธึมเพื่อจัดกลุ่มโหนดในแผนภาพ SOM จากปัญหาข้างต้นโดยทำการสร้างแผนภาพเพิ่มขึ้นอีก 1 เลเยอร์ เพื่อใช้สำหรับการจัดกลุ่มโหนดที่มีลักษณะของข้อมูลคล้ายกันแต่อยู่ห่างกันดังแสดงในรูปที่ 4.2



รูปที่ 4.2 แสดงการสร้างแผนภาพเลเยอร์ สำหรับการจัดกลุ่มโหนด

การกำหนดรูปแบบของโครโมโซมดังรูปที่ 4.3 โดยให้ความยาวของโครโมโซมเท่ากับจำนวนโหนดของแผนภาพ SOM นั่นคือ $R_k = \{C_1, C_2, C_3, \ldots, C_m\}$ เมื่อ m แทนจำนวนโหนด โดยที่ $C_i \in \{0,1\}$ เช่น

$R_1 = \{11110000\ldots\ldots\}$

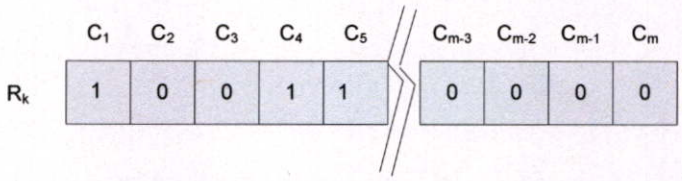
แสดงว่า ใน R_1 จะมีการจับกลุ่ม C_1, C_2, C_3 และ C_4 เข้าด้วยกัน

และ $R_2 = \{1111110000\ldots.\}$

แสดงว่า ใน R_2 จะมีการจับกลุ่ม C_1, C_2, C_3, C_4, C_5 และ C_6 เข้าด้วยกัน

และ $R_3 = \{1111100000\ldots.\}$

แสดงว่า ใน R_3 จะมีการจับกลุ่ม C_1, C_2, C_3, C_4 และ C_5 เข้าด้วยกัน

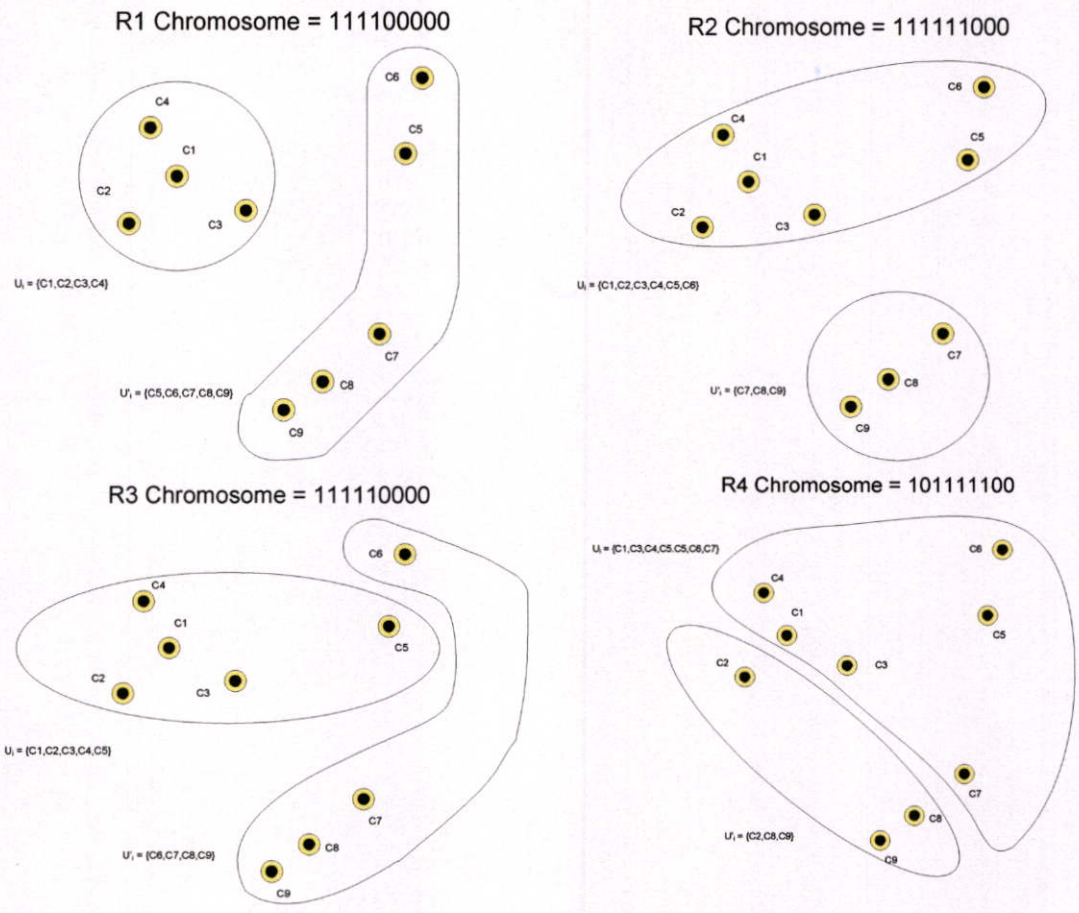


รูปที่ 4.3 แสดงโครงสร้างการสุ่มโครโมโซม และ C_1, C_4 และ C_5 ที่มีโอกาสในการจัดกลุ่มรวมกัน

กำหนดให้ U_k และ $\overline{U_k}$ คือเซตของตำแหน่งบิตในโครโมโซม R_k ที่มีค่าเป็น 1 และ 0 ตามลำดับดังแสดงตามสมการที่ 4.1

$$U_k = \{j \mid j^{th} \text{ bit of } R_k \text{ is } 1\}$$
$$\overline{U_k} = \{j \mid j^{th} \text{ bit of } R_k \text{ is } 0\} \quad (4.1)$$

โดย k แทนลำดับของโครโมโซม



รูปที่ 4.4 ตัวอย่างการสุ่มประชากรของเจเนติกจำนวน 4 โครโมโซมและถูกกำหนดกลุ่มโดยเซตค่าตาม U_k และ $\overline{U_k}$

กำหนดให้เมทริกซ์ $M_{m\times m}$ คือเมทริกซ์แสดงระยะห่างระหว่างโหนดต่างๆ เมื่อ m แทนขนาดของเมทริกซ์ ในแผนภาพ SOM มีลักษณะกราฟสมบูรณ์ (Complete Graph) ดังนั้นสามารถหาเมทริกซ์ได้จากสมการ Adjacency Matrix เพื่อบันทึกระยะห่างระหว่างข้อมูลไว้เป็น Lookup Table และให้ง่ายสำหรับการเขียนโปรแกรม ดังนี้

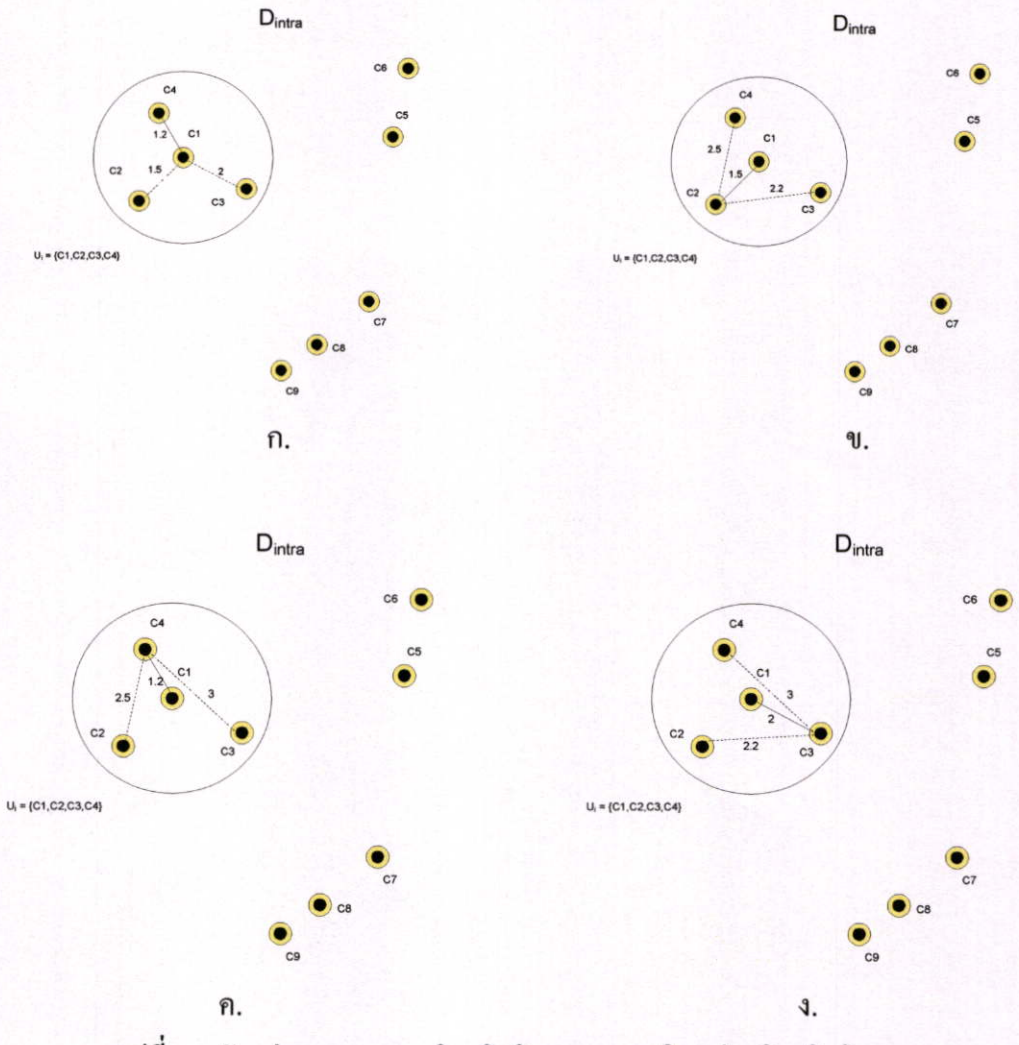
$$D(i,j) = \left\| C_i - C_j \right\|, i \neq j \tag{4.2}$$

เมื่อ i,j แทนลำดับของโหนดในแผนภาพ SOM

ในการกำหนดฟังก์ชันความเหมาะสมพิจารณาจาก 2 ปัจจัยหลักคือ D_{intra} และ D_{inter} ดังแสดงสมการที่ 4.3 และ 4.4 ตามลำดับ

$$D_{\text{int ra}}(R_i) = \max_{j \in U_i} \min_{\substack{k \in U_i \\ k \neq U_i'}} D(j,k) \tag{4.3}$$

ให้ D_{intra} ประกอบไปด้วยเซต $\{C_1, C_2, C_3, \ldots, C_m\}$ cjtสมมติให้ R_1={111100000} ซึ่งโหนดภายใน U_k เท่ากับ $\{C_1, C_2, C_3, C_4\}$ จะได้ระยะห่างของแต่ละโหนดจาก C_1 กับ C_2, C_3 และ C_4 ที่สั้นที่สุดคือ 1.2, จากโหนด C_2 กับ C_1, C_3, C_4 เท่ากับ 1.5, จากโหนด C_3 กับ C_1, C_2, C_4 เท่ากับ 1.2 และจากโหนด C_4 กับ C_1, C_2, C_3 เท่ากับ 2 ซึ่งจะได้ค่าสูงสุดของระยะห่างที่สั้นที่สุด เท่ากับ 2 ดังรูปที่ 4.5

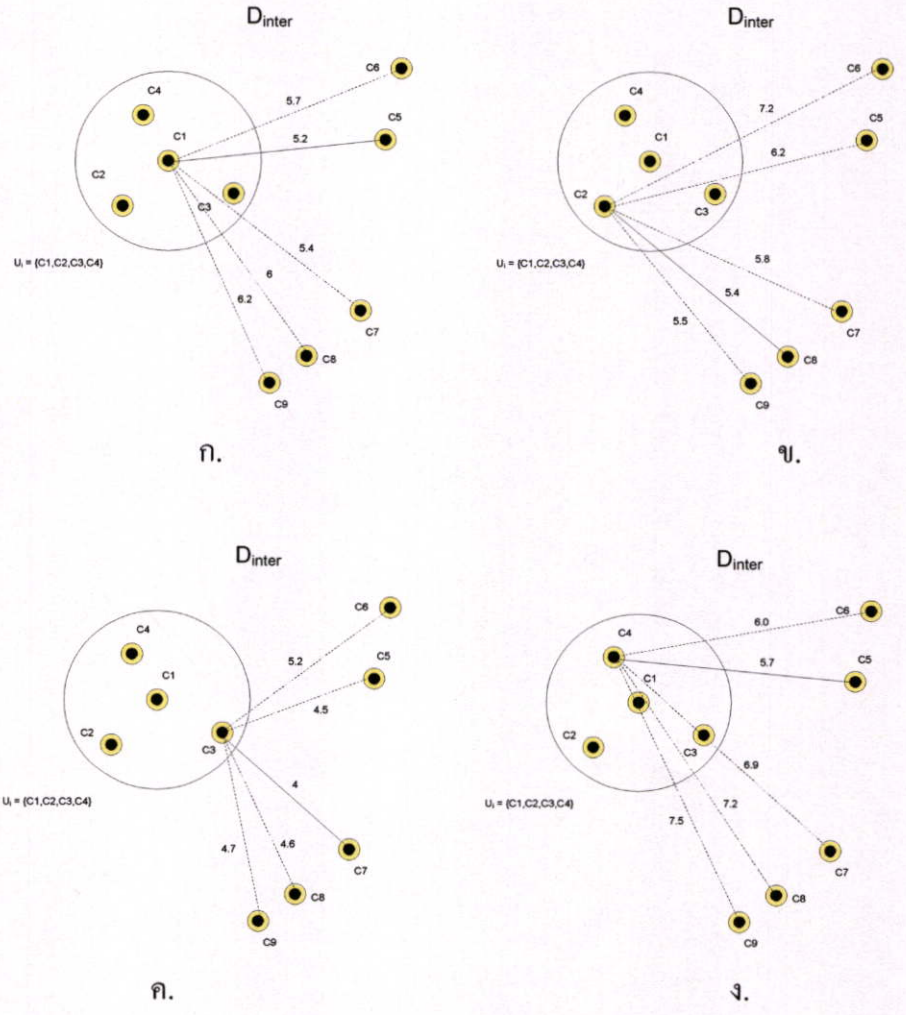


รูปที่ 4.5 ตัวอย่างแสดง D_{intra} โครโมโซม ของ U_k ในแต่ละโครโมโซม R_k

และ D_{inter} แทนระยะทางที่สั้นที่สุดระหว่างซับเซตที่อยู่ภายใน U_k กับซับเซตที่อยู่ภายนอก U_k

$$D_{\text{inter}}(R_i) = \min_{\substack{j \in U_i \\ k \in U_i'}} D(j,k) \tag{4.4}$$

จากรูปที่ 4.6 ให้โครโมโซม R_1={111100000} เซตของ U_k จะประกอบด้วย {C_1, C_2, C_3, C_4} และ $\overline{U_k}$ มี {C_5, C_6, C_7, C_8, C_9} D_{inter} จะเป็นการหาระยะห่างที่สั้นที่สุดของกลุ่มภายใน U_k กับ $\overline{U_k}$ ซึ่งเป็นการหา C_1 กับ {C_5, C_6, C_7, C_8, C_9}จะเท่ากับ 5.2 , C_2 กับ {C_5, C_6, C_7, C_8, C_9}เท่ากับ 5.4, C_3 กับ {C_5, C_6, C_7, C_8, C_9} เท่ากับ 4 และ C_4 กับ {C_5, C_6, C_7, C_8, C_9} เท่ากับ 5.7 ดังนั้นเมื่อพิจารณาผลลัพธ์ระยะห่างความเหมือนที่สั้นที่สุดเกิดขึ้นที่ C_3 กับ C_7 ที่มีค่าเท่ากับ 4 ดังรูป 4.6 ค.



รูปที่ 4.6 ตัวอย่างแสดง D_{inter} เลือกค่าระยะห่างที่น้อยที่สุด จะได้เท่ากับ C_3 และ C_7

เมื่อ $D_{\text{intra}}(R_k)$ ระบุคความเหนียวแน่นของโหนดภายในโครโมโซม R_k โดยที่ *D(i,j)* คือ ระยะห่างระหว่าง Ci,Cj $\in U_k$ และ $D_{\text{inter}}(R_k)$ แทนระยะทางที่สั้นที่สุดระหว่าง Ci$\in U_k$, Cj$\in \overline{U_k}$

ตัวอย่างเช่น ถ้า R_k มีค่าบิตเป็น 0 ทั้งหมด $D_{intra}(R_k)$ และ $D_{inter}(R_k)$ จะมีค่าเป็น 0 ถ้า R_k มีค่าบิตเป็น 1 เพียงแค่ 1 บิต ทั้ง $D_{intra}(R_k)$ และ $D_{inter}(R_k)$ จะมีค่าเป็น 0

ดังนั้นให้ฟังก์ชันความเหมาะสมสามารถนิยามได้ดังสมการที่ 4.5

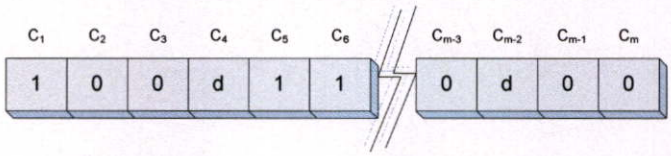
$$fitness(R_k) = w \times D_{inter}(R_k) - D_{intra}(R_k) \tag{4.5}$$

เมื่อ w $\in[1,3]$ แทนค่าน้ำหนักเพื่อกำหนดความสำคัญ [4] ถ้าค่าน้ำหนักมากแสดงว่าให้ความสำคัญกับ D_{inter} สูง

ถ้าค่า w มีค่ามากแล้วจะให้ความสำคัญกับ D_{inter} หมายถึงโอกาสจะเกิดกลุ่มน้อยลงไป

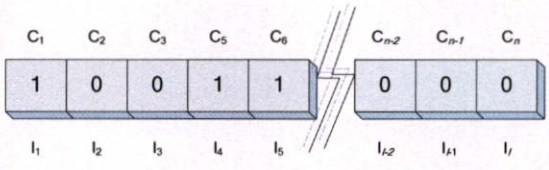
และ ถ้าค่า w มีค่าน้อยแล้วจะให้ความสำคัญกับ D_{intra} หมายถึงโอกาสจะจับกลุ่มได้แน่นขึ้น แต่ถ้าน้อยเกินไปจะทำให้สมาชิกบางตัวหลุดออกจากกลุ่มได้เช่นเดียวกัน

เนื่องจากแผนภาพ SOM อาจจะมีโหนดที่ไม่มีข้อมูลใดๆ ตกในโหนดนั้นๆ เลยหรือในบางครั้งจะเรียกว่า "โหนดตาย" (Dead Node) ในทางปฏิบัติจึงไม่จำเป็นต้องคำนึงถึงโหนดตายในการกำหนดรูปแบบของโครโมโซม ดังนั้นจึงทำให้ความยาวของโครโมโซมมีขนาดสั้นลง ตัวอย่างเช่น แผนภาพขนาดสองมิติ 30×30 จะมีจำนวนความยาวเท่ากับ 900 บิตและโหนดตาย 200 โหนด ซึ่งสามารถจัดรูปแบบของโครโมโซมใหม่ได้ โดยไม่พิจารณาโหนดตาย จำนวนความยาวของโครโมโซมจะลดลงเหลือ 700 บิต เป็นต้น ดังนั้นสามารถกำหนดรูปแบบของโครโมโซมใหม่ได้ โดยให้โหนดตายแทนด้วย Don't Care ดังรูป



รูปที่ 4.7 การกำหนดรูปแบบใหม่ของโครโมโซมโดยไม่พิจารณาโหนดตาย

จากที่กล่าวมาข้างต้นการจัดกลุ่มโหนดจำเป็นต้องบันทึกตำแหน่งโหนดของแต่ละโหนดบนแผนภาพ SOM จึงต้องเพิ่มตัวแปรดัชนีเพื่อบันทึกตำแหน่งบิตของโครโมโซมด้วย ส่งผลให้รูปแบบโครโมโซมนั้นจะสั้นลงได้ดังรูป



เมื่อ $n < m$

รูปที่ 4.8 การกำหนดรูปแบบของโครโมโซมที่สั้นลง

และสามารถหาความยาวของโครโมโซมได้โดย

$$l = m - d \tag{4.6}$$

เมื่อ C แทนตำแหน่งของโหนดบนแผนภาพ SOM

I แทนลำดับของบิตในโครโมโซม

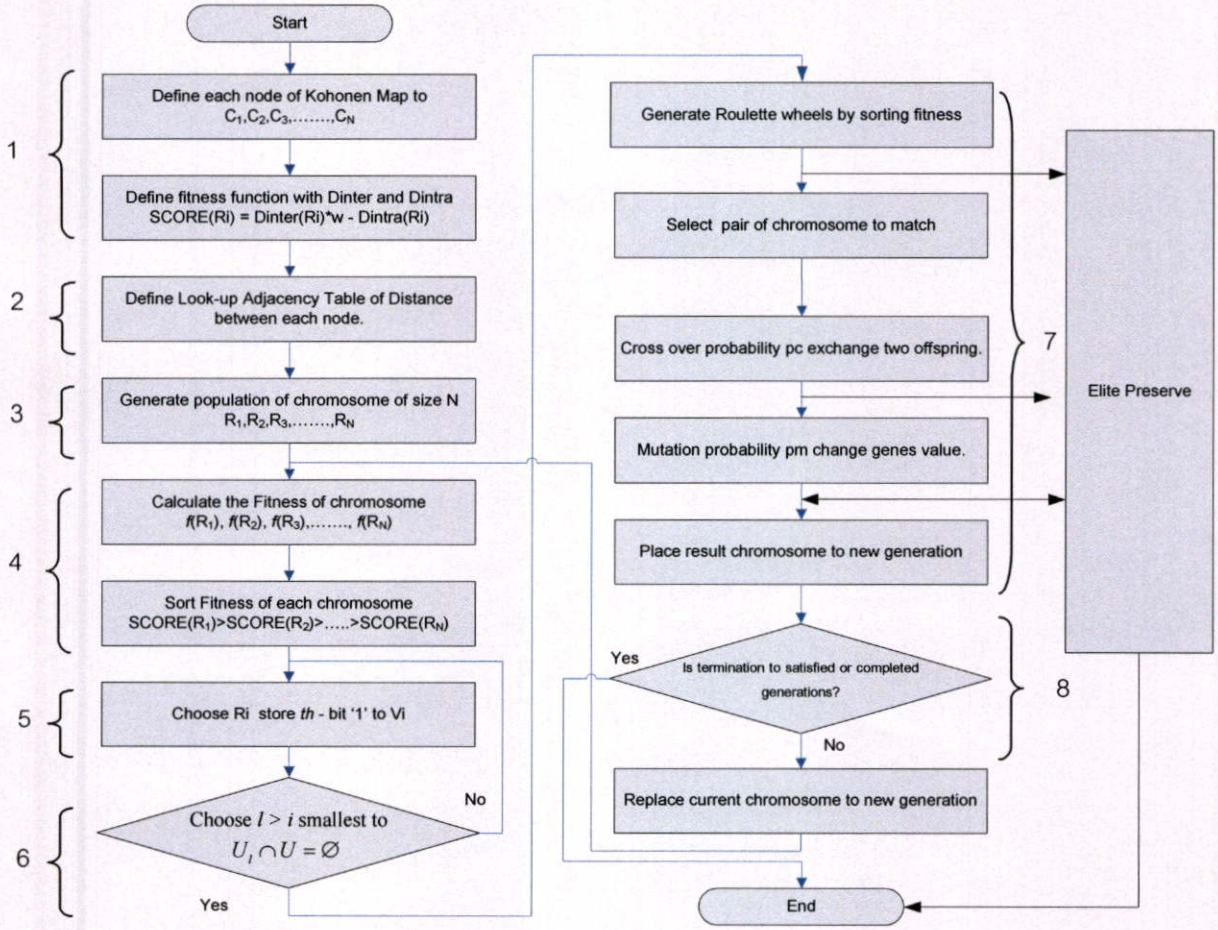
l แทนความยาวของโครโมโซม

n แทนจำนวนโหนดชนะ

m แทนจำนวนโหนดที่เกิดขึ้นบนแผนภาพ SOM และ

d แทนจำนวนโหนดตายบนแผนภาพ SOM

4.3 ขั้นตอนการจัดกลุ่มด้วยวิธีเจเนติกอัลกอริธึม



รูปที่ 4.9 การทำงานของการจัดกลุ่มข้อมูลจากแผนภาพโคโฮเนนด้วยวิธีเจเนติก

ในการจัดกลุ่มโดยใช้เจเนติกอัลกอริธึม จะมีขั้นตอนดังแสดงในรูปที่ 4.9

ขั้นที่ 1 กำหนดให้โหนดแต่ละโหนดของแผนภาพ SOMแทนด้วย $C_1, C_2, C_3, \ldots, C_N$

ขั้นที่ 2 สร้างตารางแสดงระยะห่างความเหมือนของข้อมูลในแต่ละโหนด

ขั้นที่ 3 ในการทดลองจะสร้างโครโมโซมจำนวน *popsize*

ขั้นที่ 4 หาค่าความเหมาะสมของแต่ละโครโมโซมโดยใช้สมการที่ 4.5 แล้วนำมาเรียงลำดับตามค่าความเหมาะสมจากมากไปหาน้อย ดังตารางที่ 4.1 และกำหนดให้ k=1, U=∅

ตารางที่ 4.1 แสดงการจัดเรียงค่าความเหมาะสมสูงสุด

Chromosome(R_i)	Subset Cluster {$C_1, C_2, C_3, \ldots\ldots, C_m$}	Fitness Value
R_2	11111110000…..	20
R_{10}	10111001101…..	15
R_{25}	11001111000…..	13
R_{15}	00001110110…..	12
R_{30}	10101011010…..	9
⋮	⋮	⋮
R_n	1111111000…..	0.2

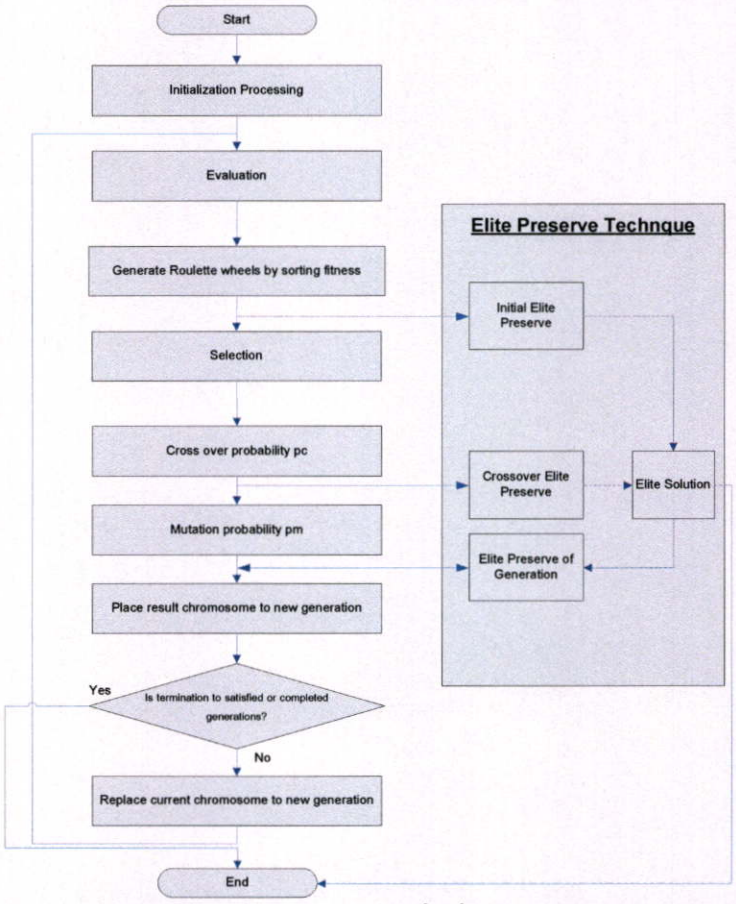
ขั้นที่ 5 ทำการ Reproduction Cross Over และ Mutation ระหว่างขั้นตอนนี้จะเก็บคำตอบที่ดีที่สุดไว้ (Elite preserve solution: ดูหัวข้อที่ 4.4)

การทำ *Reprod*uction หรือการกำเนิดกลุ่มประชากร สามารถสร้างขึ้นได้หลายวิธีด้วยกัน วิธีที่ง่ายวิธีการหนึ่งคือการสร้างจากวงล้อรูลเล็ทที่มีจำนวนช่องเท่ากับจำนวนประชากรของโคโมโซม และหาขนาดของช่องเป็นสัดส่วนกับค่าความเหมาะสม

การทำ Cross Over เริ่มต้นโดยการสุ่มเลือกประชากรหรือโครโมโซม มาสองตัวให้เป็นคู่เป็นโครโมโซมพ่อและแม่ แล้วทำการสุ่มตำแหน่งที่ทำการครอสโอเวอร์ เพื่อสลับตำแหน่งระหว่างโครโมโซมพ่อและแม่ ซึ่งจะได้ประชากรรุ่นถัดไปจำนวน 2 โครโมโซม ในแต่ละรอบการทำครอสโอเวอร์จะพิจารณาจากค่า Pc และเกิดประชากรเพิ่มเป็น 2 เท่า

การทำ Mutation เป็นสิ่งที่จำเป็น ถึงแม้ว่าการกำเนิดกลุ่มประชากรและครอสโอเวอร์ช่วยให้การค้นหาเป็นไปอย่างมีประสิทธิภาพ แต่ในบางครั้งก็มีการสูญเสียส่วนที่สำคัญของข้อมูลไป (ค่า 1 หรือ 0 ในบางตำแหน่ง) การทำมิวเตชั่นจะป้องกันส่วนที่สูญเสียที่ไม่อาจเรียกคืนได้ ในบางครั้งการหาคำตอบของเจเนติกอัลกอริธึมแบบง่ายๆ คำตอบอาจติดอยู่ใน Local optimal การทำมิวเตชั่นด้วยอัตราส่วนที่เหมาะสมจะทำให้คำตอบสามารถหลุดออกจาก Local Optimal หรืออาจกล่าวได้ว่าการดำเนินการของมิวเตชั่นเป็นการเปลี่ยนแปลงค่าตำแหน่งโครโมโซมแบบสุ่ม จาก

เนื่องจากโครโมโซมคำตอบที่ได้จากการทำครอสโอเวอร์และการทำมิวเตชั่น อาจเป็นคำตอบที่แย่กว่าคำตอบที่เคยปรากฏในเจเนอเรชั่นที่ผ่านๆ มา ดังนั้นจึงต้องมีการเก็บค่าที่ดีที่สุดเอสไว้เพื่อใช้เปรียบเทียบกับค่าที่ดีที่สุดของโครโมโซมคำตอบชุดใหม่ ถ้าหาก Elite Preserve ให้ค่าความเหมาะสมที่ดีกว่าค่าที่ดีที่สุดของโครโมโซมชุดใหม่ก็ให้เอา Elite Preserve แทนที่ค่าที่แย่ที่สุด ทั้งนี้เพื่อให้โครโมโซมคำตอบที่ดีที่สุดเท่าที่พบยังคงอยู่ในกระบวนการของ GAs ต่อไป ดังรูปที่ 4.11 แสดงขั้นตอนการเก็บคำตอบที่ดีที่สุดโดยแทรกระหว่างขั้นตอนการกำเนิดกลุ่มประชากร



รูปที่ 4.11 โครงสร้างการทำงานการเก็บค่าคำตอบที่ดีที่สุดในเจเนติกอัลกอริธึม เทคนิคการเก็บคำตอบที่ดีที่สุดจะถูกนำไปใช้งานระหว่างกระบวนการของ GAs 3 ครั้ง

1) **เทคนิคเก็บคำตอบเริ่มต้น** (Initial Elite Preserve)

เป็นจุดเริ่มต้นของเทคนิคการเก็บคำตอบที่ดีที่สุด ซึ่งจะกระทำครั้งแรกเพียงครั้งเดียว ภายหลังจากการสร้างโครโมโซมคำตอบเบื้องต้นในตอนต้นของกระบวนการ GAs และผ่านการประเมินค่าเรียบร้อยแล้ว ค่าความเหมาะสมของโครโมโซมแต่ละตัวที่ได้จากการประเมินค่าจะถูกเรียงลำดับจากมากไปหาน้อย โครโมโซมคำตอบเพียงตัวเดียวที่ให้ค่าเหมาะสมมากที่สุดก็จะถูกเลือกไปเป็นคำตอบที่ดีที่สุดที่เก็บไว้ จากนั้นโครโมโซมคำตอบทั้งหมดรวมทั้งตัวที่เลือกไปเป็นคำตอบที่ดีที่สุดจะเข้าสู่ขั้นตอนต่างๆ ของ GAs ต่อไป

2) **เทคนิคการเก็บคำตอบหลังครอสโอเวอร์** (Crossover Elite Preserve)

เป็นเทคนิคการเก็บค่าที่ดีที่สุดที่ใช้ภายหลังจากเสร็จสิ้นกระบวนการครอสโอเวอร์แล้ว ทั้งนี้เนื่องจากว่าโครโมโซมคำตอบที่ได้จากการครอสโอเวอร์อาจเป็นคำตอบที่ดีกว่าคำตอบอื่นๆ ที่เคยพบมา แต่เมื่อผ่ากระบวนการมิวเตชั่นแล้ว โครโมโซมคำตอบตัวนี้จะเปลี่ยนไปและอาจให้ได้คำตอบที่ดีน้อยกว่าเดิม ดังนั้นเพื่อป้องกันไม่ให้โครโมโซมคำตอบที่ดีหลังจากการทำครอสโอเวอร์สูญหายไป จึงต้องทำการประเมินค่าโครโมโซมคำตอบภายหลังการครอสโอเวอร์ของโครโมโซมทั้งหมด แล้วนำโครโมโซมคำตอบที่ดีที่สุดภายหลังการครอสโอเวอร์ไปเปรียบเทียบกับ Elite Preserve Solution ถ้าหากโครโมโซมคำตอบภายหลังครอสโอเวอร์ดีกว่า ก็ให้โครโมโซมคำตอบที่ดีที่สุดนั้นไปเป็น Elite Preserve Solution แทน แต่ถ้า Elite Preserve Solution นั้นดีกว่า ก็ให้นำโครโมโซมคำตอบภายหลังจากครอสโอเวอร์ของโครโมโซมทั้งหมด ไปผ่านกระบวนการมิวเตชั่นต่อไป

ตัวอย่างเช่นภายหลังการทำครอสโอเวอร์มีโครโมโซมคำตอบ 10 ตัว ซึ่งเมื่อนำไปผ่านกระบวนการประเมินค่าแล้วได้ค่าความเหมาะสมของโครโมโซมคำตอบแต่ละตัวเป็น 2 6 8 7 9 4 5 12 6 และ 4 ค่าความเหมาะสมที่ดีที่สุดในโครโมโซม 10 ตัวนี้ คือค่า 12 ของโครโมโซมคำตอบตัวที่ 8 ก็เอาค่า 12 นี้ ไปเปรียบเทียบกับค่าความเหมาะสมของ Elite Preserve Solution ถ้าหากค่าดังกล่าวน้อยกว่า 12 ก็ให้เอาโครโมโซมคำตอบตัวที่ 8 นี้ไปใช้เป็น Elite Preserve Solution ตัวใหม่แทน แต่ถ้าค่าดังกล่าวมากกว่าหรือเท่ากับ 12 ของ Elite Preserve Solution ไว้แล้วนำโครโมโซมคำตอบทั้ง 10 ตัวนี้ไปทำการมิวเตชั่นต่อไป

3) **เทคนิคการเก็บคำตอบของเจเนอเรชั่น** (Elite Preserve of Generation)

เป็นเทคนิคการเก็บค่าที่ดีที่สุดที่ใช้ภายหลังการทำมิวเตชั่น ซึ่งถือว่าเป็นการเก็บค่าที่ดีที่สุดของเจเนอเรชั่นนั้นๆ ด้วย การเก็บค่าที่ดีที่สุดของเจเนอเรชั่นจะช่วยให้ได้คำตอบที่ดีที่สุดเท่าที่เคยปรากฏขึ้นมายังคงมีอยู่ในเจเนอเรชั่นต่อไป การเก็บค่าในขั้นตอนนี้จะทำหลังจากที่มีการทำมิว-เตชั่นเรียบร้อยแล้ว โครโมโซมคำตอบที่ได้ภายหลังการทำมิวเตชั่นของจำนวนประชากรทั้งหมดจะถูกประเมินค่า จากนั้นก็ให้เอาโครโมโซมคำตอบหลังที่ดีที่สุดจากการทำมิวเตชั่น มาเปรียบเทียบกับ Elite Preserve Solution เช่นเดียวกับในขั้นตอนของการเก็บคำตอบที่ดีที่สุดหลังการทำครอสโอเวอร์ แต่แตกต่างกันตรงที่มีการเอา Elite Preserve Solution มาแทนที่คำตอบที่แย่ที่สุดของโครโมโซมของคำตอบชุดนี้ เมื่อ Elite Preserve Solution เป็นคำตอบที่ดีกว่า

ตัวอย่างเช่นภายหลังการทำมิวเตชั่นได้โครโมโซมคำตอบจำนวน 10 ตัว ที่มีค่าความเหมาะสมเป็น 5 6 8 3 1 9 4 6 7 และ 7 จะได้ว่าค่าความเหมาะสมที่ดีที่สุดคือ 9 ของโครโมโซมคำตอบตัวที่ 6 ซึ่งถ้าค่าความเหมาะสมของ Elite Preserve Solution น้อยกว่า 9 โครโมโซมคำตอบ

ตัวที่ 6 จะกลายเป็น Elite Preserve Solution ตัวใหม่ แต่ถ้าค่าความเหมาะสมของ Elite Preserve Solution มากกว่า 9 ก็ให้ตัดโครโมโซมคำตอบตัวที่ 5 ซึ่งมีค่าความเหมาะสมต่ำสุดนั้นทิ้งไป แล้วเอาโครโมโซมคำตอบที่เป็น Elite Preserve Solution ขณะนั้นไปใส่แทน โครโมโซมคำตอบที่ได้ภายหลังจากขั้นตอนนี้จะกลายเป็นโครโมโซมคำตอบของพ่อแม่ที่แท้จริงในเจเนอเรชั่นถัดไป

4.5 การวัดประสิทธิภาพหลังการจัดกลุ่มโหนด

ในการวัดประสิทธิภาพการทำงานหลังการจัดกลุ่มโหนด โดยเลือกใช้ตัววัดค่าเอนโทรพี (Entropy) โดยตัววัดเอนโทรพีเป็นตัววัดเพื่อวัดการทับซ้อนกันของกลุ่มข้อมูล

เอนโทรพี (Entropy)

ตัววัดเอนโทรพีเป็นการวัดการซ้อนทับกันของกลุ่ม ซึ่งเป็นการวัดคุณภาพอย่างหนึ่งของการทำงานอย่างหนึ่ง โดยค่าเอนโทรพีที่ดีที่สุดเมื่อมีค่าเป็น 0 นั่นคือไม่มีการซ้อนทับกันของข้อมูลเลยหรือผลลัพธ์ของแต่ละคลัสเตอร์ประกอบด้วยข้อมูลที่เป็นสมาชิกเพียงคลัสเตอร์เดียวเท่านั้น การหาค่าเอนโทรพี เริ่มต้นจากการคำนวณหาค่าความน่าจะเป็นที่สมาชิกจากผลการทดลองของคลัสเตอร์ i เป็นสมาชิกของคลัสเตอร์ j ซึ่งแทนความน่าจะเป็นนี้ด้วย p_{ij} ทั้งนี้ค่าเอนโทรพีของแต่ละคลัสเตอร์หาได้จาก

$$E_j = -\sum_i p_{ij} \log(p_{ij}) \tag{4.8}$$

และผลรวมของค่า Entropy

$$E_{cs} = \sum_{j-1}^{m} \frac{n_j \times E_j}{n} \tag{4.9}$$

เมื่อ p_{ij} เป็นความน่าจะเป็นของสมาชิก j ในกลุ่ม i

n_j เป็นเอกสารในกลุ่ม j

m เป็นจำนวนกลุ่มข้อมูล

และ n เป็นจำนวนของข้อมูลทั้งหมด

ถ้าค่าเอนโทรพีที่ได้มีค่าน้อยจะบ่งบอกว่าผลจากจัดกลุ่มข้อมูลมีการซ้อนทับของข้อมูลน้อยมาก และในทางตรงข้ามค่าเอนโทรพีจะมีค่ามากเมื่อผลจากการจัดกลุ่มข้อมูลมีการซ้อนทับกันของข้อมูลมาก

ตัวอย่างเช่น เรามีข้อมูลที่ถูกจัดกลุ่มแล้วโดยผู้เชี่ยวชาญ 3 กลุ่ม ดังนี้ C1={d1, d2, d3}, C2={ d4, d5, d6}, C2={ d7, d8, d9, d10} หลังจากที่เราจัดกลุ่มโดยใช้อัลกอริธึมใด ๆ แล้วได้ดังนี้ K1= {d1, d2, d3}, K2= {d4, d6, d7}, K3= {d5, d8, d9, d10} เราสามารถวัดค่าเอนโทรพีของ K1 K2 K3 ได้ดังนี้

E1 = -(3/3 * log(3/3)) = 0

E2 = - (2/3*log(2/3)) +(1/3*log1/3)) = -(-0.11739-0.15904) = 0.27643

E3 = - (3/4*log(3/4))+(1/4*log(1/4)) = -(-0.09370-0.15051) = 0.24421

เราสามารถหาค่าเอนโทรพีรวมของทั้ง 3 กลุ่มได้ดังนี้

Ecs = 0 + (3*0.27643/10) + (4*0.24421/10) = 0.180613

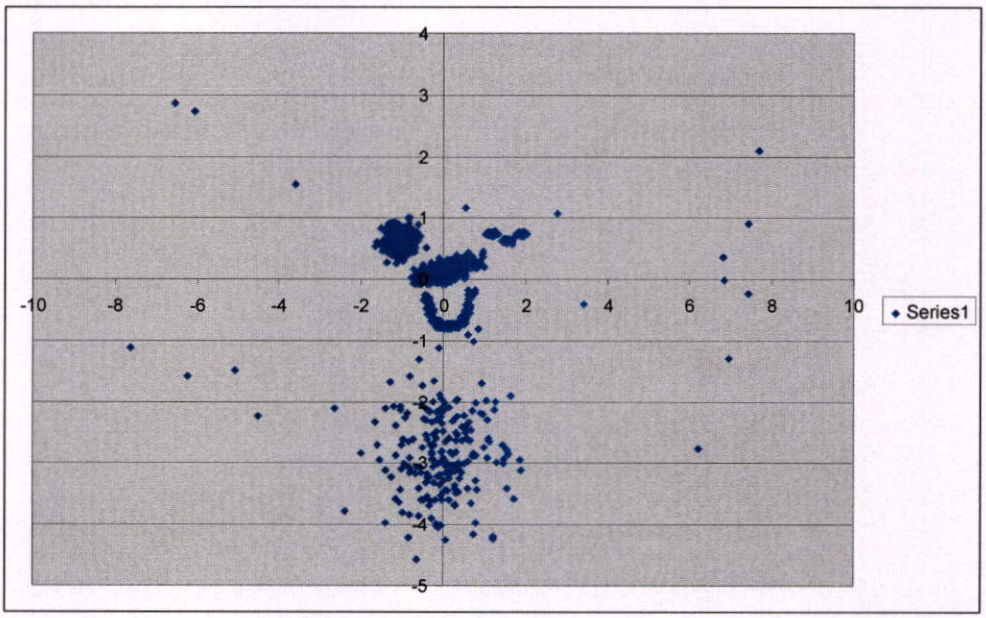
บทที่ 5

ผลการทดลอง

ในบทนี้จะกล่าวถึงขั้นตอนการทดลองของงานวิจัยในวิทยานิพนธ์ เพื่อแก้ปัญหาการจัดกลุ่มข้อมูลที่ไม่สมบูรณ์หลังจากการเรียนรู้ SOM ด้วยวิธีเจเนติกอัลกอริธึม ในการทดลองจะทำการเปรียบเทียบประสิทธิภาพการจัดกลุ่มข้อมูลกับวิธีต่างๆ ดังนี้ เปรียบเทียบการจัดกลุ่มด้วยวิธี K-Mean เพียงอย่างเดียว เปรียบเทียบวิธี Hierarchical Clustering เพียงอย่างเดียว เปรียบเทียบวิธี SOM กับวิธี K-Mean และเปรียบเทียบวิธี SOM กับวิธี Hierarchical Clustering ในการเปรียบเทียบจะใช้ตัววัดประสิทธิภาพจากค่าเอนโทรพีเพื่อตรวจสอบการซ้อนทับกัน

5.1 การทดลองชุดข้อมูลที่ 1 จัดกลุ่มข้อมูลสองมิติ

การทดลองที่ 1 เพื่อทดลองความเร็วในการจัดกลุ่มเปรียบเทียบระหว่างวิธีการจัดกลุ่มสองระดับโดยการจัดกลุ่มหลังการเรียนรู้ SOM ด้วยเจเนติกอัลกอริธึมและการจัดกลุ่มด้วยวิธีเพียงระดับเดียว ทำการทดลองกับชุดข้อมูล Clown [18] มีรูปร่างข้อมูลคล้ายตัวตลก ที่มีขนาดสองมิติ จำนวน 2,220 ชุดข้อมูล ดังรูปที่ 5.1



รูปที่ 5.1 แสดงชุดข้อมูล Clown

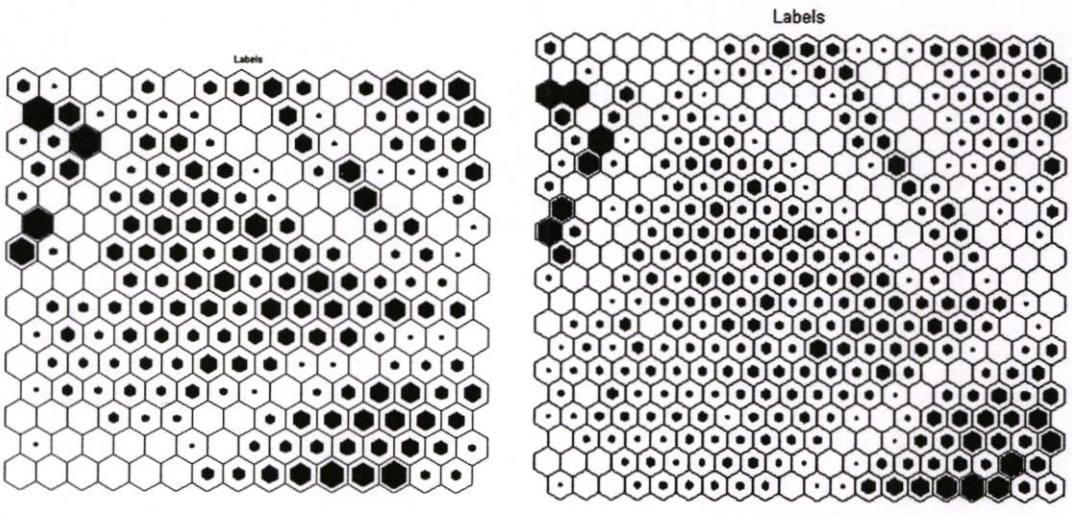
ชุดข้อมูล Clown สามารถแบ่งข้อมูลออเป็นได้จำนวน 7 กลุ่มดังนี้

1. กลุ่มข้อมูลย่อย 3 กลุ่ม (ตาขวา)
2. กลุ่มข้อมูลรูปทรงกลม (ตาซ้าย)
3. กลุ่มข้อมูลรูปทรงวงรี (จมูก)
4. กลุ่มข้อมูลไม่เป็นรูปทรง (มีลักษณะแบบตัว U)
5. กลุ่มข้อมูลขนาดพื้นที่ใหญ่

5.1.1 การจัดกลุ่มข้อมูลด้วย SOM

กำหนดให้แผนภาพมีขนาด 15×15 และ 20×20 โดยไม่กำหนด Label ให้อัตราการเรียนรู้มีค่าเท่ากับ 0.2 น้ำหนักประจำแต่ละโหนดจะเป็นตัวแทนของกลุ่มอินพุทข้อมูลที่ตกในโหนด จากรูปที่ 5.2 แสดงข้อมูลที่ตกลงในแต่ละโหนดหรือโหนดชนะ โหนดใดที่มีสีดำเต็มโหนด แสดงให้เห็นว่ามีข้อมูลตกอยู่ที่โหนดนั้นมาก

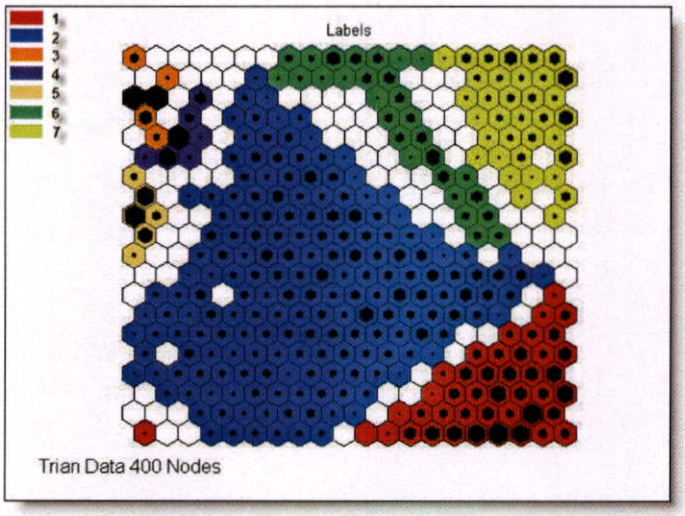
เมื่อพิจารณาแผนภาพขนาด 15×15 ด้วยการเรียกดู (Visualization) จะพบจำนวนกลุ่มที่ไม่ชัดเจนอยู่ 2 กลุ่มจากรูป 5.2 (ก) บริเวณบนซ้าย ซึ่งมีโอกาสที่จะสรุปผลของการจัดกลุ่มได้เพียง 6 กลุ่มเท่านั้น สาเหตุมาจากข้อมูลใกล้กันมากจำนวน 2 กลุ่ม ดังรูป 5.2 (ก) สามารถสรุปการเรียกดูกลุ่มข้อมูลได้ดังนี้ กลุ่มแรกตกที่โหนดที่ 2 และข้อมูลกลุ่มที่ 2 ไปตกโหนดที่ 33 มากจนเกินไปทำให้มีการขยายออกจากโหนดนั้น (การนับจำนวนโหนดให้นับเป็นคอลัมน์จากคอลัมน์ที่ 1 แถวที่ 1 เป็นโหนดที่ 1 และนับคอลัมน์ที่ 1 แถวที่ 2 เป็นโหนดที่ 2 จนถึงแถวสุดท้ายแล้วเริ่มคอลัมน์ที่ 2 แถวที่ 1 ใหม่เป็นโหนดที่ 16 และนับไปเรื่อยๆ จนครบ 225 โหนด) พิจารณาจากโหนดที่ 17 จะเกิดการซ้อนทับกันของข้อมูลจำนวน 2 กลุ่ม ซึ่งเป็นผลให้พิจารณาผลการจัดกลุ่มได้ไม่ชัดเจน



(ก) (ข)

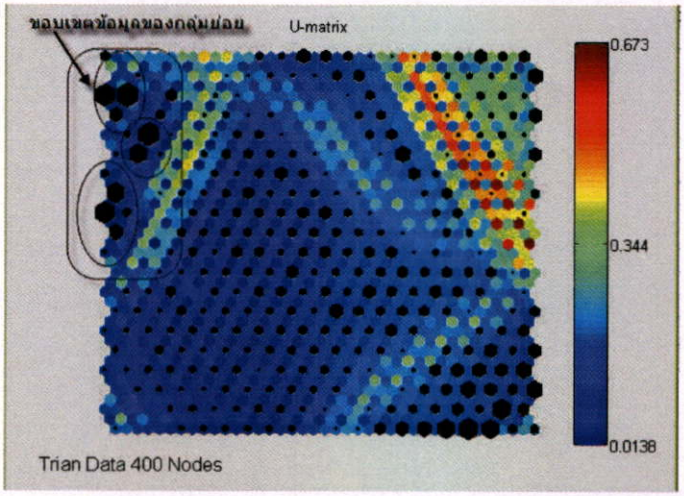
รูปที่ 5.2 แสดงแผนภาพหลังกระบวนการเรียนรู้ด้วยชุดข้อมูล Clown ที่มีแผนภาพขนาด (ก) 15×15 และ (ข) 20×20

แต่เมื่อขยายขนาดของแผนภาพให้ใหญ่ขึ้นเป็นขนาด 20×20 ดังรูป 5.3 (ข) จะพบความชัดเจนของกลุ่มได้จำนวน 7 กลุ่ม การแบ่งกลุ่มข้อมูลจะถูกแบ่งออกอย่างชัดเจนมากยิ่งขึ้นทำให้สามารถสำรวจและเรียกดูได้ง่าย แต่ความเร็วในการจัดกลุ่มจะลดลงเมื่อขยายแผนภาพให้มีขนาดใหญ่ขึ้นเรื่อยๆ เนื่องจากเป็นการเพิ่มการคำนวณให้กับ SOM มากขึ้น นอกจากนี้การขยายแผนภาพใหญ่มากขึ้น ทำให้พบปัญหาข้อมูลแยกตัวกระจายห่างออกไป ดังรูปที่ 5.3



รูปที่ 5.3 การกระจายข้อมูล Clown จากแผนภาพ SOM ขนาด 20×20

จากความชัดเจนของ SOM ไม่เพียงพอในการกำหนดกลุ่มโหนดอื่นๆ รอบๆ โหนดที่มีข้อมูลตกจำนวนมากและมักเกิดขึ้นกับขอบเขตของข้อมูล พิจารณาจากรูปที่ 5.4 แสดงขอบของข้อมูลด้วย U-Matrix จึงต้องทำการหาข้อมูลรอบๆ กลุ่มหลักนั้นเป็นกลุ่มข้อมูลใด ดังนั้นจึงต้องมีการจัดกลุ่มอีกครั้งหนึ่งด้วยวิธีเจเนติกอัลกอริธึม



รูปที่ 5.4 แสดง U-Matrix ของข้อมูลย่อยที่ไม่สามารถบอกกลุ่มได้

5.1.2 การจัดกลุ่มโหนดโดยใช้เจเนติกอัลกอริธึม

จากการทดลองด้วย SOM จะเกิดโหนดชนะจำนวน 308 โหนด และโหนดตายจำนวน 92 โหนด จาก 400 โหนด ดังนั้นการจัดรูปแบบของโครโมโซมในขั้นตอนของเจเนติกอัลกอริธึมจะมีความยาวเท่ากับ 308 บิตตามจำนวนของโหนดชนะ กำหนดให้จำนวนประชากรเท่ากับ 25 โครโมโซม (การพิจารณาจำนวนของโครโมโซมได้จากการทดลองซ้ำ ซึ่งจะพบว่าหลังจากเพิ่มจำนวนของโครโมโซมมากกว่า 25 ตัวจะเกินความจำเป็นในการจัดกลุ่มข้อมูล) กำหนดให้ประชากรไว้จำนวน 50 รุ่น มีการเก็บค่าที่ดีที่สุดเอาไว้เพื่อใช้เปรียบเทียบกับค่าที่ดีที่สุดของโครโมโซมคำตอบชุดใหม่ ให้ค่า Pc และ Pm เท่ากับ 0.8 และ 0.01 ตามลำดับและกำหนดค่าน้ำหนักจากสมการฟังก์ชันความเหมาะสมจากสมการที่ 4.5 เท่ากับ 0.3 ซึ่งเป็นค่าที่ดีที่สุดสำหรับกลุ่มข้อมูล Clown จากตารางที่ 5.1 แสดงการทดลองค่าน้ำหนักในฟังก์ชันความเหมาะสมที่ 1.3 ***จากการทดลองจะพบจำนวนกลุ่มที่เกิดขึ้นจำนวน 7 กลุ่มข้อมูล*** สามารถบอกกลุ่มบริเวณขอบของกลุ่มย่อยได้ชัดเจนมากยิ่งขึ้น

ตารางที่ 5.1 ผลการจัดกลุ่มโหนดด้วยวิธีเจเนติกอัลกอริธึม

	กลุ่มข้อมูลย่อย 3 กลุ่ม			กลุ่มข้อมูลรูปทรงกลม	กลุ่มข้อมูลรูปทรงวงรี	กลุ่มข้อมูลไม่เป็นรูปทรง	กลุ่มข้อมูลขนาดพื้นที่ใหญ่
	กลุ่มที่ 1	กลุ่มที่ 2	กลุ่มที่ 3				
จำนวนข้อมูลที่จัดกลุ่มได้	110	100	100	507	1000	204	199

ทำการทดลองปรับค่าน้ำหนัก (w) ต่างๆ จากสมการฟังก์ชันความเหมาะสมจากสมการที่ 4.5 ชุดข้อมูล Clown จะได้ผลลัพธ์การจัดกลุ่มดังตารางที่ 5.2

ตารางที่ 5.2 ผลการจัดกลุ่มโหนดโดยให้ค่าน้ำหนักจากสมการความเหมาะสม

ลำดับที่	w = 0. 1		w = 0.5		w = 1.0		w = 1.3		w = 2.0	
	จำนวนกลุ่ม	ผิดพลาด (กลุ่ม)	จำนวนกลุ่ม	ผิดพลาด (กลุ่ม)	จำนวนกลุ่ม	ผิดพลาด (กลุ่ม)	จำนวนกลุ่ม	ผิดพลาด (กลุ่ม)	จำนวนกลุ่ม	ผิดพลาด (กลุ่ม)
1	14	0	12	2	8	4	7	6	4	16
2	14	0	11	3	9	5	7	6	4	16
3	14	0	11	2	8	4	7	5	4	16
4	14	0	12	3	9	5	7	5	4	17
5	14	0	11	2	8	4	7	5	3	16
6	15	0	11	2	9	4	7	5	4	17
7	14	0	11	2	8	5	8	6	4	16
8	16	0	13	3	10	5	7	5	3	16
9	15	0	12	2	8	4	7	5	4	17
10	14	0	11	2	8	4	7	5	4	17

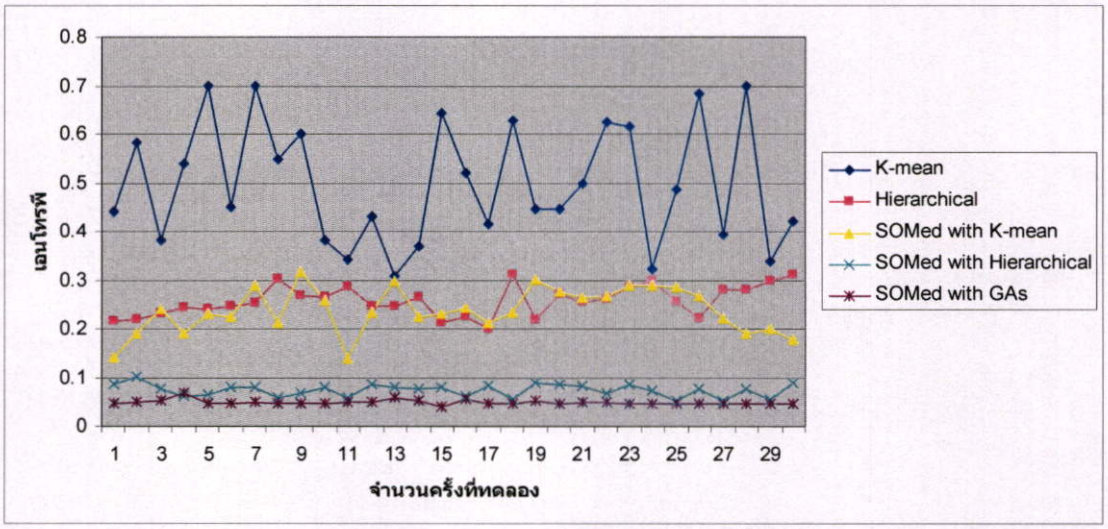
ทำการทดลองเปรียบเทียบระหว่างวิธีการจัดกลุ่มระดับเดียว โดยใช้ K-mean และ Hierarchical และวิธีการจัดกลุ่ม 2 ระดับ โดยใช้ SOM เพื่อสร้างต้นแบบแล้วจัดกลุ่มโหนดจากแผนภาพด้วย K-mean, Hierarchical และเจเนติกอัลกอริธึม แล้ววัดประสิทธิภาพจากค่าเอนโทรพี เวลาที่ใช้ในการคำนวณ และจำนวนที่จัดกลุ่มได้ ดังตารางที่ 5.3

ตารางที่ 5.3 แสดงการเปรียบเทียบวิธีการจัดกลุ่มข้อมูล Clown แบบต่างๆ จากค่าเอนโทรพี เวลาในการคำนวณ และจำนวนที่จัดกลุ่มได้

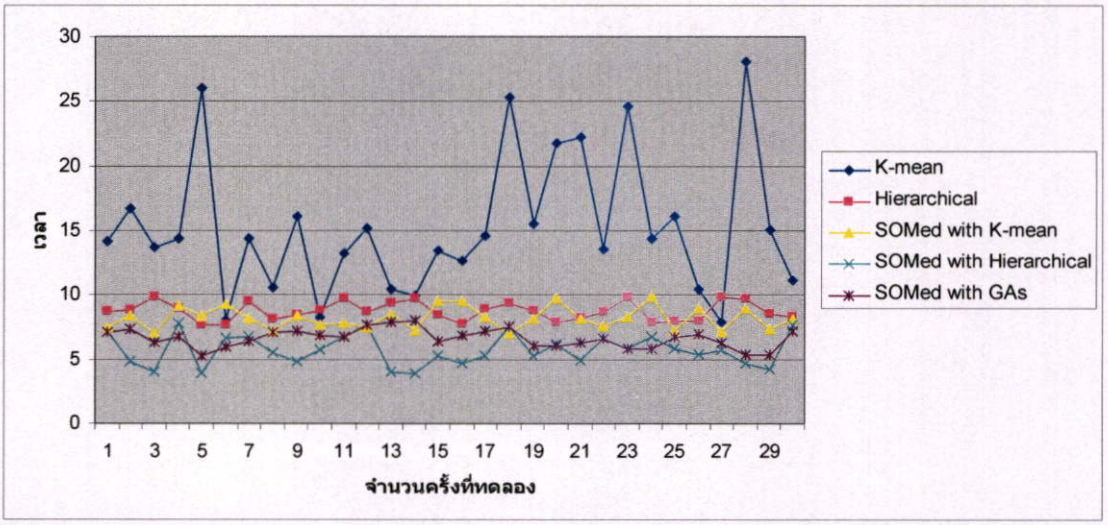
ลำดับที่	จำนวนกลุ่ม	K-Mean		Hierarchical		SOM-based และ K-mean		SOM-based และ Hierarchical		SOM-based และ GAs	
		เอนโทรพี	เวลา (วินาที)	เอนโทรพี	เวลา (วินาที)	เอนโทรพี	เวลา (วินาที)	เอนโทรพี	เวลา (วินาที)	เอนโทรพี	เวลา (วินาที)
1	7	0.44	14.161	0.214	8.77	0.14	7.46	0.086	7.19	0.047	7.21
2	7	0.581	16.684	0.219	8.92	0.192	8.46	0.101	4.79	0.0483	7.33
3	7	0.382	13.7	0.231	9.97	0.24	7	0.078	3.99	0.051	6.4
4	7	0.538	14.41	0.242	9.05	0.19	9.2	0.062	7.88	0.067	6.78
5	7	0.699	26.057	0.239	7.68	0.231	8.48	0.065	3.87	0.0467	5.31
6	7	0.45	8.031	0.246	7.69	0.224	9.29	0.081	6.68	0.0465	5.99
7	7	0.698	14.41	0.251	9.57	0.289	8.14	0.08	6.78	0.0486	6.51
8	7	0.547	10.595	0.301	8.22	0.211	7.3	0.059	5.54	0.0469	7.11
9	7	0.601	16.153	0.267	8.56	0.317	8.4	0.068	4.88	0.047	7.25
10	7	0.383	8.322	0.265	8.91	0.258	7.74	0.081	5.73	0.0467	6.94
11	7	0.342	13.249	0.287	9.86	0.138	7.87	0.058	6.77	0.0502	6.77
12	7	0.43	15.272	0.245	8.8	0.234	7.48	0.086	7.78	0.0477	7.78
13	7	0.307	10.515	0.245	9.44	0.3	8.5	0.081	4.09	0.06	7.93
14	7	0.368	10.004	0.265	9.85	0.225	7.26	0.078	3.97	0.0532	8.02
15	7	0.644	13.449	0.213	8.51	0.232	9.6	0.079	5.36	0.04	6.44
16	7	0.521	12.744	0.226	7.88	0.243	9.63	0.061	4.77	0.0567	6.87
17	7	0.414	14.601	0.199	9.04	0.211	8.26	0.082	5.35	0.0468	7.29
18	7	0.627	25.407	0.31	9.48	0.234	7	0.054	7.58	0.0474	7.58
19	7	0.447	15.528	0.22	8.94	0.302	8.21	0.088	5.29	0.051	6.12
20	7	0.445	21.841	0.271	7.92	0.276	9.79	0.086	6.24	0.0472	6.15
21	7	0.498	22.232	0.254	8.27	0.265	8.22	0.082	4.94	0.0488	6.32
22	7	0.626	13.639	0.261	8.77	0.267	7.56	0.068	6.7	0.0495	6.75
23	7	0.614	24.73	0.287	9.98	0.289	8.33	0.087	5.85	0.0467	5.85
24	7	0.323	14.43	0.298	7.91	0.29	9.95	0.073	6.82	0.0467	5.91
25	7	0.486	16.193	0.255	8.11	0.285	7.31	0.052	5.91	0.0465	6.82
26	7	0.682	10.475	0.223	8.08	0.267	8.96	0.076	5.4	0.0466	7.04
27	7	0.393	7.991	0.281	9.92	0.223	7.2	0.052	5.82	0.0467	6.4
28	7	0.699	28.19	0.279	9.78	0.19	8.95	0.076	4.69	0.046	5.44
29	7	0.338	15.071	0.299	8.6	0.2	7.37	0.054	4.22	0.0451	5.37
30	7	0.44	11.226	0.31	8.44	0.18	8.21	0.089	7.71	0.0467	7.32

จากการทดลองที่ 1 สามารถสรุปการจัดกลุ่มแบบระดับเดียวเปรียบเทียบการจัดกลุ่มแบบ 2 ระดับจากค่าเอนโทรพีและความเร็วที่ใช้ในการจัดกลุ่มข้อมูล จะพบว่าเอนโทรพีของ K-mean จะมีค่ามาก นั้นหมายถึงข้อมูลในกลุ่มๆ หนึ่งมีข้อมูลหลากหลายชนิดมาก สาเหตุมาจากในการคำนวณจะอาศัยศูนย์กลางข้อมูล ในการหาค่าเอนโทรพีจากการจัดกลุ่มข้อมูลด้วย Hierarchical ค่าเอนโทร-พีไม่ค่อยมีการเปลี่ยนแปลงมากนักและมีค่าน้อยกว่าวิธี K-mean

หลังจากนั้นสร้างต้นแบบจาก SOM แล้วทำการจัดกลุ่มข้อมูลอีกครั้ง พบว่า การจัดกลุ่มโหนดด้วย K-mean จะได้ค่าเอนโทรพีลดลงเมื่อเทียบกับการจัดกลุ่มข้อมูลระดับเดียวด้วย K-mean แต่ประสิทธิภาพจะด้อยกว่าการจัดกลุ่มโหนดจากแผนภาพ SOM ด้วยเจเนติกอัลกอริธึม ซึ่งจะพบว่ามีค่าเอนโทรพีจากการจัดกลุ่มจะน้อยกว่าการจัดกลุ่มด้วยวิธีอื่น ดังรูปที่ 5.5 และความเร็วในการคำนวณจะใกล้เคียงกับการจัดกลุ่มโหนดจากแผนภาพ SOM ด้วย Hierarchical ดังรูปที่ 5.6



รูปที่ 5.5 แสดงกราฟเปรียบเทียบค่าเอนโทรพีจากการจัดกลุ่มด้วยวิธีต่างๆ



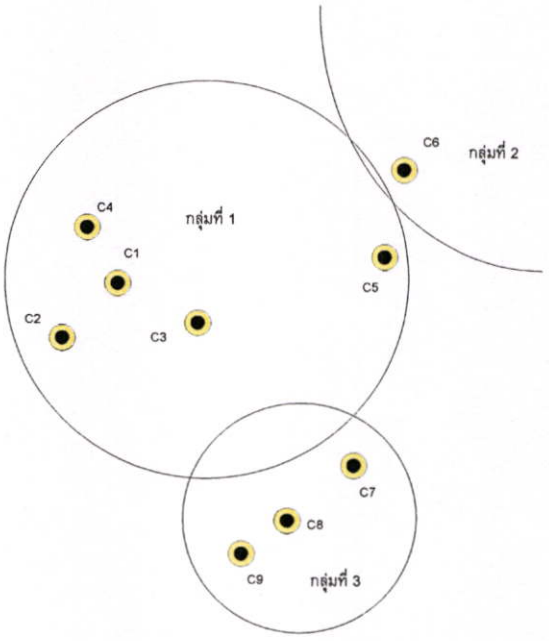
รูปที่ 5.6 แสดงกราฟเปรียบความเร็วที่ใช้ในการจัดกลุ่มด้วยวิธีต่างๆ

5.1.3 เปรียบเทียบการจัดกลุ่มข้อมูลสองระดับ

การเปรียบเทียบวิธีการจัดกลุ่มสองระดับ จุดประสงค์เพื่อเปรียบเทียบประสิทธิภาพในการจัดกลุ่มข้อมูลด้วย

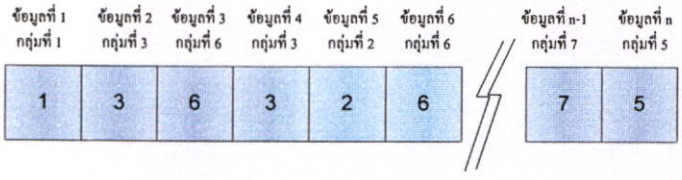
1. วิธี SOM และการจัดกลุ่มโหนดด้วยเจเนติกอัลกอริธึม
2. การจัดกลุ่มข้อมูลหลังจากวิธี K-mean ด้วยเจเนติกอัลกอริธึม

ผลการจัดกลุ่มข้อมูลหลังวิธี K-Mean ซึ่งผลลัพธ์จะมีลักษณะเป็นทรงกลม แต่ผลลัพธ์ดังกล่าวอาจจะไม่ถูกต้อง เพราะข้อมูล Clown เป็นข้อมูลมีลักษณะหลายรูปแบบ เช่น ข้อมูลลักษณะทรงกลม ข้อมูลทรงวงรี ข้อมูลมีลักษณะกระจัดกระจาย เป็นต้น ดังหัวข้อที่ 5.1 ผลลัพธ์การจัดกลุ่มข้อมูลด้วยวิธี K-Mean แสดงได้ดังรูปที่ 5.7



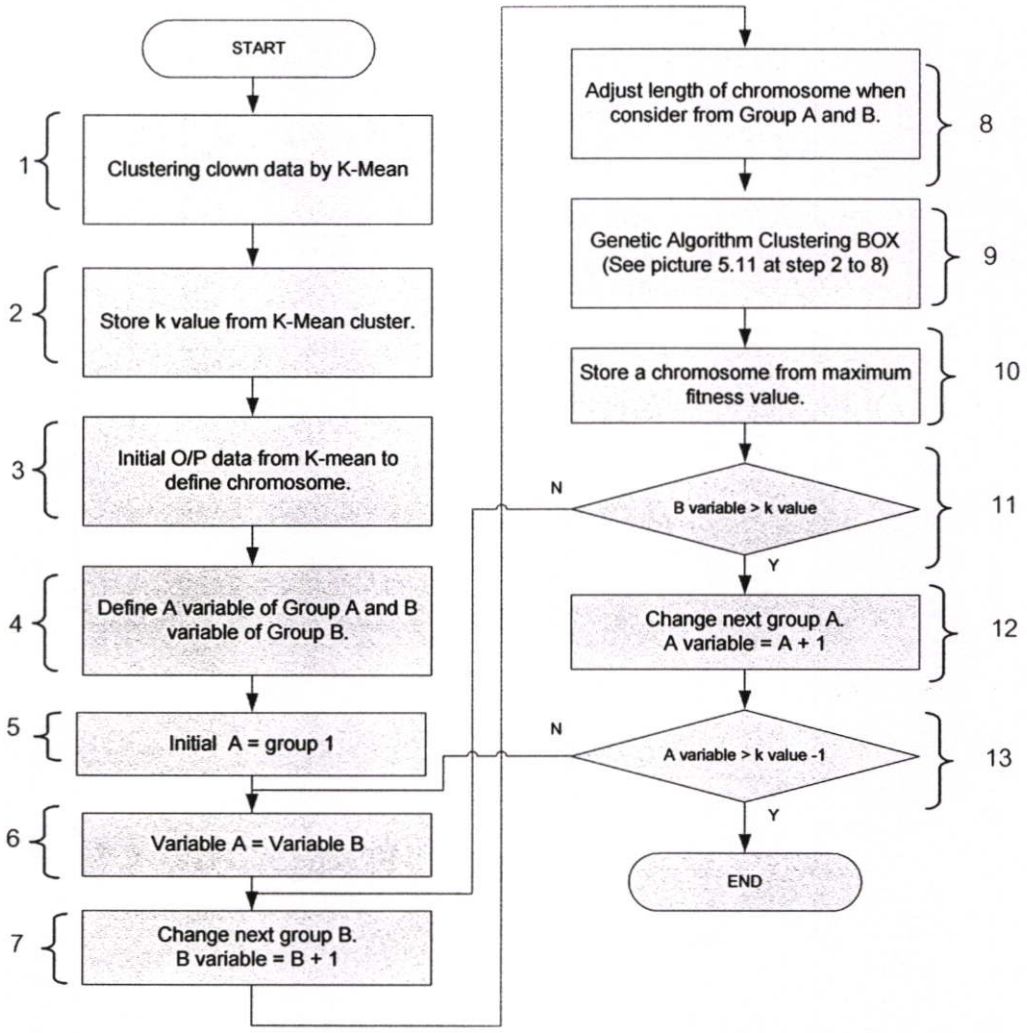
รูปที่ 5.7 ตัวอย่างผลการจัดกลุ่มข้อมูลหลังวิธี K-Mean

การจัดกลุ่มข้อมูลหลังวิธีการจัดกลุ่มด้วยวิธี K-mean และทำการจัดกลุ่มข้อมูลด้วยวิธีเจเนติกอัลกอริธึมอีกครั้ง การกำหนดรูปแบบโครโมโซมในขั้นตอนของเจเนติกอัลกอริธึม โดยให้แต่ละบิตแทนด้วยกลุ่มข้อมูลที่ถูกจัดไว้แล้วด้วยวิธี K-Mean สามารถกำหนดได้ดังรูปที่ 5.8



รูปที่ 5.8 การกำหนดรูปแบบของโครโมโซมหลังการจัดกลุ่มด้วยวิธี K-Mean

จากข้อมูล Clown ซึ่งมีขนาด 2,220 ข้อมูล ดังนั้นความยาวของโครโมโซมจะเท่ากับ 2,220 บิต กำหนดให้เมทริกซ์ $M_{m\times m}$ คือเมทริกซ์แสดงระยะห่างระหว่างข้อมูลเช่นเดียวกับสมการที่ 4.2 และสรุปขั้นตอนการทำงานได้ดังรูปที่ 5.9



รูปที่ 5.9 ขั้นตอนการทำงานของการจัดกลุ่มข้อมูลด้วยวิธี K-Mean และเจเนติกอัลกอริธึม

ขั้นตอนที่ 1 ทำการจัดกลุ่มข้อมูล Clown ด้วยวิธี K-Mean (ดูหัวข้อ 2.3.2)

ขั้นตอนที่ 2 บันทึกค่า k ในวิธีการ K-Mean ซึ่งข้อมูล Clown จะเท่ากับ 7 หรือกลุ่มข้อมูล 7 กลุ่ม

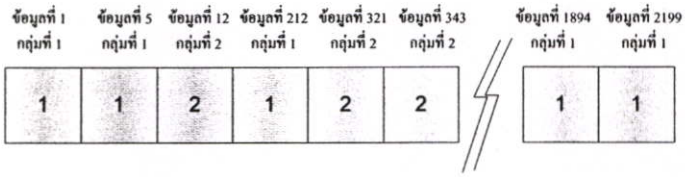
ขั้นตอนที่ 3 นำข้อมูลที่จัดกลุ่มด้วยวิธี K-Mean เพื่อกำหนดรูปแบบของโครโมโซมดังรูปที่ 5.8

ขั้นตอนที่ 4 กำหนดตัวแปรเพื่อหาความสัมพันธ์ระหว่างกลุ่มข้อมูล 2 กลุ่ม โดยตัวแปร A แทนด้วยลำดับกลุ่มข้อมูลที่ 1 และ ตัวแปร B แทนลำดับของกลุ่มข้อมูลที่ 2

ขั้นตอนที่ 5 กำหนดค่าเริ่มต้นให้ตัวแปร A แทนด้วยกลุ่มที่ 1

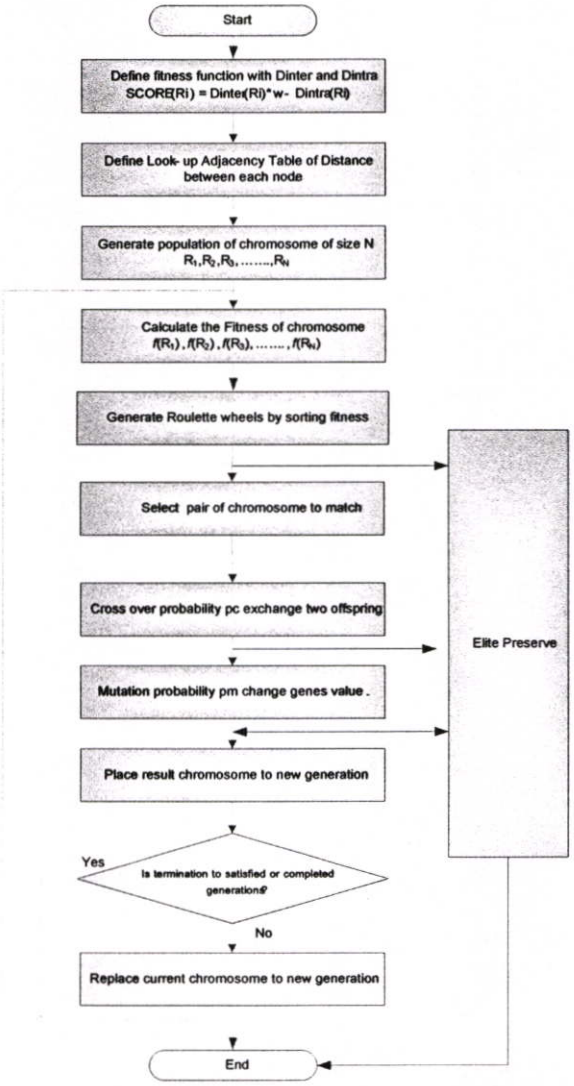
ขั้นตอนที่ 6 และ 7 กำหนดให้ตัวแปร A เท่ากับตัวแปร B จะกล่าวให้ขั้นตอนถัดไป และกำหนดตัวแปร B เป็นข้อมูลของกลุ่มที่ 2

ขั้นตอนที่ 8 พิจารณาข้อมูลภายในกลุ่ม A และกลุ่ม B แล้วปรับความยาวของโรโมโซมใหม่ เช่น การหาความสัมพันธ์ของข้อมูลในกลุ่มที่ 1 และ กลุ่มที่ 2 ดังนั้นความยาวของโครโมโซมจะขึ้นอยู่กับข้อมูลของกลุ่มที่ 1 และข้อมูลกลุ่มที่ 2 แสดงดังรูป 5.10



รูปที่ 5.10 แสดงโครโมโซมเฉพาะข้อมูลในกลุ่มที่ 1 และข้อมูลในกลุ่มที่ 2

ขั้นตอนที่ 9 นำรูปแบบโครโมโซมจากขั้นตอนที่ 8 ทำการจัดกลุ่มข้อมูลด้วยวิธีเจเนติกอัลกอริธึมดังรูปที่ 5.11 ระหว่างขั้นตอนที่ 2 ถึงขั้นตอนที่ 8 โดยให้สุ่มโครโมโซมเริ่มต้นเฉพาะข้อมูลในกลุ่มที่ 1 และกลุ่มที่ 2 เท่านั้น



รูปที่ 5.11 แสดงการจัดกลุ่มข้อมูลด้วยวิธีเจเนติกอัลกอริธึม

ขั้นตอนที่ 10 บันทึกโครโมโซมที่ความเหมาะสมสูงสุดเพื่อเป็นคำตอบของข้อมูลในกลุ่ม A และข้อมูลในกลุ่ม B

ขั้นตอนที่ 11 และ 12 ตรวจสอบกลุ่ม B ครบจำนวน k กลุ่มหรือไม่ (ในข้อมูล Clown มีค่า k = 7) ถ้ายังไม่ครบให้กลับไปขั้นตอนที่ 7 ใหม่ จนกว่าจะครบจำนวน k

ขั้นตอนที่ 13 หาความสัมพันธ์ของกลุ่มถัดไป เช่น เมื่อทำการหาความสัมพันธ์ของกลุ่มที่ 1 กับกลุ่มที่ 2, 3, 4, 5, 6 และ7 แล้วให้หาความสัมพันธ์ของกลุ่มที่ 2 กับกลุ่มที่ 3, 4, 5, 6 และ 7 หาความสัมพันธ์ของกลุ่มที่ 3 กับกลุ่มที่ 4, 5, 6 และ 7 หาความสัมพันธ์ของกลุ่มที่ 4 กับ 5, 6 และ 7 หาความสัมพันธ์ของกลุ่มที่ 5 กับ 6 และ 7 หาความสัมพันธ์ของกลุ่มที่ 6 กับ 7

เมื่อครบกระบวนการจัดกลุ่มข้อมูลแล้ว ทำการหาคำตอบโดยพิจารณาจากโครโมโซมที่มีค่าความเหมาะสูงสุด ของข้อมูลในกลุ่ม A และกลุ่ม B จากการทดลองจะพบข้อมูลที่ถูกจัดกลุ่มได้ทั้งหมด 7 กลุ่มข้อมูล โดยมีข้อมูลจัดเป็นกลุ่มต่างๆ ได้ดังตารางที่ 5.4

ตารางที่ 5.4 ผลการจัดกลุ่มข้อมูล 2 ระดับด้วยวิธี K-Mean และเจเนติกอัลกอริธึม

	กลุ่มข้อมูลย่อย 3 กลุ่ม			กลุ่มข้อมูลรูปทรงกลม	กลุ่มข้อมูลรูปทรงวงรี	กลุ่มข้อมูลไม่เป็นรูปทรง	กลุ่มข้อมูลขนาดพื้นที่ใหญ่
	กลุ่มที่ 1	กลุ่มที่ 2	กลุ่มที่ 3				
จำนวนข้อมูล Clown ที่มีการจัดกลุ่มไว้แล้ว	100	100	100	500	1000	210	210
จำนวนข้อมูลที่จัดกลุ่มได้	100	100	100	500	946	294	180

การวัดประสิทธิภาพจากค่าเอนโทรพีและเวลาในการจัดกลุ่มข้อมูล เพื่อเปรียบเทียบระหว่างการจัดกลุ่มโหนดใน SOM ด้วยเจเนติกอัลกอริธึม และวิธีการจัดกลุ่มสองระดับด้วย K-Mean และเจเนติกอัลกอริธึม ในการทดลองจะทำการทดลอง 30 ครั้ง แล้วบันทึกค่าเอนโทรพีและเวลาของทั้งสองวิธี ในแต่ละครั้ง ซึ่งจะพบว่าสามารถจัดกลุ่มข้อมูลได้ทั้งหมด 7 กลุ่มเท่ากัน แต่ผลของค่าเอนโทรพีในวิธีการจัดกลุ่มโหนดใน SOM ด้วยเจเนติกอัลกอริธึม จะมีประสิทธิภาพดีกว่ากับวิธีการจัดกลุ่มสองระดับด้วย K-Mean และเจเนติกอัลกอริธึม ดังตารางที่ 5.5 และรูปที่ 5.12 เนื่องจากในบางครั้งของผลการจัดกลุ่มข้อมูลด้วยวิธี K-Mean จะจัดกลุ่มข้อมูลย่อย 2 กลุ่มถูกรวมเป็นกลุ่มเดียวกัน แต่พบว่าข้อมูลย่อยอีก 1 กลุ่มไปรวมกับข้อมูลที่มีรูปทรงแบบวงรีดังนั้นทำให้ผลของค่าเอนโทรพีในผลการทดลองบางครั้งจะมีค่าสูงมาก

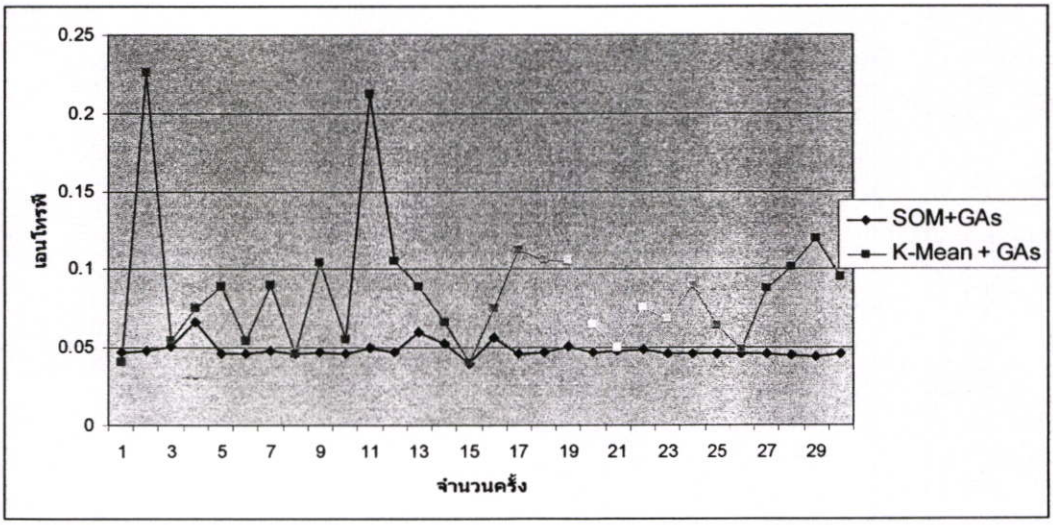
ผลของเวลาที่ใช้ในการจัดกลุ่มข้อมูลจะพบว่าวิธีการจัดกลุ่มโหนดใน SOM ด้วยเจเนติกอัลกอริธึมจะมีใช้เวลาการจัดกลุ่มข้อมูลน้อยกว่าวิธีการจัดกลุ่มสองระดับด้วย K-Mean และเจเนติกอัลกอริธึมอย่างเห็นได้ชัด เนื่องจากวิธีการจัดกลุ่มสองระดับนั้นจะมีรูปแบบของโครโมโซมที่แตกต่างกว่าวิธีการจัดกลุ่มโหนดใน SOM ด้วยเจเนติกอัลกอริธึม ซึ่งจำเป็นต้องคงไว้สำหรับกลุ่ม

ข้อมูลที่เกิดจากผลของการจัดกลุ่มด้วยวิธี K-Mean ซึ่งจะเกิดลูปการทำงานมากกว่าวิธีแรก และสามารถสรุปผลการวัดประสิทธิภาพได้ดังตารางที่ 5.5 และรูปที่ 5.13

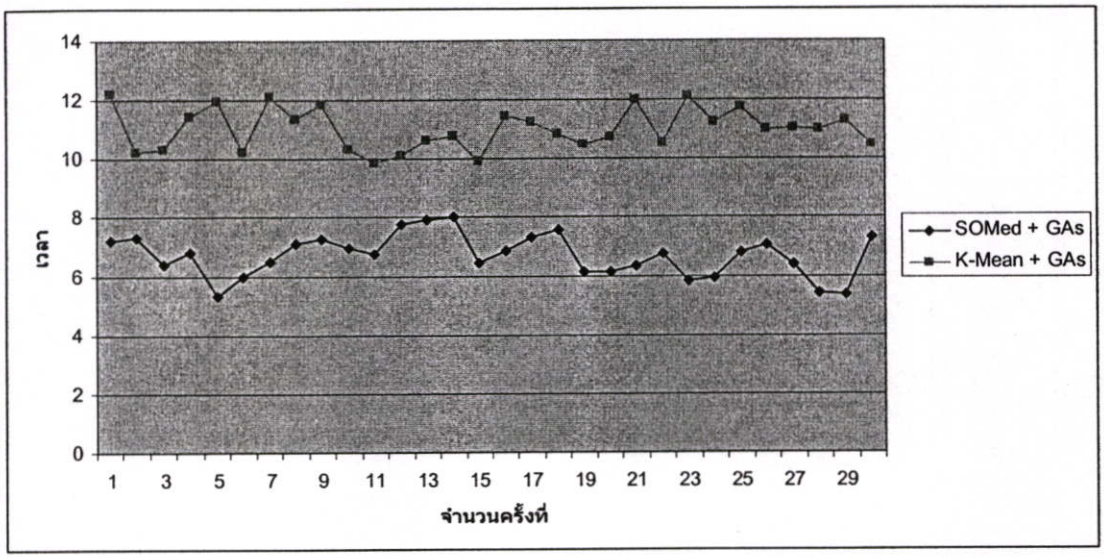
ตารางที่ 5.5 แสดงการเปรียบเทียบวิธีการจัดกลุ่มข้อมูล Clown ด้วยการจัดกลุ่มโหนดใน SOM ด้วยเจเนติกอัลกอริธึม และวิธีการจัดกลุ่มสองระดับด้วย K-Mean และเจเนติกอัลกอริธึม

ลำดับที่	จำนวนกลุ่ม	SOM-based และ GAs		K-Mean และ GAs	
		เอนโทรพี	เวลา (วินาที)	เอนโทรพี	เวลา (วินาที)
1	7	0.047	7.21	0.0415	12.21
2	7	0.0483	7.33	0.2265	10.23
3	7	0.051	6.4	0.0545	10.34
4	7	0.067	6.78	0.0755	11.45
5	7	0.0467	5.31	0.0895	11.94
6	7	0.0465	5.99	0.0545	10.24
7	7	0.0486	6.51	0.0905	12.1
8	7	0.0469	7.11	0.0465	11.32
9	7	0.047	7.25	0.1045	11.87
10	7	0.0467	6.94	0.0555	10.34
11	7	0.0502	6.77	0.2125	9.86
12	7	0.0477	7.78	0.1055	10.14
13	7	0.06	7.93	0.0895	10.65
14	7	0.0532	8.02	0.0665	10.78
15	7	0.04	6.44	0.0415	9.91

ลำดับที่	จำนวนกลุ่ม	SOM-based และ GAs		K-Mean และ GAs	
		เอนโทรพี	เวลา (วินาที)	เอนโทรพี	เวลา (วินาที)
16	7	0.0567	6.87	0.0755	11.43
17	7	0.0468	7.29	0.1135	11.22
18	7	0.0474	7.58	0.1065	10.83
19	7	0.051	6.12	0.1055	10.45
20	7	0.0472	6.15	0.0645	10.75
21	7	0.0488	6.32	0.0505	12.02
22	7	0.0495	6.75	0.0755	10.54
23	7	0.0467	5.85	0.0685	12.11
24	7	0.0467	5.91	0.0905	11.23
25	7	0.0465	6.82	0.0645	11.73
26	7	0.0466	7.04	0.0495	10.99
27	7	0.0467	6.4	0.0885	11.06
28	7	0.046	5.44	0.1025	11
29	7	0.0451	5.37	0.1205	11.3
30	7	0.0467	7.32	0.0955	10.45



รูปที่ 5.12 แสดงกราฟเปรียบเทียบค่าเอนโทรพีจากการจัดกลุ่มด้วยวิธีการจัดกลุ่มโหนดใน SOM ด้วยเจเนติกอัลกอริธึม และวิธีการจัดกลุ่มสองระดับด้วย K-Mean และเจเนติกอัลกอริธึม



รูปที่ 5.13 แสดงกราฟเปรียบความเร็วที่ใช้ในการจัดกลุ่มด้วยวิธีวิธีการจัดกลุ่มโหนดใน SOM ด้วยเจเนติกอัลกอริธึม และวิธีการจัดกลุ่มสองระดับด้วย K-Mean และเจเนติกอัลกอริธึม

5.2 การทดลองชุดข้อมูลที่ 2 จัดกลุ่มข้อมูลหลายมิติ

ข้อมูลที่ใช้ทำการทดลองนำมาจากชุดข้อมูล KDD cup 1999 ซึ่งเป็นรูปแบบของการบุกรุกเครือข่าย [5] ได้มีการจัดกลุ่มไว้ก่อนแล้ว 11 กลุ่ม ได้แก่ back, ipsweep, land, neptune nmap, normal, pod, portsweep, satan, smurf และ teardrop มีจำนวนข้อมูลทั้งหมด 42,574 ชุดข้อมูล แต่ละข้อมูลมีคุณสมบัติหรือมิติ 41 มิติ ได้แก่

1. *Basic feature* เป็นคุณลักษณะพื้นฐานที่ได้จากแพคเกจข้อมูลที่สื่อสารในเครือข่าย มีจำนวน 9 features

2. *Content feature* เป็นคุณลักษณะที่เก็บรวบรวมข้อมูลที่แสดงให้เห็นถึงพฤติกรรมน่าสงสัย เช่น ความผิดพลาดในการล็อกอิน จำนวน 13 features

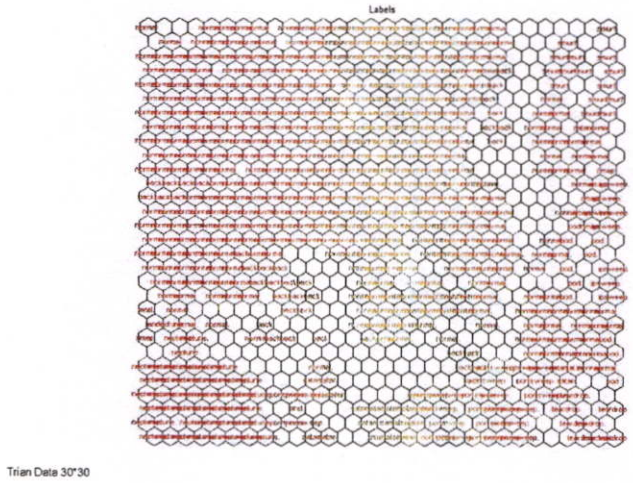
3. *Traffic feature* เป็นคุณลักษณะที่เก็บรวบรวมข้อมูลที่ลักษณะของการสื่อสารจำนวน 9 features

4. *Host based feature* เป็นคุณลักษณะที่เก็บรวบรวมข้อมูลที่แสดงลักษณะของการสื่อสารไปยังเครื่องปลายทางเครื่องเดิมตลอดเวลา จำนวน 10 features

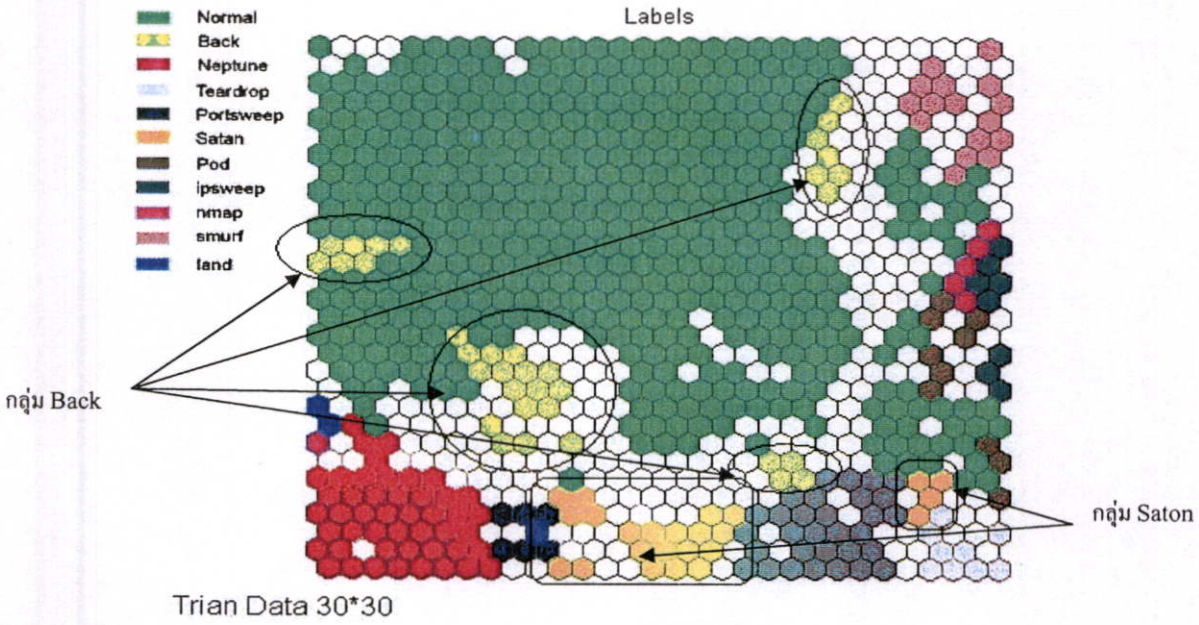
ในการทดลองจะนำ Label ของแต่ละชุดข้อมูลออกไปเพื่อให้สามารถจัดกลุ่มโดยไม่อาศัยผู้สอน ข้อมูลที่นำมาทดสอบจะถูกสร้างแผนภาพ SOM สองมิติ

5.2.1 การจัดกลุ่มข้อมูลด้วย SOM

โดยกำหนดให้แผนภาพขนาด 30×30 อัตราการเรียนรู้มีค่าเท่ากับ 0.2 น้ำหนักประจำแต่ละโหนดจะเป็นตัวแทนของกลุ่มอินพุทข้อมูลที่ตกในโหนดนั้นเพื่อสร้างต้นแบบสำหรับการจัดกลุ่มข้อมูล 2 ระดับ จากรูปที่ 5.14 (ก) แสดงแผนภาพ SOM ที่เรียนรู้แล้วโดยแทนข้อมูลตกบนโหนดด้วยชื่อกลุ่ม และเพื่อให้ชัดเจนยิ่งขึ้นจะแทนข้อมูลที่ตกบนโหนดด้วยสีดังรูป 5.14 (ข) ซึ่งจะพบว่าข้อมูลการบุกรุกเครือข่ายในรูปแบบ Back จะกระจายอยู่ในแผนภาพที่ไม่ได้อยู่กลุ่มเดียวกัน เช่นเดียวกับข้อมูลรูปแบบ Satan ซึ่งจะทำให้การสำรวจในแผนภาพนั้นกระโดดไปมาขาดประสิทธิภาพและไม่สามารถมองเห็นภาพรวมของข้อมูลได้



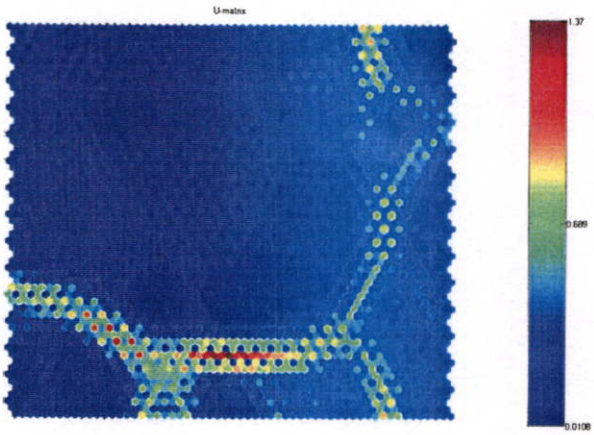
(ก)



(ข)

รูปที่ 5.14 (ก) แสดงการแผนภาพ SOM หลังจากผ่านการเรียนรู้ จากข้อมูลรูปแบบการบุกรุกเครือข่าย (ข) แสดงผลลัพธ์อย่างชัดเจนโดยแยกกลุ่มต่างๆ ด้วยสี

รูปที่ 5.15 แสดงแผนภาพ U-matrix ของแผนภาพ SOM ที่ได้โดยแผนภาพ U-matrix คือ แผนภาพแสดงระยะห่างระหว่างโหนดใกล้เคียง โหนดที่สว่างจะเป็นโหนดไม่ใช่โหนดชนะ และโหนดที่มีสีเข้มแสดงความหนาแน่นของข้อมูลมาก โดยอาศัยแผนภาพ U-matrix ที่ได้จะเห็นได้ว่าไม่สามารถกำหนดขอบเขตได้ชัดเจน และไม่สามารถจัดกลุ่มที่กระจายในแผนภาพ SOM ได้



รูปที่ 5.15 U-Matrix แสดงการกระจายของข้อมูลหลังการเรียนรู้ด้วย SOM จากข้อมูลการบุกรุกเครือข่าย

5.2.2 การจัดกลุ่มโหนดโดยใช้เจเนติกอัลกอริธึม

แผนภาพโคโฮเนนหลังการเรียนรู้แล้วจะกำหนดให้โหนด 900 โหนด (30×30) สร้างเป็นโครโมโซมจำนวน 1 สตริง จำนวน 50 โครโมโซม ในการกำหนดจำนวนประชากรของโครโมโซมจะพิจารณาจากจำนวนกลุ่มที่เกิดขึ้น 11 กลุ่ม โดยเผื่อการเกิดโครโมโซมที่ซ้ำกันและทำการทดลองซ้ำเพื่อใช้พิจารณากำหนดจำนวนประชากรที่เหมาะสม กำหนดให้เกิดโครโมโซมรุ่นใหม่ 50 รุ่น มีการเก็บค่าที่ดีที่สุดเอาไว้เพื่อใช้เปรียบเทียบกับค่าที่ดีที่สุดของโครโมโซมคำตอบชุดใหม่ กำหนดค่า Pc=0.8 และ Pm=0.01

การวัดประสิทธิภาพในการจัดกลุ่มเราจะใช้ค่าเอนโทรพี(Entropy)[6][7] จากการทดลองการจัดกลุ่มโหนดโดยใช้เจเนติกอัลกอริธึมจะสามารถจัดกลุ่มแบบอัตโนมัติได้กลุ่มออกมาทั้งหมด 13 กลุ่ม ซึ่งค่าเอนโทรพีรวม ดังตารางที่ 5.6 เมื่อเทียบกับการจัดกลุ่มโดยใช้วิธี K-mean และกำหนดค่า K ให้เท่ากับ 11 และ 15 ซึ่งหมายถึงจำนวนกลุ่มที่ถูกจัดกลุ่มจะมี 11 และ 15 กลุ่มข้อมูล จะเห็นว่าการจัดกลุ่มโดยใช้เจเนติกอัลกอริธึมจะให้ค่าเอนโทรพีที่น้อยกว่าซึ่งหมายถึงประสิทธิภาพการจัดกลุ่มที่ดีกว่าเมื่อเทียบจำนวนกลุ่มที่เท่ากัน เมื่อทดลองเพิ่มกลุ่ม K ให้กับวิธี K-mean จะเห็นว่าค่าเอนโทรพีที่ได้จะค่อย ๆ มีค่าลดลง แต่จะเกิดจำนวนโหนดในเลเยอร์ของเจเนติกมากขึ้นซึ่งแสดงว่าข้อมูลจะถูกจัดให้เป็นกลุ่มย่อยมากขึ้นทำให้เอนโทรพีรวมลดลงด้วย

ตารางที่ 5.6 แสดงค่า Entropy เปรียบเทียบวิธี GA วิธี K-mean และวิธี Hierarchical

SOM Clustered with	ค่า Entropy
GA (จัดกลุ่มได้ 13 กลุ่ม)	0.2250
K-Mean (k=11)	0.356
K-Mean (k=15)	0.317
K-Mean (k=20)	0.2725
K-Mean (k=25)	0.2190
Hierarchical	0.325

แต่การกำหนดกลุ่ม K มากขึ้นทำให้กลุ่มของข้อมูลถูกแบ่งเป็นกลุ่มย่อย ๆ มากขึ้นตาม ทำให้เราไม่สามารถระบุกลุ่มที่แท้จริงได้อย่างชัดเจน ปัญหาอีกประการในการใช้ K-mean คือเราไม่สามารถรู้ค่า K ที่เหมาะสมที่สุดสำหรับกลุ่มข้อมูลนั้นต้องทดลองหาค่า K เองซึ่งจะทำให้เสียเวลามาก และที่สำคัญ K-mean ไม่สามารถที่จะจัดกลุ่มของข้อมูลที่กระจายตามแผนภาพ SOM ได้อย่างถูกต้อง

ทำการทดลองปรับค่าน้ำหนักต่างๆ จากสมการฟังก์ชันความเหมาะสมกับชุดข้อมูลการบุกรุกเครือข่ายจะได้ผลลัพธ์การจัดกลุ่มดังตารางที่ 5.7

ตารางที่ 5.7 ผลการจัดกลุ่มโหนดโดยให้ค่าน้ำหนักจากสมการความเหมาะสม

Experiment	w=1		w=1.2		w=1.5		w=2.5		w=3	
	#cluster GA	#miss class	**#cluster GA**	#miss Class	**#cluster GA**	#miss Class	**#cluster GA**	#miss Class	**#cluster GA**	#miss Class
1	**15**	0	**14**	3	**13**	9	**8**	24	**6**	30
2	**16**	0	**14**	3	**13**	9	**8**	22	**6**	32
3	**16**	0	**15**	0	**13**	8	**8**	21	**7**	25
4	**16**	0	**15**	0	**14**	8	**9**	20	**7**	24

ค่าน้ำหนักจากสมการความเหมาะสมในสมการ 4.5 นั้น พบว่าค่าน้ำหนักที่ดีที่สุดสำหรับปัญหาของชุดข้อมูลการบุกรุกเครือข่ายเท่ากับ 1.5 ซึ่งจะสามารถจัดกลุ่มได้ใกล้เคียงกับข้อมูลการบุกรุกเครือข่ายที่เคยแบ่งกลุ่มข้อมูลไว้ก่อนหน้านั้นแล้ว (จาก Data set การบุกรุกเครือข่ายได้กำหนดชื่อกลุ่มสำหรับข้อมูลไว้แล้ว) และได้จำนวนข้อมูลที่จัดกลุ่มผิดน้อย แต่เมื่อเพิ่มค่าน้ำหนัก

ขึ้นเรื่อยๆ จะพบว่าจำนวนกลุ่มข้อมูลน้อยลง แต่ข้อมูลจะถูกจัดผิดกลุ่มมีมากขึ้น และสามารถยอมรับได้ที่ 1.5 เมื่อพิจารณาจากจำนวนข้อมูลที่ถูกจัดผิดกลุ่ม ซึ่งสรุปได้ว่าการจัดกลุ่มที่เกิดขึ้นกับเลเยอร์เจเนติก (Genetic Layer) ได้ 13 กลุ่มดังตารางที่ 5.8

จากตารางที่ 5.8 พบว่าข้อมูลการบุกรุกเครือข่ายแบบ Back และ Satan บางตัวมีลักษณะคล้ายกับการบุกรุกแบบ Normal เป็นสาเหตุที่ทำให้ข้อมูลทั้งสองรูปแบบกระจายใกล้บริเวณข้อมูลกลุ่ม Normal มาก ในการวิเคราะห์จากแผนภาพนั้นแสดงว่า การบุกรุกเครือข่ายแบบ Satan และ Back จะพยายามเลียนแบบการทำงานให้คล้ายคลึงกับการรับส่งข้อมูลปกติ แต่การวิเคราะห์และสำรวจกลุ่มข้อมูลทำได้ยาก เมื่อเปรียบเทียบการจัดกลุ่มข้อมูลด้วย SOM เพียงอย่างเดียวกับการจัดกลุ่มโหนดจากแผนภาพ SOM ด้วยเจเนติกอีกครั้ง ทำให้สามารถสรุปกลุ่มข้อมูลได้ง่ายขึ้น

ตารางที่ 5.8 แสดงผลการจัดกลุ่มด้วยวิธีเจเนติกอัลกอริธึมหลังการเรียนรู้ SOM จากชุดข้อมูลที่ 2

Genetic Layer	จำนวนข้อมูลที่เกิดขึ้น										
	Normal	Back	Neptune	Land	Saton	Smurf	Ipsweep	Portsweep	Teardrop	Pod	nmap
1	470					1					
2	11	31									
3			47								
4				2							
5	8				20						
6						15					
7	3				3		31				
8				1				8			
9	3								9		
10	2									8	
11											6
12	8									4	
13	11	3									

บทที่ 6
สรุปผลการวิจัย และข้อเสนอแนะ

6.1 สรุปผลการวิจัย

จุดประสงค์ของการจัดกลุ่มข้อมูลคือ พยายามจัดข้อมูลที่มีความคล้ายกันมาอยู่กลุ่มเดียวกันให้มากที่สุด ซึ่งการจัดกลุ่มข้อมูลจากผลลัพธ์ของแผนภาพ SOM แต่จะพบว่าโหนดบนแผนภาพมีการกระจายของข้อมูลยังโหนดที่ห่างออกไปทั้งที่ควรจะถูกจัดอยู่ในโหนดหรือกลุ่มบริเวณเดียวกัน โดยทั่วไปปัญหานี้มักเกิดขึ้นกับข้อมูลที่มีมิติมากๆ และซับซ้อน ซึ่งเป็นผลให้การสำรวจและเลือกดูข้อมูลบนแผนภาพ SOM ทำได้ยาก ดังนั้นวิธีการจัดกลุ่มจากแผนภาพจะต้องไม่อาศัยจุดศูนย์กลางข้อมูล (Centriod) ซึ่งต่างจากวิธีของ K-mean ที่ต้องใช้ค่าเฉลี่ยหรือจุดศูนย์กลางข้อมูลและจำเป็นต้องรู้กลุ่มที่แน่นอนเสียก่อน พิจารณาจากค่าความเหนี่ยวแน่นภายในกลุ่มหรือเอนโทรพี วิธี K-mean จะมีค่ามากกว่าวิธีเจเนติกอัลกอริธึม ซึ่งแสดงว่าวิธี K-mean จะมีโอกาสที่ข้อมูลชนิดเดียวกันมีความหนาแน่นน้อยกว่า วิธีเจเนติกอัลกอริธึมเมื่อเทียบกับจำนวนกลุ่มที่เท่ากัน เนื่องจากวิธี K-mean จะสุ่มจุดเริ่มต้นมาก่อนและเวลาที่ใช้ในการคำนวณจะเพิ่มขึ้นเป็นแฟคเทอเรียลตามจำนวนข้อมูล

การจัดกลุ่มด้วยวิธี Hierarchical นั้นจะเป็นการจัดกลุ่มโดยจับข้อมูลสองตัวที่ใกล้เคียงกันที่สุดและรวมเป็นกลุ่มเดียวกัน ทำเช่นนี้ไปจนจบกระบวนการทำงาน ดังนั้นเวลาที่ใช้จะลดลงเมื่อเทียบกับการจัดกลุ่มด้วย K-Mean แต่โอกาสในการจัดกลุ่มผิดจะเกิดขึ้นเมื่อมีการรวมกลุ่มใดกลุ่มหนึ่งไปแล้ว การแก้ปัญหาดังกล่าวด้วยวิธีเจเนติกอัลกอริธึมเพื่อเพิ่มโอกาสการจัดกลุ่มข้อมูลให้ถูกต้องมากขึ้น

จากการทดลองที่ 1 ซึ่งเป็นการทดลองที่มีขนาดมิติข้อมูลเพียง 2 มิติที่ไม่มีความสลับซับซ้อน การจัดกลุ่มจึงไม่พบความชัดเจนที่เกิดปัญหาการกระจายข้อมูลเท่าใดนัก จากการทดลองจะพบว่ามีเพียงโหนดเดียวที่ถูกแยกออกมา ดังรูป 5.2(ข) (บริเวณมุมซ้ายล่างของแผนภาพหลังกระบวนการเรียนรู้ด้วย SOM) นอกจากนี้ข้อมูลที่ตกบริเวณโหนดขอบข้อมูลจะไม่สามารถแยกได้ว่าเป็นกลุ่มใด จากวิธีการของเจเนติกอัลกอริธึมที่ใช้จะช่วยเพิ่มกระบวนการหาคำตอบที่ดีที่สุดซึ่งแตกต่างจากวิธีเจเนติกอัลกอริธึมทั่วๆ ไป ทำให้ได้คำตอบที่ถูกต้องมากขึ้น

การทดลองที่ 2 จะพบปัญหาการกระจายข้อมูลอย่างชัดเจนมากขึ้น ดังรูป 5.4 (ข) โดยวิธีการของ SOM ในระหว่างการเรียนรู้แผนภาพ SOM จะสมบูรณ์ก็ต่อเมื่อมีการเรียนรู้จนแผนภาพมีเสถียรภาพหรือมีการเปลี่ยนแปลงน้อยหรือไม่มีการเปลี่ยนแปลงใดๆ เกิดขึ้นอีก แต่สำหรับข้อมูลที่มีความสลับซับซ้อนหลายๆ มิติเช่นเดียวกับการทดลองที่ 2 จะไม่สามารถคาดเดาความสมบูรณ์ของแผนภาพว่าจะเกิดขึ้นเมื่อใด อาจจะใช้เวลาเป็นอนันต์ก็ได้ เพราะสาเหตุมาจากการแยกตัวข้อมูล

ชนิดเดียวออกจากกลุ่มหลัก จากโจทย์ปัญหาดังกล่าวจึงเป็นผลให้ต้องมีการจัดกลุ่มอีกครั้งหนึ่ง จากการทดลองจัดกลุ่มด้วยวิธีเจเนติกอัลกอริธึมซึ่งจะเห็นได้จากการวัดค่าประสิทธิภาพด้วยค่าเอนโทรพีที่ได้ค่าถูกต้องมากกว่าวิธีอื่นๆ และทำการเปรียบเทียบกับข้อมูลที่เคยระบุกลุ่มไว้ก่อนหน้านั้นแล้ว ซึ่งจะได้จำนวนกลุ่มที่เท่ากันคือ 7 กลุ่มและ 11 กลุ่ม ในชุดข้อมูลการทดลองที่ 1 และ 2 ตามลำดับ

6.2 ข้อเสนอแนะ

การจัดกลุ่มโหนดจากปัญหาของแผนภาพ SOM นั้นเป็นเหตุเป็นผลมาจากการเรียนรู้ที่ไม่สมบูรณ์ของ SOM ผู้วิจัยได้พยายามปรับปรุงการจัดกลุ่มจากปัญหาดังกล่าวหลังกระบวนการเรียนจึงทำการจัดกลุ่มจากแผนภาพ แต่การแก้ปัญหานั้นสามารถที่จะปรับปรุงกระบวนการจัดกลุ่มด้วยวิธีอื่น เช่นวิธีเจเนติกอัลกอริธึมเข้าร่วมอยู่ในกระบวนการเรียนรู้ด้วย SOM เพื่อลดเวลาในการคำนวณ เนื่องจากกระบวนการทำสองขั้นตอนจะเสียเวลาในการส่งถ่ายข้อมูลระหว่างขั้นตอนการเรียนรู้และจัดกลุ่มโหนดหลังการเรียนรู้ด้วยเจเนติกอัลกอริธึม โดยยึดหลักการสร้างต้นแบบจาก Juha Vesanto [3] และปรับปรุงรูปแบบของโครโมโซมในเจเนติกอัลกอริธึมโดยไม่จำเป็นต้องกำหนดให้เป็น 0 หรือ 1 แต่อาจจะใช้เป็นตัวเลขจำนวนจริงแทน เพื่อลดการเปรียบเทียบ ซึ่งทำให้สามารถตัดขั้นตอนการสร้าง Adjacency Matrix ออกไปได้ เช่นวิธีการจัดกลุ่มด้วยวิธี Particle Swarm Optimization :PSO [13, 14]

โดยปกติวิธีการของ PSO จะคล้ายนำวิธีของ K-Mean กล่าวคือเป็นการหาจุดศูนย์กลางของข้อมูลแล้วใช้วิธีการหาคำตอบที่ดีที่สุดร่วมกัน ดังนั้นความเหมาะสมกับการแก้ปัญหาจากโจทย์ที่ไม่สมบูรณ์ของแผนภาพจะต้องปรับปรุงวิธีของ PSO ให้สามารถจัดกลุ่มข้อมูลที่ไม่เป็นรูปแบบเสียก่อน

เอกสารอ้างอิง

[1] T. Kohonen. "The Self-organizing map." Proceeding of the IEEE, Volume 78, NO.9, Sep 1990.

[2] A.K. Jain, M.N. Murty and P.J. Flynn, "Data Clustering: A Review." ACM Computing Surveys, Vol. 31, NO. 3, Sept. 1999.

[3] J. Vesanto and E. Alhoniemi, "Clustering of the Self-organizing Map." IEEE, NO. 3, May. 2000.

[4] L. Y. Tseng and S. B. Yang. "A genetic clustering algorithm for data with Non-spherical-shape clusters." Pattern Recognition Society, Elsevier Science. Dec. 1999.

[5] C. Bezdek and Nikhil R.Pal, "Some New Indexes of Cluster Validity", IEEE Transactions on System, Man and Cybernetics, Vol. 28, NO. 3, June. 1998.

[6] J. Vesanto, "SOM-based data visualization methods," Intelligent Data Analysis., Vol. 3, NO. 2, pp. 111–126, 1999.

[7] L.Boudjeloud and F.Poulet, "Attribute Selection for High Dimensional Data Clustering.", IEEE, Vol. 26, pp.387-395, 1998.

[8] S.Stolfo, The Third International Knowledge Discovery and Data Mining Tool Competition. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[9] M.Koskela, J. Laaksonen and E.Oja, "Entropy-based measures for clustering and SOM topology preservation applied to content-based image indexing and retrieval", Proceeding of the 17th ICPR'04 IEEE, June 2001

[10] พรเทพ โรจนวสุ, "การจัดกลุ่มเอกสารโดยใช้ Self-Organizing Map แบบความเร็วสูง", ภาควิชาวิศวกรรมคอมพิวเตอร์, คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, พ.ศ.2547.

[11] อภิญญา สุวรรณละมัย วรพจน์ กรีสุระเดช, "การจัดกลุ่มเอกสารโดยใช้โคโฮเนนนิวรอลเน็ตเวิร์คร่วมกับกฎระหว่างสองข้อมูล", วิทยานิพนธ์หลักสูตรวิทยาศาสตรมหาบัณฑิต, สาขาวิชาเทคโนโลยีสารสนเทศ, บัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, พ.ศ.2548.

[12] B. Kruatrachue, K. Warunsin and K. Siriboon, "The classified method for overlapping data", ICCA 2004

[13] M. Halkidi, Y. Batistakis and M. Vazirgiannis, "Clustering Validity Checking Method:Part II", ACM., Vol. 31, NO.3, Sept 2002.

[14] C. Y. Chen and F. Ye, "Particle Swarm Optimization Algorithm and Its Application to Clustering Analysis", Proceeding of the 2004 IEEE, ICNSC'04, March 2004.

[15] X. Xiao, R. Dow, R. Eberhart, B. Miled and J. Oppelt, "Gene Clustering Using Self-Organizing Maps and Particle Swarm Optimization", IEEE, IPDPS'03, 2003.

[16] J. Vesanto and E. Alhoniemi., Clown Dataset in Clustering of Self-Organizing Map., http://www.cis.hut.fi/projects/ide/publications/other/clown.zip

ภาคผนวก งานวิจัยที่ได้รับการตีพิมพ์

ISSN 0125-1724

วิศวสาร

ลาดกระบัง

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

LADKRABANG ENGINEERING JOURNAL

ปีที่ 22 ฉบับที่ 4 ธันวาคม 2548

การจัดกลุ่มโหนดใน Self- Organizing Map โดยใช้เจเนติกอัลกอริทึ่ม

Node Clustering in Self-Organizing Map using Genetic Algorithms

กสานติ์ ศรีกุลนาถ พรเทพ โรจนวสุ ไพฑูรย์ ศรีนิล เอื้อน ปิ่นเงิน
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สำนักวิจัยการสื่อสารและเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ 10520
E-mail: {s4061633, s8060022, s8060027, kpouen}@kmitl.ac.th

บทคัดย่อ

การจัดกลุ่มข้อมูลโดยใช้ Self-Oranizing Map (SOM) ข้อมูลจะถูกจัดให้อยู่ในรูปแบบของแผนภาพ 2 มิติ จุดเด่นคือกลุ่มของข้อมูลที่มีลักษณะคล้ายกันจะอยู่ในโหนดใกล้ที่เคียงกัน แต่ในกรณีที่แผนภาพมีขนาดใหญ่ข้อมูลที่อยู่ในกลุ่มเดียวกันอาจจะแตกออกเป็นกลุ่มย่อยอยู่ในโหนดที่ห่างออกไป ทำให้ไม่สามารถระบุกลุ่มได้อย่างชัดเจนและทำการสำรวจแผนภาพเป็นไปด้วยความยากลำบาก ดังนั้นจำเป็นต้องจัดกลุ่มของโหนดในแผนภาพ SOM หลังจากเสร็จสิ้นกระบวนการเรียนรู้ งานวิจัยนี้นำเสนอการจัดกลุ่มโดยใช้เจเนติกอัลกอริทึ่มกับแผนภาพ SOM เพื่อแก้ไขปัญหาการกระจายของข้อมูล โดยแบ่งการทำงานออกเป็น 2 ขั้นตอนหลักคือ ขั้นตอนแรกเป็นการจัดกลุ่มข้อมูลโดยใช้แผนภาพ SOM ขั้นตอนที่สองเป็นการจัดกลุ่มโหนดของแผนภาพ SOM โดยใช้เจเนติกอัลกอริทึ่ม ในการทดลองกับข้อมูลรูปแบบพฤติกรรมการบุกรุกเครือข่าย และเปรียบเทียบกับการจัดกลุ่มโหนดโดยใช้วิธี K-mean ผลปรากฏว่าการจัดกลุ่มโดยใช้เจเนติกอัลกอริทึ่มที่นำเสนอนั้นให้ประสิทธิภาพในการจัดกลุ่มที่ดีกว่า โดยวัดประสิทธิภาพจากค่าเอนโทรพี

Abstract

Data clustering using self-organizing map (SOM) is represented as a two dimensional map. The advantage of this method is that similar feature of data are clustered into the neighbor node. In any case of a large map, the SOM will separate those data into sub-groups that are in the other nodes. It is difficult to identify the appropriate groups. Therefore, after finished the training process it is necessary to cluster the node in SOM's map again. In this research we present the clustering method using genetic algorithm in SOM's map. Our algorithm has two processes. First, cluster data using SOM and second, cluster nodes in SOM using genetic algorithm. In experiment, we applied our algorithm to intrusion network detection dataset comparing between genetic algorithms and K-mean method. The clustering result of genetic algorithm yields a better performance than that of K-mean based on Entropy value.

1. บทนำ

Self-Oranizing Map (SOM) [1] นิยมนำมาใช้ในการจัดกลุ่มข้อมูล n มิติ เนื่องจากผลที่ได้จะถูกแสดงอยู่ในรูปแบบของแผนภาพ 2 มิติ โดยข้อมูลที่มีลักษณะคล้ายกันจะอยู่ในโหนดบริเวณใกล้เคียงกัน ซึ่งเป็นคุณลักษณะเด่นที่ทำให้ SOM ถูกนำมาใช้กันอย่างกว้างขวางในหลาย ๆ ด้าน [2]

อย่างไรก็ตามในกรณีที่แผนภาพมีขนาดใหญ่ทำให้เลือกดูข้อมูลนั้นทำได้ค่อนข้างลำบากเนื่องจากเราไม่ทราบขอบเขตของกลุ่มข้อมูลที่แน่นอน อีกทั้งในบางกรณีข้อมูลที่อยู่ในกลุ่มเดียวกันอาจจะแยกออกเป็นกลุ่มย่อยอยู่ในโหนดที่ห่างแยกออกไป ทำให้ผู้ใช้เกิดความยากในการเลือกดูข้อมูลและไม่สามารถระบุกลุ่มของข้อมูลได้ จึงได้มีการจัดกลุ่มโหนดของแผนภาพ SOM ใน [4] โดยนำเสนอวิธีการจัดกลุ่มโหนดโดยใช้ การจัดกลุ่มแบบ K-mean แต่ยังไม่สามารถจัดกลุ่มข้อมูลที่กระจายห่างออกไปไกลได้ ใน [3] ได้นำเสนอการจัดกลุ่มข้อมูลโดยใช้เจเนติกอัลกอริทึม สามารถช่วยจัดการกับกลุ่มข้อมูลที่ไม่มีรูปแบบและกระจายกันได้เป็นอย่างดี

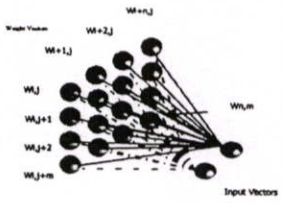
งานวิจัยนี้นำเสนอการจัดกลุ่มโหนดของแผนภาพ SOM โดยใช้การจัดกลุ่มแบบเจเนติกเพื่อแก้ไขปัญหาการกระจายของกลุ่มข้อมูล ซึ่งแบ่งการนำเสนอออกเป็นหัวข้อดังนี้ การจัดกลุ่มข้อมูลโดยใช้แผนภาพ SOM ปัญหาของแผนภาพ SOM การจัดกลุ่มโหนดของแผนภาพ SOM ด้วยวิธีเจเนติกอัลกอริทึม ผลการทดลองรวมทั้งข้อเสนอแนะ

2. Self-Organizing Map

SOM คือนิวรอนเน็ตเวิร์กแบบไม่มีผู้สอนประกอบ ด้วยอินพุทเวคเตอร์ที่มีขนาด n มิติ และมีโหนดของนิวรอนขนาดสองมิติ ซึ่งแต่ละโหนดของนิวรอนประกอบไปด้วยเวคเตอร์น้ำหนักแทนด้วย w_i โดยที่ขนาดของเวคเตอร์น้ำหนักจะต้องมี ขนาดเท่ากับอินพุตเวคเตอร์

$$x(t) = \{x_1, x_2, x_3, \ldots\ldots, x_n\}$$

$$w_i(t) = \{w_1, w_2, w_3, \ldots\ldots\ldots, w_n\}$$



รูปที่ 1 แบบแผนภาพจำลอง SOM

กระบวนการเรียนรู้ของ SOM เกิดขึ้นจากการปรับค่าของเวคเตอร์น้ำหนักที่มีต่ออินพุตเวคเตอร์ ในแต่ละรอบ t ของการเรียนรู้จะทำการสุ่มเลือกอินพุตเวคเตอร์ $x(t)$ ทำการเปรียบเทียบกับโหนดทุกโหนดเพื่อที่จะหาโหนดชนะสำหรับอินพุตเวคเตอร์นั้น ฟังก์ชั่นที่มักใช้ในการเปรียบเทียบหาความห่างของข้อมูลคือฟังก์ชั่นวัดระยะห่างแบบยูคลิดเดียน(Euclidean distance) ระยะห่างระหว่างอินพุตเวคเตอร์กับโหนดน้อยที่สุดจะเป็นโหนดชนะดังสมการที่ 1

$$\|x(t) - w_c(t)\| = \min \|x(t) - w_i(t)\| \qquad (1)$$

จากนั้นเวคเตอร์น้ำหนักของโหนดชนะจะถูกปรับค่า โดยการปรับจะพิจารณาจากผลต่างของอินพุตเวคเตอร์และเวคเตอร์น้ำหนัก โดยอัตราการเรียนรู้แต่ละรอบจะค่อย ๆ ลดลง นอกจากการเรียนรู้ที่เกิดขึ้นที่โหนดชนะแล้ว โหนดใกล้เคียง (neighborhood nodes) จะเกิดการเรียนรู้ด้วย โดยเวคเตอร์น้ำหนักของโหนดใกล้เคียงจะปรับค่าให้เข้าใกล้กับอินพุตเวคเตอร์เดียวกัน เพื่อเพิ่มโอกาสให้อินพุตใหม่ที่ใกล้เคียงกับอินพุตเดิมสามารถที่จะมี โหนดชนะใหม่ใกล้กับโหนดชนะเดิมได้ สมการในการปรับค่าสามารถแสดงได้ดังสมการที่ 2

$$w_i(t+1) = w_i(t) + \alpha(t) h_{ci}(t)[x(t) - w_i(t)] \qquad (2)$$

โดย $x(t)$ คืออินพุตเวคเตอร์ $w_i(t)$ คือเวคเตอร์น้ำหนักของโหนด i $\alpha(t)$ คืออัตราการเรียนรู้ในแต่ละรอบและ ที่ t คือรอบในการเรียนรู้ ซึ่งแสดงเป็นสมการเชิงเส้นได้ดังสมการที่ 3

$$\alpha(t+1) = \alpha(0) e^{-\frac{t}{T}} \qquad (3)$$

เมื่อ T คือจำนวนรอบทั้งหมด และ t คือจำนวนรอบที่ปัจจุบัน, $h_{ci}(t)$ คือฟังก์ชั่นที่ใช้ในการกำหนดขนาดของโหนดใกล้เคียงโดยทั่วไปแล้วโดยใช้ฟังก์ชั่น Gaussian ดังแสดงตามสมการที่ 4

$$h_{ci}(t) = e^{-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}} \quad (4)$$

เมื่อ $\|r_c - r_i\|$ คือระยะห่างระหว่างโหนด i กับโหนดชนะ c $\sigma(t)$ คือรัศมีของบริเวณโหนดใกล้เคียง โดยปกติรัศมีจะค่อยๆ ลดลงตามจำนวนรอบในการเรียนรู้ ดังแสดงตามสมการที่ 5

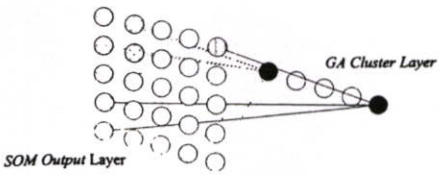
$$\sigma(t+1) = 1+(\sigma(t)-1)\times\frac{T-t}{T} \quad (5)$$

3. ปัญหาของแผนภาพ SOM

แผนภาพ SOM ที่ผ่านกระบวนการเรียนรู้แล้ว ยังไม่สามารถบอกกลุ่มของข้อมูลได้อย่างชัดเจน เนื่องจากกลุ่มของข้อมูลกระจายไปตามโหนดต่างๆ ในกรณีที่แผนภาพมีขนาดใหญ่ข้อมูลที่อยู่ในกลุ่มเดียวกันอาจจะแยกออกเป็นกลุ่มย่อยอยู่ในโหนดที่ห่างออกไปได้ ทำให้เราไม่สามารถระบุกลุ่มได้อย่างชัดเจนและการเลือกดูข้อมูลจากแผนภาพทำได้ยาก ดังนั้นเราจำเป็นต้องจัดกลุ่มของโหนดในแผนภาพ SOM หลังจากเสร็จสิ้นกระบวนการเรียนรู้เพื่อให้การสำรวจและการวิเคราะห์มีประสิทธิภาพมากขึ้น

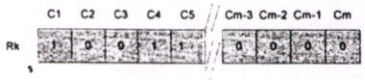
4. จัดกลุ่มข้อมูลจากแผนภาพ SOM ด้วย GA

ในงานวิจัยนี้ได้นำเสนอการใช้เจเนติกอัลกอริทึมเพื่อจัดกลุ่มโหนดในแผนภาพ SOM โดยสร้างแผนภาพเลเยอร์ขึ้นอีก 1 ชั้น เพื่อใช้สำหรับจัดกลุ่มโหนดที่คล้ายกันแต่กลุ่มห่างกัน ดังแสดงในรูปที่ 2



รูปที่ 2 แสดงการสร้างแผนภาพเลเยอร์เจเนติก สำหรับจัดกลุ่มโหนด SOM

การกำหนดรูปแบบของโครโมโซม โดยให้ขนาดของโครโมโซมเท่ากับจำนวนโหนดของแผนภาพ SOM นั่นคือ $R_k=\{C_1, C_2,\ldots,C_m\}$ เมื่อ m คือจำนวนโหนด โดยที่ $C_i \in \{0,1\}$



รูปที่ 3 แสดงโครงสร้างการสุ่มของโครโมโซม

กำหนดให้ U_k และ $\overline{U_k}$ คือเซตของตำแหน่งบิตในโครโมโซม R_k ที่มีค่าเป็น 1 และ 0 ตามลำดับดังสมการที่ 6

$$\begin{aligned} U_k &= \{j \mid j^{th} \text{ bit of } R_k \text{ is } 1\} \\ \overline{U_k} &= \{j \mid j^{th} \text{ bit of } R_k \text{ is } 0\} \end{aligned} \quad (6)$$

กำหนดให้เมทริกซ์ $M_{m\times m}$ คือเมทริกซ์แสดงระยะห่างระหว่างโหนดต่างๆ ในแผนภาพ SOM ตามสมการที่ 7

$$D(i,j) = \|C_i - C_j\|, i \neq j \quad (7)$$

ในการกำหนดฟังก์ชั่นความเหมาะสมพิจารณาจาก 2 ปัจจัยหลักคือ D_{intra} และ D_{inter} ดังแสดงสมการที่ 8 และ 9 ตามลำดับ

$$D_{intra}(R_k) = \max_{i\in U_k} \min_{\substack{j\in U_k \\ j\neq i}} D(i,j) \quad (8)$$

$$D_{inter}(R_k) = \min_{\substack{i\in U_k \\ j\in \overline{U_k}}} D(i,j) \quad (9)$$

เมื่อ $D_{intra}(R_k)$ ระบุความเหนียวแน่นของโหนดภายในโครโมโซม R_k โดยที่ $D(i,j)$ คือระยะห่างระหว่าง $C_i, C_j \in U_k$ และ $D_{inter}(R_k)$ แทนระยะทางที่สั้นที่สุดระหว่าง $C_i \in U_k, C_j \in \overline{U_k}$

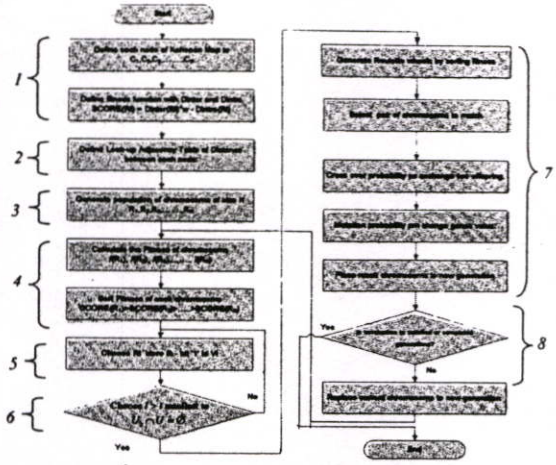
ดังนั้นให้ฟังก์ชั่นความเหมาะสมสามารถนิยามได้ดังสมการที่ 10

$$fitness(R_k) = w\times D_{inter}(R_k) - D_{intra}(R_k) \quad (10)$$

เมื่อ $w\in[1,3]$ แทนค่าน้ำหนักเพื่อกำหนดความสำคัญให้กับ D_{inter} ถ้าค่าน้ำหนักมากแสดงว่าให้ความสำคัญสูง ค่าน้ำหนักจะมีค่าระหว่าง 1 ถึง 3 ซึ่งได้จากการทดลอง

5. ขั้นตอนการจัดกลุ่มโดยใช้เจเนติกอัลกอริทึม

ในการจัดกลุ่มโดยใช้เจเนติกอัลกอริทึม จะมีขั้นตอนดังแสดงในรูปที่ 4



รูปที่ 4 การทำงานของการจัดกลุ่มข้อมูลจากแผนภาพ SOM ด้วยวิธีเจเนติกอัลกอริทึม

ขั้นที่ 1 กำหนดให้โหนดแต่ละโหนดของแผนภาพ SOM แทนด้วย $C_1, C_2, C_3, \ldots, C_N$

ขั้นที่ 2 สร้างตารางแสดงระยะห่างความเหมือนของข้อมูลในแต่ละโหนด

ขั้นที่ 3 ในการทดลองจะสร้างโครโมโซมจำนวน 50 ตัว

ขั้นที่ 4 หาค่าความเหมาะสมของแต่ละโครโมโซมโดยใช้สมการที่ 10 แล้วนำมาเรียงลำดับตามค่าความเหมาะสมจากมากไปหาน้อย และกำหนดให้ k=1, U=∅

ขั้นที่ 5 พิจารณา R_k เพื่อสร้าง V_k โดย

$$V_k = \{C_j \mid j \in U_k\} \qquad (11)$$

ให้ $U = U \cup U_k$

ขั้นที่ 6 เลือก $l > i$ ที่น้อยที่สุดที่ทำให้ $U_l \cap U = \varnothing$ ถ้าไม่มีให้ข้อนกลับทำขั้นตอนที่ 5 ใหม่

ตัวอย่างเมื่อ R1 มีค่าความเหมาะสมมากกว่า R2 และ R2 มีค่าความเหมาะสมมากกว่า R3 โครโมโซม R1 มี U_1 = {C1, C2, C3 C4} แล้ว R2 มี U_2 = {C1, C4, C5, C6} และ R3 มี U_3 = {C5, C6} จะเห็นว่า $U_1 \cap U_2 \neq \varnothing$ ดังนั้น R2 จะสามารถตัดทิ้งไปได้ และพิจารณา R3 จะเห็นว่า C5 และ C6 ไม่เป็นสมาชิกของ R1 ดังนั้นสามารถรวมโหนด C5 และ C6 เข้าด้วยกันได้ พิจารณาจนครบ R_N เมื่อทำกระบวนการเจเนติกอัลกอริทึมแล้ว จะได้ผลของการจัดกลุ่มของแต่ละโหนดอยู่ใน Vi ที่ทำให้สามารถระบุกลุ่มข้อมูลได้อย่างชัดเจนขึ้น

ขั้นที่ 7 ทำการ Cross Over และ Mutation

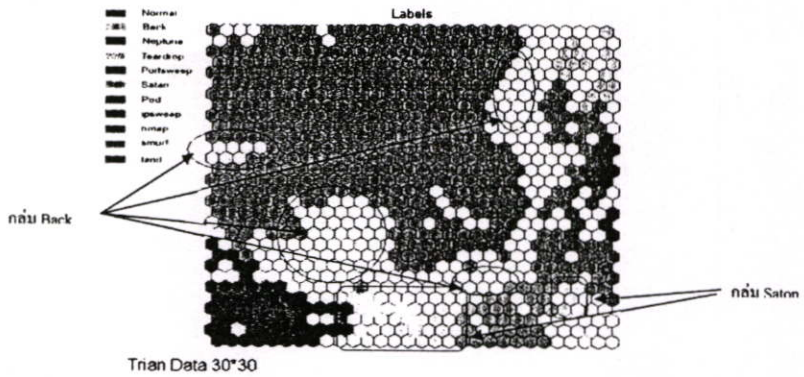
การทำ Cross Over เริ่มต้นโดยการสุ่มเลือกประชากรหรือโครโมโซม มาสองตัวให้เป็นคู่เป็นโครโมโซมพ่อและแม่ แล้วทำการสุ่มตำแหน่งที่ทำการ Crossover สลับตำแหน่งระหว่างพ่อและแม่ ซึ่งจะได้ประชากรรุ่นถัดไปจำนวน 2 โครโมโซม ในแต่ละรอบการทำ Crossover จะพิจารณาจากค่า Pc และเกิดประชากรเพิ่มเป็น 2 เท่า

การทำ Mutation เป็นการเปลี่ยนค่าระดับบิต จาก 0 เป็น 1 ของแต่ละโครโมโซมที่ถูกเลือกมา โดยอาศัยความเป็นไปได้จากค่า Pm เพื่อกำหนดการทำ mutation

ขั้นที่ 8 ทำขั้นตอนที่สี่จนกว่าจะครบเจนเนอเรชั่นที่กำหนดไว้

6. การทดลอง

ข้อมูลที่ทำการทดลองเป็นรูปแบบของการบุกรุกเครือข่าย [5] ซึ่งได้มีการจัดกลุ่มไว้ก่อนแล้ว 11 กลุ่ม (Classify) ได้แก่ back, ipsweep, land, neptune nmap, normal, pod, portsweep, satan, smurf และ teardrop มีข้อมูลทั้งหมด 42,574 ชุดข้อมูล แต่ละข้อมูลมีมิติ 41 มิติ ได้แก่ *1. Basic feature* เป็นคุณลักษณะพื้นฐานที่ได้จากแพคเกจข้อมูลที่สื่อสารในเครือข่าย มีจำนวน 9 features 2. *Content feature* เป็นคุณลักษณะที่เก็บรวบรวมข้อมูลที่แสดงให้เห็นถึงพฤติกรรมน่าสงสัย เช่น ความผิดพลาดในการล็อกอิน จำนวน 13 features *3. Traffic feature* เป็นคุณลักษณะที่เก็บรวบรวมข้อมูลที่ลักษณะของการสื่อสารจำนวน 9 features *4. Host based feature* เป็นคุณลักษณะที่เก็บรวบรวมข้อมูลที่แสดงลักษณะของการสื่อสารไปยังเครื่องปลายทางเครื่องเดิมตลอดเวลา จำนวน 10 features ในการทดลองจะนำ Label ของแต่ละชุดข้อมูลออกไปเพื่อให้สามารถจัดกลุ่ม

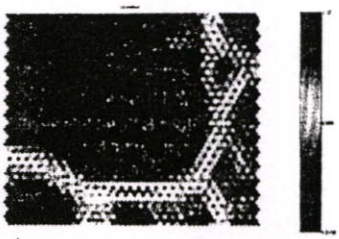


รูปที่ 5 แสดงการแผนภาพ SOM หลังจากผ่านการเรียนรู้ จากข้อมูลรูปแบบการบุกรุกเครือข่าย

โดยไม่อาศัยผู้สอน ข้อมูลที่นำมาทดสอบจะถูกสร้างแผนภาพ SOM สองมิติ ดังรูปที่ 5

6.1 ผลการทดลอง

โดยกำหนดให้แผนภาพขนาด 30×30 อัตราการเรียนรู้มีค่าเท่ากับ 0.2 น้ำหนักประจำแต่ละโหนดจะเป็นตัวแทนของกลุ่มอินพุทข้อมูลที่ตกในโหนดนั้น จากรูปที่ 5 แสดงแผนภาพ SOM ที่เรียนรู้แล้ว ซึ่งจะพบว่าข้อมูลการบุกรุกเครือข่ายในรูปแบบ Back จะกระจายอยู่ในแผนภาพที่ไม่ได้อยู่กลุ่มเดียวกัน เช่นเดียวกับข้อมูลรูปแบบ Satan ซึ่งจะทำให้การสำรวจในแผนภาพนั้น กระโดดไปมา ขาดประสิทธิภาพและไม่สามารถมองเห็นภาพรวมของข้อมูลได้



รูปที่ 6 U-Matrix แสดงการกระจายของข้อมูลหลังการเรียนรู้ด้วย SOM โดยใช้ข้อมูลรูปแบบการบุกรุกเครือข่าย

รูปที่ 6 แสดงแผนภาพ U-matrix ของแผนภาพ SOM ที่ได้โดยแผนภาพ U-matrix คือแผนภาพแสดงระยะห่างระหว่างโหนดใกล้เคียง โหนดที่สว่างจะเป็นโหนดไม่ใช่โหนดชนะ และโหนดที่มีสีเข้มแสดงความหนาแน่นของข้อมูลมาก โดยอาศัยแผนภาพ U-matrix ที่ได้จะเห็นได้ว่าไม่สามารถกำหนดขอบเขตได้ชัดเจน และไม่สามารถจัดกลุ่มที่กระจายในแผนภาพ SOM ได้

6.2 การจัดกลุ่มโหนดโดยใช้เจเนติกอัลกอริทึม

แผนภาพโคโฮเนนหลังการเรียนรู้แล้วจะกำหนดให้โหนด 900 โหนด (30×30) สร้างเป็นโครโมโซมจำนวน 1 สตริง จำนวน 50 โครโมโซม ในการกำหนดจำนวนประชากรของโครโมโซมจะพิจารณาจากจำนวนกลุ่มที่เกิดขึ้น 11 กลุ่ม โดยเผื่อการเกิดโครโมโซมที่ซ้ำกันและทำการทดลองซ้ำเพื่อใช้พิจารณากำหนดจำนวนประชากรที่เหมาะสม กำหนดให้เกิดโครโมโซมรุ่นใหม่ 50 รุ่น และยึดโครโมโซมรุ่นเดิมที่มีค่าฟิตเนสที่ดีที่สุดจำนวน 10 เปอร์เซ็นต์ ผสมกับจำนวนโครโมโซมรุ่นใหม่ที่จัดเรียงฟิตเนส 90 เปอร์เซ็นต์ของจำนวน 50 โครโมโซม กำหนดค่า Pc=0.8 และ Pm=0.1

การวัดประสิทธิภาพในการจัดกลุ่มเราจะใช้ค่าเอนโทรพี(Entropy)[6][7] ในการวัดซึ่งมีสมการเป็น

$$E_j = -\sum_i p_{ij} \log(p_{ij}) \qquad (13)$$

เรียนรู้ด้วย SOM โดยใช้ข้อมูลรูปแบบการบุกรุกเครือข่ายและผลรวมของค่า Entropy

$$E_{cs} = \sum_{j-1}^{m} \frac{n_j \times E_j}{n} \qquad (14)$$

เมื่อ p_{ij} เป็นความน่าจะเป็นของสมาชิก j ในกลุ่ม i n_j เป็นเอกสารในกลุ่ม j, m เป็นจำนวนกลุ่มเอกสาร และ n เป็นจำนวนของเอกสารทั้งหมด

จากการทดลองการจัดกลุ่มโหนดโดยใช้เจเนติกอัลกอริทึมจะสามารถจัดกลุ่มแบบอัตโนมัติได้กลุ่มออกมาทั้งหมด 13 กลุ่ม ซึ่งค่าเอนโทรพีรวม ดังตารางที่ 1 เมื่อเทียบกับการจัดกลุ่มโดยใช้วิธี K-mean และกำหนดค่า K ให้เท่ากับ 11 และ 15 ซึ่งหมายถึงจำนวนกลุ่มที่ถูกจัดกลุ่มจะมี 11 และ 15 กลุ่มข้อมูล จะเห็นว่าการจัดกลุ่มโดยใช้เจเนติกอัลกอริทึมจะให้ค่าเอนโทรพีที่น้อยกว่าซึ่งหมายถึงประสิทธิภาพการจัดกลุ่มที่ดีกว่าเมื่อเทียบจำนวนกลุ่มที่เท่ากัน เมื่อทดลองเพิ่มกลุ่ม K ให้กับวิธี K-mean จะเห็นว่าค่าเอนโทรพีที่ได้จะค่อย ๆ มีค่าลดลง แต่จะเกิดจำนวนโหนดในเลเยอร์ของเจเนติกมากขึ้นซึ่งแสดงว่าข้อมูลจะถูกจัดให้เป็นกลุ่มย่อยมากขึ้นทำให้เอนโทรพีรวมลดลงด้วย

ตารางที่ 1 แสดงค่า Entropy เปรียบเทียบ GA และ K-mean

Clustering Method	ค่า Entropy
GA (จัดกลุ่มได้ 13 กลุ่ม)	0.2250
K-Mean (k=11)	0.356
K-Mean (k=15)	0.317
K-Mean (k=20)	0.2725
K-Mean (k=25)	0.2190

แต่การกำหนดกลุ่ม K มากขึ้นทำให้กลุ่มของข้อมูลถูกแบ่งเป็นกลุ่มย่อย ๆ มากขึ้นตาม ทำให้เราไม่สามารถระบุกลุ่มที่แท้จริงได้อย่างชัดเจน ปัญหาอีกประการในการใช้ K-mean คือเราไม่สามารถรู้ค่า K ที่เหมาะสมที่สุดสำหรับกลุ่มข้อมูลนั้นต้องทดลองหาค่า K เองซึ่งจะทำให้เสียเวลามาก และที่สำคัญ K-mean ไม่สามารถที่จะจัดกลุ่มของข้อมูลที่กระจายตามแผนภาพ SOM ได้อย่างถูกต้อง

7. สรุป

จุดประสงค์ของการจัดกลุ่มจะพยายามจัดข้อมูลที่มีความคล้ายกันมาอยู่กลุ่มเดียวกันให้มากที่สุด การจัดกลุ่มข้อมูลจากผลลัพธ์ SOM ซึ่งพบว่าโหนดบนแผนภาพมีโหนดที่เป็นกลุ่มเดียวกันกระจายออกไป ดังนั้นวิธีการจัดกลุ่มจากแผนภาพต้องไม่อาศัยจุดศูนย์กลางข้อมูล (Centriod) ซึ่งต่างจากวิธีของ K-mean ที่ต้องใช้ค่าเฉลี่ยหรือจุดศูนย์กลางข้อมูลและจำเป็นต้องรู้กลุ่มที่แน่นอนเสียก่อน พิจารณาจากค่าความเหนี่ยวแน่นภายในกลุ่มหรือเอนโทรพี วิธี K-mean จะมีค่ามากกว่าวิธีเจเนติกอัลกอริทึม ซึ่งแสดงว่าวิธี K-mean จะมีโอกาสที่ข้อมูลชนิดเดียวกันมีความหนาแน่นน้อยกว่า วิธีเจเนติกอัลกอริทึมเมื่อเทียบกับจำนวนกลุ่มที่เท่ากัน เนื่องจากวิธี K-mean จะสุ่มจุดเริ่มต้นมาก่อน

8. เอกสารอ้างอิง

[1] T Kohonen. "The Self-organizing map." Proceeding of the IEEE, Volume 78, No.9, Sep 1990.

[2] A K Jain, M N Murty and P.J. Flynn, "Data Clustering A Review " ACM Computing Surveys, Vol 31, No 3, Sept 1999

[3] Lin Yu Tseng and Shiueng Bien Yang. "A genetic clustering algorithm for data with non-spherical-shape clusters " Pattern Recognition Society, Elsevier Science Dec 1999.

[4] J Vesanto and E Alhoniemi, "Clustering of the Self-organizing Map." IEEE, No. 3, May. 2000.

[5] S Stolfo et al , The Third International Knowledge Discovery and Data Mining Tool Competition. http //kdd ics.uci.edu/databases/kddcup99/kddcup99.html

[6] M Koskela, J. Laaksonen and E.Oja, "Entropy-based measures for clustering and SOM topology preservation applied to content-based image indexing and retrieval", Proceeding of the 17th ICPR'04 IEEE, June 2001

[7] พรเทพ โรจนวสุ, "การจัดกลุ่มเอกสารโดยใช้ Self-Organizing Map แบบความเร็วสูง", ภาควิชาวิศวกรรมคอมพิวเตอร์, คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, พ.ศ 2547

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายกสานติ์ ศรีกุลนาถ
วันเดือนปีเกิด	วันที่ 4 กุมภาพันธ์ พ.ศ. 2520 ที่จังหวัดกรุงเทพมหานคร
ที่อยู่	31/859 ม.2 ตำบลคลองสาม อำเภอคลองหลวง จังหวัดปทุมธานี 12120
ประวัติการศึกษา	2542 จบการศึกษาครุศาสตร์บัณฑิต สาขาอิเล็กทรอนิกส์คอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประสบการณ์ทำงาน

พ.ศ. 2542-2545	วิศวกรด้านเครื่องมือหาค่าพิกัดดาวเทียม GPS บริษัทจีโอโซลูชั่นส์จำกัด
พ.ศ. 2545-2548	วิศวกรด้านเครื่องมือหาค่าพิกัดดาวเทียม GPS บริษัทอัลทิเมทโพซิชั่นนิ่ง จำกัด ให้คำปรึกษาทางด้านเครื่องมือหาค่าพิกัดให้กับ สปก. กรมที่ดิน กรมผังเมืองและสถานที่ราชการอื่นๆ - ซอฟท์แวร์สนับสนุนเครื่องมือหาค่าพิกัดดาวเทียม
พ.ศ. 2546-ปัจจุบัน	อาชีพอิสระทำงานด้าน Website และระบบคอมพิวเตอร์ให้กับหน่วยงานราชการต่างๆ