การนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม

DATA CLUSTERING BASED ON GENETIC ALGORITHM AND ITS HISTORITICAL INFORMATION

อุไรวรรณ กะกุลพิมพ์
URAIWAN KAKULPHIMP

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2549

ISBN 974-15-2080-6

การนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม

DATA CLUSTERING BASED ON GENETIC ALGORITHM AND ITS HISTORITICAL INFORMATION

อุไรวรรณ กะกุลพิมพ์

URAIWAN KAKULPHIMP

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2549

ISBN 974-15-2680-6

DATA CLUSTERING BASED ON GENETIC ALGORITHM AND ITS HISTORITICAL INFORMATION

URAIWAN KAKULPHIMP

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN INFORMATION TECHNOLOGY
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2006

ISBN 974-15-2680-6

หัวข้อวิทยานิพนธ์	การนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม
นักศึกษา	นางสาวอุไรวรรณ กะกุลพิมพ์
รหัสนักศึกษา	44067033
ปริญญา	หลักสูตรวิทยาศาตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
พ.ศ.	2549
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร.อาริต ธรรมโน

บทคัดย่อ

วิทยานิพนธ์นี้เสนอการจัดกลุ่มข้อมูล (Data Clustering) โดยการนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม งานวิจัยในครั้งนี้ได้เล็งเห็นถึงความสำคัญของกระบวนการมิวเตชั่น (mutation) ซึ่งเป็นกระบวนการเปลี่ยนแปลงค่ายีน เนื่องจากการมิวเตท (mutate) โดยทั่วไปเป็นการสุ่มค่าที่ต้องการเปลี่ยนแปลง ซึ่งวิธีการดังกล่าวเป็นการสุ่มแบบไม่มีทิศทาง ดังนั้นเพื่อให้การมิวเตททำการเปลี่ยนแปลงค่ายีนอย่างมีทิศทาง จึงมีการเก็บข้อมูลประวัติ (history) ของโครโมโซมแต่ละชุด โดยการเปลี่ยนแปลงค่ายีนในโครโมโซม ยีนปัจจุบันจะปรับเข้าหายีนในโครโมโซมในอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มมาก และปรับออกจากยีนในโครโมโซมอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มน้อย ซึ่งวิธีการเช่นนี้ ทำให้การจัดกลุ่มมีความเหมาะสมและรวดเร็วยิ่งขึ้น

Thesis Title	Data Clustering Based on Genetic Algorithm and its Historical Information
Student	Ms. Uraiwan Kakulphimp
Student ID.	44067033
Degree	Master of Science
Programme	Information Technology
Year	2006
Thesis Advisor	Assoc. Prof. Dr. Arit Thammano

ABSTRACT

This paper proposes an extension to the original GA-clustering algorithm by introducing a new way to mutate the chromosome. Instead of deciding randomly to add or subtract a small random number to/from the original value of the selected position, the historical information of each chromosome is taken into consideration when mutating the chromosome. In this research, the histories of the chromosome are divided into 2 groups: "good" and "bad." The good consists of all previous stages of the chromosome that have higher fitness value than the current stage, while the bad is composed of all previous stages of the chromosome that have lower fitness value than the current stage.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก รศ.ดร.อาริต ธรรมโน ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ได้ประสิทธิ์ประสาทวิชาให้อย่างต่อเนื่อง ข้าพเจ้ารู้สึกขอบพระคุณในความกรุณาจากท่านอาจารย์เป็นอย่างมาก และขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ขอกราบพระคุณคณาจารย์ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณทบวงมหาวิทยาลัย และมหาวิทยาลัยบูรพา ที่สนับสนุนทุนการศึกษาในการศึกษาและวิจัยในครั้งนี้

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในคณะเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

ขอขอบคุณเจ้าหน้าที่ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้ความช่วยเหลือ และคำแนะนำต่างๆ ในการทำวิทยานิพนธ์

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจและให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

อุไรวรรณ กะกุลพิมพ์

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญตาราง (ต่อ)

ตารางที่ หน้า

สารบัญตาราง (ต่อ)

ตารางที่ หน้า

สารบัญรูป

รูปที่ หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การจัดกลุ่มข้อมูล (Clustering) เป็นส่วนหนึ่งซึ่งมีความสำคัญต่อการแก้ปัญหาต่างๆ เช่น ด้านการตลาด การจัดกลุ่มข้อมูลช่วยให้นักการตลาดสามารถจัดกลุ่มข้อมูลของลูกค้า เพื่อนำข้อมูลดังกล่าวไปพัฒนาจุดมุ่งหมายของแผนการตลาด ในธุรกิจการประกันภัยมีการจัดกลุ่มข้อมูลของผู้ถือประกันภัยรถยนต์ตามค่าเฉลี่ยของค่าชดใช้ความเสียหาย ในการวางผังเมืองมีการจัดกลุ่มของบ้านตามประเภทของบ้าน ราคาและสถานที่ตั้ง เป็นต้น

Clustering เป็นกระบวนการรวมกลุ่มข้อมูลให้อยู่ในรูปคลัสเตอร์ (cluster) โดยการพิจารณาจัดกลุ่มจะพิจารณาจากความคล้ายคลึงหรือความแตกต่าง อาจกล่าวได้ว่า สมาชิกที่อยู่ในกลุ่มเดียวกันจะมีความคล้ายคลึงกันมากที่สุด ในขณะที่สมาชิกที่อยู่ต่างกลุ่มกันจะมีความแตกต่างกันมากที่สุด

เทคนิคในการจัดกลุ่มข้อมูลมีหลายวิธีการและในแต่ละวิธีการมีอัลกอริทึมที่มีผู้นำเสนอไว้จำนวนมาก ในงานวิจัยนี้ได้ใช้แนวคิดในการจัดกลุ่มข้อมูลแบบ partitioning method ซึ่งเป็นการจัดกลุ่มข้อมูลจากข้อมูลจำนวน n ข้อมูล แบ่งข้อมูลออกเป็น partition จำนวน k กลุ่ม ตัวอย่างอัลกอริทึมที่ใช้หลักการนี้ ได้แก่ k-means, k-medoids, CLARA และ CLARANS เป็นต้น

งานวิจัยต่าง ๆ ที่เกี่ยวข้องกับวิธีหรือกระบวนการจัดกลุ่มข้อมูลแบบ partitioning method นั้น ได้มีการศึกษาและพัฒนาอย่างกว้างขวาง อาทิเช่น

Ujjwal Maulik และ Sanghamitra Bandyopadhyay [1] ได้เสนอเทคนิคการจัดกลุ่มข้อมูลด้วยจีเนติกอัลกอริทึม (GA-Clustering) โดยโครโมโซมแสดงสายอักขระตัวเลขจำนวนจริงซึ่งเป็นจุดศูนย์กลางของกลุ่ม และการทำงานของ อัลกอริทึมนี้มีพื้นฐานมาจาก k-means อัลกอริทึม

Sanghamitra Bandyopadhyay และ Ujjwal Maulik [2] ได้เสนอเทคนิคการจัดกลุ่มข้อมูลด้วยจีเนติกอัลกอริทึมสำหรับการจำแนกข้อมูลรูปภาพ อัลกอริทึมดังกล่าวสามารถจัดกลุ่มของข้อมูลได้โดยไม่กำหนดจำนวนกลุ่มที่แน่นอน แต่เป็นการกำหนดจำนวนกลุ่ม k ให้อยู่ในช่วง $[k_{mim}, k_{max}]$ โดยโครโมโซมแสดงสายอักขระตัวเลขจำนวนจริงหรือสัญลักษณ์ # และใช้ Davies-Bouldin index สำหรับวัดความเที่ยงตรงของกลุ่มข้อมูล

Weiguo Sheng และ Xiaohui Liu [3] ได้เสนอเทคนิค local search heuristic และนำเทคนิคดังกล่าวมาผสานกับจีเนติกอัลกอริทึมและ k-medoid clustering สำหรับการจัดกลุ่มชุดข้อมูลขนาดใหญ่ ซึ่งเป็นการแก้ปัญหาแบบ NP-hard optimization โดย local search heuristic จะทำการเลือก medoid จำนวน k medoid จากชุดข้อมูลและพยายามปรับค่าความแตกต่างของแต่ละกลุ่มให้มี

ผลรวมความแตกต่างกันภายในกลุ่มน้อยที่สุด การจัดกลุ่มข้อมูลด้วย k-medoid อัลกอริทึมแบบใหม่นี้เรียกว่า Hybrid K-medoid Algorithm (HKA)

Hwei-Jen Lin, Fu-Wen Yang และ Yang-Ta Kao [4] เสนอเทคนิคการจัดกลุ่มด้วยจีเนติกอัลกอริทึมโดยการเลือกจุดศูนย์กลางของกลุ่มจากข้อมูลโดยตรงและเพิ่มความรวดเร็วในการคำนวณค่าความเหมาะสมด้วยการสร้างตารางเพื่อเก็บข้อมูล (look-up table) ซึ่งเก็บข้อมูลระยะห่างระหว่างทุกๆคู่ของข้อมูล และใช้การแสดงแทนข้อมูลด้วยข้อมูลแบบไบนารี

การทำงานของจีเนติกอัลกอริทึมประกอบไปด้วยขั้นตอนสำคัญในการเปลี่ยนแปลงค่าของข้อมูลได้แก่ การครอสโอเวอร์ (crossover) และการมิวเตชั่น (mutation) โดยการครอสโอเวอร์คือการไขว้กันของยีนในโครโมโซม โดยยีนจากคู่ของพ่อและแม่จะถูกสลับตำแหน่งกัน ประชากรพ่อ-แม่ 1 คู่ หลังจากการดำเนินการครอสโอเวอร์จะได้ประชากรลูก (offspring) 2 ประชากร การมิวเตชั่นคือการกลายพันธุ์ของโครโมโซม โดยยีนในโครโมโซมจะถูกเปลี่ยนแปลงค่าไปจากค่าเดิม ประชากร 1 ประชากรที่ผ่านกระบวนการมิวเตชั่นจะได้ประชากรใหม่ 1 ประชากร

งานวิจัยครั้งนี้ได้เล็งเห็นถึงความสำคัญของกระบวนการกลายพันธุ์ของโครโมโซม (mutation) เนื่องจากการมิวเตท (mutate) เป็นการสุ่มค่าที่ต้องการเปลี่ยนแปลง ซึ่งวิธีการดังกล่าวเป็นการสุ่มแบบไม่มีทิศทาง ดังนั้นงานวิจัยครั้งนี้จึงมีการเก็บข้อมูลประวัติของโครโมโซมแต่ละชุดเพื่อช่วยในการปรับทิศทางการเปลี่ยนแปลงของค่ายีน โดยยีนปัจจุบันจะปรับเข้าหายีนในโครโมโซมในอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มมาก และปรับออกจากยีนในโครโมโซมอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มน้อย ซึ่งวิธีการเช่นนี้ ทำให้การจัดกลุ่มมีความเหมาะสมและรวดเร็วยิ่งขึ้น

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1.2.1 เพื่อศึกษาการจัดกลุ่มของข้อมูลด้วยจีเนติกอัลกอริทึมแบบใหม่ โดยศึกษาเปรียบเทียบจากวิธีต่าง ๆ ที่ผ่านการวิจัยมาแล้ว

1.2.2 เพื่อศึกษาวิธีการที่เหมาะสมของการนำจีเนติกอัลกอริทึมมาใช้กับการจัดกลุ่มข้อมูล

1.2.3 เสนออัลกอริทึมที่ใช้ในการจัดกลุ่มข้อมูลโดยใช้พื้นฐานของจีเนติกอัลกอริทึม

1.3 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย

งานวิจัยนี้เสนอส่วนขยายของ อัลกอริทึมการจัดกลุ่มด้วยจีเนติก (Original GA-clustering algorithms) ซึ่งนำเสนอโดย Maulik และ Bandyopadhyay ด้วยการเสนอวิธีการใหม่สำหรับการมิวเตทโครโมโซมจะถูกเก็บไว้เพื่อทำการพิจารณา ในกระบวนการมิวเตชั่น ประวัติของโครโมโซมจะถูกแบ่งออกเป็น 2 กลุ่มได้แก่ "โครโมโซมกลุ่มดี" และ "โครโมโซมกลุ่มไม่ดี" โครโมโซมกลุ่มดี ประกอบด้วย โครโมโซมประวัติของโครโมโซมปัจจุบันซึ่งมีค่าความเหมาะสม

มากกว่าค่าความเหมาะสมของโครโมโซมปัจจุบัน และโครโมโซมกลุ่มไม่ดี ประกอบด้วยโครโมโซมประวัติของโครโมโซมปัจจุบันซึ่งมีค่าความเหมาะสมน้อยกว่าค่าความเหมาะสมของโครโมโซมปัจจุบัน จากนั้นค่าของโครโมโซมทั้งสองกลุ่มจะถูกนำมาพิจารณาเพื่อมิวเตทโครโมโซมปัจจุบัน โดยการมิวเตทยีนจะพิจารณาค่ายีนปัจจุบันและค่ายีนในอดีต ยีนปัจจุบันจะปรับเข้าหายีนในโครโมโซมในอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มมาก และปรับออกจากยีนในโครโมโซมอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มน้อย การพิจารณาเช่นนี้มีผลให้การปรับทิศทางของยีนมีความน่าจะเป็นที่จะเข้าถึงค่าที่ดีที่สุดได้เร็วยิ่งขึ้น

1.4 ขอบเขตของการวิจัย

1.4.1 งานวิจัยนี้ใช้การทดลองแบบ off-line

1.4.2 ชุดข้อมูลที่ใช้ในการทดลองประกอบด้วยชุดข้อมูลมาตรฐานที่ใช้ทดสอบประสิทธิภาพการจัดกลุ่มของอัลกอริทึมในกลุ่ม Clustering

1.4.3 งานวิจัยนี้มุ่งเน้นการจัดกลุ่มข้อมูลสำหรับข้อมูลที่เป็นตัวเลขเท่านั้น

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ทราบถึงบทความงานวิจัยต่าง ๆ ที่เกี่ยวข้องกับงานวิจัยครั้งนี้

1.5.2 เป็นพื้นฐานในการจัดกลุ่มข้อมูล เพื่อใช้ในการพัฒนาต่อไปในอนาคตให้เหมาะกับการใช้งานจริง

1.6 ขั้นตอนของการศึกษา

วิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของการวิจัย และขั้นตอนการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการวิจัย ซึ่งประกอบด้วยเทคนิคการจัดกลุ่มข้อมูลประเภทต่างๆ การวัดค่าความเหมือนและความแตกต่างของข้อมูลเพื่อนใช้ในการจัดกลุ่ม พื้นฐานการจัดกลุ่ม การจัดกลุ่มข้อมูลด้วยอัลกอริทึม k-means จีเนติก อัลกอริทึม (genetic algorithm) การจัดกลุ่มด้วยจีเนติกอัลกอริทึม และงานวิจัยที่เกี่ยวข้องกับการจัดกลุ่มข้อมูลด้วยอัลกอริทึมแบบต่างๆ

บทที่ 3 กล่าวถึงการนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม เริ่มจากขั้นตอนการจัดเก็บข้อมูลประวัติของโครโมโซม การนำข้อมูลประวัติดังกล่าวมาใช้ในการปรับปรุงการจัดกลุ่มในขั้นตอนของการดำเนินการมิวเตชั่น พร้อมทั้งแสดงตัวอย่างการนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม

บทที่ 4 กล่าวถึงการทดสอบการนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึมกับชุดข้อมูลทดสอบ 10 ชุด และแสดงผลการทดสอบอัลกอริทึมที่นำเสนอเปรียบเทียบกับอัลกอริทึม k-means และจีเนติกอัลกอริทึม

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะ

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 การจัดกลุ่ม (Clustering)

การจัดกลุ่ม (Clustering) เป็นกระบวนการรวมกลุ่มข้อมูลให้อยู่ในรูปคลัสเตอร์ (cluster) โดยการพิจารณาจัดกลุ่มจะพิจารณาจากความคล้ายคลึงหรือความแตกต่าง อาจกล่าวได้ว่า สมาชิกที่อยู่ในกลุ่มเดียวกันจะมีความคล้ายคลึงกันมากที่สุด ในขณะที่สมาชิกที่อยู่ต่างกลุ่มกันจะมีความแตกต่างกันมากที่สุด

โดยทั่วไปแล้วเทคนิคการจัดกลุ่มข้อมูลสามารถจำแนกออกเป็นประเภทต่างๆ ดังนี้

1. Partitioning methods เป็นเทคนิคการจัดกลุ่มข้อมูล จากข้อมูลจำนวน n ข้อมูล สร้างกลุ่มจำนวน k *กลุ่ม* โดยที่จำนวนข้อมูลจะต้องมากกว่าหรือเท่ากับจำนวนกลุ่ม ($k \leq n$) โดย การจัดข้อมูลให้อยู่ในกลุ่มใดๆ ต้องเป็นไปตามเงื่อนไขดังนี้
   a. แต่ละกลุ่มจะต้องประกอบไปด้วยสมาชิกอย่างน้อย 1 ข้อมูล
   b. แต่ละข้อมูลจะต้องมีกลุ่มที่แน่นอน

   ตัวอย่างอัลกอริทึมที่จัดกลุ่มข้อมูลด้วยเทคนิคนี้ได้แก่ k-means, k-medoids, CLARA และ CLARANS เป็นต้น
2. Hierarchical methods เป็นเทคนิคการจัดกลุ่มข้อมูลด้วยการสร้างโครงสร้างแบบลำดับชั้นของกลุ่มข้อมูล ซึ่งโครงสร้างแบบลำดับนี้อาจสร้างขึ้นในรูปของ bottom-up หรือ top-down ตัวอย่างอัลกอริทึมที่จัดกลุ่มข้อมูลด้วยเทคนิคนี้ได้แก่ agglomerative และ divisive
3. Density-based methods ส่วนมากแล้วการจัดกลุ่มด้วยเทคนิค partitioning methods มีการจัดกลุ่มของข้อมูลโดยพิจารณาค่าระยะห่างระหว่างข้อมูล วิธีการดังกล่าวจะทำให้กลุ่มที่ได้อยู่ในรูปทรงกลม แต่เทคนิคการจัดกลุ่มแบบ Density-based methods นี้เป็นการขยายขนาดของกลุ่มไปเรื่อยๆจนกระทั่งค่าความจุ (density) ของกลุ่มที่อยู่ใกล้กันมีค่ามากกว่าค่า threshold ที่กำหนดไว้
4. Grid-based methods เป็นเทคนิคการจัดกลุ่มข้อมูลโดยใช้โครงสร้างข้อมูลแบบ multiresolution grid โดยข้อมูลจะถูกเก็บลงในส่วนย่อยของโครงสร้างแบบตาราง ข้อดีของเทคนิคนี้คือทำงานได้อย่างรวดเร็วโดยไม่ขึ้นกับจำนวนข้อมูล แต่ขึ้นกับจำนวนของเซลในแต่ละมิติใน quantized space ตัวอย่างอัลกอริทึมที่จัดกลุ่มข้อมูลด้วยเทคนิคนี้ได้แก่ STRING

5. Model-based methods เป็นเทคนิคการจัดกลุ่มข้อมูลที่มีการสร้างแบบจำลองเพื่อใช้ในการจัดกลุ่มข้อมูล โดยแบบจำลองจะถูกสร้างขึ้นด้วยการพิจารณาความเหมาะสมของรูปแบบข้อมูลและแบบจำลองทางคณิตศาสตร์ การสร้างแบบจำลองด้วยเทคนิค Model-based methods นี้มีสองรูปแบบใหญ่ๆได้แก่ การสร้างแบบจำลองด้วยวิธีการทางสถิติ (statistical approach) และการสร้างแบบจำลองด้วยวิธีการทางนิวรอลเน็ตเวิร์ค (neural network approach)

จะเห็นว่าเทคนิคในการจัดกลุ่มข้อมูลมีหลายวิธีการและในแต่ละวิธีการมีอัลกอริทึมที่มีผู้นำเสนอไว้จำนวนมาก การคัดเลือกอัลกอริทึมเพื่อนำไปใช้งานขึ้นอยู่กับ ประเภทของข้อมูล และวัตถุประสงค์ของการจัดกลุ่ม

ในงานวิจัยนี้ได้ใช้แนวคิดในการจัดกลุ่มข้อมูลแบบ partitioning method ซึ่งเป็นการจัดกลุ่มข้อมูลจากข้อมูลจำนวน n ข้อมูล แบ่งข้อมูลออกเป็น partition จำนวน k กลุ่ม

2.2 การวัดค่าความเหมือนหรือความแตกต่าง

ความเหมือนหรือความแตกต่างระหว่างวัตถุ อธิบายได้ด้วยการคำนวณระยะห่างระหว่างวัตถุ 2 วัตถุ วิธีการวัดระยะห่างที่เป็นที่นิยม คือ Euclidean distance

$$d(i,j)=\sqrt{\left|x_{i1}-x_{j1}\right|^2+\left|x_{i2}-x_{j2}\right|^2+\ldots+\left|x_{in}-x_{jn}\right|^2} \tag{2.1}$$

เมื่อ $i=(x_{i1},x_{i2},\ldots,x_{in})$ และ $j=(x_{j1},x_{j2},\ldots,x_{jn})$ เป็นวัตถุที่ประกอบไปด้วยข้อมูล n มิติ

ตัวอย่างการหาค่า Euclidean distance ของข้อมูล 2 ข้อมูล $i=(1,2)$ และ $j=(1,4)$ เป็นดังนี้

$$\begin{aligned} d(i,j) &= \sqrt{\left|1-1\right|^2+\left|2-4\right|^2} \\ &= \sqrt{0+4} \\ &= 2 \end{aligned}$$

อีกหนึ่งวิธีที่เป็นที่รู้จักในการวัดระยะทาง คือ Manhattan (หรือ city block) distance

$$d(i,j)=\left|x_{i1}-x_{j1}\right|+\left|x_{i2}-x_{j2}\right|+\ldots+\left|x_{in}-x_{jn}\right| \tag{2.2}$$

ตัวอย่างการหาค่า Manhattan distance ของข้อมูล 2 ข้อมูล $i=(1,2)$ และ $j=(1,4)$ เป็นดังนี้

$$\begin{aligned} d(i,j) &= |1-1| + |2-4| \\ &= 0+2 \\ &= 2 \end{aligned}$$

ทั้ง Euclidean distance และ Manhattan distance มีคุณสมบัติดังนี้

1. $d(i,j) \geq 0$ ระยะห่างไม่เป็นจำนวนลบ
2. $d(i,i) = 0$ ระยะห่างของวัตถุและตัววัตถุเองมีค่าเป็น 0
3. $d(i,j) = d(j,i)$ ระยะห่างถือเป็น ฟังก์ชันสมมาตร (symmetric function)
4. $d(i,j) \leq d(i,h) + d(h,j)$ ระยะทางจากตรง i ไป j มีค่าน้อยกว่าหรือเท่ากับระยะทางจาก i ไป j โดยผ่าน h

Minkowski distance เป็นรูปแบบทั่วไปของทั้ง Euclidean distance และ Manhattan distance

$$d(i,j) = \left(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \ldots + |x_{in} - x_{jn}|^q \right)^{1/q} \qquad (2.3)$$

เมื่อ q เป็นเลขจำนวนเต็มบวก

จากสมการ (2.3) แสดงให้เห็นว่า Manhattan distance มีค่า $q = 1$ และ Euclidean distance มีค่า $q = 2$

2.3 พื้นฐานการจัดกลุ่ม

การจัดกลุ่มข้อมูลขนาด N มิติ เริ่มต้นจากข้อมูลจำนวน n ข้อมูล หากต้องการจัดกลุ่มข้อมูลจำนวน k กลุ่ม

จากเงื่อนไขการกำหนดกลุ่ม การพิจารณาความเป็นสมาชิกกลุ่มจะพิจารณาจากความเหมือนหรือความแตกต่างกันของข้อมูล โดยพิจารณาศูนย์กลางของกลุ่มเป็นหลัก ถ้าข้อมูลมีความคล้ายคลึงกับศูนย์กลางของกลุ่มใดมากที่สุด ข้อมูลจะถูกจัดให้เป็นสมาชิกของกลุ่มนั้น

การจัดกลุ่มข้อมูลเซต S โดย S ประกอบด้วยข้อมูลจำนวน n ข้อมูลได้แก่ $\{x_1, x_2, \ldots, x_n\}$ จัดกลุ่มข้อมูลออกเป็น k กลุ่ม โดยมีศูนย์กลางกลุ่มเป็น $C_1, C_2, \ldots, C_k$ ตามลำดับ ดังนั้นคุณสมบัติของการจัดกลุ่มจะเป็นดังนี้

$$C_i \neq \phi \qquad \text{เมื่อ } i = 1,2,\ldots,k \qquad (2.4)$$

$$C_i \cap C_j = \phi \quad \text{เมื่อ } i = 1,2,\ldots,k \text{ , } j = 1,2,\ldots,k \text{ และ } i \neq j \qquad (2.5)$$

$$\bigcup_{i=1}^{K} C_i = S \tag{2.6}$$

2.4 การจัดกลุ่มด้วยอัลกอริทึม k-means

อัลกอริทึม k-means เป็นเทคนิคการจัดกลุ่มข้อมูลที่ใช้กันอย่างแพร่หลาย โดยอัลกอริทึม k-means เป็นการจัดกลุ่มข้อมูลแบบ partitioning method แบบหนึ่ง ประกอบด้วยขั้นตอนการทำงานดังนี้

ขั้นตอนที่ 1 : กำหนดจำนวนของกลุ่ม (k)

ขั้นตอนที่ 2 : สุ่มเลือกศูนย์กลางของแต่ละกลุ่ม $z_1, z_1, ..., z_k$ จำนวน k ศูนย์กลาง จากข้อมูลทั้งหมด n ข้อมูล $\{x_1, x_2, ..., x_n\}$

ขั้นตอนที่ 3 : จัดข้อมูล $x_i, i = 1,2,...,n$ เข้าเป็นสมาชิกของกลุ่ม $C_j, j \in \{1,2,...,k\}$ โดยสมาชิกอยู่ใกล้ศูนย์กลางใดมากที่สุด จะถูกจัดให้เป็นสมาชิกของศูนย์กลางนั้นตามสมการที่ (2.7)

$$\|x_i - z_j\| < \|x_i - z_p\|, p = 1,2,...,k \text{ และ } j \neq p \tag{2.7}$$

เมื่อ x_i คือ ข้อมูลสมาชิกของกลุ่มที่ C_j

z_j คือ ศูนย์กลางของกลุ่มที่ C_j

z_p คือ ศูนย์กลางของกลุ่มที่ C_p

ขั้นตอนที่ 4 : คำนวณ clustering metric M สำหรับข้อมูลจำนวน k กลุ่ม $C_1, C_2, ..., C_k$ ซึ่งเป็นการหาระยะห่างของทุกสมาชิกในกลุ่มกับศูนย์กลางของกลุ่ม ตามสมการที่ (2.5)

$$M(C_1, C_2, ..., C_k) = \sum_{i=1}^{k} \sum_{x_j \in C_i} \| x_j - z_i \| \tag{2.8}$$

ขั้นตอนที่ 5 : คำนวณค่าศูนย์กลางใหม่ $z_1^*, z_2^*, ..., z_k^*$ ตามสมการที่ (2.9)

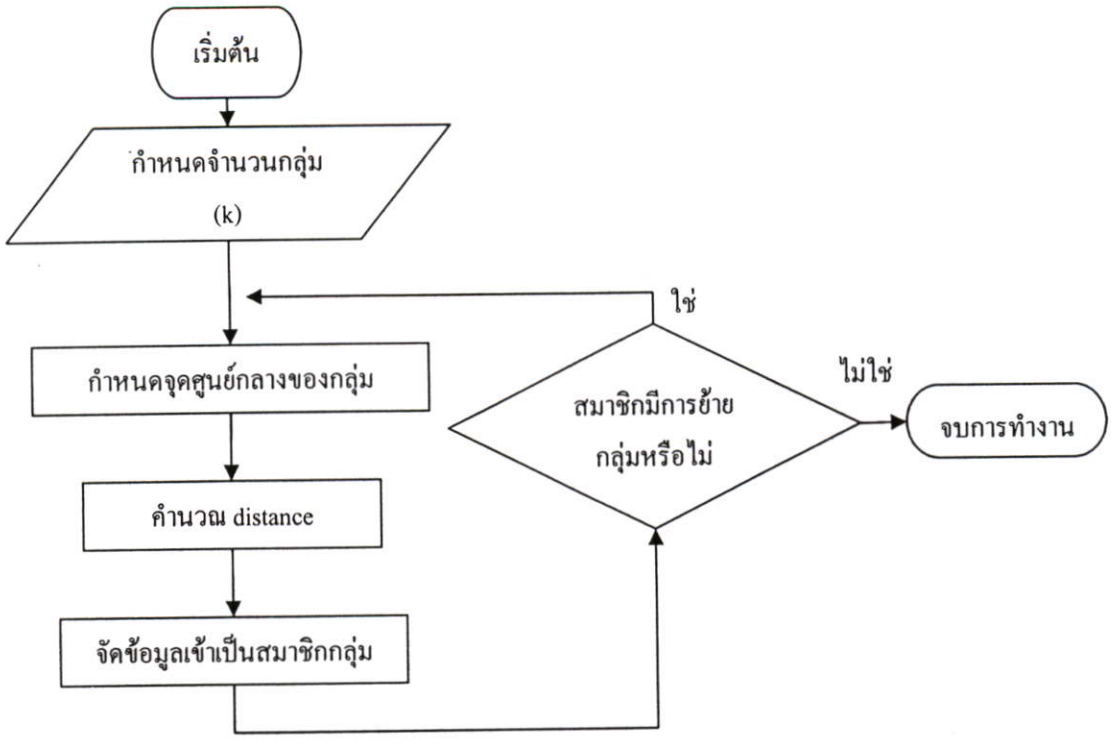
$$z_i^* = \frac{1}{n_i} \sum_{x_j \in C_i} x_j \text{ , } i = 1,2,\ldots,\text{k} \tag{2.9}$$

เมื่อ x_j คือ ข้อมูลสมาชิกของกลุ่มที่ C_i

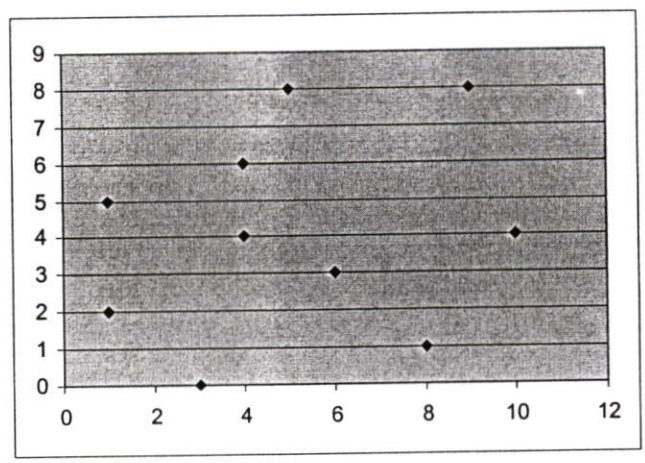
n_i คือ จำนวนสมาชิกในกลุ่มที่ C_i

z_i คือ ศูนย์กลางของกลุ่มที่ C_i

ขั้นตอนที่ 5 : ทำซ้ำขั้นตอนที่ 3 จนสมาชิกในกลุ่ม ไม่มีการย้ายกลุ่ม



รูปที่ 2.1 แผนภาพแสดงการทำงานของอัลกอริทึม k-means



รูปที่ 2.2 กราฟแสดงชุดข้อมูล A ที่ต้องการจัดกลุ่ม

จากกราฟในรูปที่ 2.2 ชุดข้อมูล A ประกอบด้วยข้อมูลดังนี้ (3,0), (1,5), (1,2), (9,8), (6,3), (4,6), (8,1), (4,4), (5,8) และ (10,4)

นำชุดข้อมูล A มาจัดกลุ่มข้อมูล โดยกำหนดให้จำนวนกลุ่มเท่ากับ 2 มีจุดศูนย์กลางเริ่มต้นคือ (2,1) และ (8,4) แสดงขั้นตอนการจัดกลุ่มด้วยอัลกอริทึม k-means ได้ดังนี้

ขั้นตอนที่ 1 : กำหนดจำนวนของกลุ่ม ให้ k = 2

ขั้นตอนที่ 2 : เลือกศูนย์กลางของแต่ละกลุ่ม ให้ $z_1 = (2,1)$ และ $z_2 = (8,4)$

ขั้นตอนที่ 3 : จัดข้อมูลให้แต่ละกลุ่ม $C_j, j \in \{1,2\}$ โดยการคำนวณหาค่าระยะห่างระหว่างข้อมูลและจุดศูนย์กลาง ด้วย Euclidean distance ดังสมการที่ 2.1

ระยะห่างของข้อมูล (3,0) และศูนย์กลาง (2,1) คำนวณได้ดังนี้

$$\begin{aligned} d(i,j) &= \sqrt{|3-2|^2 + |0-1|^2} \\ &= \sqrt{1+1} \\ &= 1.414214 \end{aligned}$$

คำนวณหาระยะห่างของทุกๆข้อมูลและทุกศูนย์กลางกลุ่มด้วย Euclidean distance ได้ผลดังตารางที่ 2.1

ตารางที่ 2.1 แสดงผลการคำนวณ Euclidean distance ระหว่างข้อมูลทุกข้อมูลจากชุดข้อมูล A กับจุดศูนย์กลาง (2,1) และ (8,4)

x	y	(2,1)	(8,4)
3	0	1.414214	6.403124
1	5	4.123106	7.071068
1	2	1.414214	7.28011
9	8	9.899495	4.123106
6	3	4.472136	2.236068
4	6	5.385165	4.472136
8	1	6	3
4	4	3.605551	4
5	8	7.615773	5
10	4	8.544004	2

เมื่อทราบระยะห่างระหว่างทุกๆ ข้อมูลและทุกๆ ศูนย์กลางกลุ่มแล้ว ทำการจัดข้อมูลเข้าเป็นสมาชิกของกลุ่มโดย ข้อมูลอยู่ใกล้ศูนย์กลางใดมากที่สุด จะถูกจัดให้เป็นสมาชิกของกลุ่มนั้น เช่น ข้อมูล (3,0) ห่างจากศูนย์กลาง (2,1) และ (8,4) เป็นระยะห่าง 1.414214 และ 6.403124

ตามลำดับ ดังนั้นข้อมูล (3,0) อยู่ใกล้ศูนย์กลาง (2,1) มากที่สุด (3,0) จึงถูกจัดให้เป็นสมาชิกกลุ่มซึ่งมี (2,1) เป็นศูนย์กลางกลุ่ม

ตารางที่ 2.2 แสดงการจัดข้อมูลจากชุดข้อมูล A เข้าเป็นสมาชิกของกลุ่ม (2,1) และ (8,4)

x	y	(2,1)	(8,4)
3	0	1.414214	6.403124
1	5	4.123106	7.071068
1	2	1.414214	7.28011
9	8	9.899495	4.123106
6	3	4.472136	2.236068
4	6	5.385165	4.472136
8	1	6	3
4	4	3.605551	4
5	8	7.615773	5
10	4	8.544004	2

หลังจากจัดข้อมูลเข้าเป็นสมาชิกของกลุ่มเรียบร้อยแล้ว จากตารางที่ 2.2 จะได้ข้อมูลสองกลุ่ม กลุ่มแรกประกอบด้วยสมาชิกได้แก่ (3,0) (1,5) (1,2) และ (4,4) และกลุ่มที่สองประกอบด้วยสมาชิกได้แก่ (9,8) (6,3) (4,6) (8,1) (5,8) และ (10,4)

ขั้นตอนต่อไปจะทำการปรับค่าศูนย์กลางของกลุ่มใหม่ โดยการหาค่าเฉลี่ยของทุกข้อมูลในกลุ่มนั้นๆ

$$x_1 = \frac{3+1+1+4}{4} = 2.25$$
$$y_1 = \frac{0+5+2+4}{4} = 2.75$$

$$x_2 = \frac{9+6+8+4+4+5}{6} = 6$$
$$y_2 = \frac{8+3+1+6+4+8}{6} = 5$$

ดังนั้นได้จุดศูนย์กลางใหม่เป็น (2.25,2,75) และ (6,5) ตามลำดับ

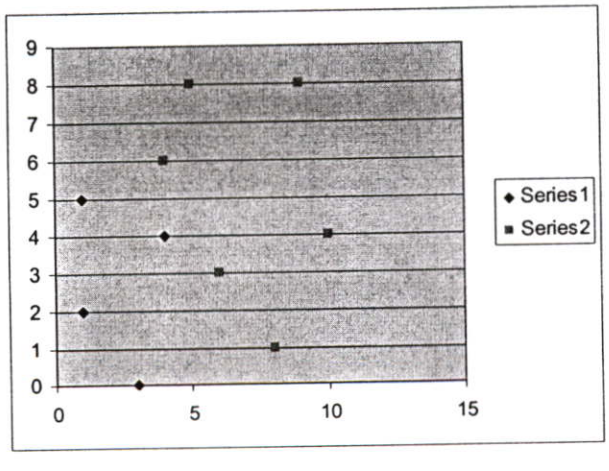
ทำซ้ำขั้นตอนที่ 3 โดยการจัดข้อมูลให้แต่ละกลุ่มตามค่าศูนย์กลาง พิจารณาการจัดข้อมูลให้เป็นสมาชิกของกลุ่มจากค่า Euclidean distance

ตารางที่ 2.3 แสดงการจัดข้อมูลจากชุดข้อมูล A เข้าเป็นสมาชิกของกลุ่ม (2.25,2.75) และ (6,5)

x	y	(2.25,2.75)	(6,5)
3	0	2.850439	5.830952
1	5	2.573908	5
1	2	1.457738	5.830952
9	8	8.551316	4.242641
6	3	3.758324	2
4	6	3.691206	2.236068
8	1	6.010408	4.472136
4	4	2.150581	2.236068
5	8	5.926635	3.162278
10	4	7.850159	4.123106

ตรวจสอบการย้ายกลุ่มของข้อมูล พบว่าไม่มีสมาชิกในกลุ่มใดเปลี่ยนแปลง ดังนั้นถือว่าสิ้นสุดการดำเนินการ การจัดกลุ่มข้อมูลด้วย อัลกอริทึม k-means

ผลการจัดกลุ่มชุดข้อมูล A ได้ข้อมูลสองกลุ่มดังนี้ กลุ่มแรกประกอบไปด้วย (3,0), (1,5), (1,2) และ (4,4) กลุ่มที่สองประกอบด้วยข้อมูล (9,8), (6,3), (4,6), (8,1), (5,8) และ (10,4)



รูปที่ 2.3 กราฟแสดงชุดข้อมูล A หลังจากจัดกลุ่มด้วยอัลกอริทึม k-means เมื่อกำหนด k=2

แม้ว่า อัลกอริทึม k-means จะเป็นอัลกอริทึมการจัดกลุ่มข้อมูลที่นิยมนำมาใช้จัดกลุ่มข้อมูล แต่ผลการจัดกลุ่มที่ได้ขึ้นอยู่กับการกำหนดค่าศูนย์กลางเริ่มต้น ซึ่งการจัดกลุ่มที่ได้อาจไม่ใช่การจัดกลุ่มที่ดีที่สุด เพราะการสุ่มเลือกศูนย์กลางเพียงครั้งเดียว อาจไม่สามารถเจอคำตอบของปัญหาได้

2.5 จีเนติกอัลกอริทึม (Genetic algorithms)

จีเนติกอัลกอริทึมถูกพัฒนาขึ้นโดย Jonh Holland จาก มหาวิทยาลัยมิชิแกน (University of Michigan) โดยอัลกอริทึมจำค้นหาคำตอบของปัญหาโดยเลียนแบบการคัดเลือกสายพันธุ์ของสิ่งมีชีวิตตามธรรมชาติ และธรรมชาติทางพันธุกรรม โดยใช้หลักการสุ่มในการค้นหาคำตอบของปัญหา

การดำรงชีวิตอยู่ในธรรมชาติ สิ่งมีชีวิตใดแข็งแรงกว่าย่อมมีโอกาสอยู่รอดได้มากกว่า สิ่งมีชีวิตที่มีสายพันธุ์ที่ดีจะมีโอกาสถูกคัดเลือกเพื่อถ่ายทอดลักษณะทางพันธุกรรมที่ดี ไปยังรุ่นต่อๆ ไป ทำให้สิ่งมีชีวิตที่มีสายพันธ์ที่ดีหรือมีความเหมาะสมที่จะดำรงชีวิตอยู่ในธรรมชาติมีโอกาสอยู่รอดได้มากกว่า ส่วนสายพันธุ์ที่ไม่ดีอาจไม่ถูกเลือกเพื่อถ่ายทอดคุณลักษณะ ดังนั้นสายพันธุ์ที่ไม่ดีก็จะสูญพันธุ์ไปในที่สุด โดยกระบวนการคัดเลือกตามธรรมชาติเป็นกระบวนการเปลี่ยนแปลงให้สิ่งมีชีวิตมีความเหมาะสมยิ่งขึ้นที่จะดำรงชีวิตอยู่ได้ในธรรมชาติ ด้วยการดำเนินการทางพันธุกรรม เช่น การครอสโอเวอร์ (crossover) หรือการมิวเตชั่น (mutation)

จีเนติกอัลกอริทึมเป็นการจำลองทางคอมพิวเตอร์เพื่อแก้ปัญหา การหาค่าที่เหมาะสมที่สุดโดยการแทนคำตอบที่มีอยู่ในลักษณะโครโมโซม (chromosome) แล้วปรับปรุงคำตอบนั้นด้วยวิธีการต่างๆคล้ายกับการดำเนินการทางพันธุกรรม เป็นการเปลี่ยนแปลงยีนแบบสุ่มด้วยการดำเนินการทางพันธุกรรม เพื่อให้ได้คำตอบที่ดีขึ้น

จีเนติกอัลกอริทึมประกอบด้วยกระบวนการดำเนินการ 3 กระบวนการที่สำคัญได้แก่

2.5.1 การคัดเลือก (selection)

การคัดเลือกประชากรเป็นไปตามแนวคิดของ กฎการคัดเลือกตามธรรมชาติ ในธรรมชาติสิ่งมีชีวิตใดมีความเหมาะสมที่จะดำรงชีวิตอยู่ในธรรมชาติมากที่สุด จะถูกคัดเลือกโดยธรรมชาติให้สามารถดำรงชีวิตอยู่ได้ ส่วนสิ่งมีชีวิตที่อ่อนแอก็จะไม่สามารถดำรงชีวิตอยู่ได้และสูญพันธุ์ไปในที่สุด เช่นเดียวกัน การคัดเลือกโครโมโซมในจีเนติกอัลกอริทึม จะพิจารณาค่าความเหมาะสม (fitness value) ของแต่ละโครโมโซม โดยโครโมโซมที่มีค่าความเหมาะสมมาก อาจถูกเลือกให้เป็นประชากรรุ่นต่อไป และอาจถูกเลือกเป็นโครโมโซมพ่อ-แม่ เพื่อถ่ายทอดคุณลักษณะไปยังรุ่นลูกต่อไป

เทคนิคการสุ่มเลือกโครโมโซมเพื่อเป็นโครโมโซมพ่อ-แม่ที่เป็นที่นิยมได้แก่ การสุ่มโดยการหมุนวงล้อถ่วงน้ำหนัก (roulette wheel) โดยกำหนดขนาดของแต่ละช่องภายในวงล้อตามความน่าจะเป็นที่จะถูกเลือก ซึ่งคำนวณได้จากสมการ (2.10)

$$P_i = \frac{f_i}{f_1 + f_2 + \ldots + f_n} \tag{2.10}$$

เมื่อ P_i คือค่าความน่าจะเป็นที่โครโมโซม i จะถูกเลือก

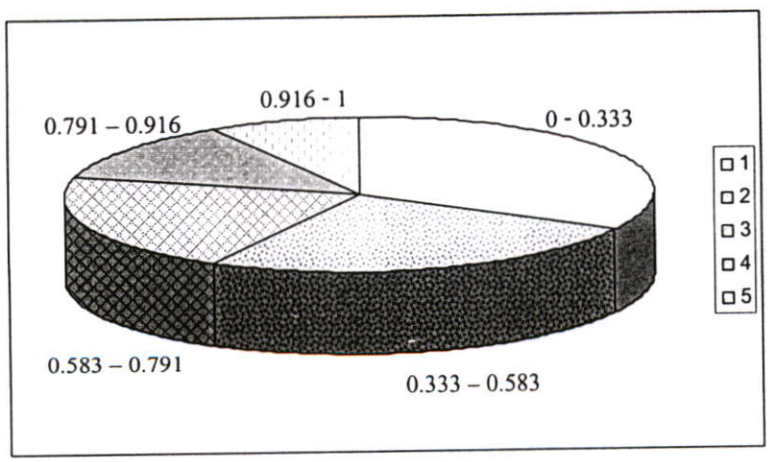
f_i คือค่าความเหมาะสมของโครโมโซม i

$f_1 + f_2 + \ldots + f_n$ คือผลรวมของค่าความเหมาะสมของทุกๆโครโมโซมในรุ่นประชากร

ตารางที่ 2.4 การคัดเลือกด้วยวิธีการหมุนวงล้อ

โครโมโซม	ค่าความเหมาะสม	ค่าความน่าจะเป็นที่จะถูกเลือก
1	0.8	0.333
2	0.6	0.250
3	0.5	0.208
4	0.3	0.125
5	0.2	0.084

ความน่าจะเป็นที่จะถูกเลือกถูกกำหนดโดยเปรียบเทียบระหว่างค่าความเหมาะสมกับผลรวมของค่าความเหมาะสมรวมทั้งหมด จากตารางที่ 2.4 จะได้ว่าโครโมโซม 1, 2, 3, 4 และ 5 มีโอกาสจะถูกเลือกเป็นโครโมโซมพ่อ-แม่ มีค่าเท่ากับ 0.333, 0.250, 0.208, 0.125 และ 0.083 ตามลำดับ



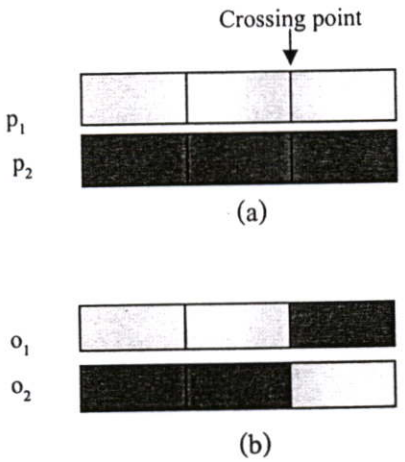
รูปที่ 2.4 วงล้อถ่วงน้ำหนัก

การสุ่มเลือกโครโมโซมจากวงล้อถ่วงน้ำหนัก เริ่มต้นจากการสุ่มเลือกค่าตัวเลขจากการหมุนวงล้อ และเลือกโครโมโซมในตำแหน่งค่าตัวเลขที่สุ่มนั้นๆ เช่น จากวงล้อถ่วงน้ำหนักในรูปที่ 2.4 สมมติว่า ค่าตัวเลขที่สุ่มได้มีค่าเป็น 0.560 ดังนั้น จะได้โครโมโซมจากการสุ่มคือ โครโมโซมที่ 3

2.5.2 การครอสโอเวอร์

การครอสโอเวอร์คือการถ่ายทอดคุณสมบัติบางส่วนของโครโมโซมพ่อ-แม่สองประชากรไปสู่โครโมโซมลูกด้วยการสลับตำแหน่งยีนในโครโมโซมพ่อและยีนในโครโมโซมแม่ การครอสโอเวอร์แบบดั้งเดิมจะเลือกจุดในการสลับตำแหน่ง (crossing point) ด้วยการสุ่มเท่านั้น โดย crossing point จะชี้ว่าตำแหน่งใดในโครโมโซมจะถูกตัดและแลกเปลี่ยนข้อมูลกันระหว่าง พ่อ-แม่

ในกระบวนการครอสโอเวอร์ โครโมโซมพ่อ-แม่ 2 ประชากร จะสามารถสร้างโครโมโซมลูกได้ 2 โครโมโซม



รูปที่ 2.5 แสดงตัวอย่างการครอสโอเวอร์

จากรูปที่ 2.5 โครโมโซมพ่อ-แม่ ได้แก่ p_1 และ p_2 และจุดในการสลับตำแหน่งคือ 2 ดังรูปที่ 2.5 (a) ดำเนินการครอสโอเวอร์โดย

o_1 จะได้ข้อมูล 2 ค่าแรกจาก p_1 และ 1 ข้อมูลหลังจาก p_2

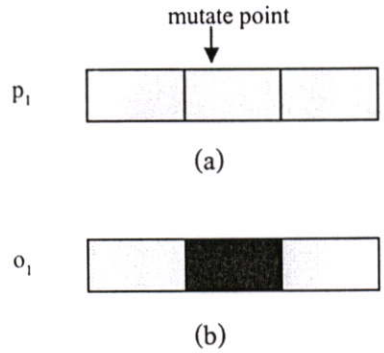
o_2 จะได้ข้อมูล 2 ค่าแรกจาก p_2 และ 1 ข้อมูลหลังจาก p_1

ดังนั้นจะได้โครโมโซมลูกที่เกิดจากการครอสโอเวอร์ได้แก่ o_1 และ o_2 ดังรูปที่ 2.3 (b)

2.5.3 การมิวเตชั่น

กระบวนการมิวเตชั่นเป็นกระบวนการหนึ่งซึ่งสำคัญในทางจีเนติก โดยการมิวเตชั่นจะทำให้เกิดประชากรใหม่ที่แตกต่างออกไป หรืออาจเรียกกว่าการผ่าเหล่าในทางชีววิทยา การมิวเตชั่นเป็นการเปลี่ยนแปลงค่ายีนในบางตำแหน่งของโครโมโซม โดยสุ่มเลือกตำแหน่งของยีนที่ต้องการ

มิวเตท (mutate) จากนั้นทำการสุ่มเพื่อเปลี่ยนแปลงค่ายีน โครโมโซมพ่อ-แม่ 1 โครโมโซมหลังจากผ่านการดำเนินการมิวเตชั่นแล้วจะได้โครโมโซมลูก 1 โครโมโซม



รูปที่ 2.6 แสดงตัวอย่างการมิวเตชั่น

จากรูปที่ 2.6 โครโมโซมที่ต้องการมิวเตทได้แก่ โครโมโซม p_1 เลือกตำแหน่งที่ต้องการมิวเตท (mutate point) คือตำแหน่งที่ 2 จากนั้นทำการสุ่มค่าเพื่อเปลี่ยนแปลงค่ายีน ในตำแหน่งที่สอง

2.6 ขั้นตอนการทำงานของจีเนติกอัลกอริทึม

ขั้นตอนการทำงานของจีเนติกอัลกอรึทึมเป็นดังนี้

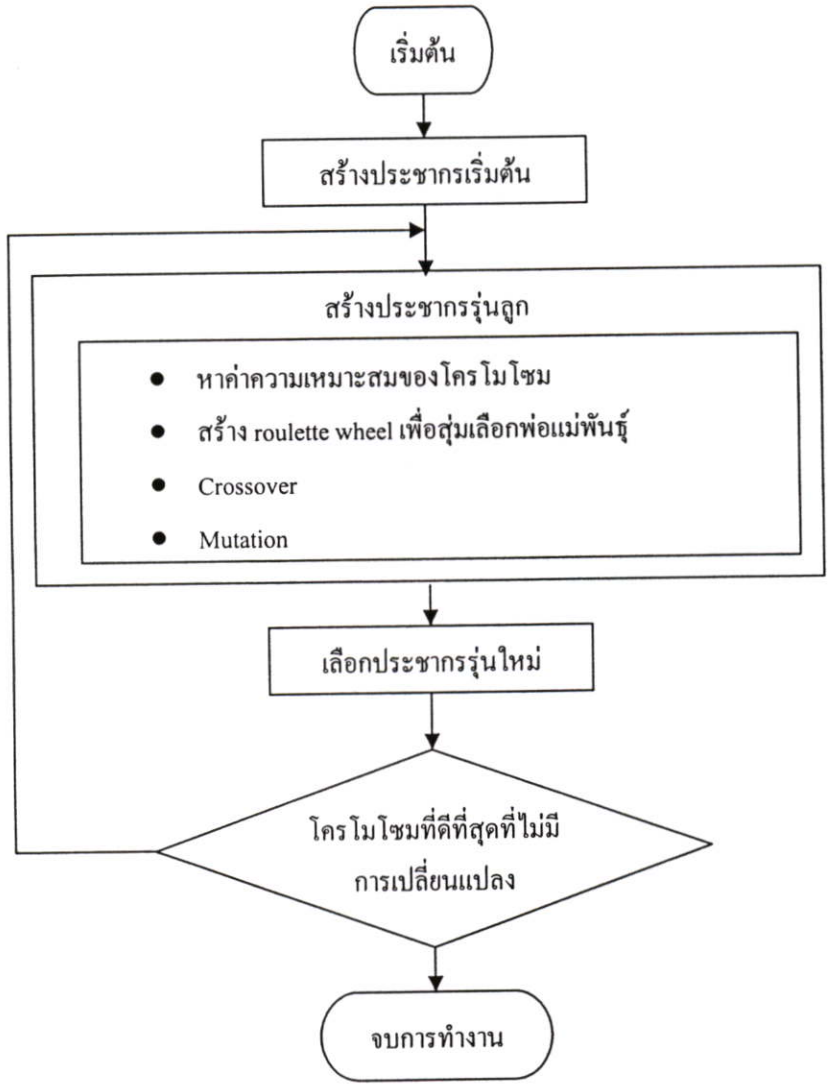
ขั้นตอนที่ 1 : สร้างประชากรเริ่มต้นตามจำนวนประชากร (population) ที่กำหนด

ขั้นตอนที่ 2 : สร้างโครโมโซมรุ่นลูกโดย

1. คำนวณค่าความเหมาะสมของโครโมโซม ตามฟังก์ชั่นความเหมาะสม (fitness function) ที่กำหนดไว้
2. สร้าง roulette wheel เพื่อสุ่มเลือกประชากรพ่อ-แม่
3. สุ่มค่าเพื่อดำเนินการครอสโอเวอร์ หากค่าที่สุ่มมีค่าน้อยกว่าค่าความน่าจะเป็นของการการครอสโอเวอร์ (μ_c) จะทำการครอสโอเวอร์ โดยสุ่มตำแหน่งในโครโมโซมที่ต้องการครอสโอเวอร์ แล้วทำการสลับตำแหน่งยีนของโครโมโซมพ่อและแม่ ในขั้นตอนนี้ หากมีการครอสโอเวอร์เกิดขึ้น จะได้ประชาการลูกจำนวน 2 ประชากร
4. สุ่มค่าเพื่อดำเนินการมิวเตชั่น หากค่าที่สุ่มมีค่าน้อยกว่าค่าความน่าจะเป็นของมิวเตชั่น (μ_m) ทำการมิวเตชั่น โดยสุ่มตำแหน่งยีนในโครโมโซมที่ต้องการมิวเตท แล้วทำการปรับเปลี่ยนค่ายีนดังกล่าว

ขั้นตอนที่ 3 : คัดเลือกประชากรรุ่นใหม่ โดยนำโครโมโซมพ่อ-แม่ และลูกมาคำนวณค่าความเหมาะสมของแต่ละโครโมโซม เลือกประชากรที่ให้ค่าความเหมาะสมสูงสุดเท่ากับจำนวนประชากร เพื่อเป็นประชากรรุ่น (generation) ใหม่

ขั้นตอนที่ 4 : ทำซ้ำขั้นตอนที่ 2 จนกว่าจะได้ประชากรที่ให้ค่าความเหมาะสมสูงสุด ที่ไม่เปลี่ยนแปลง



รูปที่ 2.7 แผนภาพแสดงการทำงานของจีเนติกอัลกอริทึม

จีเนติกอัลกอริทึม ได้ถูกนำไปใช้ในงานต่างๆอย่างหลากหลาย เช่น การออกแบบ VLSI (VLSI design) การรู้จำรูปแบบ (pattern recognition) การประมวลผลภาพ (image processing) นิวรอลเน็ตเวิร์ก (neural network) และ การเรียนรู้ของเครื่องจักร (machine learning) เป็นต้น

2.7 งานวิจัยที่เกี่ยวข้อง

2.7.1 Genetic algorithm-based clustering technique [1]

Ujjwal Maulik และ Sanghamitra Bandyopadhyay ได้เสนอเทคนิคการจัดกลุ่มข้อมูลด้วยจีเนติกอัลกอริทึม (GA-Clustering) โดยโครโมโซมแสดงถึงสายอักขระตัวเลขจำนวนจริงซึ่งเป็นจุดศูนย์กลางของกลุ่ม และการทำงานของอัลกอริทึมมีพื้นฐานมาจากอัลกอริทึม k-means

จากอัลกอริทึม k-means พบว่าในแต่ละรุ่นของประชากรจากอัลกอริทึม k-means จะถูกนำมาปรับค่าศูนย์กลางใหม่ทุกครั้ง และจุดศูนย์กลางจะถูกแทนที่ด้วยจุดศูนย์กลางใหม่ จากการดำเนินการเช่นนี้แสดงให้เห็นถึงความละโมบของ อัลกอริทึม k-means ซึ่งอาจทำให้เกิดการหยุดทำงานก่อนเวลาอันควร เนื่องจากอัลกอริทึมจะหยุดทำงานเมื่อพบพบโครโมโซมที่เหมาะสมที่สุดในขณะนั้น ซึ่งค่าดังกล่าวอาจไม่ใช่คำตอบที่ดีที่สุดของปัญหา อาจเป็นเพียงคำตอบที่ดีที่สุดที่เกิดจากชุดศูนย์กลางเริ่มต้นชุดหนึ่งที่อัลกอริทึมได้สุ่มเลือกในตอนเริ่มต้น ดังนั้นเพื่อหลีกเลี่ยงปัญหานี้จีเนติกอัลกอริทึมถูกนำมาใช้ร่วมกันกับอัลกอริทึม k-means

ขั้นตอนการทำงานของ GA-Clustering อัลกอริทึมเป็นดังนี้

ขั้นตอนที่ 1 : สร้างประชากรเริ่มต้นตามจำนวนประชากรที่กำหนด โดยในแต่ละประชากรมีโครโมโซมซึ่งแสดงถึงสายอักขระตัวเลขจำนวนจริงอันเป็นจุดศูนย์กลางของกลุ่ม

เลือกศูนย์กลางของแต่ละกลุ่ม $z_1, z_1, ..., z_k$ จำนวน k กลุ่ม จากสมาชิก n สมาชิก $\{x_1, x_2, ..., x_n\}$ โดยวิธีการสุ่ม

จัดข้อมูล $x_i, i = 1,2,...,n$ ให้แต่ละกลุ่ม $C_j, j \in \{1,2,...,k\}$ ถ้า

$$\|x_i - z_j\| < \|x_i - z_p\|, p = 1,2,...,k \text{ และ } j \neq p \tag{2.11}$$

เมื่อ x_i คือ ข้อมูลสมาชิกของกลุ่มที่ C_j

z_j คือ ศูนย์กลางของกลุ่มที่ C_j

z_p คือ ศูนย์กลางของกลุ่มที่ C_p

ขั้นตอนที่ 2 : คำนวณค่าศูนย์กลางใหม่ $z_1^*, z_2^*, ..., z_k^*$

$$z_i^* = \frac{1}{n_i} \sum_{x_j \in C_i} x_j \text{ , } i = 1,2,\ldots,\text{k} \tag{2.12}$$

เมื่อ x_j คือ ข้อมูลสมาชิกของกลุ่มที่ C_i

n_i คือ จำนวนสมาชิกในกลุ่มที่ C_i

z_i คือ ศูนย์กลางของกลุ่มที่ C_i

คำนวณค่าความเหมาะสมของโครโมโซม โดยการพิจารณาระยะห่างระหว่างสมาชิกกับศูนย์กลางกลุ่ม

คำนวณหาค่า clustering metric M สำหรับข้อมูลจำนวน k กลุ่ม $C_1, C_2, ..., C_k$ จากสมการ

$$M(C_1, C_2, ..., C_k) = \sum_{i=1}^{k} \sum_{x_j \in C_i} \| x_j - z_i \| \tag{2.13}$$

คำนวณหาค่าความเหมาะสมของชุดศูนย์กลาง

$$f = \frac{1}{M} \tag{2.14}$$

ขั้นตอนที่ 3 : สร้าง roulette wheel เพื่อสุ่มเลือกพ่อ-แม่พันธุ์

ขั้นตอนที่ 4 : สุ่มค่าเพื่อดำเนินการครอสโอเวอร์ หากค่าที่สุ่มมีค่าน้อยกว่าค่าความน่าจะเป็นของการครอสโอเวอร์ (μ_c) ทำการครอสโอเวอร์ โดยสุ่มตำแหน่งในโครโมโซมที่ต้องการครอสโอเวอร์ แล้วทำการสลับตำแหน่งยีนของโครโมโซมพ่อและแม่

ขั้นตอนที่ 5 : สุ่มค่าเพื่อดำเนินการมิวเตชั่น หากค่าที่สุ่มมีค่าน้อยกว่าค่าความน่าจะเป็นของมิวเตชั่น (μ_m) ทำการมิวเตชั่น โดยสุ่มตำแหน่งยีนในโครโมโซมที่ต้องการมิวเตท แล้วทำการปรับเปลี่ยนค่ายีนดังกล่าว โดยมีเงื่อนไขในการปรับเปลี่ยนค่าเป็นดังสมการ (2.15) และ (2.16)

$$v \pm 2 * \delta * v, \quad v \neq 0 \tag{2.15}$$

$$v \pm 2 * \delta, \quad v = 0 \tag{2.16}$$

เมื่อ δ เป็นค่าที่เกิดจากการสุ่ม โดยกำหนดให้มีค่าอยู่ระหว่าง [0,1]

v คือค่าข้อมูล ณ จุดที่ต้องการมิวเตท

การเลือกค่า '+' หรือ '-' ขึ้นอยู่กับการสุ่มเลือก

ขั้นตอนที่ 6 : คำนวณค่าศูนย์กลางใหม่ของกลุ่ม ดังสมการที่ (2.12) และ คำนวณค่าความเหมาะสมของประชากรรุ่นลูก ดังสมการที่ (2.13) และ (2.14)

ขั้นตอนที่ 7 : คัดเลือกประชากรรุ่นใหม่โดยพิจารณาโครโมโซมพ่อ-แม่ และลูก เลือกประชากรที่ให้ค่าความเหมาะสมสูงสุดเท่ากับจำนวนประชากร เพื่อเป็นประชากรรุ่นใหม่

ขั้นตอนที่ 8 : ทำซ้ำขั้นตอนที่ 3 จนกว่าจะได้ประชากรที่ให้ค่าความเหมาะสมสูงสุด ที่ไม่เปลี่ยนแปลง

2.7.2 An evolutionary technique based on k-means algorithm for optimal clustering in R^N [5]

Sanghamitra Bandyopadhyay และ Ujjwal Maulik ได้เสนอเทคนิคการจัดกลุ่มข้อมูลด้วยจีเนติกอัลกอริทึมโดยมีพื้นฐานจาก อัลกอริทึม k-means เรียกอัลกอริทึมดังกล่าวนี้ว่า KGA-clustering

ขั้นตอนการทำงานของ KGA-clustering อัลกอริทึมแสดงดังรูปที่ 2.8

ในกระบวนการครอสโอเวอร์ งานวิจัยนี้เลือกใช้วิธีการครอสโอเวอร์แบบจุดเดี่ยว (single-point crossover) โดยมีการกำหนดค่าความน่าจะเป็นของการครอสโอเวอร์เท่ากับ μ_c จากโครโมโซมที่มีความยาว l ($l = nk$) จะทำการสุ่มเลขจำนวนเต็ม 1 ค่าซึ่งมีค่าอยู่ระหว่าง [1, l-1] เป็นตำแหน่งของจุดครอสโอเวอร์

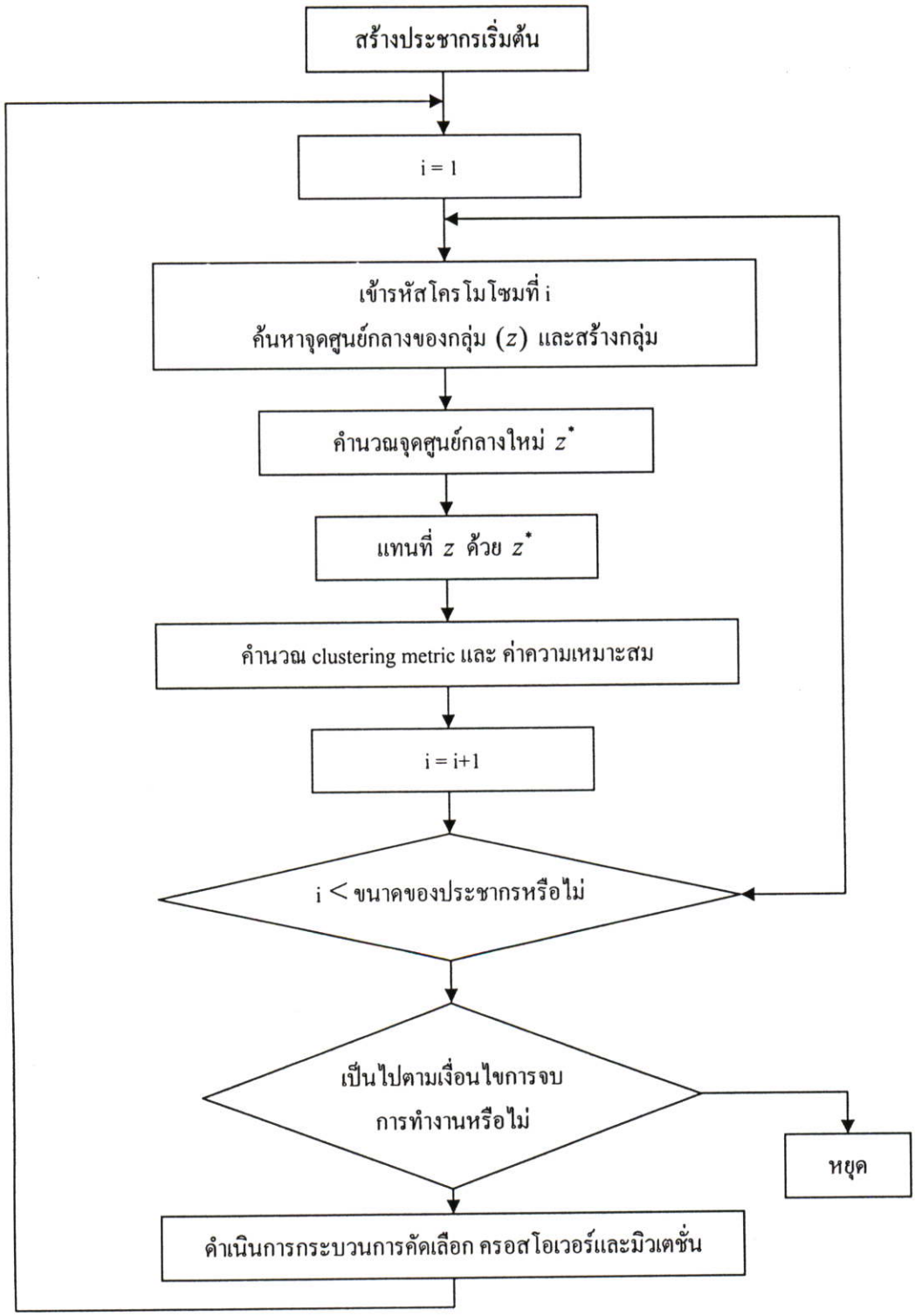
ในกระบวนการมิวเตชั่นแต่ละโครโมโซมจะมีการมิวเตชั่นตามความน่าจะเป็น μ_m กำหนดให้ $M_{\min}$ และ $M_{\max}$ คือค่าต่ำสุดและสูงสุดของ clustering metric การมิวเตทโครโมโซมที่มีค่า clustering metric เท่ากับ M และค่า δ อยู่ในช่วง $[-R, \ +R]$ โดยค่า R คำนวณได้จากสมการ

$$R = \begin{cases} \dfrac{M - M_{\min}}{M_{\max} - M_{\min}} & \text{ถ้า} \quad M_{\max} > M \\ 1 & \text{ถ้า} \quad M_{\min} > M_{\max} \end{cases} \tag{2.17}$$

ถ้าค่าต่ำสุดของชุดข้อมูลในมิติที่ i $(i = 1,2,...,n)$ มีค่าเป็น $x^i_{\min}$ และ $x^i_{\max}$ ตามลำดับ และตำแหน่งที่ต้องการมิวเตทมีค่าเป็น x^i หลังจากมิวเตทจะได้ค่าดังสมการ

$$x^i + \delta \times \left(x^i_{\max} - x^i\right) \text{ ถ้า } \delta \geq 0 \tag{2.18}$$

$$x^i + \delta \times \left(x^i - x^i_{\min}\right) \text{ ถ้า } \delta < 0 \tag{2.19}$$



รูปที่ 2.8 ลำดับขั้นตอนการทำงานของอัลกอริทึม KGA-clustering

2.7.3 A Hybrid Algorithm for k-medoid Clustering of Large Data Sets [3]

Weiguo Sheng และ Xiaohui Liu ได้เสนอเทคนิค local search heuristic และนำเทคนิคดังกล่าวมาผสานกับจีเนติกอัลกอริทึมและ k-medoid clustering สำหรับการจัดกลุ่มชุดข้อมูลขนาดใหญ่ ซึ่งเป็นการแก้ปัญหาแบบ NP-hard optimization โดย local search heuristic จะทำการเลือก medoid จำนวน k medoid จากชุดข้อมูลและพยายามปรับค่าความแตกต่างของแต่ละกลุ่มให้มีผลรวมความแตกต่างกันภายในกลุ่มน้อยที่สุด การจัดกลุ่มข้อมูลด้วย k-medoid algorithm แบบใหม่นี้เรียกว่า Hybrid k-medoid Algorithm (HkA)

ขั้นตอนการทำงานของ local search heuristic เป็นดังนี้

ขั้นตอนที่ 1 : กำหนดจำนวนกลุ่ม k และ nearest neighbor p

ขั้นตอนที่ 2 : สุ่มเลือก medoid จำนวน k medoid จากชุดข้อมูล X = $\{x_1, x_2, ..., x_n\}$

ขั้นตอนที่ 3 : จัดข้อมูล $x_i, i = 1,2,...,n$ ให้แต่ละกลุ่ม $C_j, j \in \{1,2,...,k\}$ โดยจัดให้อยู่ใกล้ medoid ที่อยู่ใกล้ที่สุดตามค่า Euclidean Distance

คำนวณค่าผลรวมของ Euclidean Distance (SED)

$$\text{SED} = \sum_{i=1}^{n} \sum_{j=1, x_i \in C_j}^{k} d(x_i, m_j) \tag{2.20}$$

เมื่อ m_j คือ medoid ของกลุ่ม C_j $(j = 1,2,...,k)$

ขั้นตอนที่ 4 : ปรับปรุงค่า k-medoid โดย

(a) ในกลุ่ม C_j เลือกสับเซต (subset) C_{subset} ที่สัมพันธ์กับ m_j และ p nearest neighbors

(b) คำนวณค่า medoid ใหม่ ถ้าค่า medoid ถูกเปลี่ยน โดยการแทนที่ q

$$q = \arg\min_{x_k \in C_{subset}} \sum_{x_i \in C_j} d(x_k, x_i) \tag{2.21}$$

(c) ดำเนินการซ้ำ (a) และ (b) จนกว่า medoid ไม่เปลี่ยนแปลง

ขั้นตอนที่ 5 : ดำเนินการซ้ำขั้นตอนที่ 3 และ 4 จนกว่า medoid ทั้ง k medoid ไม่เปลี่ยนแปลง

แม้ว่าการทำงานของ local search k-medoid clustering heuristic นี้จะสามารถทำงานได้อย่างมีประสิทธิภาพ อย่างไรก็ตามการจัดกลุ่มข้อมูลอาจไม่อาจรับประกันได้ว่าจะสามารถค้นพบการจัดกลุ่มที่ดีได้ทุกครั้ง ดังนั้นเพื่อให้ได้การจัดกลุ่มข้อมูลที่ดียิ่งขึ้น จึงได้นำเทคนิค local search heuristic มาทำงานร่วมกันกับจีเนติกอัลกอริทึม

ขั้นตอนการทำงานของ Hybrid k-medoid อัลกอริทึม เป็นดังนี้

ขั้นตอนที่ 1 : สุ่มเลือกประชากรจำนวน p

ขั้นตอนที่ 2 : คำนวณค่า SED ตามสมการ (2.18) และคำนวณค่าความเหมาะสมของแต่ละประชากร

$$f = \frac{1}{SED} \tag{2.22}$$

ขั้นตอนที่ 3 : ดำเนินการซ้ำ (a) ถึง (e) จนกว่าจะพบคำตอบ

(a) เลือกประชากรพ่อ-แม่ จำนวน *p/2* เพื่อทำการสืบพันธุ์

(b) ครอสโอเวอร์และมิวเตทประชากรพ่อ-แม่ที่ได้คัดเลือกไว้

(c) ดำเนินการ local search heuristic กับประชากรรุ่นลูกที่ได้จากการครอสโอเวอร์

(d) คำนวณค่า SED ตามสมการ (2.20) และคำนวณค่าความเหมาะสมของแต่ละประชากรในประชากรรุ่นลูก ตามสมการ (2.22)

(e) สร้างประชากรรุ่นใหม่ โดยเลือกประชากรที่ให้ค่าความเหมาะสมสูงสุดจำนวน *p* ประชากร

ขั้นตอนที่ 4 : เลือกประชากรที่ให้ค่าความเหมาะสมสูงสุดเป็นลำดับ

2.7.4 Genetic clustering for automatic evolution of cluster and application to image classification [2]

Sanghamitra Bandyopadhyay และ Ujjwal Mualik นำเสนอวิธีการจำแนกข้อมูลรูปภาพด้วยจีเนติกอัลกอริทึม โดยอัลกอริทึมดังกล่าวสามารถจัดกลุ่มของข้อมูลได้โดยไม่กำหนดจำนวนกลุ่มของข้อมูล (k) แต่เป็นการกำหนดจำนวนกลุ่ม k ให้อยู่ในช่วง $[k_{mim}, k_{max}]$ โดยโครโมโซมแสดงสายอักขระตัวเลขจำนวนจริงหรือสัญลักษณ์ # และใช้ Davies-Bouldin index สำหรับวัดความเที่ยงตรงของกลุ่มข้อมูล

โครโมโซมแสดงสายอักขระของตัวเลขจำนวนจริงและสัญลักษณ์ # ซึ่งเป็นสัญลักษณ์ที่ถูกกำหนดขึ้นเพื่อใช้แทนยีนที่ว่างในโครโมโซม โดยสัญลักษณ์ดังกล่าวจะไม่มีผลต่อการคำนวณเมื่อจัดกลุ่มเสร็จแล้วใช้ Davies – Bouldin index วัดค่าความเที่ยงตรงของการจัดกลุ่ม ในการทดลองเป็นการทดสอบอัลกอริทึมกับข้อมูลที่สังเคราะห์ขึ้นและข้อมูลจริง โดยเทคนิคที่นำเสนอสามารถแยกแยะความแตกต่างของประเภทของพื้นที่ในรูปภาพได้

ใน GCUK – Clustering โครโมโซมจะถูกสร้างจากจำนวนจริง ซึ่งเป็นศูนย์กลางของกลุ่ม ค่าของ k ถูกกำหนดให้อยู่ในช่วง $[k_{mim}, k_{max}]$ เมื่อ k_{mim} เท่ากับ 2 และ k_{max} คือความยาวของสายอักขระ การสร้างโครโมโซมใน GCUK – Clustering เริ่มต้นจากสุ่มเลือกศูนย์กลางของกลุ่มจำนวน k กลุ่ม จากนั้นสุ่มเลือกตำแหน่งยีนที่แต่ละศูนย์กลางจะถูกนำไปบรรจุ เมื่อบรรจุศูนย์กลางของกลุ่มครบแล้วแทนที่ยีนที่เหลือด้วย #

ตัวอย่างเช่น กำหนดให้ k_{mim} = 2 และ k_{max} = 10 สุ่มเลือกค่า k ได้ค่า k เท่ากับ 4 ดังนั้นโครโมโซมจะถูกสร้างขึ้นจาก ศูนย์กลางจำนวน 4 ศูนย์กลาง สมมติว่าศูนย์กลางที่สุ่มมามีค่าดังนี้ (10.0, 5.0) (20.4, 13.2) (15.8, 2.9) (22.7, 17.7)

จากนั้นสุ่มเลือกตำแหน่งของยีนที่จะนำศูนย์กลางแต่ละศูนย์กลางไปบรรจุ หากสุ่มเลือกตำแหน่งยีนได้เป็น 7 2 5 และ 8 สำหรับศูนย์กลางที่ (10.0, 5.0) (20.4, 13.2) (15.8, 2.9) และ (22.7, 17.7) ตามลำดับ

ดังนั้น (10.0, 5.0) จะถูกบรรจุลงในยีนตำแหน่งที่ 7 (20.4, 13.2) จะถูกบรรจุลงในยีนตำแหน่งที่ 2 (15.8, 2.9) จะถูกบรรจุลงในยีนตำแหน่งที่ 5 และ (22.7, 17.7) จะถูกบรรจุลงในยีนตำแหน่งที่ 8 ตามลำดับ ดังรูปที่ 2.9

#	(20.4, 13.2)	#	#	(15.8, 2.9)	#	(10.0, 5.0)	(22.7, 17.7)	#	#

รูปที่ 2.9 สายอักขระของโครโมโซมที่ถูกเข้ารหัสด้วย GCUK Clustering

ค่าความเหมาะสมของโครโมโซม พิจารณา Davies – Bouldin index โดยค่าชี้วัดนี้แสดงถึงอัตราส่วนของ ความสัมพันธ์ภายในกลุ่ม และความสัมพันธ์ระหว่างกลุ่มโดยความสัมพันธ์ภายในกลุ่ม C คำนวณได้จากสมการ (2.23)

$$S_{i,q} = \left(\frac{1}{|c|} \sum_{x \in c_i} \left\{ \| x - z_i \|_2^q \right\} \right)^{1/q} \tag{2.23}$$

เมื่อ z_i เป็นจุดศูนย์กลางของ C_i และ $z_i = \frac{1}{n_i} \sum_{x \in c_i} x$ และ n เป็นลำดับของ C_i

ระยะห่างระหว่างจุดศูนย์กลาง C_i และ C_i ถูกกำหนดโดย

$$d_{ij,t} = \left\{ \sum_{s-1}^{p} \| z_{is} - z_{js} \|^t \right\}^{\frac{1}{t}} = \| z_i - z_j \|_t \tag{2.24}$$

$$R_{i,} \, qt = \max_{j, j \neq i} \left\{ \frac{S_{i,q} + S_{j,q}}{d_{ij,t}} \right\} \tag{2.25}$$

$$DB = \frac{1}{K} \sum_{j=1}^{k} R_{i,qt} \tag{2.26}$$

จุดมุ่งหมายของกระบวนการทำงาน คือ ค้นหาโครโมโซมที่มีค่า DB น้อยที่สุด เพื่อหาจำนวนกลุ่มที่เหมาะสม ดังนั้นฟังก์ชั่นความเหมาะสมสำหรับ โครโมโซม j คือ $\frac{1}{DB_j}$ เมื่อ DB_j คือ Davies – Bouldin index ที่ถูกคำนวณสำหรับ โครโมโซม j (ค่าความเหมาะสมที่สูงที่สุด แสดงถึง ค่า DB index ที่ต่ำที่สุด)

การดำเนินการทางจีเนติกอัลกอริทึมประกอบด้วยสามส่วนได้แก่การคัดเลือกประชากร การครอสโอเวอร์ และการมิวเตชั่น แต่ละการดำเนินการมีขั้นตอนการทำงานดังนี้

การคัดเลือกประชากร ประชากรจะถูกเลือก ตามค่าความเหมาะสม

ในกระบวนการครอสโอเวอร์ ค่าที่ถูกพิจารณาคือค่าของแต่ละยีน โดยในงานวิจัยนี้พิจารณาการครอสโอเวอร์แบบจุดเดียว (single point crossover) และค่าความน่าจะเป็น μ_c ดังตัวอย่างแสดง

(20.4, 13.2) # # (15.8, 2.9) | # (10.0, 5.0) (22.7, 17.7) #
(13.2, 15.6) # # # (5.3, 13.7) | # (10.5, 16.2) (7.9, 15.3) # (18.3, 14.5)

เมื่อครอสโอเวอร์ตำแหน่งที่ 5 จะได้โครโมโซมใหม่ 2 โครโมโซม ดังนี้

(20.4, 13.2) # # (15.8, 2.9) # (10.5, 16.2) (7.9, 15.3) # (18.3, 14.5)
(13.2, 15.6) # # # (5.3, 13.7) # (10.0, 5.0) (22.7, 17.7) # #

ในการดำเนินการมิวเตชั่น แต่ละตำแหน่งในโครโมโซมที่มีค่าข้อมูล (ซึ่งไม่ใช่ #) จะถูกมิวเตท ด้วยค่าความน่าจะเป็น μ_c ตามวิธีการดังนี้ กำหนดค่า δ เป็นค่าตัวเลขที่อยู่ระหว่าง [0,1] ค่า v คือค่า ณ ตำแหน่งที่ต้องการมิวเตท ดังนั้น v หลังจากมิวเตท จะกลายเป็น

$$vx(1 \pm 2\delta),\ v \neq 6 \tag{2.27}$$

$$\pm 2\delta \quad ,\ v = 0 \tag{2.28}$$

สัญลักษณ์ + หรือ - ขึ้นอยู่กับการสุ่ม

การทำงานจะสิ้นสุด เมื่อโครโมโซมที่ให้ค่าความเหมาะสมสูงสุด คือ โครโมโซมเดิมที่ให้ค่าความเหมาะสมสูงสุดจากทุกประชากรรุ่นล่าสุด

บทที่ 3
การนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม

การทำงานของจีเนติกอัลกอริทึมประกอบไปด้วยขั้นตอนสำคัญในการเปลี่ยนแปลงค่าของข้อมูลได้แก่ การครอสโอเวอร์ (crossover) และการมิวเตชั่น (mutation) โดยการครอสโอเวอร์คือการไขว้กันของยีนในโครโมโซม โดยยีนจากคู่ของพ่อและแม่จะถูกสลับตำแหน่งกัน การมิวเตชั่นคือการกลายพันธุ์ของโครโมโซม โดยยีนในโครโมโซมจะถูกเปลี่ยนแปลงค่าไปจากค่าเดิม

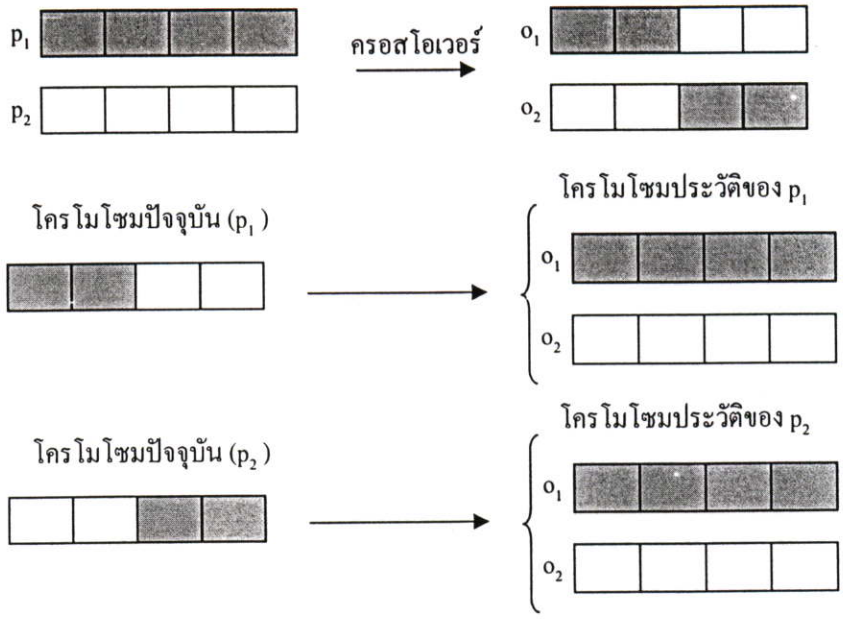
งานวิจัยครั้งนี้จึงได้เล็งเห็นถึงความสำคัญของกระบวนการกลายพันธุ์ของโครโมโซม เนื่องจาก มิวเตท (mutate) เป็นการสุ่มค่าที่ต้องการเปลี่ยนแปลง ซึ่งวิธีการดังกล่าวเป็นการสุ่มแบบไม่มีทิศทาง ดังนั้นงานวิจัยครั้งนี้จึงมีการเก็บข้อมูลประวัติของโครโมโซมแต่ละชุด เพื่อช่วยในการปรับทิศทางการเปลี่ยนแปลงของค่ายีน โดย ยีนปัจจุบันจะปรับเข้าหายีนในโครโมโซมในอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มมาก และปรับออกจากยีนในโครโมโซมอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มน้อย ซึ่งวิธีการเช่นนี้ ทำให้การจัดกลุ่มมีความเหมาะสมและรวดเร็วยิ่งขึ้น

ในบทที่ 3 นี้ จะกล่าวถึงวิธีการในการจัดกลุ่มแนวทางใหม่โดยใช้จีเนติกอัลกอริทึม และวิธีการพิจารณาการมิวเตทยีนในโครโมโซมที่ได้นำเสนอ

3.1 การเก็บประวัติของโครโมโซม

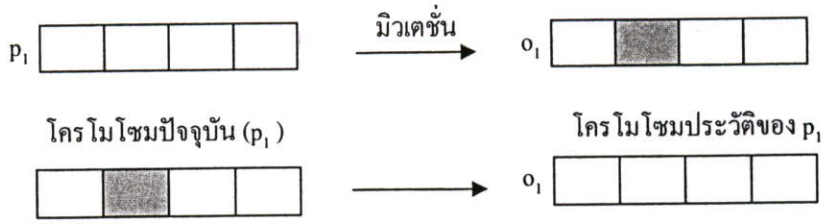
ในงานวิจัยนี้ได้นำข้อมูลประวัติของโครโมโซมมาพิจาณาการปรับเปลี่ยนยีนในโครโมโซม ดังนั้นทุกครั้งที่โครโมโซมถูกเปลี่ยนแปลงจากกระบวนการดำเนินการทางพันธุกรรม ซึ่งได้แก่กระบวนการครอสโอเวอร์ และกระบวนการวิมเตชั่น โครโมโซมเก่าจะถูกเก็บในรายการโครโมโซมประวัติ

การเก็บประวัติของโครโมโซมหลังจากโครโมโซมเกิดการครอสโอเวอร์ โครโมโซมพ่อ-แม่ทั้งสองโครโมโซม จะถูกเก็บเป็นโครโมโซมประวัติของโครโมโซมลูกทั้งสองโครโมโซม เช่น จากรูปที่ 3.1 โครโมโซม p_1 และ p_2 เป็นโครโซมพ่อ-แม่ และ โครโมโซม o_1 และ o_2 คือโครโมโซมลูกที่เกิดจากการครอสโอเวอร์ ดังนั้นโครโมโซม p_1 และ p_2 จะถูกเก็บเป็นโครโมโซมประวัติของทั้งโครโมโซม o_1 และ o_2



รูปที่ 3.1 การเก็บโครโมโซมประวัติจากกระบวนการครอสโอเวอร์

การเก็บประวัติของโครโมโซมหลังจากโครโมโซมเกิดการมิวเตชั่น โครโมโซมพ่อ-แม่ จะถูกเก็บเป็นโครโมโซมประวัติของโครโมโซมลูก เช่น จากรูปที่... โครโมโซม p_1 เป็นโครโซมพ่อ-แม่ และ o_1 คือโครโมโซมลูกที่เกิดจากการมิวเตชั่น ดังนั้น โครโมโซม p_1 จะถูกเก็บเป็นโครโมโซมประวัติของโครโมโซม o_1



รูปที่ 3.2 การเก็บโครโมโซมประวัติจากกระบวนการมิวเตชั่น

3.2 การจัดกลุ่มแนวทางใหม่โดยใช้จีเนติกอัลกอริทึม

3.2.1 การแสดงค่าของสตริง (String representation)

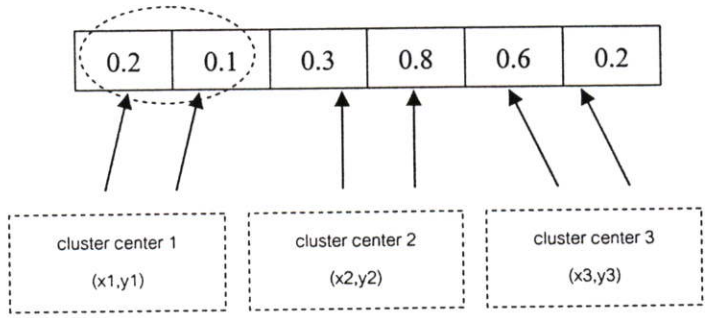
ในธรรมชาติโครโมโซมประกอบด้วยยีนลักษณะต่างๆจำนวนมาก ยีนเหล่านั้นเป็นตัวแสดงลักษณะทางพันธุกรรมของสิ่งมีชีวิต ยีนเป็นสิ่งที่บ่งบอกลักษณะที่เหมาะสมและความอยู่รอดในธรรมชาติ ยีนที่ดี มีความแข็งแรงจะสามารถดำรงชีวิตอยู่ในธรรมชาติได้ การสร้าง

โครโมโซมในจีเนติกอัลกอริทึมได้เลียนแบบการแสดงค่าของสตริงหรือสายอักขระจากโครโมโซมในธรรมชาติ โดยโครโมโซมในจีเนติกอัลกอริทึม เกิดจากการเข้ารหัสข้อมูล

โครโมโซมที่ใช้สำหรับการจัดกลุ่มด้วยจีเนติกอัลกอริทึมเกิดจากการเข้ารหัสข้อมูลโดยในแต่ละยีนแสดงแทนศูนย์กลางของกลุ่ม และการสร้างโครโมโซมของข้อมูลขนาด n มิติ ที่ต้องการจัดกลุ่มจำนวน k กลุ่ม จะได้โครโมโซมที่มีจำนวนยีนเท่ากับ k ยีน ความยาวของยีนมีขนาดเท่ากับ n

ในวิทยานิพนธ์นี้นำเสนอการจัดกลุ่มข้อมูลโดยข้อมูลที่ใช้ในการจัดกลุ่มเป็นข้อมูลตัวเลขจำนวนจริง ดังนั้นโครโมโซมจึงเกิดจาการเข้ารหัสข้อมูลตัวเลขจำนวนจริง

รูปที่ 3.3 แสดงตัวอย่างของโครโมโซม 1 โครโมโซม ซึ่งประกอบด้วย n = 2 และ k = 3 นั่นคือใน 1 โครโมโซมประกอบด้วยข้อมูลจำนวน 2 มิติ มีจำนวนกลุ่มเท่ากับ 3 โครโมโซมแสดงค่าศูนย์กลาง 3 กลุ่มคือ (0.2,0.1) (0.3,0.8) และ (0.6,0.2)



รูปที่ 3.3 แสดงตัวอย่างของโครโมโซม 1 โครโมโซม

3.2.2 การสร้างประชากร (Population initialization)

ประชากร (population) คือชุดของโครโมโซมที่สร้างขึ้น เพื่อจะถูกคัดเลือกเป็นพ่อหรือแม่ให้กับประชากรในรุ่นต่อไป

รูปที่ 3.4 แสดงตัวอย่างของเจนเนอเรชั่น (generation) ซึ่งประกอบด้วยประชากรจำนวน 5 ประชากร (p = 5) แต่ละประชากรประกอบด้วยข้อมูลขนาด 2 มิติ (n = 2) และมีจำนวนกลุ่ม (k) เท่ากับ 3

0.2	0.1	0.3	0.8	0.6	0.2
0.3	0.2	0.4	0.1	0.9	0.8
0.1	0.1	0.3	0.6	0.6	0.9
0.1	0.6	0.3	0.1	0.5	0.7
0.2	0.2	0.4	0.6	0.9	0.2

รูปที่ 3.4 เจนเนอเรชั่น (generation) ของประชากรจำนวน 5 ประชากร

3.2.3 การคำนวณค่าความเหมาะสม (Fitness function)

ก่อนการคำนวณหาค่าความเหมาะสมของแต่ละโครโมโซมจะต้องทำการจัดข้อมูลให้เป็นสมาชิกของกลุ่ม ซึ่งการจัดสมาชิกให้อยู่ในกลุ่มใดๆ นั้นสามารถทำได้โดยสมาชิกที่อยู่ใกล้กับศูนย์กลางของกลุ่มใดมากที่สุด จะถูกนำไปรวมกับกลุ่มนั้น

โดยมีขั้นตอนในการกำหนดแต่ละสมาชิกให้กับกลุ่มดังนี้

1. เลือกศูนย์กลางของแต่ละกลุ่ม $z_1, z_1, ..., z_k$ จำนวน k กลุ่ม จากสมาชิก m สมาชิก $\{x_1, x_2, ..., x_m\}$

2. จัดข้อมูล $x_i, i = 1,2,...,m$ ให้แต่ละกลุ่ม $C_j, j \in \{1,2,...,k\}$ ถ้า

$$\|x_i - z_j\| < \|x_i - z_p\|, p = 1,2,...,k \text{ และ } j \neq p \tag{3.1}$$

เมื่อ x_i คือ ข้อมูลสมาชิกของกลุ่มที่ C_j

z_j คือ ศูนย์กลางของกลุ่มที่ C_j

z_p คือ ศูนย์กลางของกลุ่มที่ C_p

คำนวณหาค่า clustering metric M สำหรับข้อมูลจำนวน k กลุ่ม $C_1, C_2, ..., C_k$ จากสมการ

$$M(C_1, C_2, ..., C_k) = \sum_{i=1}^{k} \sum_{x_j \in C_i} \| x_j - z_i \| \tag{3.2}$$

จากคุณสมบัติการจัดกลุ่มข้อมูล การจัดกลุ่มที่ดีประกอบด้วยคุณสมบัติ 2 ข้อดังนี้

1. สมาชิกที่อยู่ในกลุ่มเดียวกันจะมีความคล้ายคลึงกันมากที่สุด

2. สมาชิกที่อยู่ต่างกลุ่มกันจะมีความแตกต่างกันมากที่สุด

การคำนวณค่าความเหมาะสมของโครโมโซมพิจารณาจากระยะห่างระหว่างข้อมูลและศูนย์กลางกลุ่ม ดังนั้นการคำนวณหาค่าความเหมาะสมของโครโมโซมคำนวณได้จากสมการ (3.3)

$$f = \frac{1}{M} \tag{3.3}$$

โครโมโซมที่มีค่า clustering metric M ต่ำสุดนั่นคือโครโมโซมที่มีความเหมาะสมที่จะเป็นคำตอบของปัญหามากที่สุด

3.2.4 การคัดเลือกประชากร (Selection)

ในธรรมชาติการคัดเลือกประชากรเพื่อสืบพันธ์จะถูกคัดเลือกโดยพิจารณาค่าความเหมาะสมของแต่ละประชากร เช่นกันกับการคัดเลือกประชากรในจีเนติกอัลกอริทึม โครโมโซมหรือประชากรที่มีค่าความเหมาะสมมากก็มีความน่าจะเป็นที่จะถูกเลือกเพื่อเป็นประชากรต้นแบบต่อไป

วิธีการที่นิยมใช้คัดเลือกประชากรต้นแบบในกระบวนการจีเนติกอัลกอริทึม คือการใช้เทคนิคการหมุนวงล้อถ่วงน้ำหนัก (roulette wheel) โดยเทคนิคดังกล่าวจะแบ่งช่องว่างแต่ละช่องว่างในวงล้อให้มีขนาดเท่ากับความน่าจะเป็นที่โครโมโซมนั้นจะถูกเลือก ซึ่งคำนวณได้จากสมการ (3.4) จากนั้นหมุนวงล้อเพื่อเลือกโครโมโซม

$$P_i = \frac{f_i}{f_1 + f_2 + \ldots + f_n} \tag{3.4}$$

เมื่อ P_i คือค่าความน่าจะเป็นที่โครโมโซม i จะถูกเลือก

f_i คือค่าความเหมาะสมของโครโมโซม i

$f_1 + f_2 + \ldots + f_n$ คือผลรวมของค่าความเหมาะสมของทุกๆโครโมโซมในรุ่นประชากร

3.2.5 การครอสโอเวอร์ (Crossover)

โครโมโซมพ่อ-แม่ จะถูกครอสโอเวอร์เมื่อทำการสุ่มค่าหนึ่งค่าขึ้นมาแล้วพบว่า ค่าดังกล่าวมีค่าน้อยกว่าค่าความน่าจะเป็นในการครอสโอเวอร์ (μ_c)

ทำการสลับค่าตำแหน่งยีนของโครโมโซมพ่อและแม่ ตามตำแหน่งที่สุ่มเลือก โดยตำแหน่งที่สามารถทำการครอสโอเวอร์ได้ มีค่าอยู่ระหว่าง 1 ถึง ค่าความยาวของยีนในโครโมโซม (l)

3.2.6 การมิวเตชั่น (Mutation)

งานวิจัยนี้เสนอส่วนขยายของ อัลกอริทึมการจัดกลุ่มด้วยจีเนติก (Original GA-clustering algonithms) ซึ่งนำเสนอโดย Maulik และ Bandyopadhyay ด้วยการเสนอการนำประวัติของโครโมโซมมาใช้ในการจัดกลุ่มด้วยจีเนติกอัลกอริทึม ในกระบวนการมิวเตชั่นประวัติของโครโมโซมจะถูกแบ่งออกเป็น 2 กลุ่มได้แก่ "โครโมโซมกลุ่มดี" และ "โครโมโซมกลุ่มไม่ดี"

การกำหนดโครโมโซมกลุ่มดีหรือโครโมโซมกลุ่มไม่ดี พิจารณาจากค่าความเหมาะสม โดยถ้าโครโมโซมในอดีตมีค่าความเหมาะสมมากกว่าค่าความเหมาะสมของโครโมโซมปัจจุบัน โครโมโซมอดีตนั้นจะถูกจัดอยู่ในกลุ่มของ "โครโมโซมกลุ่มดี" แต่ถ้าโครโมโซมในอดีตมีค่าความเหมาะสมน้อยกว่าค่าความเหมาะสมของโครโมโซมปัจจุบัน โครโมโซมอดีตนั้นจะถูกจัดอยู่ในกลุ่มของ "โครโมโซมกลุ่มไม่ดี"

C	0.1	0.2	0.3	0.4
h_1	0.2	0.1	0.3	0.6
h_2	0.2	0.2	0.1	0.2
h_3	0.1	0.2	0.3	0.8
	0.4	0.9	0.6	0.1

รูปที่ 3.5 ตัวอย่างโครโมโซม C และโครโมโซมประวัติของโครโมโซม C

จากรูปที่ 3.5 แสดงตัวอย่างโครโมโซมปัจจุบัน คือ โครโมโซม C มีค่าความเหมาะสมเท่ากับ 0.8 และโครโมโซม C ประวัติดังโครโมโซม h_1, h_2, h_3, h_4 แต่ละโครโมโซมประวัติมีค่าความเหมาะสมเป็น 0.7, 0.5, 0.9 และ 0.85 ตามลำดับ

ดังนั้นสามารถแบ่งโครโมโซมในอดีตของ C ได้ดังรูปที่ 3.6

h_1	0.2	0.1	0.3	0.6
h_2	0.2	0.2	0.1	0.2

(a)

h_3	0.1	0.2	0.3	0.8
h_4	0.4	0.9	0.6	0.1

(b)

รูปที่ 3.6 แสดงการแบ่งประวัติของโครโมโซม C ออกเป็นโครโมโซมกลุ่มไม่ดี (a) และกลุ่มดี (b)

โครโมโซม h_1 และ h_2 มีค่าความเหมาะสมน้อยกว่าโครโมโซม C ดังนั้นจึงจัดให้ h_1 และ h_2 เป็นโครโมโซมในกลุ่มไม่ดี ส่วนโครโมโซม h_3 และ h_4 มีค่าความเหมาะสมมากกว่าโครโมโซม C ดังนั้นจึงจัดให้ h_3 และ h_4 เป็นโครโมโซมในกลุ่มดี

สมการสำหรับคำนวณค่ายีนที่ดำเนินการมิวเตชั่น ประกอบด้วย 2 คอมโพเนนท์ ซึ่งได้แก่ ส่วนของคอมโพเนนท์ที่เกิดขึ้นจากการสุ่ม (random component) และคอมโพเนนท์คงที่ (deterministic component)

$$z^{t+1} = z^t + \Delta z^t \tag{3.5}$$

$$\Delta z^t = \text{random component} + \text{deterministic component} \tag{3.6}$$

$$\Delta z^t = \left[\alpha \delta z^t\right] + \left[\beta_1 \sum_{i \in g} w_i \left(z_i - z^t\right) - \beta_2 \sum_{k \in b} w_k \left(z_k - z^t\right)\right] \tag{3.7}$$

$$w_i = \frac{f_i}{\sum_{j \in g} f_j} \tag{3.8}$$

$$w_k = \frac{f_k}{\sum_{j \in b} f_j} \tag{3.9}$$

เมื่อ z^t คือ ค่าจุดศูนย์กลางปัจจุบันที่ถูกคัดเลือก

$\alpha = 0.1 - \beta_1 - \beta_2$ ค่า α คือ ค่าสัมประสิทธิ์น้ำหนักสำหรับคอมโพเนนท์แบบสุ่ม

δ คือ ค่าตัวเลขที่อยู่ระหว่าง -1 และ 1

β_1 และ β_2 คือ ค่าสัมประสิทธิ์น้ำหนักสำหรับ "โครโมโซมกลุ่มดี" และ "โครโมโซมกลุ่มไม่ดี" ตามลำดับ ในงานวิจัยนี้ กำหนดค่าเริ่มต้นของทั้ง β_1 และ β_2 ไว้ที่ 0.05 อย่างไรก็ตาม เมื่อโครโมโซมที่ให้ค่าความเหมาะสมสูงสุดในชุดประชากรปัจจุบัน เป็นโครโมโซมเดียวกับโครโมโซมที่ให้ค่าความเหมาะสมสูงสุดในประชากรก่อนหน้านี้ ค่าของ β_1 และ β_2 จะถูกลดลงตามค่าที่กำหนดไว้ในเบื้องต้น หากประชากรยังคงไม่มีการเปลี่ยนแปลงอีก ค่า β_1 และ β_2 ก็จะถูกลดลงเรื่อยๆ

z_i และ z_k คือ ค่าจุดศูนย์กลางในอดีตของจุดศูนย์กลางในปัจจุบัน

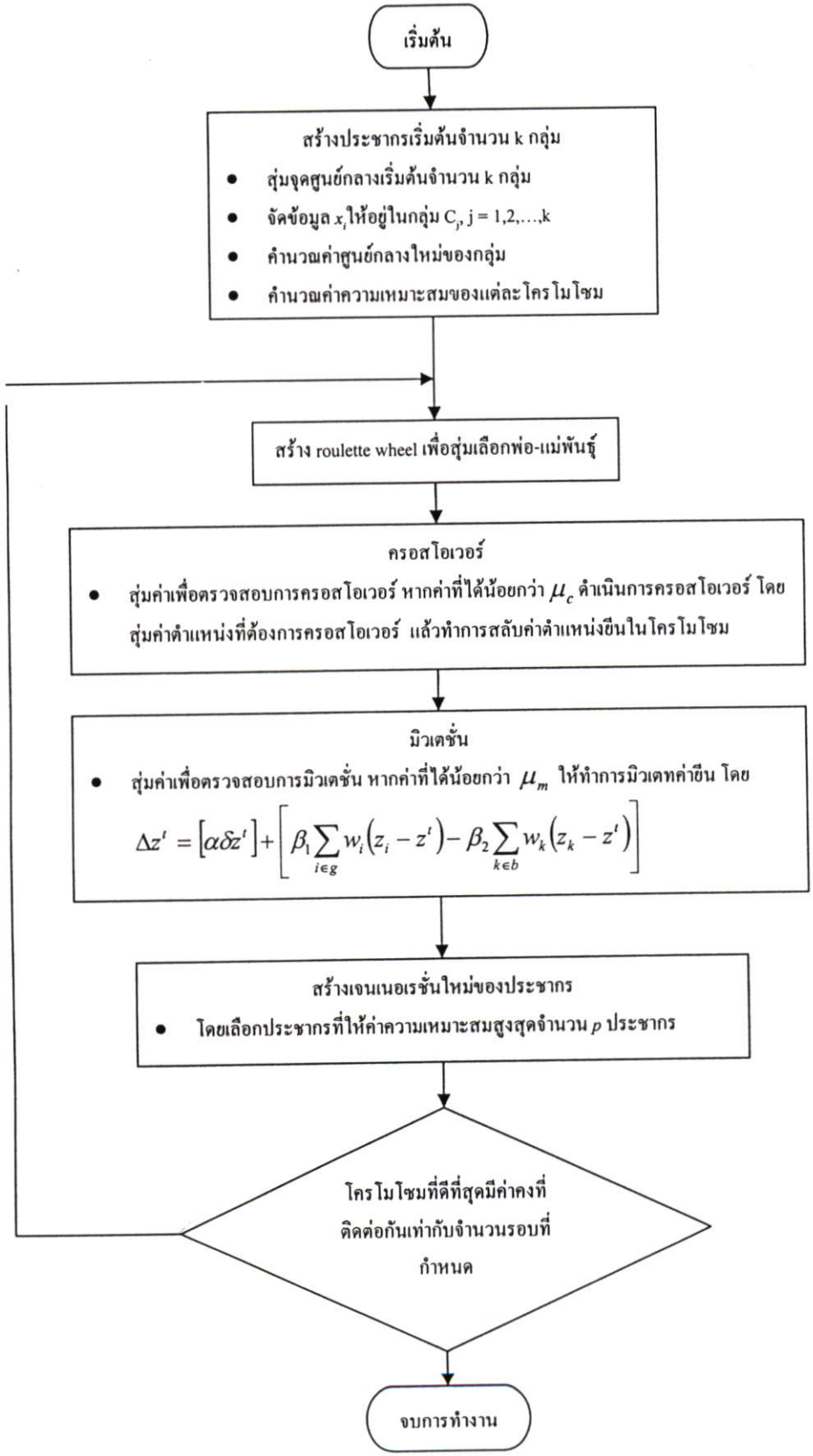
g แสดงแทน " โครโมโซมกลุ่มดี "

b แสดงแทน "โครโมโซมกลุ่มไม่ดี "

3.2.7 การสิ้นสุดกระบวนการ (Termination criterion)

กระบวนการจัดกลุ่ม จะสิ้นสุดลงเมื่อได้โครโมโซมในรุ่นสุดท้ายที่ดีที่สุด ที่ให้จุดศูนย์กลางของกลุ่มไม่เปลี่ยนแปลงติดต่อกันเท่ากับจำนวนรอบที่กำหนด

ขั้นตอนการทำงานทั้งหมดของการจัดกลุ่มด้วยจีเนติกอัลกอริทึมที่นำเสนอ สามารถนำมาเขียนเป็นแผนภูมิแสดงการทำงานตามขั้นตอนต่างๆได้ดังรูปที่ 3.7



รูปที่ 3.7 ขั้นตอนการทำงานของอัลกอริทึมที่นำเสนอ

3.3 ตัวอย่างการจัดกลุ่มด้วยจีเนติกอัลกอริทึม

กำหนดให้ข้อมูลชุด A คือชุดข้อมูลที่ต้องการจัดกลุ่มประกอบด้วยข้อมูลดังนี้ (3,0), (1,5), (1,2), (9,8), (6,3), (4,6), (8,1), (4,4), (5,8) และ (10,4) กำหนดจำนวนกลุ่ม k = 2 จำนวนประชากร p = 4

สุ่มเลือกศูนย์กลางของกลุ่มเท่ากับจำนวนกลุ่มที่กำหนดให้ คือ 2 และสร้างประชากรเท่ากับจำนวนประชากรที่กำหนดไว้คือ 4 ดังนั้นสร้างประชากรได้ดังรูปที่ 3.8

2	1	8	4
10	6	5	7
2	2	9	7
1	4	5	7

รูปที่ 3.8 แสดงประชากรเริ่มต้นของชุดข้อมูล A

ตารางที่ 3.1 แสดงค่า Euclidean distance ของทุกข้อมูลจากชุดข้อมูล Aกับจุดศูนย์กลาง (2,1) และ (8,4)

x	y	(2,1)	(8,4)
3	0	1.41	6.40
1	5	4.12	7.07
1	2	1.41	7.28
9	8	9.90	4.12
6	3	4.47	2.24
4	6	5.39	4.47
8	1	6.00	3.00
4	4	3.61	4.00
5	8	7.62	5.00
10	4	8.54	2.00

จัดกลุ่มข้อมูลให้กับศูนย์กลางแต่ละศูนย์กลาง ตารางที่ 3.1 แสดงผลการคำนวณค่า Euclidean distance ระหว่างข้อมูลทุกข้อมูลและศูนย์กลาง (2,1) และ (8,4) จากนั้นพิจารณาว่าข้อมูลแต่ละข้อมูล อยู่ใกล้จุดศูนย์กลางใดมากที่สุด จัดให้เป็นสมาชิกของจุดศูนย์กลางนั้น

หาค่าเฉลี่ยของทุกข้อมูลในแต่ละศูนย์กลาง จะได้จุดศูนย์กลางใหม่ของชุดข้อมูล เป็น (2.25,2.75) และ (7,4.5) ดำเนินการเช่นนี้กับทุกโครโมโซม จะได้โครโมโซมชุดใหม่ดังรูปที่ 3.9

2.25	2.75	7	4.5
9	4.33	3.43	4
8	6.67	6	3
7	5	2.25	2.75

รูปที่ 3.9 แสดงโครโมโซมชุดใหม่หลังจากปรับค่าจุดศูนย์กลาง

คำนวณค่า Clustering metric M ของทุกๆโครโมโซม ตามสมการ (3.2)

ตารางที่ 3.2 แสดงการหาค่า Clustering metric M ของชุดข้อมูล A เมื่อโครโมโซมที่มีจุดศูนย์กลางเป็น (2.25,2.75) และ(7,4.5)

x	y	(2.25,2.75)	(7,4.5)
3	0	2.85	6.02
1	5	2.57	6.02
1	2	1.46	6.50
9	8	8.55	4.03
6	3	3.76	1.80
4	6	3.69	3.35
8	1	6.01	3.64
4	4	2.15	3.04
5	8	5.93	4.03
10	4	7.85	3.04

จัดสมาชิกเข้ากลุ่มโดยข้อมูลอยู่ใกล้ศูนย์กลางใดมากที่สุด จะถูกจัดให้เป็นสมาชิกของศูนย์กลางนั้น จากตารางที่ 3.2 จะได้ว่าศูนย์กลาง (2.25,2.75) มีสมาชิก ได้แก่ (3,0) (1,5) (1,2) และ (4,4) และศูนย์กลาง (7,4.5) มีสมาชิกได้แก่ (9,8) (6,3) (4,6) (8,1) (5,8) และ (10,4)

คำนวณค่า Clustering metric M โดยหาค่าผลรวมของระยะห่างของสมาชิกทุกตัวไปยังศูนย์กลางกลุ่มของสมาชิกนั้นๆ จากตารางที่ 3.2 จะได้ค่า Clustering metric M ดังนี้

Clustering metric M $= 2.85 + 2.57 + 1.46 + 4.03 + 1.80 + 3.35 + 3.64 + 2.15 + 4.03 + 3.04$

$= 28.93$

คำนวณค่าความเหมาะสมจากค่า Clustering metric M

ค่าความเหมาะสม $= 1/28.93$

$= 0.034562$

คำนวณค่าความเหมาะสมของทุกๆโครโมโซม จะได้ค่าความเหมาะสมของแต่ละโครโมโซมดังรูปที่ 3.10

2.25	2.75	7	4.5	0.034562
9	4.33	3.43	4	0.036098
8	6.67	6	3	0.031567
7	5	2.25	2.75	0.034569

รูปที่ 3.10 แสดงโครโมโซมและค่าความเหมาะสมของแต่ละโครโมโซม

เลือกประชากรพ่อ-แม่ด้วยวงล้อถ่วงน้ำหนักและค่าความเหมาะสมของโครโมโซม โดยสร้างวงล้อถ่วงน้ำหนักสำหรับสุ่มเลือกประชากร เพื่อเข้าสู่กระบวนการครอสโอเวอร์และมิวเตชั่นในลำดับต่อไป

จากค่าความเหมาะสมของแต่ละโครโมโซมหาค่าความน่าจะเป็นที่โครโมโซมนั้นจะถูกเลือก(P_i) ของแต่ละโครโมโซมจากสมการ (3.4)

$$P_1 = \frac{0.034562}{0.034562 + 0.036098 + 0.031567 + 0.034569}$$

$= 0.2527$

$$P_2 = \frac{0.036098}{0.034562 + 0.036098 + 0.031567 + 0.034569}$$

$= 0.2639$

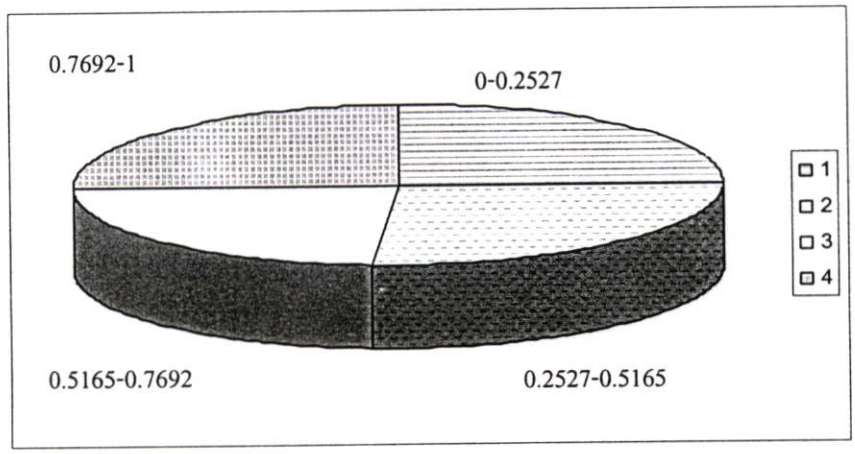
$$P_3 = \frac{0.031567}{0.034562 + 0.036098 + 0.031567 + 0.034569}$$

$$= 0.2308$$

$$P_4 = \frac{0.034569}{0.034562 + 0.036098 + 0.031567 + 0.034569}$$

$$= 0.2527$$

จากค่าความน่าจะเป็นที่โครโมโซมจะถูกเลือกสร้างเป็นวงล้อถ่วงน้ำหนักได้ดังรูปที่ 3.11

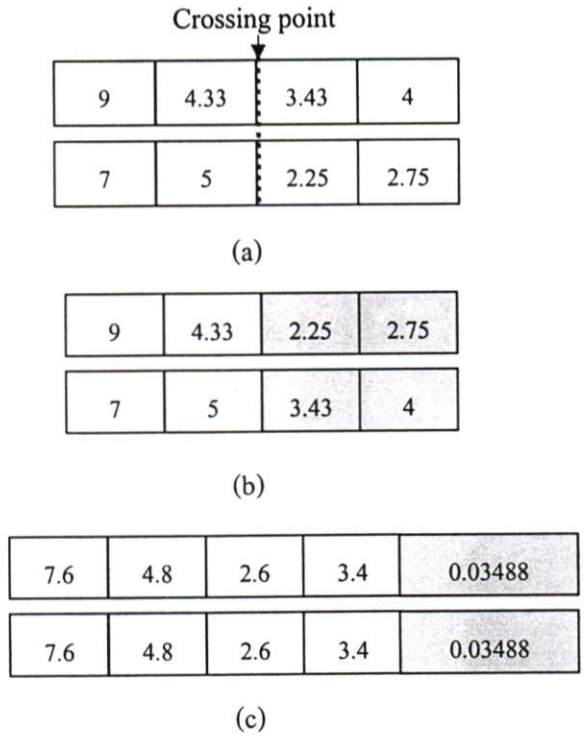


รูปที่ 3.11 แสดงการสร้างวงล้อถ่วงน้ำหนักจากชุดประชากร

จากการสุ่มเลือกประชากร ได้ประชากรพ่อ-แม่ดังรูปที่ 3.12

9	4.33	3.43	4	0.036098
7	5	2.25	2.75	0.034569

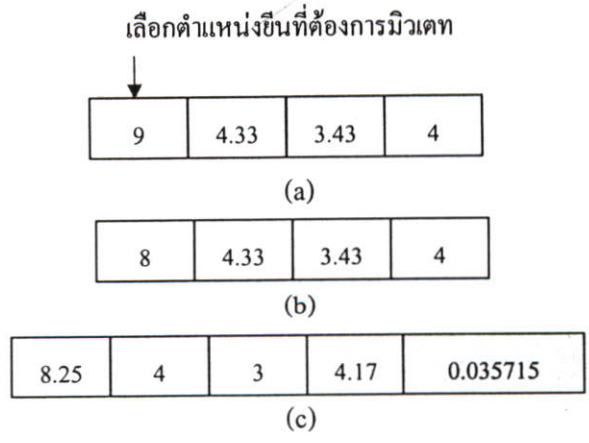
รูปที่ 3.12 ประชากรพ่อ-แม่ที่ถูกเลือก



รูปที่ 3.13 แสดงขั้นตอนการครอสโอเวอร์

รูปที่ 3.13 แสดงขั้นตอนการครอสโอเวอร์เริ่มต้นจากสุ่มเลือกตำแหน่งที่ต้องการครอสโอเวอร์ (a) แล้วทำการครอสโอเวอร์ โดยสลับตำแหน่งยีนระหว่างยีนโครโมโซมพ่อและโครโมโซมแม่ (b) จากนั้นทำการปรับค่าศูนย์กลางใหม่ โดยหาค่าเฉลี่ยของทุกสมาชิกในกลุ่ม (c)

รูปที่ 3.14 แสดงขั้นตอนการมิวเตชั่น หลังจากสุ่มเลือกตำแหน่งยีนที่ต้องการมิวเตทได้ค่าตำแหน่งเป็น 1 (a) จากนั้นสุ่มค่าการเปลี่ยนแปลง สมมติว่าสุ่มค่าการเปลี่ยนแปลงยีนเท่ากับ -1 ดังนั้นค่ายีนใหม่ที่ได้จะมีค่าเท่ากับ 8 (b) คำนวณจุดศูนย์กลางใหม่และค่าความเหมาะสมของกลุ่ม (c)



รูปที่ 3.14 แสดงขั้นตอนการมิวเตชั่น

หลังจากดำเนินการครอสโอเวอร์และมิวเตชั่นจะได้ประชากรลูกเพิ่มมาอีก 3 ประชากรนำประชากรลูกที่ได้ไปรวมกับประชากรรุ่นพ่อ-แม่ และเรียงลำดับประชากรตามค่าความเหมาะสมดังรูปที่ 3.15 จากนั้นคัดเลือกประชากรที่มีค่าความเหมาะสมสูงสุด จำนวน p ประชากร

9	4.33	3.43	4	0.036098
8.25	4	3	4.17	0.035715
7.6	4.8	2.6	3.4	0.03488
7	5	2.25	2.75	0.034569
2.25	2.75	7	5	0.034569
2.25	2.75	7	4.5	0.034562
8	6.67	6	3	0.031567

(a)

9	4.33	3.43	4	0.036098
8.25	4	3	4.17	0.035715
7.6	4.8	2.6	3.4	0.03488
7	5	2.25	2.75	0.034569

(b)

รูปที่ 3.15 แสดงการคัดเลือกประชากร

เปรียบเทียบโครโมโซมที่ให้ค่าความเหมาะสมสูงสุดในประชากรรุ่นใหม่กับโครโมโซมที่ให้ความเหมาะสมสูงสุดในประชากรรุ่นเก่า ว่าเป็นโครโมโซมเดียวกันหรือไม่ หากใช่แสดงว่าได้โครโมโซมที่มีความเหมาะสม หรือ ได้คำตอบของปัญหา ดังนั้นถือว่าสิ้นสุดการทำงาน

9	4.33	3.43	4	0.036098
7	5	2.25	2.75	0.034569
2.25	2.75	7	4.5	0.034562
8	6.67	6	3	0.031567

(a)

9	4.33	3.43	4	0.036098
8.25	4	3	4.17	0.035715
7.6	4.8	2.6	3.4	0.03488
7	5	2.25	2.75	0.034569

(b)

รูปที่ 3.16 แสดงการเปรียบเทียบ ประชากรชุดใหม่ (a) และประชากรชุดเก่า (b)

3.4 ตัวอย่างการนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม

วิทยานิพนธ์นี้เสนอวิธีการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม โดยการพัฒนากระบวนการปรับปรุงยีนแบบใหม่ เพื่อใช้แทนกระบวนการมิวเตชั่นแบบเดิม

พิจารณาประวัติโครโมโซมในอดีต โดยแบ่งโครโมโซมออกเป็น 2 กลุ่ม ตามค่าความเหมาะสมของประชากร

ตัวอย่าง ต้องการปรับปรุงโครโมโซม A โดยโครโมโซมอดีตของโครโมโซม A ได้แก่ โครโมโซม B, C, D และ E ดังแสดงในรูปที่ 3.17

A	5	0	7	5	0.15
B	6	1	7	3	0.12
C	10	2	3.5	4	0.13
D	2.25	4	8	3	0.14
E	4	5	2.25	2.75	0.16

รูปที่ 3.17 โครโมโซม A และโครโมโซมประวัติของโครโมโซม A

สุ่มตำแหน่งของยีนที่ต้องการปรับปรุง ได้เป็น 1 ดังนั้นสามารถแบ่งโครโมโซมออกเป็นสองกลุ่มดังนี้ โครโมโซมกลุ่มดีได้แก่ โครโมโซม E ส่วนโครโมโซมกลุ่มไม่ดีได้แก่ โครโมโซม B,C และ D

การปรับปรุงค่ายีนในแต่ละโครโมโซมเป็นไปตามสมการ (3.7)

ตารางที่ 3.3 แสดงผลการคำนวณค่าน้ำหนักของโครโมโซมในอดีตของโครโมโซม A

z	f	w	$z-z^t$	$w_g(z-z^t)$	$w_b(z-z^t)$
6	0.12	0.31	-1		-0.31
10	0.13	0.34	-5		-1.7
2.25	0.14	0.36	2.75		0.99
4	0.16	1	1	1	
				1	**-1.02**

จากตารางที่ 3.3 จะได้ค่า f_{good} และ f_{bad} เท่ากับ 0.39 และ 0.16 ตามลำดับ

กำหนดให้ δ , α , β_1 และ β_2 มีค่าเป็น 0.5, 0.02, 0.04 และ 0.04 ตามลำดับ

$$\Delta z^t = [0.02*0.5*5] + [(0.04*1) - (0.04*(-1.02))]$$
$$= 0.05 + [0.04-0.0408]$$
$$= 0.0492$$

$$z^{t+1} = 5+ 0.0492$$
$$= 5. 0492$$

ดังนั้น ค่ายีนเดิมคือ 5 จะถูกแทนที่ด้วยค่ายีนใหม่คือ 5.0492

บทที่ 4
การทดสอบการนำประวัติของโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึม

4.1 ข้อมูลทดสอบ

ข้อมูลที่ใช้ในการทดสอบอัลกอริทึมประกอบไปด้วยชุดข้อมูลมาตรฐานจำนวน 10 ชุดข้อมูล ได้แก่

4.1.1 Iris data set ชุดข้อมูลดอกไอริส [6] เป็นชุดข้อมูลที่รวบรวมข้อมูลดอกไอริสจำนวน 150 ข้อมูล แต่ละข้อมูลประกอบด้วย 4 แอททริบิว ได้แก่ ความกว้างและความยาวของกลีบเลี้ยงและความกว้างและความยาวของกลีบดอก โดยดอกไอริสที่นำมาทดสอบ สำรวจมาจากดอกไอริส 3 สปีชีส์ ได้แก่ setosa, versicolor และ virginica สปีชีส์ละ 50 ข้อมูล ดังนั้น จำนวน k ของชุดข้อมูลนี้มีค่าเท่ากับ 3

(a) (b) (c)

รูปที่ 4.1 ภาพตัวอย่างดอกไอริส สปีชีส์ setosa, versicolor และ virginica ตามลำดับ

Name	exp1	exp2	exp3	exp4
Setosa	5.1	3.5	1.4	0.2
Versicolor	5.5	2.3	4.0	1.3
Virginica	6.7	3.0	5.2	2.3

รูปที่ 4.2 ตัวอย่างข้อมูลจาก iris data set

4.1.2 Indian Telugu Vowel data set ชุดข้อมูลรู้จำเสียงสระ [7] โดย Deterding เป็นผู้รวบรวมบันทึกเสียงสระภาษาอังกฤษ ทั้งหมด 11 ระดับเสียง ซึ่งรวบรวมจากผู้ให้การบันทึกเสียงทั้งหมด 15 คน ทำการบันทึกเสียงแต่ละสระจำนวน 6 ครั้ง ข้อมูลทั้งหมดมีจำนวน 990 ข้อมูล ดังนั้นค่า k ของชุดข้อมูลนี้มีค่าเท่ากับ 11

4.1.3 Pima Indians diabetes data set ชุดข้อมูลคนไข้โรคเบาหวาน ชนเผ่าอินเดียแดง จากคลังข้อมูล UCI machine learning [6] เป็นชุดข้อมูลสำหรับทำนายผลตรวจสอบคนไข้ว่าเป็นโรคเบาหวานหรือไม่ โดยการอ้างอิงข้อมูลจากองค์กรอนามัยโลก ข้อมูลประกอบด้วย ข้อมูลของคนไข้เพศหญิง อายุไม่ต่ำกว่า 21 ปี จากชนเผ่าอินเดียนแดง ที่อาศัยอยู่ใกล้ เมืองโฟนิก มลรัฐอาริโซน่า ประเทศสหรัฐอเมริกา ข้อมูลทั้งหมดจำนวน 768 ข้อมูล แต่ละข้อมูลประกอบด้วยแอททริบิว 8 แอททริบิว ข้อมูลมีทั้งหมด 2 กลุ่ม โดยกลุ่มคนไข้ที่เป็นโรคเบาหวานมีทั้งหมด 268 คน และไม่เป็นโรคเบาหวานจำนวน 500 คน ดังนั้นค่า k ของชุดข้อมูลนี้มีค่าเท่ากับ 2

4.1.4 Heart Statlog data set ชุดข้อมูลคนไข้โรคหัวใจจากฐานข้อมูลโครงการ Statlog [8] เป็นชุดข้อมูลสำหรับทำนายคนไข้ว่าเป็นโรคหัวใจหรือไม่ โดยพิจารณาจากข้อมูลผลการตรวจสอบทางการแพทย์ จำนวน 13 แอททริบิว ข้อมูลทั้งหมด 270 ข้อมูลประกอบด้วยคนไข้ที่เป็นโรคหัวใจจำนวน 120 คน และคนไข้ที่ไม่เป็นโรคหัวใจจำนวน 150 คน ดังนั้นค่า k ของชุดข้อมูลนี้มีค่าเท่ากับ 2

4.1.5 Sonar data set ชุดข้อมูลระบบกระบวนการหาตำแหน่งของวัตถุใต้น้ำโดยการส่งคลื่นเสียงและรับเสียงสะท้อน [6] โดย แบ่งแยกระหว่าง แร่ หรือ ก้อนหิน ชุดข้อมูลกลุ่มนี้ประกอบด้วยข้อมูลจำนวน 111 ข้อมูลที่ทดสอบได้ว่าเป็นแร่ และ 97 ข้อมูลที่ทดสอบได้ว่าเป็นก้อนหิน แต่ละข้อมูลประกอบด้วย แอททริบิว จำนวน 60 แอททริบิว มีค่าระหว่าง 0.0 ถึง 1.0 กลุ่มของข้อมูลนี้คือ 2

4.1.6 Image Segmentation data set ชุดข้อมูลรูปภาพจากฐานข้อมูลโครงการ Statlog [8] จำนวน 2,310 ข้อมูล แต่ละข้อมูลประกอบด้วยแอททริบิวจำนวน 19 แอททริบิวโดยเป็นข้อมูลรูปภาพนอกสถานที่จำนวน 7 ภาพ ได้แก่ ภาพพื้นผิวอิฐ ภาพท้องฟ้า ภาพใบไม้ ภาพพื้นผิวซีเมนต์ ภาพหน้าต่าง ภาพพื้นผิวคอนกรีต ภาพทางเดิน ภาพทุ่งหญ้า ดังนั้นจำนวนกลุ่มของชุดข้อมูลชุดนี้คือ 7

4.1.7 Ionosphere data set ข้อมูลชั้นบรรยากาศของโลกชั้นไอโอโนสเฟียร์ (ชั้นบรรยากาศซึ่งห่างจากผิวโลกระหว่าง 80-1000 กิโลเมตร) [6] จำนวน 351 ข้อมูล แต่ละข้อมูลประกอบด้วยแอททริบิวจำนวน 34 แอททริบิว โดยแบ่งข้อมูลออกเป็น 2 กลุ่ม ได้แก่ กลุ่มที่สภาพบรรยากาศดี และกลุ่มที่สภาพบรรยากาศไม่ดี กลุ่มละ 123 และ 24 ข้อมูลตามลำดับ ดังนั้นจำนวนกลุ่มของชุดข้อมูลชุดนี้คือ 2

4.1.8 Breast cancer data set ข้อมูลคนไข้มะเร็งเต้านม จากคลังข้อมูล UCI machine learning [6] (Wisconsin breast cancer) เป็นชุดข้อมูลสำหรับตรวจสอบว่าคนไข้ป่วยเป็นโรคมะเร็งเต้านมในระดับใด จากข้อมูลทั้งหมด 9 แอททริบิว 699 ข้อมูล แบ่งข้อมูลเป็น 2 กลุ่มได้แก่กลุ่มที่เป็นเนื้องอกในเต้านมจำนวน 458 คน และกลุ่มที่เป็นโรคมะเร็งเต้านมจำนวน 241 คน ดังนั้นจำนวนกลุ่มของชุดข้อมูลชุดนี้คือ 2

4.1.9 Satellite image data set ข้อมูลภาพถ่ายจากดาวเทียมพื้นผิวโลก [8] โดย แบ่งลักษณะของภาพถ่ายออกเป็น 6 ประเภทตามลักษณะของพื้นผิวโลก และข้อมูลประกอบด้วยแอททริบิวทั้งหมด 36 แอททริบิว ดังนั้นจำนวนกลุ่มของชุดข้อมูลชุดนี้คือ 6

4.1.10 Letter recognition data set ข้อมูลภาพตัวอักษร[6] ในภาษาอังกฤษจำนวน 26 ตัวอักษร (จาก A ถึง Z) จำนวน 20,000 ภาพ โดยข้อมูลประกอบด้วยแอททริบิวทั้งหมด 17 แอททริบิว ดังนั้นจำนวนกลุ่มของชุดข้อมูลชุดนี้คือ 26

4.2 การกำหนดค่าเริ่มต้น

ในการทดสอบการจัดกลุ่มด้วยจีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอดำเนินการด้วยค่าพารามิเตอร์ดังนี้

ขนาดของจำนวนประชากร $p = 5$

ความน่าจะเป็นการการครอสโอเวอร์ $\mu_c = 0.8$

ความน่าจะเป็นในการมิวเตชั่น $\mu_m = 1$

เนื่องจากงานวิจัยนี้ได้มุ่งเน้นการพัฒนาการดำเนินการมิวเตชั่น ดังนั้นจึงกำหนดให้ดำเนินการมิวเตชั่นกับทุกโครโมโซมที่ถูกเลือกให้เป็นโครโมโซมพ่อ-แม่

การปรับปรุงค่ายีนในแต่ละโครโมโซมเป็นไปตามสมการ (4.1)

$$\Delta z^t = \left[\alpha \delta z^t\right] + \left[\beta_1 \sum_{i \in g} w_i \left(z_i - z^t\right) - \beta_2 \sum_{k \in b} w_k \left(z_k - z^t\right)\right] \quad (4.1)$$

ค่า α คือ ค่าสัมประสิทธิ์น้ำหนักสำหรับคอมโพเนนท์แบบสุ่ม กำหนดให้มีค่าเท่ากับ 0

ค่า δ คือ ค่าตัวเลขที่อยู่ระหว่าง -1 และ 1

ค่า β_1 และ β_2 คือ ค่าสัมประสิทธิ์น้ำหนักสำหรับ "โครโมโซมกลุ่มดี" และ "โครโมโซมกลุ่มไม่ดี" ตามลำดับ กำหนดให้มีค่าเป็น 0.05 และ 0.05 ตามลำดับ

ค่า β_1 และ β_2 จะลดลงครั้งละ 0.005 หากค่าศูนย์กลางที่ดีที่สุดไม่เปลี่ยนแปลงอัลกอริทึมจะหยุดทำงานเมื่อค่า β_1 และ β_2 ลดลงต่อเนื่องกันเป็นจำนวน 20 รอบ ถ้าค่า β_1 และ β_2 ลดค่าลงยังไม่ถึง 20 รอบแต่ค่าศูนย์กลางที่ดีที่สุดถูกเปลี่ยนค่าแล้วค่า β_1 และ β_2 จะถูกตั้งค่าให้มีค่าเท่ากับค่าเริ่มต้น นั่นคือ ค่า β_1 และ β_2 มีค่า 0.05 และ 0.05 ตามลำดับ

4.3 ผลการทดสอบอัลกอริทึม

ผลการทดสอบเปรียบเทียบการจัดกลุ่มด้วยอัลกอริทึม k-means การจัดกลุ่มด้วยจีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ โดยมีข้อมูลทดสอบ 10 ชุดได้แก่ iris data set, Indian Telugu vowel data set, Pima Indians diabetes data set, heart Statlog data set, sonar data set , image segmentation data set, ionosphere data set, breast cancer data set, satellite image data set และ letter recognition data set ได้ผลการทดสอบดังนี้

การจัดกลุ่มข้อมูล iris data set ด้วยอัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอแสดงดังตารางที่ 4.1, 4.2 และ 4.3 ตามลำดับ ตารางที่ 4.4, 4.5, และ 4.6 แสดงผลการจัดกลุ่มข้อมูล Indian Telugu Vowel data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ ตารางที่ 4.7, 4.8, และ 4.9 แสดงผลการจัดกลุ่มข้อมูล Pima Indians diabetes data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ ตารางที่ 4.10, 4.11, และ 4.12 แสดงผลการจัดกลุ่มข้อมูล heart Statlog data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ ตารางที่ 4.13, 4.14, และ 4.15 แสดงผลการจัดกลุ่มข้อมูล sonar data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ ตารางที่ 4.16, 4.17, และ 4.18 แสดงผลการจัดกลุ่มข้อมูล image segmentation data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ ตารางที่ 4.19, 4.20, และ 4.21 แสดงผลการจัดกลุ่มข้อมูล ionosphere data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ ตารางที่ 4.22, 4.23, และ 4.24 แสดงผลการจัดกลุ่มข้อมูล breast

cancer data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ ตารางที่ 4.25, 4.26, และ 4.27 แสดงผลการจัดกลุ่มข้อมูล satellite image data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ ตารางที่ 4.28, 4.29, และ 4.30 แสดงผลการจัดกลุ่มข้อมูล letter recognition data set อัลกอริทึม k-means จีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ ตามลำดับ

ตารางที่ 4.1 ผลการจัดกลุ่ม iris data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=3

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	97.34622	0.0310
2	97.34622	0.0320
3	97.325924	0.0310
4	97.325924	0.0310
5	122.478706	0.0310

ตารางที่ 4.2 ผลการจัดกลุ่ม iris data set ด้วยจีเนติกอัลกอริทึม เมื่อกำหนด k=3

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	97.325924	0.7030
2	97.325924	1.8590
3	97.325924	0.8430
4	97.325924	1.2500
5	97.325924	0.6870

ตารางที่ 4.3 ผลการจัดกลุ่ม iris data set ด้วยอัลกอริทึมที่นำเสนอ เมื่อกำหนด k=3

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	97.133887	0.2500
2	97.287073	0.2500
3	96.993731	0.4220
4	97.037044	0.4690
5	97.21852	0.2500

ตารางที่ 4.4 ผลการจัดกลุ่ม Indian Telugu vowel data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=6

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	151605.6001	0.0790
2	149912.5029	0.1400
3	153962.6897	0.0780
4	149773.9404	0.1090
5	161523.3703	0.1090

ตารางที่ 4.5 ผลการจัดกลุ่ม Indian Telugu vowel data set ด้วยจีเนติกอัลกอริทึม เมื่อกำหนด k=6

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	149405.9451	5.3750
2	149423.5982	4.6250
3	149383.9932	5.2030
4	149408.9149	6.1250
5	149369.6381	6.2660

ตารางที่ 4.6 ผลการจัดกลุ่ม Indian Telugu vowel data set ด้วยอัลกอริทึมที่นำเสนอ เมื่อกำหนด k=6

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	149383.9932	0.9370
2	149363.2346	1.0470
3	149394.804	0.8590
4	149336.0663	1.4690
5	149398.2631	0.8910

ตารางที่ 4.7 ผลการจัดกลุ่ม Pima Indians diabetes data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	52072.24398	0.0620
2	52072.24398	0.0470
3	52072.24398	0.0620
4	52072.24398	0.0470
5	52072.24398	0.0470

ตารางที่ 4.8 ผลการจัดกลุ่ม Pima Indians diabetes data set ด้วยจีเนติกอัลกอริทึม เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	52072.24398	3.4380
2	52072.24398	2.5620
3	52072.24398	1.5320
4	52072.24398	1.2660
5	52072.24398	1.3750

ตารางที่ 4.9 ผลการจัดกลุ่ม Pima Indians diabetes data set ด้วยอัลกอริทึมที่นำเสนอ เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	48101.44628	8.7340
2	49774.43019	4.7660
3	48354.90208	3.6870
4	48114.48396	5.6100
5	48385.70868	8.7180

ตารางที่ 4.10 ผลการจัดกลุ่ม heart Statlog data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	10695.79738	0.0470
2	10695.79738	0.0310
3	10700.83855	0.0310
4	10695.79738	0.0310
5	10695.79738	0.0470

ตารางที่ 4.11 ผลการจัดกลุ่ม heart Statlog data set ด้วยจีเนติกอัลกอริทึม เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	10695.79738	0.9840
2	10695.79738	0.6560
3	10695.79738	0.7970
4	10695.79738	0.8750
5	10695.79738	0.7500

ตารางที่ 4.12 ผลการจัดกลุ่ม heart Statlog data set ด้วย อัลกอริทึมที่นำเสนอ เมื่อ กำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	10663.08096	0.6410
2	10636.06313	0.9220
3	10647.29146	0.4690
4	10652.98316	0.6090
5	10650.95658	0.7810

ตารางที่ 4.13 ผลการจัดกลุ่ม sonar data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	235.206592	0.0470
2	234.771718	0.0310
3	235.206592	0.0310
4	235.206592	0.0630
5	235.206592	0.0470

ตารางที่ 4.14 ผลการจัดกลุ่ม sonar data set ด้วย จีเนติกอัลกอริทึม เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	234.771718	0.9370
2	234.767117	0.8590
3	234.771718	0.9060
4	234.771718	0.9370
5	234.767117	0.8750

ตารางที่ 4.15 ผลการจัดกลุ่ม sonar data set ด้วย อัลกอริทึมที่นำเสนอ เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	234.698667	0.7180
2	234.771718	0.4220
3	234.549081	1.3290
4	234.562322	2.5630
5	234.741962	0.8280

ตารางที่ 4.16 ผลการจัดกลุ่ม image segmentation data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=7

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	150360.7313	0.4840
2	150234.673	0.4060
3	150089.0796	0.3290
4	152370.2903	0.2820
5	153054.0302	0.5940

ตารางที่ 4.17 ผลการจัดกลุ่ม image segmentation data set ด้วยจีเนติกอัลกอริทึม เมื่อกำหนด k=7

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	149569.1243	20.2660
2	149569.1243	12.3280
3	149914.7218	18.2190
4	149569.1243	16.3750
5	149569.1243	24.1250

ตารางที่ 4.18 ผลการจัดกลุ่ม image segmentation data set ด้วยอัลกอริทึมที่นำเสนอ เมื่อกำหนด k=7

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	148299.7584	4.5470
2	149574.2048	3.0320
3	149574.6094	3.0780
4	149191.7432	14.2030
5	149552.2229	4.0940

ตารางที่ 4.19 ผลการจัดกลุ่ม ionosphere data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	796.466676	0.0470
2	796.327083	0.0310
3	796.466676	0.0310
4	796.327083	0.0310
5	796.466676	0.0310

ตารางที่ 4.20 ผลการจัดกลุ่ม ionosphere data set ด้วย จีเนติกอัลกอริทึม เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	796.327083	1.0620
2	796.327083	1.0620
3	796.327083	1.1090
4	796.327083	1.0630
5	796.327083	1.1710

ตารางที่ 4.21 ผลการจัดกลุ่ม ionosphere data set ด้วยอัลกอริทึมที่นำเสนอ เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	795.794277	0.9380
2	795.760104	1.4690
3	795.928124	0.8750
4	795.784821	2.6870
5	795.693735	1.2030

ตารางที่ 4.22 ผลการจัดกลุ่ม breast cancer data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	3056.96602	0.0790
2	3056.96602	0.0940
3	3056.96602	0.0780
4	3056.96602	0.0790
5	3056.96602	0.0780

ตารางที่ 4.23 ผลการจัดกลุ่ม breast cancer data set ด้วย จีเนติกอัลกอริทึม เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	3056.96602	5.0790
2	3056.96602	3.5630
3	3056.96602	3.3910
4	3056.96602	3.4690
5	3056.96602	3.4370

ตารางที่ 4.24 ผลการจัดกลุ่ม breast cancer data set ด้วยอัลกอริทึมที่นำเสนอ เมื่อกำหนด k=2

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	3041.697714	4.235
2	3037.632001	3.844
3	3039.63705	1.781
4	3037.632001	3.844
5	3036.208594	4.094

ตารางที่ 4.25 ผลการจัดกลุ่ม satellite image data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=6

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	291939.6658	2.0320
2	306387.3598	1.4060
3	291939.8733	1.5630
4	291958.0493	2.2190
5	307004.764	1.4690

ตารางที่ 4.26 ผลการจัดกลุ่ม satellite image data set ด้วย จีเนติกอัลกอริทึม เมื่อกำหนด k=6

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	291939.6658	296.5620
2	291938.1466	254.0460
3	291930.8805	360.0000
4	291930.8805	368.6090
5	291938.1466	586.5000

ตารางที่ 4.27 ผลการจัดกลุ่ม satellite image data set ด้วยอัลกอริทึมที่นำเสนอ เมื่อกำหนด k=6

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	290734.4865	110.5470
2	291939.8733	29.5310
3	291489.4076	79.4530
4	291754.9195	34.4850
5	291706.7587	52.0620

ตารางที่ 4.28 ผลการจัดกลุ่ม letter recognition data set ด้วยอัลกอริทึม k-means เมื่อกำหนด k=26

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	107714.6659	28.2650
2	107280.0649	18.5630
3	107283.5448	48.2650
4	108163.3629	22.5000
5	108176.7401	29.6250

ตารางที่ 4.29 ผลการจัดกลุ่ม letter recognition data set ด้วยจีเนติกอัลกอริทึม เมื่อกำหนด k=26

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	106912.2114	4166.1870
2	106936.3717	5966.8440
3	106937.6302	3478.7650
4	106991.441	5005.5780
5	106991.445	4170.9840

ตารางที่ 4.30 ผลการจัดกลุ่ม letter recognition data set ด้วยอัลกอริทึมที่นำเสนอ เมื่อกำหนด k=26

การทดสอบครั้งที่	clustering metric M	เวลา (วินาที)
1	106923.6659	246.2970
2	107043.7487	231.9220
3	107050.0137	342.6710
4	107001.194	246.0310
5	107003.2171	330.5780

จากการทดสอบอัลกอริทึมทั้ง 3 อัลกอริทึมกับข้อมูลทดสอบ 10 ชุด จำนวนชุดข้อมูลละ 5 ครั้ง แสดงให้เห็นว่าอัลกอริทึม k-means ทำงานให้ผลการทำงานที่รวดเร็ว แต่ค่า clustering metric M ที่ต่ำที่สุดที่ได้ค่อนข้างมีความแปรปรวน ส่วนจีเนติกอัลกอริทึมให้ clustering metric M ที่ต่ำที่สุดที่แปรปรวนน้อยกว่าอัลกอริทึม k-means แต่ใช้เวลามากกว่า และอัลกอริทึมที่นำเสนอสามารถจัดกลุ่มได้อย่างรวดเร็วกว่าจีเนติกอัลกอริทึมและให้ค่า clustering metric M ที่ต่ำที่สุดที่ดีกว่าทั้งอัลกอริทึม k-means และจีเนติกอัลกอริทึม

เมื่อทำการทดสอบอัลกอริทึมกับข้อมูลทดสอบชุดเดิม โดยทดลองเป็นจำนวน 1,000 ครั้ง พบว่า อัลกอริทึมที่นำเสนอให้ค่า clustering metric M ที่ต่ำที่สุด ต่ำกว่าอัลกอริทึม k-means และจีเนติกอัลกอริทึม ดังแสดงในตารางที่ 4.31 - 4.40

ตารางที่ 4.31 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม iris data set เมื่อกำหนด k=3

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	97.3259	125.1921
จีเนติกอัลกอริทึม	97.3259	97.3259
จีเนติกอัลกอริทึมที่นำเสนอ	96.7114	97.3259

ตารางที่ 4.32 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม Indian Telugu vowel data set เมื่อกำหนด k=2

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	149369.6381	168675.2467
จีเนติกอัลกอริทึม	149362.7891	151432.5288
จีเนติกอัลกอริทึมที่นำเสนอ	149107.7242	151469.2550

ตารางที่ 4.33 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม Pima Indians diabetes data set เมื่อกำหนด k=2

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	52072.2440	52072.2440
จีเนติกอัลกอริทึม	52072.2440	52072.2440
จีเนติกอัลกอริทึมที่นำเสนอ	47697.4063	52072.2440

ตารางที่ 4.34 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม heart Statlog data set เมื่อกำหนด k=2

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	10695.7974	10700.8385
จีเนติกอัลกอริทึม	10695.7974	10695.7974
จีเนติกอัลกอริทึมที่นำเสนอ	10624.3726	10695.7974

ตารางที่ 4.35 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม sonar data set เมื่อกำหนด k=2

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	234.7671	247.1384
จีเนติกอัลกอริทึม	234.7671	234.8277
จีเนติกอัลกอริทึมที่นำเสนอ	234.1071	235.1075

ตารางที่ 4.36 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม image segmentation data set เมื่อกำหนด k=7

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	148154.9345	160055.7852
จีเนติกอัลกอริทึม	148300.8432	150953.7911
จีเนติกอัลกอริทึมที่นำเสนอ	146277.6523	150983.4004

ตารางที่ 4.37 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม ionosphere data set เมื่อกำหนด k=2

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	796.3271	974.0242
จีเนติกอัลกอริทึม	796.3271	796.3271
จีเนติกอัลกอริทึมที่นำเสนอ	795.1641	796.2373

ตารางที่ 4.38 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม breast cancer data set เมื่อกำหนด k=2

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	3056.9660	3056.9660
จีเนติกอัลกอริทึม	3056.9660	7359.1273
จีเนติกอัลกอริทึมที่นำเสนอ	3035.1664	3053.9988

ตารางที่ 4.39 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม satellite image data set เมื่อกำหนด k=6

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	291939.6658	316996.9259
จีเนติกอัลกอริทึม	291930.8805	305947.7326
จีเนติกอัลกอริทึมที่นำเสนอ	290192.1843	305931.2907

ตารางที่ 4.40 ค่า clustering metric M ต่ำสุดและสูงสุดจากการจัดกลุ่ม letter recognition data set เมื่อกำหนด k=26

	clustering metric M ต่ำสุด	clustering metric M สูงสุด
k-means อัลกอริทึม	107060.3102	108845.3471
จีเนติกอัลกอริทึม	106911.1552	107781.9854
จีเนติกอัลกอริทึมที่นำเสนอ	106910.0929	107474.0845

จากตารางที่ 4.41 – 4.50 แสดงค่าเฉลี่ยของค่า clustering metric M และเวลาที่ใช้ในการทดสอบข้อมูลทั้ง 10 ชุดข้อมูลจำนวน 1,000 ครั้ง แสดงให้เห็นว่าอัลกอริทึมที่นำเสนอให้ค่าเฉลี่ย clustering metric M ที่ต่ำกว่าอัลกอริทึม k-means และจีเนติกอัลกอริทึม และเวลาที่ใช้ในการทดสอบอัลกอริทึม k-means ใช้เวลาทดสอบน้อยที่สุด รองลงมาได้แต่อัลกอริทึมที่นำเสนอและจีเนติกอัลกอริทึมตามลำดับ

ตารางที่ 4.41 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม iris data set เมื่อกำหนด k=3

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	102.7565	10.4920	0.0299
จีเนติกอัลกอริทึม	97.3259	0.0000	0.9829
จีเนติกอัลกอริทึมที่นำเสนอ	97.0814	0.1643	0.3866

ตารางที่ 4.42 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม Indian Telugu vowel data set เมื่อกำหนด k=2

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	152942.3559	3760.9306	0.0975
จีเนติกอัลกอริทึม	149446.6858	185.4153	5.3476
จีเนติกอัลกอริทึมที่นำเสนอ	149802.7549	579.1148	1.1144

ตารางที่ 4.43 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม Pima Indians diabetes data set เมื่อกำหนด k=2

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	52072.2440	0	0.0499
จีเนติกอัลกอริทึม	52072.2440	0	1.3432
จีเนติกอัลกอริทึมที่นำเสนอ	49153.2543	877.2324	5.0095

ตารางที่ 4.44 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม heart Statlog data set เมื่อกำหนด k=2

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	10697.6018	2.3935	0.0338
จีเนติกอัลกอริทึม	10695.7974	0.0013	0.8646
จีเนติกอัลกอริทึมที่นำเสนอ	10650.9786	13.1170	0.8309

ตารางที่ 4.45 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม sonar data set เมื่อกำหนด k=2

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	235.1134	0.7458	0.0409
จีเนติกอัลกอริทึม	234.7705	0.0027	1.0123
จีเนติกอัลกอริทึมที่นำเสนอ	234.7385	0.1328	0.9959

ตารางที่ 4.46 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม image segmentation data set เมื่อกำหนด k=7

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	151092.9695	1478.8744	0.2876
จีเนติกอัลกอริทึม	149600.7532	206.9746	21.0573
จีเนติกอัลกอริทึมที่นำเสนอ	149549.1094	603.8821	7.4288

ตารางที่ 4.47 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม ionosphere data set เมื่อกำหนด k=2

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	805.2225	38.3870	0.0351
จีเนติกอัลกอริทึม	796.3271	0.0000	1.1443
จีเนติกอัลกอริทึมที่นำเสนอ	795.8559	0.1768	1.2661

ตารางที่ 4.48 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม breast cancer data set เมื่อกำหนด k=2

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	3056.9660	0	0.0883
จีเนติกอัลกอริทึม	3069.8725	235.4031	3.5641
จีเนติกอัลกอริทึมที่นำเสนอ	3039.5193	2.6024	3.7638

ตารางที่ 4.49 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม satellite image data set เมื่อกำหนด k=6

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	297464.9243	8236.6563	1.8003
จีเนติกอัลกอริทึม	291962.6099	625.5323	258.7567
จีเนติกอัลกอริทึมที่นำเสนอ	291625.1704	590.5379	29.7047

ตารางที่ 4.50 ค่าเฉลี่ย clustering metric M ค่าเบี่ยงเบนมาตรฐานและเวลาเฉลี่ยจากการจัดกลุ่ม letter recognition data set เมื่อกำหนด k=26

	ค่าเฉลี่ย clustering metric M	ค่าเบี่ยงเบนมาตรฐาน	เวลาเฉลี่ย (วินาที)
k-means อัลกอริทึม	107775.0990	368.2664	79.6387
จีเนติกอัลกอริทึม	107156.0843	154.5161	3705.6039
จีเนติกอัลกอริทึมที่นำเสนอ	107292.2251	125.6744	255.5808

บทที่ 5
สรุปผลการวิจัย และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

ในยุคที่เทคโนโลยีข่าวสารก้าวหน้าไปอย่างรวดเร็ว การจัดการข้อมูลมีบทบาทสำคัญเพื่อช่วยให้ข้อมูลเหล่านั้นถูกนำมาใช้ประโยชน์ได้มากยิ่งขึ้น โดยเฉพาะการจัดการกับข้อมูลที่มีปริมาณมาก การวิเคราะห์ข้อมูลเหล่านั้นจึงต้องอาศัยเครื่องมือช่วยในการจัดการ ดาต้าไมนิ่ง (Data mining) เป็นเครื่องมือสำหรับวิเคราะห์ข้อมูล โดยมีวิธีการวิเคราะห์ข้อมูลแบบต่าง ๆ เช่น การวิเคราะห์ความสัมพันธ์ของข้อมูล (Link Analysis), การจำแนกประเภทของข้อมูล (Classification), การทำนาย (Forecasting), การจัดกลุ่ม (Clustering) เป็นต้น

การจัดกลุ่มข้อมูลเป็นการพิจารณาคุณสมบัติของข้อมูลโดยจัดให้ข้อมูลที่มีคุณสมบัติคล้ายคลึงกันอยู่ในกลุ่มเดียวกัน อัลกอริทึมการจัดกลุ่มข้อมูลจะพิจารณาระยะห่างระหว่างข้อมูลกับศูนย์กลางของกลุ่ม ซึ่งถ้าข้อมูลนั้นอยู่ใกล้กับศูนย์กลางใดมากที่สุดก็จะถูกจัดให้เป็นสมาชิกของกลุ่มนั้น

เทคนิคในการจัดกลุ่มข้อมูลจำแนกออกเป็นประเภทต่าง ๆ ได้แก่ การจัดกลุ่มข้อมูลจากข้อมูลจำนวน n ข้อมูลโดยสร้างกลุ่มจำนวน k กลุ่ม (Partitioning methods), การจัดกลุ่มข้อมูลด้วยการสร้างโครงสร้างแบบลำดับชั้นของข้อมูล (Hierarchical methods), การจัดกลุ่มโดยพิจารณาค่าความจุของข้อมูล (Density-based methods) และการจัดกลุ่มข้อมูลโดยใช้โครงสร้างข้อมูลแบบกริด (Grid-based methods)

อัลกอริทึม k-means เป็นอัลกอริทึมสำหรับจัดกลุ่มข้อมูลตามแนวคิดของ partitioning methods ซึ่งเป็นที่นิยมอย่างแพร่หลาย การทำงานของอัลกอริทึม k-means เป็นการสุ่มเลือกศูนย์กลางเริ่มต้นชุดเดียวจากนั้นทำงานไปเรื่อยๆจนกว่าจะได้กลุ่มที่มีความเหมาะสมสูงที่สุด ซึ่งการจัดกลุ่มที่ได้อาจไม่ใช่การจัดกลุ่มที่ดีที่สุดในชุดข้อมูลนั้นๆ เพราะการสุ่มเลือกศูนย์กลางเพียงครั้งเดียว อาจไม่สามารถเจอคำตอบของปัญหาได้ ดังนั้นเพื่อเพิ่มโอกาสที่จะค้นพบคำตอบที่แท้จริง ควรทำซ้ำอัลกอริทึมดังกล่าวหลายๆรอบ ด้วยค่าศูนย์กลางเริ่มต้นที่แตกต่างกัน

ดังนั้นจีเนติกอัลกอริทึมจึงได้ถูกนำมาพิจารณาสำหรับการจัดกลุ่มข้อมูลโดยจีเนติกอัลกอริทึมทำงานร่วมกับ k-means อัลกอริทึม

จีเนติกอัลกอริทึม (Genetic algorithms) เป็นกระบวนการเลียนแบบการคัดเลือกสายพันธุ์ตามธรรมชาติ โดยในสิ่งมีชีวิตประกอบไปด้วยโครโมโซม (chromosome) ในโครโมโซมประกอบ ด้วยยีน (gene) หลายๆ ยีน การผสมพันธุ์กันระหว่างสิ่งมีชีวิตสองสิ่งอาจทำให้เกิดการครอสโอเวอร์ (crossover) หรือการมิวเตชั่น (mutation) ทั้งสองวิธีการทำให้ได้สิ่งมีชีวิตชนิด

ใหม่เกิดขึ้นกลายเป็นประชากรรุ่นลูก ประชากรเหล่านี้จะสามารถดำรงชีวิตอยู่ได้ในธรรมชาติโดยการคัดเลือกตามธรรมชาติ โดยประชากรที่มีความเหมาะสมมากที่สุดจะถูกคัดเลือก

จากปัญหาของอัลกอริทึม k-means ดังกล่าวไว้ในเบื้องต้น จีเนติกอัลกอริทึมถูกนำมาพัฒนาเพื่อทำให้การจัดกลุ่มด้วยอัลกอริทึม k-means โดยการจัดกลุ่มด้วยจีเนติกอัลกอริทึมจะสร้างประชากรเริ่มต้นเป็นจำนวน p ประชากร จากนั้นเลือกประชากรพ่อ-แม่ ด้วยวงล้อถ่วงน้ำหนักและ ดำเนินการครอสโอเวอร์และมิวเตชั่นตามลำดับ หลังจากได้ประชากรรุ่นพ่อ-แม่ และลูก ประชากรที่มีค่าความเหมาะสมสูงสุดจะถูกเลือกให้เป็นประชากรรุ่นต่อไป

เนื่องจากจีเนติกอัลกอริทึมใช้กระบวนการตัดสินใจด้วยการสุ่ม (Stochastic decision making) และพิจารณาค่าความน่าจะเป็นประกอบการตัดสินใจ กระบวนการสุ่มของจีเนติกอัลกอริทึมเป็นการสุ่มที่ไม่อาจคาดเดาได้ ดังนั้นเพื่อกำหนดขอบเขตและทิศทางของโครโมโซมในวิทยานิพนธ์ฉบับนี้ได้นำเสนอการนำประวัติจากโครโมโซมมาใช้ในการปรับปรุงการจัดกลุ่มด้วยจีเนติกอัลกอริทึมประวัติของโครโมโซมจะถูกแบ่งออกเป็น 2 กลุ่มได้แก่ "โครโมโซมกลุ่มดี" และ "โครโมโซมกลุ่มไม่ดี" โครโมโซมกลุ่มดี ประกอบด้วย โครโมโซมประวัติของโครโมโซมปัจจุบันซึ่งมีค่าความเหมาะสมมากกว่าค่าความเหมาะสมของโครโมโซมปัจจุบัน และโครโมโซมกลุ่มไม่ดี ประกอบด้วย โครโมโซมประวัติของโครโมโซมปัจจุบันซึ่งมีค่าความเหมาะสมน้อยกว่าค่าความเหมาะสมของโครโมโซมปัจจุบัน จากนั้นค่าของโครโมโซมทั้งสองกลุ่มจะถูกนำมาพิจารณาเพื่อมิวเตทโครโมโซมปัจจุบัน โดยการมิวเตทยีนจะพิจารณาค่ายีนปัจจุบันและค่ายีนในอดีต ยีนปัจจุบันจะปรับเข้าหายีนในโครโมโซมในอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มมาก และปรับออกจากยีนในโครโมโซมอดีตที่ให้ค่าความเหมาะสมในการจัดกลุ่มน้อย การพิจารณาเช่นนี้มีผลให้การปรับทิศทางของยีนมีความน่าจะเป็นที่จะเข้าถึงค่าที่ดีที่สุดได้เร็วยิ่งขึ้น

โดยการกำหนดขอบเขตและทิศทางสำหรับการปรับปรุงค่ายีนในโครโซมประกอบด้วย 2 คอมโพเนนท์ ได้แก่ส่วนของคอมโมเนนท์ที่เกิดขึ้นจากการสุ่ม (random component) คอมโพเนนท์ที่คงที่ (deterministic component)

เมื่อทำการทดสอบอัลกอริทึมที่นำเสนอกับชุดข้อมูลมาตรฐานจำนวน 10 ชุดข้อมูล ได้แก่ iris data set, Indian Telugu vowel data, Pima Indians diabetes data set, heart Statlog data set, sonar data set , image segmentation data set, ionosphere data set, breast cancer data set, satellite image data set และ letter recognition data set พบว่าอัลกอริทึมที่นำเสนอให้ค่าความเหมาะสมที่ดีกว่าอัลกอริทึมการจัดกลุ่มข้อมูลด้วย k-means และ อัลกอริทึมการจัดกลุ่มข้อมูลด้วยจีเนติก

วิธีการที่นำเสนอในวิทยานิพนธ์เป็นเทคนิคที่พัฒนาต่อเนื่องมาจากการจัดกลุ่มข้อมูลด้วยจีเนติกอัลกอริทึม จากเดิมจีเนติกอัลกอริทึมสามารถจัดกลุ่มข้อมูลมีประสิทธิภาพดีอยู่แล้ว เมื่อเทียบกับการจัดกลุ่มข้อมูลด้วยอัลกอริทึม k-means หลังจากปรับปรุงกระบวนการมิวเตชั่นของจีเนติกอัลกอริทึม ทำให้จีเนติกอัลกอริทึมมีประสิทธิภาพการทำงานดียิ่งขึ้น

5.2 ข้อเสนอแนะ

ขอบเขตในการศึกษาสำหรับการทำวิทยานิพนธ์ในครั้งนี้ เรามุ่งเน้นไปที่การจัดกลุ่มข้อมูลที่เป็นตัวเลข แต่ข้อมูลในโลกแห่งความจริง ประกอบไปด้วยข้อมูลหลากหลายชนิด ดังนั้นเพื่อการพัฒนาการจัดกลุ่มข้อมูลให้เป็นประโยชน์มากยิ่งขึ้น การพัฒนาการจัดกลุ่มข้อมูลที่มีความหลากหลายและสนองตอบกับข้อมูลชนิดต่างๆจึงเป็นส่วนที่น่าจะมีการพัฒนาต่อไปในอนาคต

บรรณานุกรม

[1] U. Maulik and S. Bandyopadhyay, **"Genetic Algorithm-Based Clustering Technique,"** Pattern Recognition, Vol. 33, pp. 1455-1465, (2000).

[2] S. Bandyopadhyay and U. Maulik, **"Genetic Clustering for Automatic Evolution of Clusters and Application to Image Classification,"** Pattern Recognition, Vol. 35, no. 2, pp. 1197-1208, (2002).

[3] W. Sheng and X. Liu, **"A Hybrid Algorithm for K-Medoid Clustering of Large Data Sets,"** IEEE Congress on Evolutionary Computation (CEC-2004), pp. 77-82, (2004).

[4] R. T. Ng and J. Han, **"Efficient and Effective Clustering Methods for Spatial Data Mining,"** Proc. 20th Conf. Very Large Databases, pp. 144–155, 1994.

[5] S. Bandyopadhyay and U. Maulik, **"An Evolutionary Technique Based on K-Means Algorithm for Optimal Clustering in R^N,"** Information Sciences-Applications: An Int'l J., Vol. 146, pp. 221-237, (2002).

[6] C. L. Blake and C. J. Merz, UCI Repository of Machine Learning Databases [http://www.ics.uci.edu/~mlearn/MLRepository.html], University of California, Deapartment of Information and Computer Science, (1998).

[7] D. H. Deterding, **"Speaker Normalization for Automatic Speech Recognition,"** Ph.D. Dissertation, (1989).

[8] Statlog Project Datasets, Retrived March 28, 2003 From http://www.liacc.up.pt/ML/statlog/datasets/.

[9] S. K. Pal and D. Dutta Majumder, **"Fuzzy sets and decision making approaches in vowel and speaker recognition,"** IEEE Transactions on Systems, Man, and Cybernetics, Vol. 7, pp. 625-629, (1977).

[10] A. Schatten, **Genetic Algorithms** . [online] Available : http://www.cs.ucdavis.edu/~vemuri/classes/ecs271/Genetic Algorithms Short Tutorial.htm.

[11] Ian H. Witten and Eibe Frank, **Data Mining**. Morgan Kaufmann Publisher, 2001.

[12] J. Han and M. Kamber, **Data Mining Concepts and Techniques** . Morgan Kaufmann Publisher, San Francisco, 2001.

[13] K. Teknomo, **K-means Clustering Tutorial** . [online] Available : http://people.revoledu.com/kardi/tutorial/kMean/index.html.

[14] Hwei-Jen Lin, Fu-Wen Yang and Yang-Ta Kao, **"An Efficient GA-based Clustering Technique,"** Tamkang Journal of Science and Engineering, Vol. 8, pp. 113–122, (2005).

[15] M. J.A. Berry and G. Linoff, **Data Mining Technique for Marketing, Sales, and Customer Support**. Wiley Computer Publishing, 1997.

[16] S. Bandyopadhyay and U. Maulik, **"Non-Parametric Genetic Clustering: Comparison of Validity Indices,"** IEEE Trans. Systems, Man, and Cybernetics, Part-C, vol. 31, no. 1, pp. 120-125, (2001).

ภาคผนวก

ภาคผนวก ก.

ตัวอย่างผลการทดสอบอัลกอริทึม

การทดสอบการจัดกลุ่มข้อมูลดอกไอริสจำนวน 150 ข้อมูล แต่ละข้อมูลประกอบด้วย 4 แอททริบิว ได้แก่ ความกว้างและความยาวของกลีบเลี้ยงและความกว้างและความยาวของกลีบดอก โดยดอกไอริสที่นำมาทดสอบ สำรวจมาจากดอกไอริส 3 สปีชีส์ ได้แก่ setosa, versicolor และ virginica สปีชีส์ละ 50 ข้อมูล

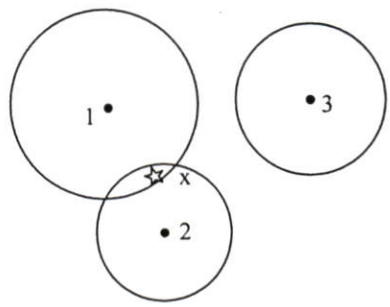
attribute1	attribute 2	attribute 3	attribute 4	GA		NGA	
				Group	TC	Group	TC
5.1	3.5	1.4	0.2	2	0.1469	2	0.1469
4.9	3	1.4	0.2	2	0.4382	2	0.4382
4.7	3.2	1.3	0.2	2	0.4123	2	0.4123
4.6	3.1	1.5	0.2	2	0.5188	2	0.5188
5	3.6	1.4	0.2	2	0.198	2	0.198
5.4	3.9	1.7	0.4	2	0.6838	2	0.6838
4.6	3.4	1.4	0.3	2	0.4152	2	0.4152
5	3.4	1.5	0.2	2	0.0599	2	0.0599
4.4	2.9	1.4	0.2	2	0.801	2	0.801
4.9	3.1	1.5	0.1	2	0.3666	2	0.3666
5.4	3.7	1.5	0.2	2	0.4878	2	0.4878
4.8	3.4	1.6	0.2	2	0.2514	2	0.2514
4.8	3	1.4	0.1	2	0.4919	2	0.4919
4.3	3	1.1	0.1	2	0.9091	2	0.9091
5.8	4	1.2	0.2	2	1.0202	2	1.0202
5.7	4.4	1.5	0.4	2	1.2131	2	1.2131
5.4	3.9	1.3	0.4	2	0.6624	2	0.6624
5.1	3.5	1.4	0.3	2	0.151	2	0.151
5.7	3.8	1.7	0.3	2	0.8285	2	0.8285
5.1	3.8	1.5	0.3	2	0.399	2	0.399
5.4	3.4	1.7	0.2	2	0.4617	2	0.4617
5.1	3.7	1.5	0.4	2	0.3376	2	0.3376
4.6	3.6	1	0.2	2	0.6444	2	0.6444
5.1	3.3	1.7	0.5	2	0.3795	2	0.3795
4.8	3.4	1.9	0.2	2	0.4846	2	0.4846
5	3	1.6	0.2	2	0.4418	2	0.4418
5	3.4	1.6	0.4	2	0.2078	2	0.2078
5.2	3.5	1.5	0.2	2	0.2182	2	0.2182
5.2	3.4	1.4	0.2	2	0.2097	2	0.2097
4.7	3.2	1.6	0.2	2	0.402	2	0.402
4.8	3.1	1.6	0.2	2	0.405	2	0.405
5.4	3.4	1.5	0.4	2	0.4257	2	0.4257
5.2	4.1	1.5	0.1	2	0.7244	2	0.7244
5.5	4.2	1.4	0.2	2	0.9282	2	0.9282
4.9	3.1	1.5	0.1	2	0.3666	2	0.3666
5	3.2	1.2	0.2	2	0.3452	2	0.3452
5.5	3.5	1.3	0.2	2	0.5288	2	0.5288
4.9	3.1	1.5	0.1	2	0.3666	2	0.3666
4.4	3	1.3	0.2	2	0.7555	2	0.7555
5.1	3.4	1.5	0.2	2	0.1113	2	0.1113
5	3.5	1.3	0.3	2	0.1918	2	0.1918
4.5	2.3	1.3	0.3	2	1.2394	2	1.2394

attribute1	attribute 2	attribute 3	attribute 4	GA		NGA	
				Group	TC	Group	TC
4.4	3.2	1.3	0.2	2	0.666	2	0.666
5	3.5	1.6	0.6	2	0.3899	2	0.3899
5.1	3.8	1.9	0.4	2	0.6076	2	0.6076
4.8	3	1.4	0.3	2	0.4737	2	0.4737
5.1	3.8	1.6	0.2	2	0.4186	2	0.4186
4.6	3.2	1.4	0.2	2	0.4673	2	0.4673
5.3	3.7	1.5	0.2	2	0.4113	2	0.4113
5	3.3	1.4	0.2	2	0.1414	2	0.1414
7	3.2	4.7	1.4	3	1.227	1	1.2125
6.4	3.2	4.5	1.5	3	0.6841	3	0.6841
6.9	3.1	4.9	1.5	1	1.019	1	0.9662
5.5	2.3	4	1.3	3	0.7315	3	0.7315
6.5	2.8	4.6	1.5	3	0.6385	3	0.6385
5.7	2.8	4.5	1.3	3	0.2694	3	0.2694
6.3	3.3	4.7	1.6	3	0.7645	3	0.7645
4.9	2.4	3.3	1	3	1.5839	3	1.5839
6.6	2.9	4.6	1.3	3	0.7558	3	0.7558
5.2	2.7	3.9	1.4	3	0.8598	3	0.8598
5	2	3.5	1	3	1.5361	3	1.5361
5.9	3	4.2	1.5	3	0.3243	3	0.3243
6	2.2	4	1	3	0.8084	3	0.8084
6.1	2.9	4.7	1.4	3	0.3967	3	0.3967
5.6	2.9	3.6	1.3	3	0.8727	3	0.8727
6.7	3.1	4.4	1.4	3	0.8731	3	0.8731
5.6	3	4.5	1.5	3	0.4123	3	0.4123
5.8	2.7	4.1	1	3	0.5358	3	0.5358
6.2	2.2	4.5	1.5	3	0.6368	3	0.6368
5.6	2.5	3.9	1.1	3	0.7125	3	0.7125
5.9	3.2	4.8	1.8	3	0.7094	3	0.7094
6.1	2.8	4	1.3	3	0.4635	3	0.4635
6.3	2.5	4.9	1.5	3	0.6937	3	0.6937
6.1	2.8	4.7	1.2	3	0.4366	3	0.4366
6.4	2.9	4.3	1.3	3	0.5459	3	0.5459
6.6	3	4.4	1.4	3	0.7431	3	0.7431
6.8	2.8	4.8	1.4	3	0.988	3	0.988
6.7	3	5	1.7	1	0.8464	1	0.7481
6	2.9	4.5	1.5	3	0.2199	3	0.2199
5.7	2.6	3.5	1	3	1.0244	3	1.0244
5.5	2.4	3.8	1.1	3	0.864	3	0.864
5.5	2.4	3.7	1	3	0.9757	3	0.9757
5.8	2.7	3.9	1.2	3	0.5576	3	0.5576
6	2.7	5.1	1.6	3	0.734	3	0.734
5.4	3	4.5	1.5	3	0.575	3	0.575
6	3.4	4.5	1.6	3	0.6879	3	0.6879
6.7	3.1	4.7	1.5	3	0.927	3	0.927
6.3	2.3	4.4	1.3	3	0.6146	3	0.6146
5.6	3	4.1	1.3	3	0.5083	3	0.5083
5.5	2.5	4	1.3	3	0.6291	3	0.6291
5.5	2.6	4.4	1.2	3	0.4879	3	0.4879
6.1	3	4.6	1.4	3	0.3827	3	0.3827

attribute1	attribute 2	attribute 3	attribute 4	GA		NGA	
				Group	TC	Group	TC
5.8	2.6	4	1.2	3	0.4919	3	0.4919
5	2.3	3.3	1	3	1.5486	3	1.5486
5.6	2.7	4.2	1.3	3	0.3856	3	0.3856
5.7	3	4.2	1.2	3	0.4428	3	0.4428
5.7	2.9	4.2	1.3	3	0.345	3	0.345
6.2	2.9	4.3	1.3	3	0.3724	3	0.3724
5.1	2.5	3	1.1	3	1.6606	3	1.6606
5.7	2.8	4.1	1.3	3	0.3839	3	0.3839
6.3	3.3	6	2.5	1	0.7773	1	0.7151
5.8	2.7	5.1	1.9	3	0.8538	3	0.8538
7.1	3	5.9	2.1	1	0.3061	1	0.4983
6.3	2.9	5.6	1.8	1	0.6529	1	0.4857
6.5	3	5.8	2.2	1	0.3846	1	0.2733
7.6	3	6.6	2.1	1	1.1423	1	1.3275
4.9	2.5	4.5	1.7	3	1.071	3	1.071
7.3	2.9	6.3	1.8	1	0.7857	1	0.952
6.7	2.5	5.8	1.8	1	0.6545	1	0.6349
7.2	3.6	6.1	2.5	1	0.8436	1	0.9815
6.5	3.2	5.1	2	1	0.7455	1	0.5986
6.4	2.7	5.3	1.9	1	0.7529	1	0.5943
6.8	3	5.5	2.1	1	0.2596	1	0.2094
5.7	2.5	5	2	3	0.8892	3	0.8892
5.8	2.8	5.1	2.4	3	1.2023	1	1.1175
6.4	3.2	5.3	2.3	1	0.6829	1	0.5248
6.5	3	5.5	1.8	1	0.5099	1	0.3537
7.7	3.8	6.7	2.2	1	1.4779	1	1.6496
7.7	2.6	6.9	2.3	1	1.5297	1	1.6967
6	2.2	5	1.5	3	0.8262	3	0.8262
6.9	3.2	5.7	2.3	1	0.2695	1	0.3611
5.6	2.8	4.9	2	3	0.8189	3	0.8189
7.7	2.8	6.7	2	1	1.3115	1	1.491
6.3	2.7	4.9	1.8	3	0.7427	3	0.7427
6.7	3.3	5.7	2.1	1	0.2763	1	0.2506
7.2	3.2	6	1.8	1	0.5277	1	0.6978
6.2	2.8	4.8	1.8	3	0.6253	3	0.6253
6.1	3	4.9	1.8	3	0.7023	3	0.7023
6.4	2.8	5.6	2.1	1	0.5463	1	0.3819
7.2	3	5.8	1.6	1	0.5943	1	0.7181
7.4	2.8	6.1	1.9	1	0.7313	1	0.9064
7.9	3.8	6.4	2	1	1.438	1	1.6179
6.4	2.8	5.6	2.2	1	0.5606	1	0.4044
6.3	2.8	5.1	1.5	3	0.8154	3	0.8154
6.1	2.6	5.6	1.4	1	1.1213	1	0.9899
7.7	3	6.1	2.3	1	0.9531	1	1.1465
6.3	3.4	5.6	2.4	1	0.7331	1	0.611
6.4	3.1	5.5	1.8	1	0.579	1	0.41
6	3	4.8	1.8	3	0.6101	3	0.6101
6.9	3.1	5.4	2.1	1	0.3479	1	0.3432
6.7	3.1	5.6	2.4	1	0.3893	1	0.3457
6.9	3.1	5.1	2.3	1	0.684	1	0.6435

attribute1	attribute 2	attribute 3	attribute 4	GA		NGA	
				Group	TC	Group	TC
5.8	2.7	5.1	1.9	3	0.8538	3	0.8538
6.8	3.2	5.9	2.3	1	0.3095	1	0.393
6.7	3.3	5.7	2.5	1	0.5094	1	0.5034
6.7	3	5.2	2.3	1	0.6117	1	0.5147
6.3	2.5	5	1.9	3	0.8975	3	0.8975
6.5	3	5.2	2	1	0.6533	1	0.4913
6.2	3.4	5.4	2.3	1	0.8357	1	0.679
5.9	3	5.1	1.8	3	0.8345	3	0.8345
ผลรวม					**97.3256**		**96.9828**

เมื่อเปรียบเทียบผลการทดสอบจีเนติกอัลกอริทึมและอัลกอริทึมที่นำเสนอ พบว่าทั้งสองอัลกอริทึมให้ผลการจัดกลุ่มที่ใกล้เคียงกัน แต่มีเพียงบางข้อมูลที่ถูกจัดกลุ่มแตกต่างกัน จากตารางแสดงผลการเปรียบเทียบนี้ พบว่ามีเพียง 1 ข้อมูล(ที่วงกลมไว้)ที่สองอัลกอริทึมให้ผลการจัดกลุ่มแตกต่างกัน ซึ่งข้อมูลดังกล่าวอาจเป็นข้อมูลที่อยู่ระหว่างรอยต่อของการจัดกลุ่มดังภาพด้านล่าง



ข้อมูล x เป็นข้อมูลที่อยู่ในช่วงรอยต่อระหว่างข้อมูลกลุ่ม A และ ข้อมูลกลุ่ม B ดังนั้นการที่จะจัดข้อมูลดังกล่าวไว้กับกลุ่มใดก็ขึ้นอยู่กับการหาศูนย์กลางของกลุ่มด้วยอัลกอริทึม อัลกอริทึมใดสามารถหาศูนย์กลางที่ดีที่สุดได้ ซึ่งจะมีผลให้ค่า Total Cost (TC) ของการจัดกลุ่มนั้นมีค่าน้อยที่สุด ซึ่งหมายถึงว่าการจัดกลุ่มนั้นมีความเหมาะสมมากที่สุดนั่นเอง

ภาคผนวก ข.

ผลงานวิจัยที่ได้รับการยอมรับตีพิมพ์เผยแพร่

1. A. Thammano and **U. Kakulphimp, "Genetic Algorithm-based Clustering and Its New Mutation Operator,"** 2006 International Conference on Intelligent Computing (ICIC 2006), Kunming Yunnan Province, China, August 16–19, 2006.

Call for Papers

2006 International Conference on Intelligent Computing (ICIC'06)

(http://www.ic-ic.org/2006/index.htm)

August 16-19, 2006

Harbour Plaza, Kunming, China

Paper Submission Deadline: March 1, 2006

Internet based paper submission opens December 15, 2005

ICIC 2006 is organized by The Yunnan University, The Institute of Intelligent Machines, and The University of Science & Technology of China, Chinese Academy of Sciences as well as The Queen's University Belfast, UK, and technically co-sponsored by The IEEE Computational Intelligence Society and The International Neural Network Society, and financially supported by The National Science Foundation of China.

The complete list of topics is available at http://www.ic-ic.org/2006/index.htm. The conference proceeding will be published by Springer Verlag in Lecture Notes in Computer Sciences (LNCS)/Lecture Notes in Artificial Intelligence(LNAI), and a small selected number of papers will be extended and revised for possible inclusion in several main stream international journals. However, the incurred fees for subsequent journal publication will be bored by the authors.

Important Deadlines

Paper submission:	~~1 March 2006~~ 1 April 2006
Decision notification:	1 May 2006
Special session proposal:	31 January 2006
Tutorial proposal:	1 March 2006
Camera-ready submission:	18 May 2006
Registration:	18 May 2006

SPECIAL SESSIONS

The ICIC 2006 Program Committee is inviting proposals for special sessions. Authors who planned papers for special sessions which will not be accepted can submit their papers as regular submissions to the ICIC 2006. Proposals should be submitted in ELECTRONIC FORM to Dr. Wen Yu, CINVESTAV-IPN, Mexico.

Important Deadlines for Special Sessions:

January 31, 2006: Special session proposal deadline

February 15, 2006: Decision notification

Manuscripts should be written in English with a single-column, single-space format. Each paper should not exceed eight pages including figures and references. The format for submitted papers should refer to the ELECTRONIC FORMAT REQUIREMENTS of the Springer-Verlag. Authors should submit their papers to the ICIC2006 by the Online Submission & Review System.

For more information, please visit the conference website:

http://www.ic-ic.org/2006/index.htm

2006 International Conference on Intelligent Computing

Genetic Algorithm-Based Clustering and Its New Mutation Operator

Arit Thammano and Uraiwan Kakulphimp

Computational Intelligence Laboratory
Faculty of Information Technology
King Mongkut's Institute of Technology Ladkrabang,
Bangkok, 10520 Thailand
arit@it.kmitl.ac.th, ukakulphimp@hotmail.com

Abstract. This paper proposes an extension to the original GA-clustering algorithm by introducing a new way to mutate the chromosome. The new mutation operator takes the previous values of the chromosome into account when mutating the chromosome. The superiority of the proposed approach over the original GA-clustering algorithm and K-means algorithm is demonstrated by using 6 benchmark data sets.

1 Introduction

Clustering is the process of grouping the data into clusters so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters [1]. Among all existing clustering algorithms, K-means algorithm is the most widely known; it is rather simple and remarkably effective. However, the K-means algorithm may get stuck at suboptimal solutions, depending on the choice of the initial cluster centers. Maulik and Bandyopadhyay [2] therefore have developed the GA-clustering algorithm, which integrates the capability of the genetic algorithm in avoiding local optima with the K-means algorithm. Their results show that the GA-clustering algorithm provides a performance that is significantly superior to that of the K-means algorithm.

The objective of this paper is to propose a new way to mutate the chromosome. The performance of the original GA-clustering algorithm is further improved when used in conjunction with this new mutation operator.

Following this introduction, section 2 and 3 briefly describe the K-means algorithm and the original GA-clustering algorithm respectively. The proposed mutation operator is presented in section 4. In section 5, the experimental results are demonstrated and discussed.

2 K-Means Algorithm

K-means is the most famous clustering algorithm. It groups a set of n objects into K clusters using Euclidean distance as the similarity measure. The steps of the K-means algorithm are described as follows:

D.-S. Huang, K. Li, and G.W. Irwin (Eds.): ICIC 2006, LNCS 4113, pp. 703 – 708, 2006.

A. Arbitrarily select K initial cluster centers ($z_1, z_2, \ldots, z_K$) from the input data.
B. Assign each data pattern x_i to the cluster C_j to which it is the most similar.
C. When all input data has been assigned, update the cluster centers as follows:

$$z_j^* = \frac{1}{n_j} \sum_{x_i \in C_j} x_i \,. \tag{1}$$

where n_j is the number of data belonging to cluster C_j.
D. If the stopping criterion is satisfied, terminate the loop. If not, go to step B.

3 GA-Clustering Algorithm

A description of the GA-clustering algorithm proposed by Maulik and Bandyopadhyay is given below.

A. Define the size (P) of the population. For each chromosome, randomly select K initial cluster centers ($z_1, z_2, \ldots, z_K$) from the data set and encode them into a chromosome. Therefore, each chromosome is structured as follows:

$$\text{Chromosome} = [z_{11}, \ldots, z_{1Q}, z_{21}, \ldots, z_{2Q}, \ldots, z_{K1}, \ldots, z_{KQ}] \,. \tag{2}$$

where Q is the dimension of each cluster center.
B. Evaluate the fitness of each chromosome in the population. The fitness computation process consists of two phases as follows:
B.1. Assign each data point x_i to the cluster C_j to which it is the most similar. The similarity is defined as

$$d(x_i, z_j) = \sqrt{\sum_{q=1}^{Q} (x_{iq} - z_{jq})^2} \,. \tag{3}$$

where $j = 1, 2, \ldots, K$.
z_j is the center of the cluster C_j.
Next, calculate the new cluster centers using

$$z_j^* = \frac{1}{n_j} \sum_{x_i \in C_j} x_i \,. \tag{4}$$

Then replace the previous cluster centers z_j encoded in the chromosome by the new centers z_j^*.
B.2. Compute the fitness of chromosomes in the current population using:

$$f = 1/M. \tag{5}$$

$$M = \sum_{j=1}^{K} \sum_{x_i \in C_j} \left\| x_i - z_j \right\| . \tag{6}$$

C. Create a new population by repeating the following steps until there is an appropriate number of chromosomes in the new population:
   C.1. Select two parent chromosomes from a current population using the roulette wheel technique.
   C.2. With a crossover probability, cross over the parent chromosomes to form the new offspring.
   C.3. Perform the mutation on the new offspring as follows:
   - Randomly select mutation positions.
   - With a mutation probability, mutate the value of the selected positions by using the following equation:

$$v^{t+1} = \begin{cases} v^t \pm 2*\delta*v^t & : \text{if } v^t \neq 0 \\ v^t \pm 2*\delta & : \text{if } v^t = 0 \end{cases}. \quad (7)$$

   where v^t is the current value of the selected gene.
   δ is a small random number in the range of 0 to 1.
   C.4. Place the new offspring into a new population.
D. If a predetermined number of iteration is reached or the end condition is satisfied, stop the loop and return the best chromosome in the current population. If not, go to step B.

4 The Proposed Mutation Operator

This paper proposes an extension to the original GA-clustering algorithm (proposed by Maulik and Bandyopadhyay) by introducing a new way to mutate the chromosome. Instead of deciding randomly to add or subtract a small random number to/from the original value of the selected position, the historical information of each chromosome is taken into consideration when mutating the chromosome. In this research, the histories of the chromosome are divided into 2 groups: "good" and "bad." The good consists of all previous stages of the chromosome that have higher fitness value than the current stage, while the bad is composed of all previous stages of the chromosome that have lower fitness value than the current stage. The formula for the proposed mutation operator is as follows:

$$z^{t+1} = z^t + \beta_1 \sum_{i \in g} w_i (z_i - z^t) + \beta_2 \sum_{k \in b} w_k (z_k - z^t). \quad (8)$$

$$w_i = \frac{f_i}{\sum_{j \in g} f_j}. \quad (9)$$

$$w_k = \frac{f_k}{\sum_{j \in b} f_j}. \quad (10)$$

where z^l is the current value of the selected cluster center.

β_1 and β_2 are the weight coefficients for "good" and "bad" groups respectively.

z_i and z_k are the past values of the selected cluster center.

5 Experimental Results

In this paper, the performance of the proposed approach is compared to the original GA-clustering algorithm and the k-means algorithm. In order to compare the three algorithms, the experiments are conducted using 6 benchmark data sets: "Pima Indians diabetes," "heart disease," "sonar, mines vs. rocks," "satellite image," "vowel recognition," and "letter image recognition." For each data set, one hundred experiments are performed for each algorithm. The performance is measured in terms of the clustering metric as defined in equation 6 and the average time to converge. The experiments are conducted on a 2.8 GHz Pentium IV PC with 1 gigabyte of main memory. Brief descriptions of the data sets are given below:

1. The first data set is the Pima Indians diabetes database. It is publicly available from the UCI machine learning database repository [3]. The problem is to predict whether a patient would test positive (1) or negative (0) for diabetes according to World Health Organization criteria. All patients represented in the data set are females at least 21 years old of Pima Indian heritage living near Phoenix, Arizona, USA. This database contains 768 examples. Each example is described by 8 numerical attributes and belongs to one of two classes (0 or 1). There are 500 examples of class 0 and 268 examples of class 1. Therefore, the value of K is chosen to be 2 for this database.
2. The second data set is the heart disease problem. It is retrieved from Statlog Project Datasets [4]. The problem concerns the prediction of the absence (1) or presence (2) of heart disease given the results of various medical tests carried out on a patient. This data set contains 13 attributes and 270 records. There are 150 records of class 1 and 120 records of class 2. The value of K used for this data set is 2.
3. The third data set is the Sonar, Mines vs. Rocks [3]. The task is to discriminate between sonar signals bounced off a metal cylinder and those bounced off a roughly cylindrical rock. This data set contains 111 patterns obtained by bouncing sonar signals off a metal cylinder at various angles and under various conditions and 97 patterns obtained from rocks under similar conditions. Sixty numerical attributes in the range 0.0 to 1.0 are used to predict the output class (mine or rock). Therefore, the value of K is chosen to be 2 for this data set.
4. The fourth data set is the satellite image data. This data is retrieved from Statlog Project Datasets [4]. This problem concerns the classification of 6 satellite images. Thirty-six numerical attributes are used to predict the output class. There are 6 classes (red soil = 1, cotton crop = 2, grey soil = 3, damp grey soil = 4, soil with vegetation stubble = 5, and very damp grey soil = 7). The value of K is therefore chosen to be 6 for this data set.
5. The fifth is the vowel recognition problem. The vowel data used in this problem was originally collected by Deterding [5]. who recorded examples of eleven steady state vowels (hid, hId, hEd, hAd, hYd, had, hOd, hod, hUd, hud, and hed) of British English spoken by fifteen speakers. Each vowel was uttered six times

by each of the fifteen speakers. Therefore, the whole data set contains 990 examples. The data set consists of 10 inputs, which are obtained by low pass filtering the speech signals at 4.7 kHz; digitizing to 12 bits with a 10 kHz sampling rate; extracting the twelfth order linear predictive coefficient (LPC) from six 512 sample Hamming windowed segments; and then using the reflection coefficients to calculate 10 log-area ratio (LAR) parameters. The value of K used for this data set is 11.

6. The sixth data set is the letter image recognition data. The data is taken from the UCI machine learning database repository [3]. Its task is to identify each of the character images as one of the 26 capital letters in the English alphabet. This data set has 20000 instances. Each instance is described by 16 numerical attributes. Therefore, the value of K is chosen to be 26 for this data set.

Tables 1 to 6 summarize the results obtained from the three clustering algorithms for the above data sets. The average time to converge of the K-means algorithm is the lowest among the three algorithms. The proposed algorithm comes second, leaving the original GA-clustering algorithm far behind. However, when considering the clustering metric, the proposed algorithm comes out to be the best among the compared algorithms. For the Pima Indians diabetes database, the heart disease problem, and the satellite image data, the proposed algorithm comes first in the competition, while the K-means and the GA-clustering algorithms tie for second place. For the sonar, mines vs. rocks data, the K-means and the proposed algorithms tie for first place. The vowel recognition problem is the only data for which the proposed algorithm does not win the competition; it comes second. For the letter image recognition data, the proposed algorithm attains the best value of 106944.2, in comparison to 107060.3 of the K-means algorithm and 107103.2 of the GA-clustering algorithm.

Table 1. Results for the Pima Indians diabetes database when K = 2

	Clustering Metric	Time (second)
K-means Algorithm	52072.24	0.0514
GA-clustering Algorithm	52072.24	1.3231
Proposed Algorithm	49388.21	0.4447

Table 2. Results for the heart disease problem when K = 2

	Clustering Metric	Time (second)
K-means Algorithm	10695.80	0.0334
GA-clustering Algorithm	10695.80	0.6539
Proposed Algorithm	10686.33	0.2238

Table 3. Results for the sonar, mines vs. rocks data when K = 2

	Clustering Metric	Time (second)
K-means Algorithm	234.7671	0.0394
GA-clustering Algorithm	235.1560	4.7724
Proposed Algorithm	234.7671	0.3067

Table 4. Results for the satellite image data when K = 6

	Clustering Metric	Time (second)
K-means Algorithm	291939.7	1.8440
GA-clustering Algorithm	291939.7	164.5408
Proposed Algorithm	291684.2	11.3747

Table 5. Results for the vowel recognition problem when K = 11

	Clustering Metric	Time (second)
K-means Algorithm	1342.710	0.2050
GA-clustering Algorithm	1343.231	274.3829
Proposed Algorithm	1343.065	1.5059

Table 6. Results for the letter image recognition data when K = 26

	Clustering Metric	Time (second)
K-means Algorithm	107060.3	33.2843
GA-clustering Algorithm	107103.2	2504.2886
Proposed Algorithm	106944.2	172.5785

References

1. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, San Francisco (2001)
2. Maulik, U., Bandyopadhyay, S.: Genetic Algorithm-based Clustering Technique. Pattern Recognition, Vol. 33 (2000) 1455-1465
3. Blake, C.L., Merz, C.J.: UCI Repository of Machine Learning Databases: http://www.ics.uci.edu/~mlearn/MLRepository.html. University of California, Department of Information and Computer Science. (1998)
4. Statlog Project Datasets: http://www.liacc.up.pt/ML/statlog/datasets/heart/. (2003)
5. Deterding, D.H.: Speaker Normalization for Automatic Speech Recognition. Ph.D. Dissertation. (1989)

1. A. Thammano and **U. Kakulphimp, "Genetic Algorithm-based Clustering and Its New Mutation Operator,"** 2006 International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC 2006), pp. 829–832, Chiang Mai, Thailand, July 10–13, 2006.

MENU
Home
Call for Paper
Committee
Keynote Speaker
Special Sessions
Login
Paper Submission
Author's Kit
Schedule
Registration
Hotel
Travel Infomation
Sightseeing
Contact Us

Last Update:
14 March 2006.
with total
032507
Hits.

ITC-CSCC2006 Chiang Mai, Thailand

Annoucement

- As of 30 May 2006, The Deadline for Early Registration Online has been extended to 2 June 2006. NEW!
- As of 28 May 2006, Formal Acceptance Letter is now ready for download, please click here to login and download your own letter. NEW!
- As of 26 May 2006, Add more information on Payment Method(Bank Transfer) in the Registration Menu. NEW!
- As of 14 May 2006, The results from the Technical Program Committee for you paper(s) have been updated. Please login using your UserID and Password to see the result, download necessary file and/or upload your camera-ready menuscript!!!. The official e-mail together with your schedule and presentation type will be send directly to your e-mail soon. Accepted paper, please visit Author's Kit page for more information on prepare the camera-ready manuscript and so on.
- Authors of the accepted papers are encouraged to submit full-length manuscripts to ECTI transactions, IEICE Transactions, or IEEK JSTS (Journal of Semiconductor Technology and Science). Papers are subject to the standard reviewing procedures of the ECTI transactions, the IEICE Transactions, or the IEEK JSTS

Author's Schedule

Deadline for proposal of special session:	4 March 2006
Deadline for paper submission:	14 April 2006
Notification of acceptance:	14 May 2006
Deadline for camera-ready manuscripts submission:	30 May 2006
Deadline for early registration:	2 June 2006

Sponsors

NECTEC
National Electronics and Computer Technology Center,Thailand

The Institute of Electronics Engineers of Korea (IEEK), Korea

The Institute of Electronics, Information and Communication Engineers (IEICE), Japan

The Electrical Engineering/Electronics, Computer, Telecommunications and Information Association(ECTI), Thailand

In association with
IEEE Thailand Section

Applying Historical Information to Improve Genetic Algorithm-based Clustering

Arit Thammano* and Uraiwan Kakulphimp**

Computational Intelligence Laboratory
Faculty of Information Technology
King Mongkut's Institute of Technology Ladkrabang
Bangkok, 10520 Thailand
e-mail: arit@it.kmitl.ac.th* and ukakulphimp@hotmail.com**

ABSTRACT

This paper proposes a new mutation operator. The proposed mutation operator consists of 2 components: a random component and a deterministic component. By combining a deterministic component with a random component, the chromosomes can perform both directional and randomized search simultaneously. The superiority of the proposed approach over the original genetic algorithm and K-means algorithm is demonstrated by using 7 benchmark data sets.

Keywords: Clustering, Data Mining, Genetic Algorithm, Mutation Operator

1. INTRODUCTION

Cluster analysis is a method for grouping a set of data objects into K clusters (C_1, C_2, ..., C_K) so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters [1]. Clustering is an unsupervised classification method, where the class label of each data object is not known. In clustering the data set X, the following conditions must be satisfied:

(1) None of the clusters is empty ($C_i \neq \varnothing$: i = 1, 2, ..., K).
(2) Every data object must belongs to exactly one cluster

$$(C_i \cap C_j = \varnothing: i \neq j \text{ and } \bigcup_{i=1}^{K} C_i = X).$$

Among all existing clustering algorithms, K-means algorithm is the most widely known; it is rather simple and remarkably effective. However, the K-means algorithm may get stuck at suboptimal solutions, depending on the choice of the initial cluster centers. Maulik and Bandyopadhyay [2] therefore have developed the GA-clustering algorithm, which integrates the capability of the genetic algorithm in avoiding local optima with the K-means algorithm. Their results show that the GA-clustering algorithm provides a performance that is significantly superior to that of the K-means algorithm.

The objective of this paper is to propose a new way to mutate the chromosome. The performance of the original GA-clustering algorithm is further improved when used in conjunction with this new mutation operator.

Following this introduction, section 2 briefly describes the original GA-clustering algorithm. The proposed mutation operator is presented in section 3. In section 4, the experimental results are demonstrated and discussed. Finally, section 5 is the conclusion.

2. GA-CLUSTERING ALGORITHM

A description of the GA-clustering algorithm proposed by Maulik and Bandyopadhyay is given below.

A. Define the size (P) of the population. For each chromosome, randomly select K initial cluster centers (z_1, z_2, ..., z_K) from the data set and encode them into a chromosome. Therefore, each chromosome is structured as follows:

$$\text{Chromosome} = [z_{11}, ..., z_{1Q}, ..., z_{K1}, ..., z_{KQ}] \quad (1)$$

where Q is the dimension of each cluster center.

B. Evaluate the fitness of each chromosome in the population. The fitness computation process consists of two phases as follows:

B.1. Assign each data point x_i to the cluster C_j to which it is the most similar. The similarity is defined as

$$d(x_i, z_j) = \sqrt{\sum_{q=1}^{Q} (x_{iq} - z_{jq})^2} \quad (2)$$

where j = 1, 2, ..., K.
z_j is the center of the cluster C_j.
Next, calculate the new cluster centers using

$$z_j^* = \frac{1}{n_j} \sum_{x_i \in C_j} x_i \quad (3)$$

Then replace the previous cluster centers z_j encoded in the chromosome by the new centers z_j^*.

B.2. Compute the fitness of chromosomes in the current population as follows:

$$f = 1/M \quad (4)$$

$$M = \sum_{j=1}^{K} \sum_{x_i \in C_j} \left\| x_i - z_j \right\| \quad (5)$$

C. Create a new population by repeating the following steps until there is an appropriate number of chromosomes in the new population:
   C.1. Select two parent chromosomes from a current population using the roulette wheel technique.
   C.2. With a crossover probability, cross over the parent chromosomes to form the new offspring.
   C.3. Perform the mutation on the new offspring as follows:
   - Randomly select mutation positions.
   - With a mutation probability, mutate the value of the selected positions by using the following equation:

$$v^{t+1} = \begin{cases} v^t \pm 2 * \delta * v^t & : \text{if } v^t \neq 0 \\ v^t \pm 2 * \delta & : \text{if } v^t = 0 \end{cases} \quad (6)$$

   where v^t is the current value of the selected gene.

   δ is a small random number in the range of 0 to 1.

   C.4. Place the new offspring into a new population.

D. If a predetermined number of iteration is reached or the end condition is satisfied, stop the loop and return the best chromosome in the current population. If not, go to step B.

3. THE PROPOSED MUTATION OPERATOR

This paper proposes an extension to the original GA-clustering algorithm (proposed by Maulik and Bandyopadhyay) by introducing a new way to mutate the chromosome. In addition to the original mutation operator that randomly adds or subtracts a small random number to/from the original value of the selected position, the historical information of each chromosome is also taken into consideration when mutating the chromosome. In this research, the histories of a chromosome are divided into 2 groups: "good" and "bad." The "good" consists of all previous stages of the chromosome that have higher fitness value than the current stage, while the "bad" is composed of all previous stages of the chromosome that have lower fitness value than the current stage. The formula for the proposed mutation operator is therefore composed of 2 components, which are a random component and a deterministic component, as follows:

$$z^{t+1} = z^t + \Delta z^t \quad (7)$$

Δz^t = random component + deterministic component (8)

$$\Delta z^t = \left[\alpha \delta z^t\right] + \left[\beta_1 \sum_{i \in g} w_i (z_i - z^t) - \beta_2 \sum_{k \in b} w_k (z_k - z^t)\right] \quad (9)$$

$$w_i = \frac{f_i}{\sum_{j \in g} f_j} \quad (10)$$

$$w_k = \frac{f_k}{\sum_{j \in b} f_j} \quad (11)$$

where z^t is the current value of the selected cluster center.

$\alpha = 0.1 - \beta_1 - \beta_2$. It is the weight coefficient for the random component.

δ is a small random number in the range of -1 to 1.

β_1 and β_2 are the weight coefficients for "good" and "bad" groups respectively. In this research, they are initially set to 0.05. However, if the highest fitness value of the current population is equal to that of the last generation, the values of both β_1 and β_2 are decreased by a predefined amount. As long as there is no improvement in the fitness value of the current population, the values of β_1 and β_2 are continuously decreased. In doing this, the algorithm will get more action from the random component and less action from the deterministic component.

z_i and z_k are the past values of the selected cluster center.

g represents "good" group.

b represents "bad" group.

4. THE EXPERIMENTAL RESULTS

In this paper, the performance of the proposed approach is compared to the original GA-clustering algorithm and the k-means algorithm. In order to compare the three algorithms, the experiments were conducted using 7 benchmark data sets: "iris," "Indian Telugu vowel," "ionosphere," "image segmentation," "Pima Indians diabetes," "heart disease," and "sonar, mines vs. rocks." For each data set, one thousand experiments are performed for each algorithm. The performance is measured in terms of the clustering metric as defined in equation 5. The experiments were conducted on a 2.8 GHz Pentium IV PC with 1 gigabyte of main memory. Brief descriptions of the data sets are given below:

1. The first data set is the well-known iris data [3]. This data set contains 150 records. The sepal length, sepal width, petal length, and petal width of 150 iris flowers from 3 species (Iris-setosa, Iris-versicolor, and Iris-virginica) are measured in centimeters, and are used as the input of the problem. Therefore, the value of K is chosen to be 3 for this data set.
2. The second is the Indian Telugu vowel data [4]. This data set consists of 871 patterns. There are six overlapping vowel classes (1 to 6) and three input features (the first, second, and third vowel formant frequencies). The value of K used for this data set is 6.
3. The third data set is the ionosphere data created by the Space Physics Group at Johns Hopkins University. The data is taken from the UCI machine learning database repository [5]. It contains the radar data collected by a system in Goose Bay, Labrador. This data set has 351 instances. Each instance is described

by 34 continuous attributes and belongs to one of two classes ("good" or "bad"). Therefore, the value of K is chosen to be 2 for this data set.

4. The fourth data set is the image segmentation database. This data is retrieved from Statlog Project Datasets [6]. Nineteen continuous attributes are used to predict the output class. There are 7 classes (brick face = 1, sky = 2, foliage = 3, cement = 4, window = 5, path = 6, and grass = 7) with 330 examples per class. The value of K is therefore chosen to be 7 for this data set.
5. The fifth data set is the Pima Indians diabetes database. It is publicly available from the UCI machine learning database repository [5]. The problem is to predict whether a patient would test positive (1) or negative (0) for diabetes according to World Health Organization criteria. All patients represented in the data set are females at least 21 years old of Pima Indian heritage living near Phoenix, Arizona, USA. This database contains 768 examples. Each example is described by 8 numerical attributes and belongs to one of two classes (0 or 1). There are 500 examples of class 0 and 268 examples of class 1. Therefore, the value of K is chosen to be 2 for this database.
6. The sixth data set is the heart disease problem. It is retrieved from Statlog Project Datasets [6]. The problem concerns the prediction of the absence (1) or presence (2) of heart disease given the results of various medical tests carried out on a patient. This data set contains 13 attributes and 270 records. There are 150 records of class 1 and 120 records of class 2. The value of K used for this data set is 2.
7. The seventh data set is the Sonar, Mines vs. Rocks [5]. The task is to discriminate between sonar signals bounced off a metal cylinder and those bounced off a roughly cylindrical rock. This data set contains 111 patterns obtained by bouncing sonar signals off a metal cylinder at various angles and under various conditions and 97 patterns obtained from rocks under similar conditions. Sixty numerical attributes in the range 0.0 to 1.0 are used to predict the output class (mine or rock). Therefore, the value of K is chosen to be 2 for this data set.

Table 1 to 7 summarize the results obtained from the three clustering algorithms for the above data sets. The proposed algorithm comes out to be the best among the compared algorithms. For the iris data, the ionosphere data, the Pima Indians diabetes database, the heart disease problem and the sonar, mines vs. rocks data, the proposed algorithm came first in the competition, while the K-means and the GA-clustering algorithms tied for second place. For the Indian Telugu vowel data, the proposed algorithm took first place; the GA-clustering algorithm and the K-means algorithm got second place and third place respectively. For the image segmentation data, the proposed algorithm attained the best value of 146277.7, in comparison to 148154.9 of the K-means algorithm and 148300.8 of the GA-clustering algorithm.

Table 1: *Results for the Iris Data When K = 3*

	Clustering Metric
K-means Algorithm	97.32592
GA-clustering Algorithm	97.32592
Proposed Algorithm	96.71142

Table 2: *Results for the Indian Telugu Vowel Data When K = 6*

	Clustering Metric
K-means Algorithm	149369.6
GA-clustering Algorithm	149362.8
Proposed Algorithm	149107.7

Table 3: *Results for the Ionosphere Data When K = 2*

	Clustering Metric
K-means Algorithm	796.3271
GA-clustering Algorithm	796.3271
Proposed Algorithm	795.1641

Table 4: *Results for the Image Segmentation Database When K = 7*

	Clustering Metric
K-means Algorithm	148154.9
GA-clustering Algorithm	148300.8
Proposed Algorithm	146277.7

Table 5: *Results for the Pima Indians Diabetes Database When K = 2*

	Clustering Metric
K-means Algorithm	52072.24
GA-clustering Algorithm	52072.24
Proposed Algorithm	47697.41

Table 6: *Results for the Heart Disease Problem When K = 2*

	Clustering Metric
K-means Algorithm	10695.80
GA-clustering Algorithm	10695.80
Proposed Algorithm	10624.37

Table 7: *Results for the Sonar, Mines vs. Rocks Data When K = 2*

	Clustering Metric
K-means Algorithm	234.7671
GA-clustering Algorithm	234.7671
Proposed Algorithm	234.1071

5. CONCLUSION

This paper proposes a new mutation operator. By combining a deterministic component with a random component, the chromosomes can adapt itself in a more reasonable manner. All of the experiments support that the performance of the genetic algorithm is further

improved when used in conjunction with this new mutation operator.

6. REFERENCES

[1] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, 2001.

[2] U. Maulik and S. Bandyopadhyay, "Genetic Algorithm-based Clustering Technique," *Pattern Recognition*, Vol. 33, pp. 1455-1465, 2000.

[3] R. A. Fisher, "The Use of Multiple Measurements in Taxonomic Problems," *Annual Eugenics*, Vol. 7, pp. 179-188, 1936.

[4] S. K. Pal and D. Dutta Majumder, "Fuzzy sets and decision making approaches in vowel and speaker recognition," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 7, pp. 625-629, 1977.

[5] C. L. Blake and C. J. Merz, *UCI Repository of Machine Learning Databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]*, University of California, Department of Information and Computer Science, 1998.

[6] Statlog Project Datasets, Retrieved March 28, 2003 From http://www.liacc.up.pt/ML/statlog/datasets/.

ประวัติผู้เขียน

นางสาวอุไรวรรณ กะกุลพิมพ์ เกิดเมื่อวันที่ 25 มิถุนายน พ.ศ.2521 ที่จังหวัดหนองคาย สำเร็จการศึกษาปริญญาตรีวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์ จากภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยบูรพา ในปีการศึกษา 2543 และเข้าศึกษาต่อในระดับปริญญาโท หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2544