



ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)

ปริญญา

วิศวกรรมคอมพิวเตอร์

วิศวกรรมคอมพิวเตอร์

สาขา

ภาควิชา

เรื่อง การค้นหารูปแบบช่วงของข้อมูลโดยใช้คลาสเพื่อลดปริมาณช่วงของรูปแบบที่ได้

DCR: Discretization Using Class Information to Reduce Number of Intervals

นามผู้วิจัย นายปรัชญา ผ่องอักษร

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(รองศาสตราจารย์กฤษณะ ไวยมัย, Ph.D.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(ผู้ช่วยศาสตราจารย์ยอดเยี่ยม ทิพย์สุวรรณ, Ph.D.)

หัวหน้าภาควิชา

(ผู้ช่วยศาสตราจารย์เข็มะทัต วิภาตะวนิช, Ph.D.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

(รองศาสตราจารย์กัญญา ชีระกุล, D.Agr.)

คณบดีบัณฑิตวิทยาลัย

วันที่ เดือน พ.ศ.

วิทยานิพนธ์

เรื่อง

การค้นหารูปแบบช่วงของข้อมูลโดยใช้คลาสเพื่อลดปริมาณช่วงของรูปแบบที่ได้

DCR: Discretization Using Class Information to Reduce Number of Intervals

โดย

นายปรัชญา ผ่องอักษร

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)

พ.ศ. 2552

ปรัชญา ผ่องอักษร 2552: การค้นหารูปแบบช่วงของข้อมูลโดยใช้คลาสเพื่อลดปริมาณ
ช่วงของรูปแบบที่ได้ ปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)
สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ อาจารย์ที่ปรึกษา
วิทยานิพนธ์หลัก: รองศาสตราจารย์กฤษณะ ไวยมัย, Ph.D. 57 หน้า

การแบ่งช่วงข้อมูล เป็นเทคนิคหนึ่งในการเตรียมข้อมูล ใช้สำหรับแปลงข้อมูลที่เป็นค่า
ต่อเนื่อง ให้เป็นค่าไม่ต่อเนื่อง โดยประโยชน์จากการนำข้อมูลที่เป็นค่าไม่ต่อเนื่องไปใช้นั้น
นอกจากจะช่วยให้ผู้ใช้เข้าใจข้อมูลได้ง่ายขึ้นแล้ว ยังช่วยลดเวลาในการนำไปใช้งาน อีกทั้งยังช่วย
เพิ่มความแม่นยำ และประสิทธิภาพเมื่อนำไปใช้ในการจำแนกประเภทอีกด้วย งานวิจัยนี้ได้
นำเสนออัลกอริทึม DCR (Discretization using Class Information to Reduce Number of
Intervals) ซึ่งเป็นอัลกอริทึมในการค้นหารูปแบบช่วง ที่ได้นำประโยชน์จากความสัมพันธ์ระหว่าง
คลาสกับคุณลักษณะต่างๆ ของฐานข้อมูล และการจัดลำดับของการแบ่งช่วงข้อมูลตามค่าเกน
สารสนเทศมาใช้ ซึ่งจำนวนช่วงของรูปแบบที่ได้นั้น จะถูกลดจำนวนลงโดยกระบวนการตัด
ระเบียบของฐานข้อมูลออกในแต่ละขั้นตอนการแบ่งช่วง จากผลการทดลองจะพบว่า อัลกอริทึม
DCR นี้ สามารถค้นหารูปแบบช่วงของข้อมูลที่มีจำนวนช่วงน้อย แต่ให้ความแม่นยำในการ
นำไปใช้จำแนกคลาสที่สูงได้ โดยใช้เวลาในการทำงานที่น้อยกว่า เมื่อเทียบกับเทคนิคอื่นๆ ที่
ใกล้เคียงกัน

Prachya Pongaksorn 2009: DCR: Discretization Using Class Information to Reduce Number of Intervals. Master of Engineering (Computer Engineering), Major Field: Computer Engineering, Department of Computer Engineering. Thesis Advisor: Associate Professor Kitsana Waiyamai, Ph.D. 57 pages.

Discretization techniques for data set features have received increasing research attention. Results using discretized features are usually more compact, shorter, and accurate than using continuous values. In this paper, an algorithm called Discretization using Class information to Reduce number of intervals (DCR) is proposed. DCR uses both class information and order between attributes to determine the discretization scheme with minimum number of intervals. Attribute discretization order is determined based on information gain of each attribute with respect to the class attribute. The number of intervals is reduced by deleting training data at each step of attribute discretization. Experiments are performed to compare the predictive accuracy and execution time of this algorithm with several well-known algorithms. Results show that discretized features generated by the DCR algorithm contain a smaller number of intervals than other supervised algorithms using less execution time, and the predictive accuracy is as high or higher.

Student's signature

Thesis Advisor's signature

____ / ____ / ____

กิตติกรรมประกาศ

ข้าพเจ้าขอกราบขอบพระคุณ รองศาสตราจารย์ ดร.กฤษณะ ไวยมัย ประธานกรรมการที่ปรึกษา ที่ได้ช่วยเหลือในการวางแผนงานวิจัยในวิทยานิพนธ์ฉบับนี้ ตลอดจนการให้คำปรึกษา พร้อมทั้งให้แนวทางและความรู้เกี่ยวกับทฤษฎีต่างๆ มากมายในการทำวิจัย รวมถึงข้อเสนอแนะที่เป็นประโยชน์ต่อการทำวิทยานิพนธ์ฉบับนี้ รวมถึงการตรวจแก้ไขข้อบกพร่องต่างๆ และขอกราบขอบพระคุณผู้ช่วยศาสตราจารย์ ดร.ยอดเยี่ยม ทิพย์สุวรรณ กรรมการที่ปรึกษาวิชาเอก ผู้ช่วยศาสตราจารย์ ดร.จิตรัทสน์ ฝักเจริญผล และผู้ช่วยศาสตราจารย์ ดร.จิรยุทธ ไชยจารุณนิช ที่กรุณาให้คำปรึกษาแนะนำและได้ให้ข้อเสนอแนะดีๆ ในการทำวิทยานิพนธ์ให้สำเร็จลุล่วงไปด้วยดี

ขอขอบคุณอาจารย์ ธนาวิทย์ รักธรรมานนท์ ที่ให้คำปรึกษาที่ดีมาโดยตลอด ไม่ว่าจะเป็นทิศทางของงานวิจัย ความรู้ต่างๆ มากมาย และขอขอบคุณพี่ณภัทร มังคะจิตร พี่สาวินิ แสงสุริยันธ์ ที่คอยช่วยให้คำปรึกษาไม่ว่าจะเป็นเรื่องงานและเรื่องอื่นๆ มากมาย และสุดท้ายขอขอบคุณสมาชิกห้องปฏิบัติการ DAKDL คุณท่านที่คอยให้คำแนะนำและกำลังใจที่ดีเสมอมา

ขอขอบคุณพี่จู้ และพี่จอย เจ้าหน้าที่ธุรการ โครงการปริญญาโท ที่ช่วยเหลือในการประสานงาน และงานด้านเอกสารต่างๆ ให้งานเป็นไปอย่างสะดวกลุล่วงไปด้วยดี รวมถึงขอขอบคุณ เจ้าหน้าที่ธุรการภาควิชาวิศวกรรมคอมพิวเตอร์มหาวิทยาลัยเกษตรศาสตร์ทุกท่าน

คุณงามความดี หรือประโยชน์อันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ขออุทิศให้แก่ บิดา มารดา บุพการี และผู้มีพระคุณทุกท่าน

ปรัชญา ผ่องอักษร
กุมภาพันธ์ 2552

สารบัญ

	หน้า
สารบัญ	(1)
สารบัญตาราง	(2)
สารบัญภาพ	(3)
คำอธิบายสัญลักษณ์และคำย่อ	(4)
คำนำ	1
วัตถุประสงค์และขั้นตอนการวิจัย	3
การตรวจเอกสาร	4
อุปกรณ์และวิธีการ	28
อุปกรณ์	28
วิธีการ	28
ผลและวิจารณ์	42
ผล	42
วิจารณ์	47
สรุปและข้อเสนอแนะ	51
สรุป	51
ข้อเสนอแนะ	51
เอกสารและสิ่งอ้างอิง	53
ประวัติการศึกษา และการทำงาน	57

สารบัญตาราง

ตารางที่		หน้า
1	ตัวอย่างข้อมูลคุณลักษณะที่เป็นค่าต่อเนื่อง	13
2	Contingency Tables สำหรับช่วง (21.5, 29.5] และ (29.5, 39.5]	17
3	Contingency Tables สำหรับช่วง (12, 21.5] และ (21.5, 72]	17
4	Critical values of chi-square	17
5	ตาราง quanta-matrix	20
6	ตาราง quanta-matrix เมื่อเซต $D = \{[12, 61], (61, 80]\}$	23
7	รูปแบบการกระจายตัวของแต่ละคลาสแบบที่ 1	26
8	รูปแบบการกระจายตัวของแต่ละคลาสแบบที่ 2	26
9	ตารางเปรียบเทียบคุณลักษณะของแต่ละอัลกอริทึม	27
10	ตัวอย่างฐานข้อมูลที่ใช้ในการค้นหารูปแบบช่วง	35
11	ผลลัพธ์ที่ได้เมื่อทำการแบ่งช่วงข้อมูลของคุณลักษณะที่ 3 ด้วยอัลกอริทึม DCR	39
12	ตารางเปรียบเทียบคุณลักษณะของอัลกอริทึม DCR กับอัลกอริทึมอื่นๆ	41
13	รายละเอียดของฐานข้อมูลแต่ละชุดจาก UCI Repository of Machine Learning Database	43
14	เปรียบเทียบจำนวนช่วงที่ได้จากการแบ่งช่วง	44
15	เปรียบเทียบเวลาที่ใช้ในการค้นหารูปแบบช่วงของข้อมูล (วินาที)	45
16	เปรียบเทียบความแม่นยำของรูปแบบช่วงที่ได้เมื่อนำไปใช้ในการจำแนกคลาส	46

สารบัญภาพ

ภาพที่		หน้า
1	ขั้นตอนของเทคนิคการจำแนกประเภทข้อมูล	5
2	โครงสร้างต้นไม้การตัดสินใจ	6
3	กระบวนการโดยทั่วไปสำหรับการแบ่งช่วงข้อมูล	10
4	การจำแนกประเภทอัลกอริทึมของการแบ่งช่วงข้อมูล	11
5	ผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม Equal width เมื่อ $k=3$	14
6	ผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม Equal frequency เมื่อ $k=3$	15
7	ผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม ChiMerge (Significance level 5%)	18
8	ขั้นตอนการทำงานของอัลกอริทึม CAIM	22
9	การสร้างเซตของจุดตัดที่เป็นไปได้ทั้งหมด (B)	23
10	ผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม CAIM	24
11	จุดตัดที่ไม่ควรนำมาพิจารณาในการหารูปแบบช่วง	25
12	วิธีการตัดระเบียบสำหรับอัลกอริทึม DCR	31
13	รหัสเทียมของอัลกอริทึม DCR	33
14	การเรียงลำดับค่าของข้อมูลที่เป็นค่าต่อเนื่อง	36
15	การหาเซตของจุดตัดสำคัญที่ต้องพิจารณา (EB) ของอัลกอริทึม DCR	37
16	ตัวอย่างการคำนวณหาค่า DCR เมื่อรูปแบบช่วงเป็น $\{[9, 20], (20, 95]\}$	38
17	เปรียบเทียบ Average score ของจำนวนช่วงที่ได้จากการแบ่งช่วง	47
18	เปรียบเทียบ Average score ของเวลาที่ใช้ในการค้นหารูปแบบช่วงของข้อมูล	48
19	เปรียบเทียบ Average score ของความแม่นยำของรูปแบบช่วงที่ได้เมื่อนำไปใช้ในการจำแนกคลาส	49

คำอธิบายสัญลักษณ์และคำย่อ

ID3	=	Iterative Dichotomiser 3
CART	=	Classification and Regression Trees
WEKA	=	Waikato Environment for Knowledge Analysis
GNU	=	General Public License
CAIM	=	Class-Attribute Interdependence Maximization
EW	=	Equal Width
EF	=	Equal Frequency
DCR	=	Discretization using Class Information to Reduce Number of Intervals
EB	=	Essential Boundary Set

การค้นหารูปแบบช่วงของข้อมูลโดยใช้คลาสเพื่อลดปริมาณช่วงของรูปแบบที่ได้

DCR: Discretization Using Class Information to Reduce Number of Intervals

คำนำ

ปริมาณของข้อมูลที่เพิ่มขึ้นอย่างมหาศาลในปัจจุบันนี้ ทำให้มนุษย์ไม่สามารถดึงองค์ความรู้ (knowledge) ออกมาจากฐานข้อมูลขนาดใหญ่ (huge datasets) ได้ ดังนั้นเทคนิคทางด้านดาต้าไมนิง (data mining) จึงเข้ามามีบทบาทสำคัญในการวิเคราะห์ฐานข้อมูลดังกล่าว จากหลายๆ เทคนิคของดาต้าไมนิงนั้น การจำแนกประเภทข้อมูล (classification) ได้ถูกนำมาประยุกต์ใช้ในหลายๆ ด้าน เช่น การจำแนกประเภทบทความ การจำแนกประเภทรูปภาพ หรือการตรวจจับไวรัส เป็นต้น ซึ่งข้อมูลต่างๆ ที่อยู่ในฐานข้อมูลนั้น มักจะเป็นข้อมูลหลายๆ ประเภทผสมกัน หรือ ฐานข้อมูลแบบผสม (mixed-mode data) โดยเราสามารถแบ่งประเภทของข้อมูลออกได้เป็น 2 กลุ่มใหญ่ๆ คือ คุณลักษณะที่เป็นค่าต่อเนื่อง (continuous) เช่น อุณหภูมิ, ส่วนสูง, น้ำหนัก, อายุ และ คุณลักษณะที่เป็นค่าไม่ต่อเนื่อง (discrete) เช่น สี, เพศ เป็นต้น จากการศึกษางานวิจัยต่างๆ พบว่าเทคนิคการจำแนกประเภทบางเทคนิค (Cios and Kurgan, 2001; Clark and Niblett, 1989; Cost and Salzberg, 1993; Kaufman and Michalski, 1999) นั้นไม่สามารถรองรับข้อมูลที่เป็นค่าต่อเนื่องได้ หรือมีบางเทคนิคที่สามารถรองรับข้อมูลประเภทนี้ได้ แต่จะให้ประสิทธิภาพที่ดีขึ้น ถ้าใช้ข้อมูลที่เป็นค่าไม่ต่อเนื่อง (Catlett, 1991; Kerber, 1992) เพื่อแก้ปัญหาดังกล่าว จึงได้มีการนำเสนออัลกอริทึมในการแบ่งช่วงข้อมูล (discretization algorithm) ออกมาเป็นจำนวนมาก (Pfahring, 1995; Lee *et al.*, 2007; Ho and Scott, 1997; Holte, 1993; Dougherty *et al.*, 1995; Quinlan, 1993)

หลังจากเทคนิคการแบ่งช่วงข้อมูลได้เริ่มถูกนำเสนอขึ้น ก็ได้มีการพัฒนาอัลกอริทึมให้มีประสิทธิภาพมากยิ่งขึ้น ทั้งในด้านของเวลาในการทำงาน จำนวนช่วงที่เหมาะสมของข้อมูล และประสิทธิภาพของผลลัพธ์ที่ได้เมื่อนำไปใช้ในกระบวนการเรียนรู้ แต่เนื่องจากเทคนิคที่มีผู้นำเสนอมาขึ้นก็ยังคงมีจุดอ่อนอยู่ ทั้งในส่วนของเวลาที่ใช้ในการทำงาน และจำนวนช่วงของผลลัพธ์ที่มากเกินไปจนเกิดความจำเอน ซึ่งส่งผลให้ผลลัพธ์ที่ได้มีประสิทธิภาพไม่สูงมากนัก เมื่อนำไปใช้งาน

ในวิทยานิพนธ์เล่มนี้ได้นำเสนอวิธีการเพิ่มประสิทธิภาพของอัลกอริทึมการแบ่งช่วงข้อมูล โดยนำประโยชน์จากความสัมพันธ์ระหว่างคลาสกับข้อมูล และความสัมพันธ์ของแต่ละคุณลักษณะ

มาช่วยพิจารณาในการหารูปแบบช่วงของข้อมูล รวมถึงยังได้มีพัฒนาทางด้านเวลาที่ใช้ในการแบ่งช่วง โดยจะแบ่งช่วงข้อมูลเท่าที่จำเป็นเท่านั้น ซึ่งจะมีผลคือ ช่วยลดเวลาในการนำไปใช้ในกระบวนการเรียนรู้ และยังเป็นการคัดคุณลักษณะที่ไม่จำเป็นออกไปอีกด้วย

วัตถุประสงค์และขั้นตอนการวิจัย

วัตถุประสงค์ของการวิจัย

1. ศึกษาทฤษฎีที่มีอยู่ของเทคนิคการแบ่งช่วงข้อมูล และเทคนิคอื่นๆที่เกี่ยวข้อง เพื่อที่จะพัฒนาระบบได้อย่างเหมาะสมและมีประสิทธิภาพ
2. พัฒนาเทคนิคการแบ่งช่วงข้อมูลในส่วนของ
 - 2.1 การหาจำนวนช่วงของผลลัพธ์ที่เหมาะสม
 - 2.2 ความแม่นยำในการจำแนกประเภท เมื่อนำผลลัพธ์ที่ได้มาใช้
 - 2.3 เวลาที่ใช้ในการทำงานของระบบ

ขั้นตอนการวิจัย

1. ศึกษาทฤษฎีต่างๆ ของเทคนิคการแบ่งช่วงข้อมูล รวมถึงศึกษาทฤษฎีที่เกี่ยวข้อง เพื่อที่จะนำความรู้ที่ได้มาใช้ในการวิจัย
2. รวบรวมฐานข้อมูลมาตรฐานเพื่อที่จะนำมาใช้ในการทดสอบเพื่อศึกษาผลของการแบ่งช่วงข้อมูล และหาสาเหตุของปัญหาที่เกิดขึ้น
3. ศึกษาผลที่ได้จากการทดลอง เพื่อวิเคราะห์ปัญหาและรวบรวมข้อดีและข้อด้อยต่างๆ ของงานก่อนหน้า เพื่อนำมาเป็นข้อมูลในการพัฒนาเทคนิคและอัลกอริทึมในการแบ่งช่วงข้อมูล
4. พัฒนาเทคนิคการแบ่งช่วงข้อมูล
5. ทดสอบและวัดผลของเทคนิคการแบ่งช่วงข้อมูลที่พัฒนาขึ้น
6. สรุปผลการวิจัยและประโยชน์ที่ได้รับ

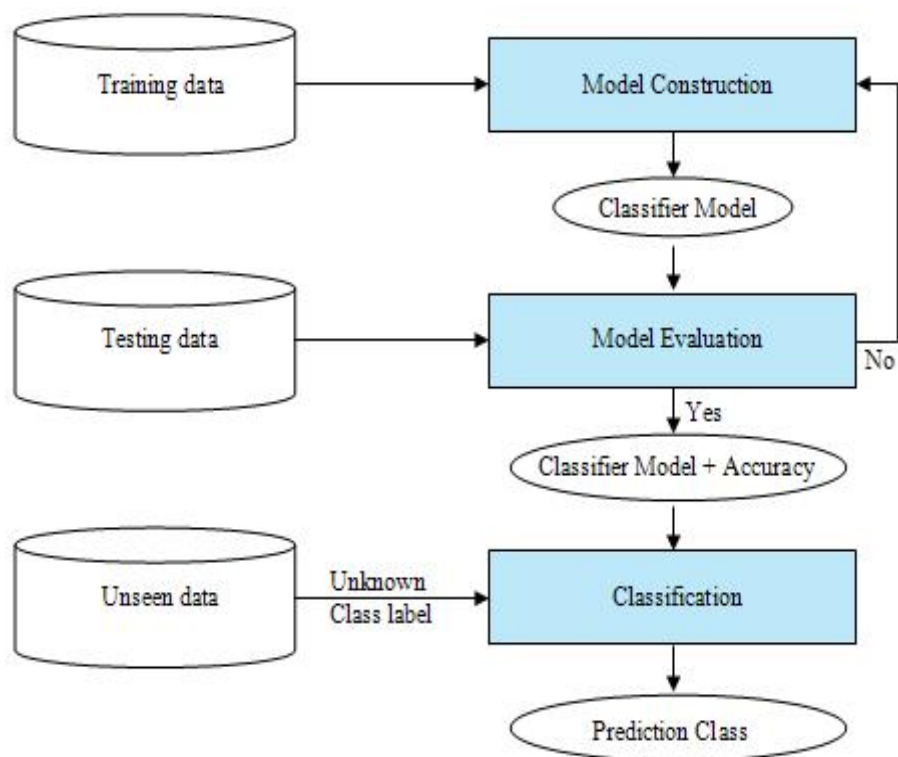
การตรวจเอกสาร

ความรู้พื้นฐานของเทคนิคดาต้าไมนิ่ง

1. การจำแนกประเภทข้อมูล (Data Classification)

การจำแนกประเภทข้อมูล (Data Classification) (กฤษณะ, 2550) เป็นหนึ่งในเทคนิคของ Data Mining ซึ่งทำหน้าที่สืบค้นความรู้เพื่อสรุปหาแบบจำลองหรือโมเดลของฐานข้อมูลนั้นๆ (Quinlan, 1993; Wang *et al.*, 2000) เพื่อใช้ในการทำนายข้อมูลใหม่ (unseen data) โดยเทคนิคนี้จะหาความสัมพันธ์ของข้อมูลจากฐานข้อมูลขนาดใหญ่ เพื่อนำมาสร้างโมเดลเพื่อใช้ในการจำแนกประเภทข้อมูล ซึ่งจะสามารถนำไปจำแนกประเภทข้อมูลใหม่ๆ ที่ยังไม่ทราบประเภทได้ (Unknown class label)

เนื่องจากการจำแนกประเภทข้อมูลเป็นเทคนิคแบบ Supervise learning นั่นคือ การจะสร้างโมเดลออกมาได้นั้นจะต้องทำการสอนระบบเสียก่อน ดังนั้นเราจำเป็นจะต้องทราบจำนวนคลาสปลายทาง (Class label) และจำนวนแอตทริบิวต์ (Attribute) ที่แน่นอน และส่วนของข้อมูลจะต้องแบ่งออกเป็นสองส่วน ส่วนหนึ่งใช้สอนระบบ (Training data) อีกส่วนหนึ่งใช้ทดสอบความแม่นยำของโมเดลที่ถูกสร้างออกมา (Testing data) โดยปกติสัดส่วนระหว่าง Training กับ Testing จะอยู่ที่ประมาณ 80 ต่อ 20 โดยในการที่จะสร้างโมเดลออกมาเพื่อใช้สำหรับทำนายข้อมูลได้นั้น จะต้องผ่านขั้นตอนดังต่อไปนี้ เริ่มจากการนำข้อมูลสอนระบบ (Training data) เข้ามาสู่กระบวนการสร้างโมเดลจำแนกประเภทข้อมูล (Model construction) เพื่อให้ได้โมเดลจำแนกประเภทข้อมูล (Classifier model) ออกมา และหลังจากได้โมเดลจำแนกประเภทข้อมูลแล้ว วิธีการทดสอบว่าโมเดลที่ถูกสร้างขึ้นมีความแม่นยำมากเพียงพอที่จะนำไปใช้ได้หรือไม่นั้น จะใช้ข้อมูลทดสอบระบบ หรือ Testing data เพื่อทดสอบความแม่นยำของโมเดลที่ถูกสร้างขึ้นมา (Model evaluation) ถ้าโมเดลที่สร้างขึ้นมีความแม่นยำไม่ผ่านเกณฑ์ที่ต้องการ ก็จะต้องกลับไปปรับปรุงในส่วน of กระบวนการสร้างโมเดลจำแนกประเภทข้อมูลเสียก่อน แต่ถ้าโมเดลที่สร้างขึ้นมีความแม่นยำผ่านเกณฑ์ที่ต้องการแล้ว ก็สามารถที่จะนำโมเดลที่สร้างมานั้นไปประยุกต์ใช้เพื่อทำนายประเภทข้อมูลใหม่ (Unseen data) ที่ไม่ทราบประเภทของข้อมูล (Unknown class label) ต่อไปได้ โดยภาพรวมขั้นตอนทั้งหมดของเทคนิคการจำแนกประเภทข้อมูลที่อธิบายไว้ข้างต้น สามารถดูได้จากภาพที่ 1

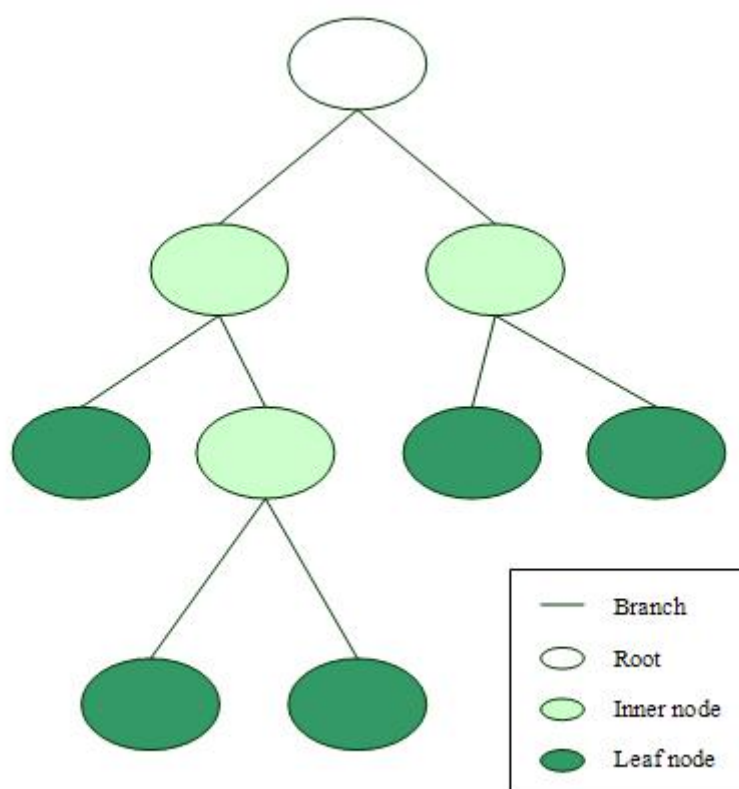


ภาพที่ 1 ขั้นตอนของเทคนิคการจำแนกประเภทข้อมูล

ที่มา: วีระพล (2549)

2. ต้นไม้การตัดสินใจ (Decision Tree)

ต้นไม้การตัดสินใจคือเทคนิคหนึ่งที่ใช้ในกระบวนการจำแนกประเภทข้อมูล ซึ่งต้นไม้การตัดสินใจจะถูกแสดงอยู่ในรูปของ flow-chart คล้ายกับโครงสร้างต้นไม้ (ภาพที่ 2) บัพภายใน (inner node) ของต้นไม้จะแสดงตัวแปร ส่วนกิ่ง (branch) จะแสดงค่าที่เป็นไปได้ของตัวแปร ส่วนบัพใบ (leaf node) จะแสดงคลาสของวัตถุ เส้นทางระหว่างบัพราก (root) ไปจนถึงบัพใบคือกฎ การจำแนกประเภทข้อมูล จะเริ่มจากบัพราก โดยทดสอบค่าตัวแปรของบัพ แล้วจึงตามกิ่งของต้นไม้ที่กำหนดค่า เพื่อไปยังบัพลูกถัดไป การทดสอบนี้จะกระทำไปจนกระทั่งเจอบัพใบซึ่งจะแสดงผลการทำนาย



ภาพที่ 2 โครงสร้างต้นไม้การตัดสินใจ

2.1 อัลกอริทึมที่ใช้ในการสร้างต้นไม้การตัดสินใจ

อัลกอริทึมที่นิยมใช้ในการสร้างต้นไม้การตัดสินใจในปัจจุบัน (ชลนิศา, 2550) คือ

2.1.1 ID3 คือวิธีการสร้างต้นไม้การตัดสินใจ โดยมีพื้นฐานจากเทคนิค Divide-and-Conquer ที่ใช้ในการสร้างต้นไม้ หรือที่เรียกว่า Top-down Induction พัฒนาโดย J. Ross Quinlan (1975) แห่งมหาวิทยาลัย Sydney, Australia การสร้างต้นไม้ด้วย ID3 ใช้หลักการของการใช้ Information Entropy ค่าที่วัดได้จะนำมาใช้ตัดสินใจเลือกตัวแปรในการสร้างเงื่อนไขในต้นไม้ตัดสินใจ

2.1.2 C4.5 เป็นอัลกอริทึมในการสร้างต้นไม้การตัดสินใจ พัฒนาโดย J. Ross Quinlan (1993) โดยนำเอา ID3 มาปรับปรุงให้มีความสามารถมากขึ้น ใช้วิธีการ Information Gain เพิ่มเติมการจัดการข้อมูลตัวเลข หรือข้อมูลที่เป็นค่าต่อเนื่อง, ข้อมูลที่ขาดหายไปและไม่สมบูรณ์ (Missing Values, Noisy Data) และการ Prune ด้วยการแทน Branch (กิ่ง) ที่ไม่ช่วยในการตัดสินใจด้วย Leaf Node ที่ตัดสินใจได้ดีกว่า

2.1.3 C5.0 คืออัลกอริทึมที่ J. Ross Quinlan พัฒนาเพิ่มจาก C4.5 มีความสามารถมากกว่า C4.5 ในด้าน ความเร็วในการประมวลผล การจัดการกับหน่วยความจำมีประสิทธิภาพมากขึ้น ต้นไม้การตัดสินใจมีขนาดเล็กลง ความถูกต้องของแบบจำลองต้นไม้มากขึ้น และมีการใช้ค่าถ่วงคูลน้ำหนักรของข้อมูล แต่สำหรับอัลกอริทึมนี้เป็นลักษณะ Commercial และ Closed-Source Product ที่ไม่ได้เปิดเผย Source Code ดังเช่น ID3, C4.5

2.1.4 CART (Classification and Regression Trees) คือต้นไม้การตัดสินใจด้านสถิติ เพื่อการทำนาย และจำแนกกลุ่มที่พัฒนาเพื่อสร้างต้นไม้ที่ใช้จัดการกับการจำแนกกลุ่มแบบตัวเลขได้ยิ่งขึ้น พัฒนาโดย Breiman *et al.* (1984) โดยอาศัยหลักการเดียวกับ C4.5 ในด้านการจัดการ Missing Values และมีการจัดการได้ดีกับข้อมูลหลายชนิดเช่น Floats, Integer และค่าที่ไม่ใช่ตัวเลข

2.2 เครื่องมือในการสร้างต้นไม้การตัดสินใจ

การวัดผลด้านความแม่นยำในการจำแนกประเภทนั้น ในงานวิจัยนี้ได้เลือกอัลกอริทึม C4.5 โดยทำการทดลองผ่าน WEKA (Waikato Environment for Knowledge Analysis) (Ian and Eibe, 2005) พัฒนาโดย University of Waikato New Zealand เพื่อเป็นเครื่องมือสำหรับการทำเหมืองข้อมูลโดยใช้หลักการของ Machine Learning มี GUI ที่สะดวกในการใช้งาน เรียกว่า Explore

และเป็น open source ภายใต้ GNU (General Public License) ทำงานบน Java VM มีฟังก์ชันการทำงานต่างๆ เช่น Pre-Processing, Evaluating Statistically, Visualizing Data Learning Algorithms ให้เลือกใช้ในปัญหาต่างๆ เช่น Classification, Regression, Clustering, Association Rules และวิธีการเรียนรู้ (Learning Methods) หรือเรียกว่า Classifiers ในงานวิจัยนี้เลือกใช้ J48 คืออัลกอริทึม C4.5 เป็นอัลกอริทึมเพื่อใช้ในการวัดผลด้านความแม่นยำในการจำแนกประเภทเมื่อนำผลลัพธ์ที่ได้จากการแบ่งช่วงข้อมูลของแต่ละอัลกอริทึมมาเป็นข้อมูลนำเข้าของระบบ (Input) และเนื่องจากอัลกอริทึมนี้มีความสามารถในการจัดการกับข้อมูลที่เป็นค่าต่อเนื่องได้ ดังนั้นงานวิจัยนี้จึงได้ทำการเปรียบเทียบทางด้านความแม่นยำในการจำแนกประเภทข้อมูลเมื่อมีข้อมูลนำเข้าของอัลกอริทึม C4.5 เป็นค่าต่อเนื่องอีกด้วย

2.3 ข้อดีของการจำแนกประเภทด้วยต้นไม้การตัดสินใจ

2.3.1 เป็นแบบจำลองที่ง่ายในการทำความเข้าใจ ทำการสร้างกฎ (Rule) แสดงความสัมพันธ์ของข้อมูลจากต้นไม้ได้ง่าย

2.3.2 สามารถทำนายหรือจำแนกกลุ่มได้ทั้งข้อมูลที่เป็นค่าต่อเนื่อง และค่าไม่ต่อเนื่อง

2.3.3 ในการสร้างแบบจำลอง จะมีการคัดตัวแปรที่ไม่มีผลในการสร้างต้นไม้ไม่ได้ โดยไม่กระทบการสร้างต้นไม้

2.3.4 แต่ละกิ่งของต้นไม้แสดงความสัมพันธ์ที่เกี่ยวของกันของข้อมูล

2.3.5 จัดการกับข้อมูลที่ไม่สมบูรณ์ที่ไม่มีผลต่อการสร้างต้นไม้ และเพื่อลดขนาดของต้นไม้

2.3.6 ใช้เวลาในการประมวลผลที่ไม่มากนัก

2.3.7 มีความแม่นยำในการจำแนกประเภทที่ดีใกล้เคียงกับเทคนิคอื่นๆ

2.4 ข้อจำกัดของการจำแนกประเภทด้วยแผนผังต้นไม้ตัดสินใจ

2.4.1 แบบจำลองต้นไม้ที่สร้างอาจมีขนาดใหญ่และมีความซับซ้อน ทำให้ยากต่อการเข้าใจ จากกฎที่สรุปได้จากแบบจำลองต้นไม้ตัดสินใจ

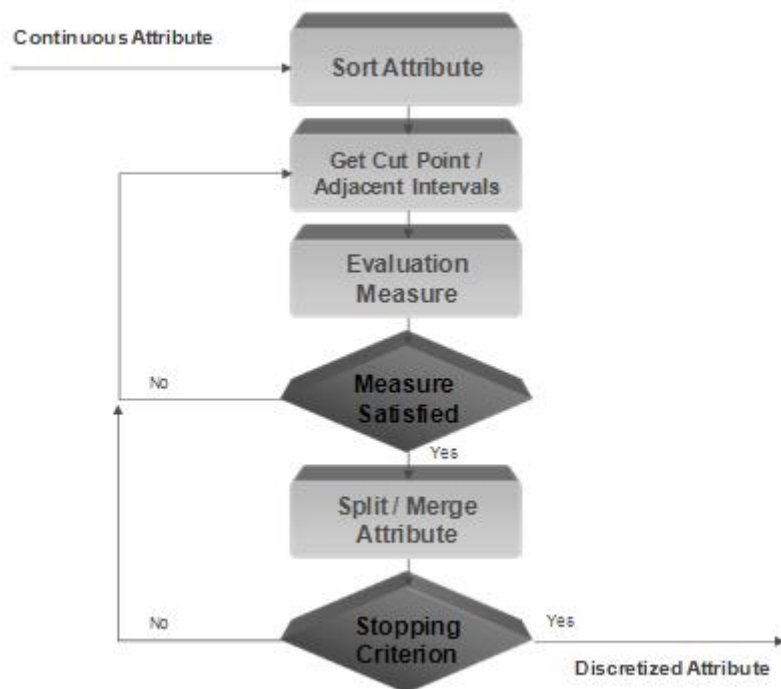
2.4.2 ไม่เหมาะในการจำแนกกลุ่มข้อมูลที่เป็นตัวเลข

2.4.3 Big O ในการสร้างแบบจำลองต้นไม้จาก Training Data ขึ้นอยู่กับขนาดของข้อมูล n อัลกอริทึมในการต้นไม้คือ $O(mn \log n) + O(n(\log n)^2)$ การเพิ่มขึ้นของข้อมูลและตัวแปรทำให้มีรูปแบบการเพิ่มเป็นแบบเชิงเส้น (Linear)

3. การแบ่งช่วงข้อมูล (Discretization)

การแบ่งช่วงข้อมูล คือ กระบวนการที่ใช้ในการแปลงข้อมูลของคุณลักษณะที่เป็นค่าต่อเนื่อง ให้เป็นค่าไม่ต่อเนื่อง ซึ่งมีจุดมุ่งหมายคือ ช่วยให้ผู้ใช้สามารถเข้าใจข้อมูลได้ง่ายขึ้น ช่วยลดเวลาในการนำไปใช้งาน อีกทั้งยังช่วยเพิ่มความแม่นยำ และประสิทธิภาพเมื่อนำไปใช้ในการจำแนกประเภท (Liu *et al.*, 2002) โดยใช้เวลาในการทำงานไม่มากนัก

เราสามารถจำแนกประเภทของการแบ่งช่วงข้อมูลได้ 5 วิธี คือ supervised กับ unsupervised, static กับ dynamic, global กับ local, top-down กับ bottom-up และ direct กับ incremental โดยการแบ่งช่วงแบบ supervised นั้น จะให้ประสิทธิภาพที่ดีกว่าแบบ unsupervised เพราะได้นำข้อมูลของคลาส (class) มาพิจารณาด้วย ต่อมาการแบ่งช่วงแบบ dynamic จะเป็นการแบ่งช่วงไปพร้อมๆกับการสร้างโมเดลการจำแนกประเภท (classifier model) ต่างจากแบบ static ที่ จะทำการแบ่งช่วงข้อมูลให้เสร็จก่อนที่จะนำข้อมูลไปใช้ ซึ่งจะทำให้เป็นอิสระต่ออัลกอริทึมในการจำแนกประเภท วิธีจัดการแบ่งช่วงข้อมูลแบบ global เป็นการหารูปแบบช่วง (discretization scheme) ที่ดีที่สุดจากรูปแบบช่วงทั้งหมดที่เป็นไปได้ ซึ่งเป็นวิธีที่ให้ประสิทธิภาพของรูปแบบช่วงสูง แต่จะใช้เวลาในการคำนวณที่มากกว่าแบบ local คือหารูปแบบช่วงที่ดีที่สุดเมื่อเทียบกับรูปแบบช่วงอื่นๆที่มีจำนวนช่วงเท่ากัน วิธีต่อการแบ่งช่วงแบบ bottom-up (merging) จะเริ่มต้นด้วยจุดตัด (cut-points) ทั้งหมดที่เป็นไปได้ แล้วจะค่อยๆทำการลบจุดตัดออกไปทีละจุดโดยการรวมช่วงที่ติดกันเข้าด้วยกัน ส่วนแบบ top-down (splitting) นั้น จะเริ่มต้นจากการไม่มีจุดตัด แล้วค่อยๆทำการเพิ่มจุดตัดทีละจุด ทำให้ใช้เวลาในการทำงานน้อยกว่า เพราะส่วนมากจำนวนช่วงที่ได้ในแต่ละคุณลักษณะจะมีค่าไม่มากนักเมื่อเทียบกับค่าต่อเนื่องที่เป็นไปได้ทั้งหมด วิธีสุดท้ายการแบ่งช่วงแบบ direct เช่น Equal Width (Wong and Chiu, 1987), Equal Frequency (Nguyen *et al.*, 1998) จะต้องให้ผู้ใช้ระบุจำนวนช่วงที่ต้องการ ต่างจากแบบ incremental ซึ่งจะหาจำนวนช่วงที่เหมาะสมเอง โดยใช้เกณฑ์ (criteria) บางอย่างในการหยุดการทำงาน



ภาพที่ 3 กระบวนการโดยทั่วไปสำหรับการแบ่งช่วงข้อมูล

ที่มา: Liu *et al.* (2002)

กระบวนการโดยทั่วไปสำหรับการแบ่งช่วงข้อมูลประกอบไปด้วย 4 ขั้นตอนหลักๆ (ภาพที่ 3) คือ

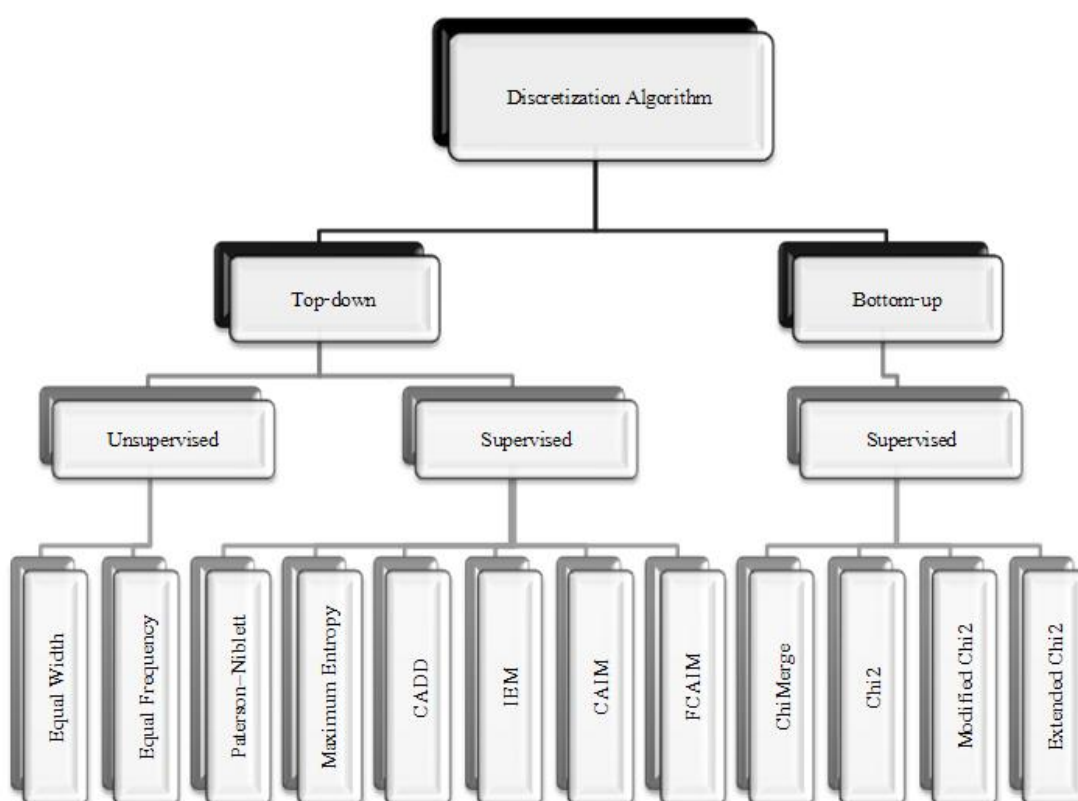
3.1 การเรียงลำดับ (Sorting) ค่าต่อเนื่องของแต่ละคุณลักษณะจะถูกเรียงลำดับจากน้อยไปมาก หรือมากไปน้อย ซึ่งขึ้นอยู่กับแต่ละอัลกอริทึมที่ใช้

3.2 การเลือกจุดตัด (Choosing a cut-point) หลังจากขั้นตอนการเรียงลำดับข้อมูล ขั้นตอนต่อไปจะเป็นการเลือกจุดตัด (cut-point) เพื่อแยกช่วงข้อมูล หรือ เลือก 2 ช่วงที่ติดกัน (adjacent intervals) เพื่อรวมช่วงข้อมูล

3.3 การแยกช่วง/การรวมช่วง (Splitting/Merging) การแบ่งช่วงแบบแยกช่วง หรือแบบ top-down นั้น จะทำการเลือกจุดตัดที่ดีที่สุดมาที่ละจุด เพื่อแยกช่วงของข้อมูล และจะทำงานลักษณะนี้ไปเรื่อยๆ (จำนวนช่วงเพิ่มขึ้นเรื่อยๆ ทีละ 1 ช่วง) จนกว่าจะถึงเกณฑ์ (criteria) ที่กำหนด

ส่วนการแบ่งช่วงแบบรวม หรือแบบ bottom-up นั้น จะมีลักษณะการทำงานตรงกันข้ามกับแบบแยกช่วง คือในแต่ละรอบการทำงาน จะทำการเลือกช่วง 2 ช่วงที่ติดกันที่ดีที่สุดมาทีละคู่ เพื่อรวมช่วงของข้อมูลเข้าด้วยกัน (จำนวนช่วงลดลงเรื่อยๆ ทีละ 1 ช่วง)

3.4 เกณฑ์การหยุด (Stopping criteria) เกณฑ์ในการหยุดการทำงานนั้น เราอาจทำได้โดยการกำหนดจำนวนช่วงของข้อมูลที่ต้องการ หรือ ใช้การคำนวณค่าของฟังก์ชันเพื่อหาจำนวนช่วงของข้อมูลที่เหมาะสมเองโดยอัตโนมัติ



ภาพที่ 4 การจำแนกประเภทอัลกอริทึมของการแบ่งช่วงข้อมูล

เมื่อเราทำการจำแนกอัลกอริทึมของการแบ่งช่วงข้อมูลออกเป็น 2 แบบ (ภาพที่ 4) คือ top-down กับ bottom-up โดยแบบ top-down นั้นยังสามารถแบ่งได้เป็นแบบ unsupervised ที่นิยมเช่น Equal Width , Equal Frequency และแบบ supervised เช่น Paterson–Niblett (Paterson and Niblett, 1987), maximum entropy (Wong and Chiu, 1987), Information Entropy Maximization (Fayyad and Irani, 1992; Fayyad and Irani, 1993), CADD (Ching *et al.*, 1995) และ CAIM (Kurgan and Cios, 2004) เป็นต้น จากผลการทดลองใน (Kurgan and Cios, 2004) เทียบกับอีก 6 top-down

อัลกอริทึม แสดงให้เห็นว่า CAIM เป็นอัลกอริทึมที่ให้ประสิทธิภาพของรูปแบบช่วงที่ดีที่สุด เมื่อเทียบกับอัลกอริทึมแบบ top-down ด้วยกัน

ในด้านของการแบ่งช่วงแบบ bottom-up นั้น มีอัลกอริทึมที่เป็นที่นิยมคือ ChiMerge (Kerber, 1992), Chi2 (Liu and Setiono, 1997) และ Modified Chi2 (Tay and Shen, 2002) ซึ่งโดยรวมจะใช้เวลาในการคำนวณที่มากกว่าแบบแบบ top-down เพราะจะเริ่มจากจุดตัดทั้งหมดที่เป็นไปได้ แล้วค่อยๆทำการลบจุดตัดออกทีละจุดโดยการรวมช่วงที่ติดกันเข้าด้วยกัน ซึ่งหลักๆจะใช้ค่า significance ในการตรวจสอบว่าช่วงใดที่ควรถูกรวมเข้าด้วยกัน โดย ChiMerge นั้นจะเป็นอัลกอริทึมพื้นฐานของการแบ่งช่วงแบบ bottom-up อื่นๆ จุดอ่อนของอัลกอริทึม ChiMerge นี้คือจะต้องให้ผู้ใช้ระบุค่าตัวแปร (parameter) ต่างๆ เช่นค่า significance level เพิ่มเติม ดังนั้นอัลกอริทึม Chi2 จึงได้แก้ปัญหาดังกล่าว โดยจะคำนวณค่า significance level ให้ แต่ยังคงให้ผู้ใช้ระบุค่า inconsistency rate เพื่อหยุดการทำงานอยู่ดี และไม่ได้นำค่า freedom ซึ่งอาจจะส่งผลต่อรูปแบบช่วงที่ได้มาพิจารณาอีกด้วย ต่อมา อัลกอริทึม Modified Chi2 ได้นำค่า freedom มาพิจารณาดู และเปลี่ยนจากการตรวจสอบค่า inconsistency ใน Chi2 เป็น quality of approximation หลังจากการหารูปแบบช่วงในแต่ละขั้นแทน

งานวิจัยที่เกี่ยวข้อง

ในวิทยานิพนธ์เล่มนี้จะนำเสนองานวิจัยที่เกี่ยวข้องกับการแบ่งช่วงข้อมูลที่เป็นค่าต่อเนื่อง และได้ทำการทดลองเปรียบเทียบกับอัลกอริทึมที่ใช้ในการแบ่งช่วงข้อมูลอื่นๆ ดังนี้

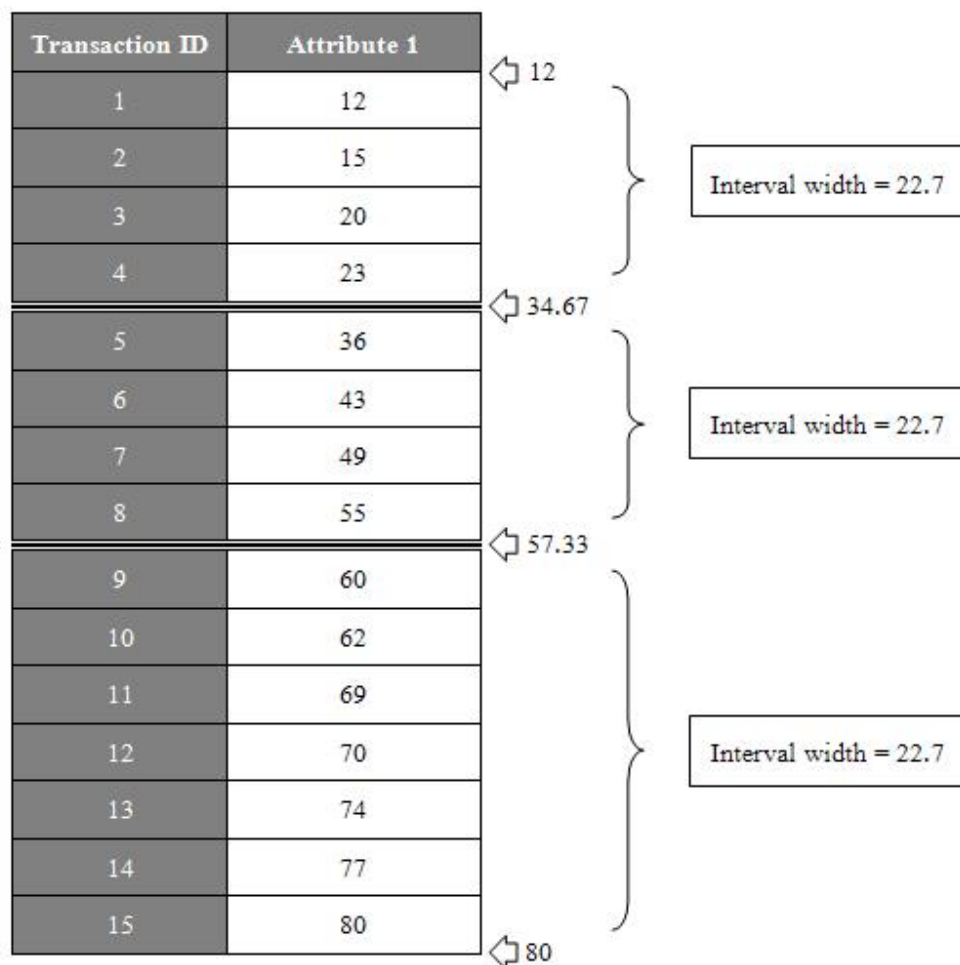
1. อัลกอริทึม Equal width และ Equal Frequency

อัลกอริทึม Equal width และ Equal Frequency เป็นอัลกอริทึมพื้นฐานที่ใช้ในการแบ่งช่วงข้อมูล จัดเป็นอัลกอริทึมแบบ unsupervised discretization จำนวนช่วงของผลลัพธ์ที่ได้ (k) จะให้ใช้งานเป็นผู้กำหนด

ตารางที่ 1 ตัวอย่างข้อมูลคุณลักษณะที่เป็นค่าต่อเนื่อง

Transaction ID	Attribute 1	Class Attribute
1	12	X
2	15	X
3	20	X
4	23	Y
5	36	Y
6	43	X
7	49	Y
8	55	Y
9	60	Z
10	62	Y
11	69	Z
12	70	Y
13	74	Z
14	77	Z
15	80	Z

อัลกอริทึม Equal width จะทำการแบ่งค่าต่อเนื่องของข้อมูลในแต่ละคุณลักษณะออกเป็น k ช่วง โดยในแต่ละช่วงจะมีขนาดความกว้างเท่าๆกัน การทำงานจะเริ่มจากการหาค่าที่น้อยที่สุดและมากที่สุด ของคุณลักษณะที่เป็นค่าต่อเนื่องที่ต้องการแบ่งช่วง จากนั้นจะทำการแบ่งช่วงของข้อมูลนี้ ออกเป็น k ช่วงเท่าๆกัน เช่น เมื่อเราทำการแบ่งช่วงของข้อมูลในตารางที่ 1 ออกเป็น 3 ช่วง ด้วย อัลกอริทึม Equal width จะได้ผลลัพธ์ดังภาพที่ 5



ภาพที่ 5 ผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม Equal width เมื่อ $k=3$

อัลกอริทึม Equal Frequency จะทำการแบ่งค่าต่อเนื่องของข้อมูลในแต่ละคุณลักษณะออกเป็น k ช่วง โดยในแต่ละช่วงจะมีจำนวนข้อมูลเท่าๆกัน การทำงานจะเริ่มจากเรียงลำดับค่าของข้อมูลคุณลักษณะที่เป็นค่าต่อเนื่องที่ต้องการแบ่งช่วง และหาจำนวนข้อมูลทั้งหมดในคุณลักษณะดังกล่าว จากนั้นจะทำการแบ่งจำนวนข้อมูลนั้นออกเป็น k ส่วนเท่าๆกัน เช่น เมื่อเราทำการแบ่งช่วงของข้อมูลในตารางที่ 1 ออกเป็น 3 ช่วง ด้วยอัลกอริทึม Equal frequency จะได้ผลลัพธ์ดังภาพที่ 6

Transaction ID	Attribute 1		
1	12	↩ 12	} Number of points = 5
2	15		
3	20		
4	23		
5	36		
6	43	↩ 39.5	} Number of points = 5
7	49		
8	55		
9	60		
10	62		
11	69	↩ 64.5	} Number of points = 5
12	70		
13	74		
14	77		
15	80	↩ 80	

ภาพที่ 6 ผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม Equal frequency เมื่อ $k=3$

เกณฑ์ในการหยุด: เมื่อมีจำนวนช่วงเท่ากับที่ผู้ใช้งานกำหนด

ทั้ง 2 อัลกอริทึมนี้จะใช้เวลาในการทำงานเพียงเล็กน้อย เนื่องจากทำการหารูปแบบช่วง (Discretization scheme) จากค่าที่ผู้ใช้งานเป็นผู้กำหนด และไม่ได้นำความสัมพันธ์ใดๆ ของข้อมูล เช่น คลาส มาร่วมพิจารณา ซึ่งถ้าผู้ใช้งานระบุจำนวนช่วงไม่เหมาะสมแล้ว อาจจะทำให้เกิดการสูญเสียของข้อมูล (Information loss) และส่งผลให้รูปแบบช่วงที่ได้ไม่มีประสิทธิภาพเมื่อนำไปใช้ในการจำแนกประเภท

2. อัลกอริทึม ChiMerge

อัลกอริทึม ChiMerge เป็นอัลกอริทึมต้นแบบของอัลกอริทึมในการแบ่งช่วงข้อมูลประเภท bottom-up อื่นๆ ซึ่งจัดเป็นอัลกอริทึมแบบ supervised discretization อัลกอริทึมนี้จะทำการค้นหาจำนวนช่วงที่เหมาะสมให้ โดยขึ้นกับสภาพของแต่ละคุณลักษณะ

อัลกอริทึมแบบ bottom-up จะมีรูปแบบช่วงเริ่มต้นเป็นช่วงของข้อมูลทั้งหมดที่เป็นไปได้ และจะค่อยๆทำการรวม 2 ช่วงที่ติดกันเข้าด้วยกันตามเงื่อนไขที่แต่ละอัลกอริทึมกำหนด จนได้จำนวนช่วงที่เหมาะสมหรือเท่าที่ต้องการ การทำงานของอัลกอริทึม ChiMerge นี้ จะเริ่มจากเรียงลำดับค่าของข้อมูลคุณลักษณะที่ต้องการแบ่งช่วง และทำการหาขอบเขต (Boundary) ที่เป็นไปได้ทั้งหมด ซึ่งประกอบไปด้วย ค่าที่น้อยที่สุด ค่ากึ่งกลางระหว่างข้อมูลแต่ละจุด และค่าที่มากที่สุด จากนั้นจะทำการคำนวณค่า Chi-Square (χ^2) ของทุกคู่ของช่วงข้อมูลที่ติดกัน และจะทำการรวมคู่ของช่วงข้อมูลที่ติดกันและมีค่า χ^2 น้อยที่สุดและต่ำกว่าเกณฑ์ที่กำหนด และจะมีลักษณะการทำงานเช่นนี้ไปเรื่อย จนกว่าจะถึงเกณฑ์ในการหยุด เช่น เมื่อเราทำการแบ่งช่วงของข้อมูลในตารางที่ 1 ด้วยอัลกอริทึม ChiMerge และกำหนดให้ค่า Significance level = 5% เราจะได้ผลลัพธ์ดังภาพที่ 7

การคำนวณหาค่า Chi-Square

$$\chi^2 = \sum_{i=1}^2 \sum_{j=1}^k \frac{(A_{ij} - E_{ij})^2}{E_{ij}}$$

โดยที่

k = จำนวนคลาสทั้งหมด

A_{ij} = จำนวนของข้อมูลที่มีค่าอยู่ในช่วงที่ i และมีคลาสเป็นคลาส j

E_{ij} = Expected frequency ของ $A_{ij} = (R_i * C_j) / N$

R_i = จำนวนของข้อมูลที่มีค่าอยู่ในช่วงที่ i

C_j = จำนวนของข้อมูลที่มีคลาสเป็นคลาส j และค่าอยู่ระหว่าง 2 ช่วงนี้

N = จำนวนของข้อมูลทั้งหมดที่มีค่าอยู่ระหว่าง 2 ช่วงนี้

หมายเหตุ: กรณีที่ค่า $E_{ij} = 0$ ให้กำหนดค่า E_{ij} นั้นเป็น 0.1

ตารางที่ 2 Contingency Tables สำหรับช่วง (21.5, 29.5] และ (29.5, 39.5]

	Class X	Class Y	Class Z	Sums
Interval 1 (21.5, 29.5]	0	1	0	1
Interval 2 (29.5, 39.5]	0	1	0	1
Sums	0	2	0	2

ตารางที่ 3 Contingency Tables สำหรับช่วง (12, 21.5] และ (21.5, 72]

	Class X	Class Y	Class Z	Sums
Interval 1 (12, 21.5]	3	0	0	3
Interval 2 (21.5, 72]	1	6	2	9
Sums	4	6	2	12

กำหนดให้ค่า Significance level = 5% และ degrees of freedom = (จำนวนคลาส - 1) = 2
จากตารางที่ 4 จะได้ว่า Threshold = 5.991

ตารางที่ 4 Critical values of chi-square

Degrees of freedom	Significance level				
	10%	5%	2.5%	1%	0.1%
1	2.706	3.841	5.024	6.635	10.828
2	4.605	5.991	7.378	9.210	13.816
3	6.251	7.815	9.348	11.345	16.266
4	7.779	9.488	11.143	13.277	18.467
:	:	:	:	:	:

จากตารางที่ 2 เมื่อทำการคำนวณค่า χ^2 จะได้ $E_{11} = 0 \sim 0.1$, $E_{12} = 1$, $E_{13} = 0 \sim 0.1$, $E_{21} = 0 \sim 0.1$, $E_{22} = 1$ และ $E_{23} = 0 \sim 0.1$ จะได้ว่า

$$\begin{aligned}
 \chi^2 &= [((0-0.1)^2/0.1) + ((1-1)^2/1) + ((0-0.1)^2/0.1)] \\
 &\quad + [((0-0.1)^2/0.1) + ((1-1)^2/1) + ((0-0.1)^2/0.1)] \\
 &= 0.4 < 5.991
 \end{aligned}$$

ดังนั้น จะทำการรวม 2 ช่วงนี้เข้าด้วยกันเป็นช่วงๆเดียว

จากตารางที่ 3 เมื่อทำการคำนวณค่า χ^2 จะได้ $E_{11} = 1$, $E_{12} = 1.5$, $E_{13} = 0.5$, $E_{21} = 3$, $E_{22} = 4.5$ และ $E_{23} = 1.5$ จะได้ว่า

$$\begin{aligned}
 \chi^2 &= [((3-1)^2/1) + ((0-1.5)^2/1.5) + ((0-0.5)^2/0.5)] \\
 &\quad + [((1-3)^2/3) + ((6-4.5)^2/4.5) + ((2-1.5)^2/1.5)] \\
 &= 7.997 > 5.991
 \end{aligned}$$

ดังนั้น จะไม่ทำการรวม 2 ช่วงนี้เข้าด้วยกัน

Transaction ID	Attribute 1	Class Attribute	
1	12	X	↩ 12
2	15	X	
3	20	X	
4	23	Y	↩ 21.5
5	36	Y	
6	43	X	
7	49	Y	
8	55	Y	
9	60	Z	
10	62	Y	
11	69	Z	
12	70	Y	
13	74	Z	↩ 72
14	77	Z	
15	80	Z	↩ 80

ภาพที่ 7 ผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม ChiMerge (Significance level 5%)

เกณฑ์ในการหยุด: เมื่อค่า χ^2 ของทุกคู่ของช่วงข้อมูลที่ติดกัน มีค่ามากกว่า Threshold ที่กำหนด

จะสังเกตเห็นว่าอัลกอริทึมประเภทนี้จะเริ่มต้นจากการที่มีจำนวนช่วงเท่ากับจำนวนค่าของข้อมูลที่เป็นไปได้ทั้งหมด ซึ่งโดยปกติแล้วคุณลักษณะที่เป็นค่าต่อเนื่องจะมีค่าที่หลากหลาย จึงส่งผลให้อัลกอริทึมนี้มักจะเริ่มต้นด้วยจำนวนช่วงที่มากมาย และค่อยๆลดจำนวนช่วงลงทีละ 1 ช่วงในแต่ละรอบการทำงาน ส่วนผลลัพธ์ที่ได้ มักมีจำนวนช่วงไม่มากนัก เช่น ถ้าคุณลักษณะที่จะทำการแบ่งช่วงมีค่าของข้อมูลที่เป็นไปได้ทั้งหมด 1,000 ค่า และจำนวนช่วงของผลลัพธ์ที่ได้มีค่าเท่ากับ 5 ดังนั้นอัลกอริทึมประเภทนี้ จะต้องทำงานทั้งหมด 995 รอบ จึงจะได้ผลลัพธ์ที่ต้องการ ส่วนอัลกอริทึมประเภท top-down จะทำงานไม่เกิน 5 รอบเท่านั้น ด้วยเหตุผลนี้อัลกอริทึมในการแบ่งช่วงประเภท bottom-up จึงมักทำงานช้ากว่าประเภท top-down

3. อัลกอริทึม CAIM

อัลกอริทึม CAIM จัดเป็นอัลกอริทึมแบบ top-down supervised discretization จากผลการทดลองของอัลกอริทึม CAIM เปรียบเทียบกับอีก 6 top-down อัลกอริทึม แสดงให้เห็นว่า CAIM เป็นอัลกอริทึมที่ให้ประสิทธิภาพของรูปแบบช่วงที่ดีที่สุด เมื่อเทียบกับอัลกอริทึมแบบ top-down ด้วยกัน ไม่ว่าจะเป็นจำนวนช่วงรวมของผลลัพธ์ที่ได้ หรือความแม่นยำเมื่อนำผลลัพธ์ที่ได้จากอัลกอริทึมนี้ไปใช้ในกระบวนการจำแนกประเภท

3.1 การทำงานของอัลกอริทึม CAIM

เมื่อกำหนดให้จำนวนข้อมูลมี M ตัวอย่าง และประกอบไปด้วยคลาส S คลาส ถ้าเราทำการแบ่งช่วงข้อมูลของคุณลักษณะ f_i ซึ่งมีค่าที่น้อยที่สุดคือ d_0 และมากที่สุดคือ d_n ให้มีจำนวนช่วง n ช่วงแล้วจะได้ว่า รูปแบบช่วงของข้อมูลในคุณลักษณะ f_i คือ

$$D_i = \{[d_0, d_1], (d_1, d_2], \dots, (d_{n-1}, d_n]\}$$

ซึ่งจากรูปแบบช่วงดังกล่าว ข้อมูลแต่ละตัวของคุณลักษณะ f_i จะถูกจำแนกให้อยู่ได้เพียงช่วงใดช่วงหนึ่งเท่านั้น โดยเราสามารถดูความสัมพันธ์ระหว่างรูปแบบช่วงของข้อมูลกับคลาสของคุณลักษณะ f_i ได้จากตาราง quanta-matrix ดัง ตารางที่ 5

ตารางที่ 5 ตาราง quanta-matrix

Class	Interval					Class Total
	$[d_{0-}d_{1-}]$	..	$(d_{r-}d_{r+}]$	..	$(d_{n-}d_{n+}]$	
C_1	q_{11}	..	q_{1r}	..	q_{1n}	q_{1+}
$\vdots$	$\vdots$	..	$\vdots$	..	$\vdots$	$\vdots$
C_i	q_{i1}	..	q_{ir}	..	q_{in}	q_{i+}
$\vdots$	$\vdots$	..	$\vdots$	..	$\vdots$	$\vdots$
C_S	q_{S1}	..	q_{Sr}	..	q_{Sn}	q_{S+}
Interval Total	q_{+1}	..	q_{+r}	..	q_{+n}	M

จากตารางที่ 5 q_{ir} คือจำนวนของข้อมูลที่เป็นคลาส C_i และมีค่าอยู่ในช่วง $(d_{r-}, d_{r+}]$, q_{i+} คือจำนวนของข้อมูลทั้งหมดที่เป็นคลาส C_i และ q_{+r} คือ จำนวนของข้อมูลทั้งหมดที่มีค่าอยู่ในช่วง $(d_{r-}, d_{r+}]$ โดยที่ i มีค่าระหว่าง $1 \dots S$ และ r มีค่าอยู่ระหว่าง $1 \dots n$

จากตารางที่ 5 อัลกอริทึม CAIM กำหนดค่าความสัมพันธ์ระหว่างรูปแบบช่วงกับคลาสของแต่ละคุณลักษณะเป็น

$$CAIM = \sum_{r=1}^n \left(\max_r^2 / q_{+r} \right) / n$$

โดย $\max_r$ เป็นค่าที่มากที่สุดเมื่อเทียบกับค่า q_{ir} ทั้งหมด ซึ่งค่าของ CAIM ยิ่งมากจะหมายถึงรูปแบบช่วงนั้นยิ่งดี

3.2 อัลกอริทึม CAIM มีขั้นตอนในการแบ่งช่วงข้อมูลในแต่ละคุณลักษณะดังนี้ (ภาพที่ 8)

3.2.1 เรียงลำดับค่าของข้อมูลจากน้อยไปหามาก

3.2.2 สร้างเซตของจุดตัดที่เป็นไปได้ทั้งหมด (B) ซึ่งประกอบไปด้วยค่าที่น้อยที่สุด (d_0) ค่ากึ่งกลางระหว่างข้อมูลแต่ละจุดเรียงจากน้อยไปหามาก และค่าที่มากที่สุด (d_n) ของข้อมูล ดังภาพที่ 9

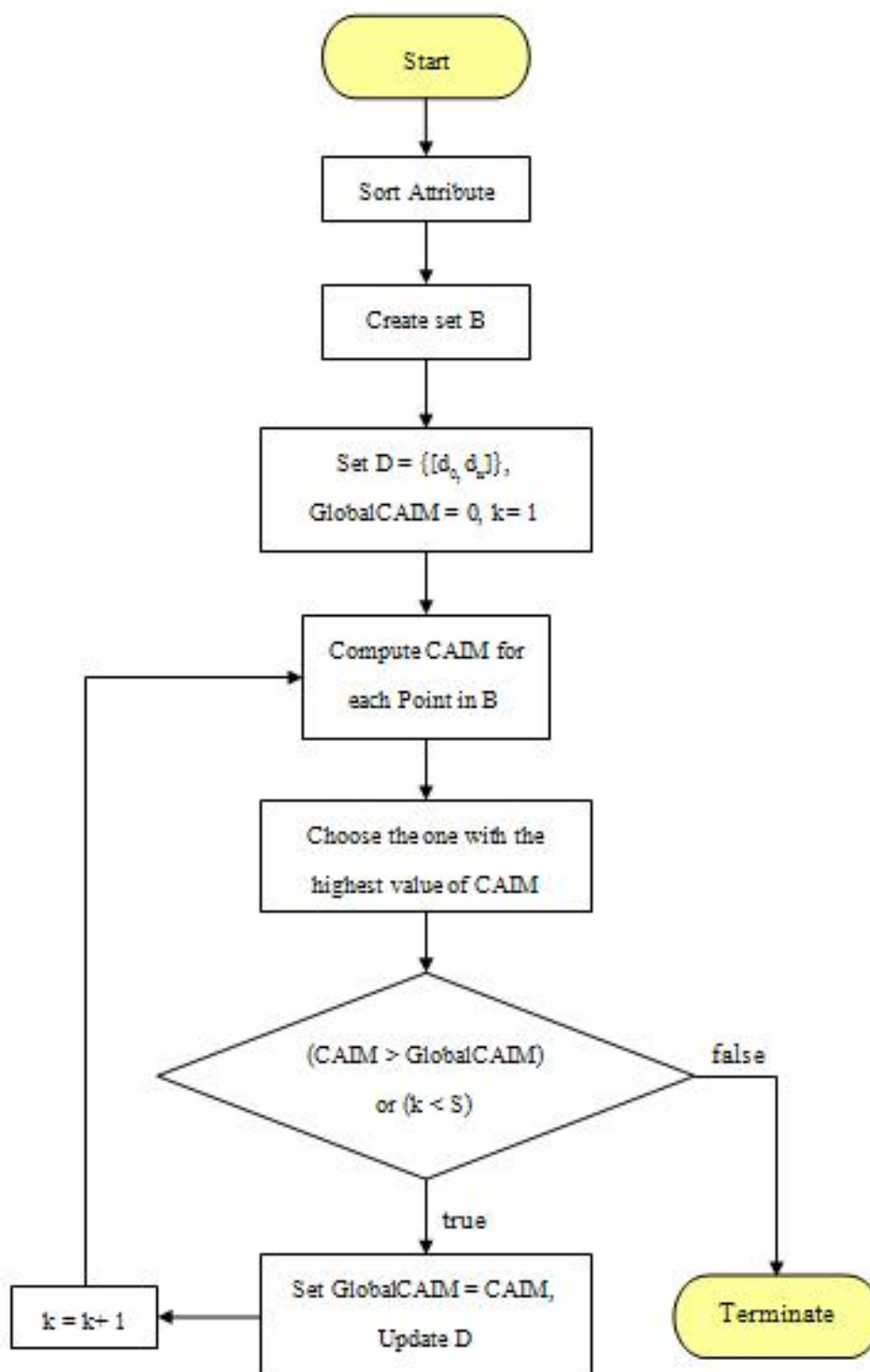
3.2.3 กำหนดให้รูปแบบช่วงเริ่มต้น (D) เป็น $\{[d_0, d_n]\}$, $\text{GlobalCAIM} = 0$ และจำนวนรอบในการทำงาน (k) เป็น 1

3.2.4 คำนวณค่า CAIM ของแต่ละจุดตัดที่เป็นไปได้ โดยนำจุดตัดที่อยู่ในเซต B และยังไม่มีในเซต D ที่ละ 1 จุด มาทดลองเป็นจุดแบ่งช่วงของข้อมูล เช่น เมื่อนำจุดตัด 61 มาทดลองเป็นจุดแบ่งช่วงข้อมูล เมื่อเซต D ในขณะนั้นเป็น $D = \{[12, 80]\}$ เราจะได้รูปแบบช่วงใหม่เป็น $D = \{[12, 61], (61, 80]\}$ ดังตารางที่ 6 และสามารถคำนวณค่า CAIM ของจุดตัดนี้ได้เป็น $[(4^2/9) + (4^2/6)] / 2 = 2.223$ เป็นต้น

3.2.5 เลือกจุดตัดที่ให้ค่า CAIM สูงที่สุด เพื่อจะนำมาเป็นจุดแบ่งช่วงของข้อมูล

3.2.6 ถ้าจุดตัดที่ได้จากข้อ 5 ให้ค่า CAIM มากกว่าค่า GlobalCAIM หรือจำนวนรอบในการทำงาน (k) ยังน้อยกว่าจำนวนคลาสที่เป็นไปได้ทั้งหมด ให้เพิ่มจุดตัดนั้นลงในเซต D และทำการกำหนดค่า $\text{GlobalCAIM} = \text{CAIM}$ แต่ถ้าไม่ผ่านเงื่อนไขนี้ จะหยุดการทำงาน

3.2.7 กำหนดค่า $k = k + 1$ และกลับไปทำขั้นตอนที่ 4



ภาพที่ 8 ขั้นตอนการทำงานของอัลกอริทึม CAIM

Transaction ID	Attribute 1	
1	12	↩ 12
2	15	↩ 13.5
3	20	↩ 17.5
4	23	↩ 21.5
5	36	↩ 29.5
6	43	↩ 39.5
7	49	↩ 46
8	55	↩ 52
9	60	↩ 57.5
10	62	↩ 61
11	69	↩ 65.5
12	70	↩ 69.5
13	74	↩ 72
14	77	↩ 75.5
15	80	↩ 78.5
		↩ 80

ภาพที่ 9 การสร้างเซตของจุดตัดที่เป็นไปได้ทั้งหมด (B)

ตารางที่ 6 ตาราง quanta-matrix เมื่อเซต $D = \{[12, 61], (61, 80]\}$

Class	Interval		Class Total
	$[12, 61]$	$(61, 80]$	
X	4	0	4
Y	4	2	6
Z	1	4	5
Interval Total	9	6	15

Transaction ID	Attribute 1	Class Attribute	
1	12	X	↩ 12
2	15	X	
3	20	X	
4	23	Y	↩ 21.5
5	36	Y	
6	43	X	
7	49	Y	
8	55	Y	
9	60	Z	
10	62	Y	
11	69	Z	
12	70	Y	
13	74	Z	↩ 72
14	77	Z	
15	80	Z	↩ 80

ภาพที่ 10 ผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม CAIM

เกณฑ์ในการหยุด: เมื่อค่า CAIM ของรูปแบบช่วงปัจจุบันที่คำนวณได้มีค่าลดลงจากค่า CAIM ของรูปแบบช่วงในรอบที่ผ่านมา โดยที่ขณะนั้นต้องมีจำนวนช่วงไม่น้อยกว่าจำนวนคลาสของข้อมูลทั้งหมด ซึ่งผลลัพธ์ของการแบ่งช่วงข้อมูลด้วยอัลกอริทึม CAIM จะเป็นดังภาพที่ 10

3.3 ข้อเสียของอัลกอริทึม CAIM

แม้ว่า CAIM เป็นอัลกอริทึมที่ให้ประสิทธิภาพของรูปแบบช่วงที่ดีที่สุด เมื่อเทียบกับอัลกอริทึมแบบ top-down อื่นๆ แต่อัลกอริทึม CAIM ก็ยังมีข้อเสียคือ

3.3.1 ไม่มีการคัดเลือกจุดตัดที่จะนำมาพิจารณาในการค้นหารูปแบบช่วง โดยจะใช้จุดตัดทุกจุดที่เป็นไปได้ (B) มาคำนวณ ทำให้ใช้เวลาในการทำงานมากกว่าที่ควรจะเป็น ยกตัวอย่างเช่น ภาพที่ 11 อัลกอริทึม CAIM มีจุดตัดที่ต้องนำมาพิจารณาถึง 14 จุด ในขณะที่มีจุดตัดบางจุดซึ่งเป็นจุดตัดซึ่งแบ่งระหว่าง 2 ช่วงของข้อมูลที่เป็นคลาสเดียวกันนั้น ไม่น่าที่จะต้องนำมาพิจารณาดังเช่นจุดตัดที่มีเครื่องหมายกากบาทอยู่ทางด้านหลังของจุด ซึ่งจะทำให้จุดที่ต้องพิจารณาในการค้นหารูปแบบช่วง ลดลงเหลือเพียง 8 จุดเท่านั้น

Transaction ID	Attribute 1	Class Attribute	
1	12	X	
2	15	X	↳ 13.5 ✗
3	20	X	↳ 17.5 ✗
4	23	Y	↳ 21.5
5	36	Y	↳ 29.5 ✗
6	43	X	↳ 39.5
7	49	Y	↳ 46
8	55	Y	↳ 52 ✗
9	60	Z	↳ 57.5
10	62	Y	↳ 61
11	69	Z	↳ 65.5
12	70	Y	↳ 69.5
13	74	Z	↳ 72
14	77	Z	↳ 75.5 ✗
15	80	Z	↳ 78.5 ✗

ภาพที่ 11 จุดตัดที่ไม่ควรนำมาพิจารณาในการหารูปแบบช่วง

3.3.2 อัลกอริทึม CAIM จะพิจารณาเฉพาะกลุ่มของคลาสที่มีมากที่สุด ($\max_c$) แต่ไม่สนใจคลาสอื่นๆ ทำให้ประสิทธิภาพของรูปแบบช่วงที่ได้ลดลง ดังตารางที่ 7 และ 8 ซึ่งจะเห็นว่า

รูปแบบช่วงของข้อมูลในตารางที่ 7 น่าจะดีกว่ารูปแบบช่วงของข้อมูลในตารางที่ 8 เพราะตารางที่ 7 นั้นมีการกระจายตัวของแต่ละคลาสที่น้อยกว่า และอาจใช้อีกเพียงคุณลักษณะเดียว ก็จะสามารถจำแนกคลาสออกจากกันได้หมด แต่ในกรณีของตารางที่ 8 ต้องใช้อีกอย่างน้อย 2 คุณลักษณะ ถึงจะสามารถจำแนกแต่ละคลาสออกจากกันได้

ตารางที่ 7 รูปแบบการกระจายตัวของแต่ละคลาสแบบที่ 1

Class	Interval			Class Total
	..	$(d_{r-p}, d_p]$	..	
C_1		0		
C_2		5		
C_3		10		
Interval Total		15		

ตารางที่ 8 รูปแบบการกระจายตัวของแต่ละคลาสแบบที่ 2

Class	Interval			Class Total
	..	$(d_{r-p}, d_p]$	..	
C_1		2		
C_2		3		
C_3		10		
Interval Total		15		

3.3.3 อัลกอริทึม CAIM นั้น เป็นอัลกอริทึมแบบ supervised ซึ่งนำประโยชน์จากคลาสของข้อมูลมาใช้ แต่ยังไม่ได้ใช้ประโยชน์จากความสัมพันธ์ระหว่างคุณลักษณะ ซึ่งจะสามารถนำมาใช้ในการลดจำนวนช่วงของผลลัพธ์ที่ไม่จำเป็นได้ ดังจะมีให้เห็นในงานวิจัยที่ได้นำเสนอ

ตารางที่ 9 ตารางเปรียบเทียบคุณลักษณะของแต่ละอัลกอริทึม

อัลกอริทึม	จำนวนช่วงของผลลัพธ์	ประเภทของอัลกอริทึม	ลักษณะการทำงาน
Equal Width	ผู้ใช้กำหนด	unsupervised	top-down
Equal Frequency	ผู้ใช้กำหนด	unsupervised	top-down
ChiMerge	อัตโนมัติ	supervised	bottom-up
CAIM	อัตโนมัติ	supervised	top-down

จากตารางที่ 9 จะเป็นตารางสรุปความแตกต่างของแต่ละอัลกอริทึมที่เกี่ยวข้องกับงานวิจัยนี้ โดยแบ่งออกเป็นอัลกอริทึมแบบไม่มีผู้สอน 2 อัลกอริทึม คือ Equal width และ Equal Frequency โดยทั้ง 2 อัลกอริทึมนี้ต้องให้ผู้ใช้งานเป็นผู้กำหนดจำนวนช่วงที่ต้องการ และอัลกอริทึมแบบมีผู้สอนอีก 2 อัลกอริทึม คือ ChiMerge ซึ่งเป็นอัลกอริทึมแบบ bottom-up และต้องให้ผู้ใช้งานเป็นผู้กำหนดค่า significance level ที่จะใช้เป็นเกณฑ์ในการทำงาน และอัลกอริทึม CAIM ซึ่งเป็นอัลกอริทึมแบบ top-down และให้ผลการทดลองที่ดีที่สุด เมื่อเทียบกับอัลกอริทึมแบบ top-down ด้วยกัน

อุปกรณ์และวิธีการ

อุปกรณ์

1. ฮาร์ดแวร์

1.1 เครื่องคอมพิวเตอร์โน้ตบุ๊ก 1 เครื่อง ประกอบด้วยอุปกรณ์ดังต่อไปนี้

1.1.1 ซีพียู (CPU) AMD Turion 64 X2 ความเร็ว 1.8 GHz

1.1.2 หน่วยความจำหลัก 1 GB

1.1.3 ฮาร์ดดิสก์ขนาด 160 GB

2. ซอฟต์แวร์

2.1 ระบบปฏิบัติการ Windows XP Professional

2.2 Borland Delphi 2006

วิธีการ

1. ภาพรวมของระบบ

จากข้อดีและข้อเสียของการแบ่งช่วงข้อมูลประเภทต่างๆ ที่ได้กล่าวถึงในส่วนของความรู้พื้นฐานข้างต้น และจากข้อเสียของแต่ละอัลกอริทึมที่ได้นำเสนอในหัวข้อที่ผ่านมา จะเห็นว่าอัลกอริทึมในการค้นหารูปแบบช่วงต่างๆ ยังสามารถพัฒนาได้ในอีกหลายๆ ด้าน ไม่ว่าจะเป็นจำนวนช่วงของผลลัพธ์ที่ได้ เวลาที่ใช้ในการประมวลผล และความแม่นยำในการจำแนกประเภทเมื่อนำผลลัพธ์ที่ได้จากอัลกอริทึมต่างๆ ไปเป็นข้อมูลนำเข้าให้กับระบบ ซึ่งถือว่าเป็นประโยชน์ที่สำคัญที่สุดของการแบ่งช่วงข้อมูล

งานวิจัยนี้จึงได้นำเสนออัลกอริทึม DCR (Discretization using Class Information to Reduce Number of Intervals) ซึ่งเป็นอัลกอริทึมที่ใช้ในการค้นหารูปแบบช่วงของข้อมูลประเภท supervised, static, global, top-down และ incremental มีวัตถุประสงค์คือ ค้นหาแบบช่วงของข้อมูลที่มีจำนวนช่วงรวมน้อยที่สุด ที่ให้ความแม่นยำสูงในการนำไปใช้จำแนกคลาส และใช้เวลาในการทำงานที่ไม่มากนัก โดยนำประโยชน์จากความสัมพันธ์ระหว่างคลาสกับข้อมูลในแต่ละ

คุณลักษณะมาช่วยในการค้นหารูปแบบช่วงที่เหมาะสม และนำความสัมพันธ์ของแต่ละคุณลักษณะ มาช่วยพิจารณาในการลดเวลาสำหรับการทำงานของระบบ ลดจำนวนช่วงของผลลัพธ์ที่จะได้ และ ยังเป็นการตัดคุณลักษณะที่ไม่จำเป็นออกไปได้อีกด้วย ซึ่งจะเป็นประโยชน์อย่างมากต่อการนำ ผลลัพธ์ที่ได้ไปใช้ในการจำแนกประเภท

การค้นหารูปแบบช่วงที่ให้ความแม่นยำในการจำแนกคลาสที่ดีนั้น เราจะทำโดยการค้นหา รูปแบบช่วงที่ทำให้คลาสของข้อมูลในแต่ละช่วงเป็นคลาสเดียวกันให้มากที่สุด ซึ่งเราจะสามารถดู ความสัมพันธ์กันระหว่าง รูปแบบช่วงของข้อมูลกับคลาส ได้จากตาราง quanta-matrix ดังที่แสดง ให้เห็นในตารางที่ 5 ส่วนขั้นตอนของการค้นหารูปแบบช่วงที่ทำให้จำนวนช่วงรวมน้อยที่สุดโดย ไม่ส่งผลกระทบต่อความแม่นยำในการจำแนกคลาสนั้น เราจะนำประโยชน์จากความสัมพันธ์กัน ของแต่ละคุณลักษณะในฐานข้อมูล มาใช้ในการตัดระเบียบ (transaction) ของฐานข้อมูลออก เพื่อ ลดจำนวนข้อมูลของคุณลักษณะต่อไปที่จะหารูปแบบช่วง ซึ่งจะมีเนื้อหาโดยละเอียดในหัวข้อ ถัดไป

2. การค้นหารูปแบบช่วง

งานวิจัยนี้ได้นำเสนอเกณฑ์ในการวัดค่าความสัมพันธ์ระหว่างรูปแบบช่วงของข้อมูลในแต่ละคุณลักษณะ กับ คลาส (DCR) ซึ่งได้ทำการปรับปรุงมาจากสูตรของอัลกอริทึม CAIM เนื่องจากที่ ได้กล่าวถึงข้อเสียของอัลกอริทึม CAIM ในข้างต้นแล้วว่า อัลกอริทึม CAIM จะพิจารณาเฉพาะกลุ่ม ของคลาสที่มีมากที่สุด ($\max_r$) แต่ไม่สนใจคลาสอื่นๆ ส่งผลทำให้ประสิทธิภาพของรูปแบบช่วงที่ ได้ลดลง งานวิจัยนี้จึงได้เสนอการใช้ค่า $\sum_{i=1}^s q_{ir}^2$ แทนค่า $\max_r^2$ เพื่อแก้ปัญหาดังกล่าว และยังคงใช้ ตาราง quanta-matrix (ตารางที่ 5) มาช่วยการดูความสัมพันธ์ระหว่างรูปแบบช่วงของข้อมูลในแต่ละคุณลักษณะ กับ คลาส โดยมีสูตรในการหาค่าความสัมพันธ์ใหม่นี้

$$DCR = \frac{\sum_{r=1}^n \left(\sum_{i=1}^s q_{ir}^2 / q_{+r} \right)}{n}$$

จากสูตรที่ได้นำเสนอจะพบว่า ค่าของ DCR จะมาจากค่าเฉลี่ยการกระจายตัวของคลาส $(\sum_{i=1}^s q_{ir}^2 / q_{+r})$ ในแต่ละช่วงของข้อมูล $(\sum_{r=1}^n)$ และจะทำการหาค่าเฉลี่ยของแต่ละช่วงโดยจะหารด้วยจำนวนช่วงทั้งหมด (n) ซึ่งค่าของ DCR จะมีคุณสมบัติดังนี้

2.1 สามารถหารูปแบบช่วงของข้อมูล ที่มีคลาสของข้อมูลในแต่ละช่วงเป็นคลาสเดียวกันมากที่สุดได้ ซึ่งจากตาราง quanta-matrix ถ้าเราพิจารณาช่วงที่ r ใดๆ จะเห็นว่าจำนวนของข้อมูลที่เป็นคลาส i จะเท่ากับ q_{ir} โดยจะมีค่าที่เป็นไปได้ในช่วง $[0, q_{+r}]$ ดังนั้นการที่จะแบ่งช่วงให้คลาสของข้อมูลในช่วงนี้เป็นคลาสเดียวกันมากที่สุดคือ การที่ q_{ir} มีค่าเท่ากับ q_{+r} เนื่องจากค่าของ DCR จะแปรผันโดยตรงกับค่าของ $\sum_{i=1}^s q_{ir}^2$ ดังนั้นจึงสรุปได้ว่า ค่าของ DCR จะสูงที่สุดเมื่อได้รูปแบบช่วงของข้อมูลที่มีคลาสในแต่ละช่วงเป็นคลาสเดียวกันมากที่สุด

2.2 การใช้ q_{ir}^2 เป็นการดูการกระจายตัวของจำนวนข้อมูลที่เป็นคลาสต่างๆในแต่ละช่วง ซึ่งจะให้ค่าที่สูงเมื่อการกระจายตัวของจำนวนข้อมูลที่เป็นคลาสต่างๆในแต่ละช่วงมีน้อย และยังทำให้สามารถแก้ปัญหของอัลกอริทึม CAIM ได้

2.3 การนำค่า q_{ir}^2 หารด้วย q_{+r} นั้น มีเหตุผลคือ เพื่อทำการปรับค่าของ q_{ir}^2 ไม่ให้มีค่าที่สูงเกินไป จนอาจทำให้คำนวณไม่ได้ เนื่องจากค่า q_{ir} จะมีค่าสูงสุดแค่ q_{+r} ดังนั้นเราจะหาค่าของ q_{ir}^2 / q_{+r} ด้วยวิธี $q_{ir} (q_{ir} / q_{+r})$ แทน ซึ่งจะทำให้มีค่าสูงสุดเพียงแค่ q_{+r} เท่านั้น

3. การลดจำนวนช่วงของข้อมูลในแต่ละคุณลักษณะ

เราได้นำเสนอประโยชน์จากความสัมพันธ์ระหว่างคุณลักษณะในฐานข้อมูลมาใช้ในส่วนนี้ โดยอัลกอริทึม DCR จะมีการตัดระเบียบของฐานข้อมูลออก เพื่อที่จะทำให้คุณลักษณะต่อไปที่จะทำการแบ่งช่วงนั้น มีจำนวนระเบียบที่จะใช้ในการพิจารณาตกลง ทำให้การแบ่งช่วงของคุณลักษณะต่อไป เป็นเพียงการค้นหารูปแบบช่วงสำหรับข้อมูลที่ยังจำแนกคลาสไม่ได้เท่านั้น ซึ่งจะทำให้จำนวนช่วงที่ได้ของคุณลักษณะต่อไปลดลง อีกทั้งยังเป็นการช่วยลดเวลาในการประมาณผลของคุณลักษณะที่เหลือ ให้ลดลงได้อีกด้วย โดยอัลกอริทึม DCR จะทำกระบวนการตัดระเบียบทุกครั้ง หลังจากทำการแบ่งช่วงข้อมูลในแต่ละคุณลักษณะ เช่น ถ้าเราสมมติให้รูปแบบช่วงของข้อมูลในตารางที่ 1 มีค่าเป็น $\{[12, 21.5], (21.5, 72], (72, 80]\}$ จะมีวิธีในการตัดระเบียบดังภาพที่ 12



Transaction ID	Attribute 1	Class
1	12	X
2	15	X
3	20	X
4	23	Y
5	36	Y
6	43	X
7	49	Y
8	55	Y
9	60	Z
10	62	Y
11	69	Z
12	70	Y
13	74	Z
14	77	Z
15	80	Z

Transaction ID	Attribute 1	Class
4	23	Y
5	36	Y
6	43	X
7	49	Y
8	55	Y
9	60	Z
10	62	Y
11	69	Z
12	70	Y

ภาพที่ 12 วิธีการตัดระเบียบสำหรับอัลกอริทึม DCR

จากภาพที่ 12 คลาสของข้อมูลที่มีค่าอยู่ในระหว่าง $[12, 21.5]$ และ $(72, 80]$ นั้น มีคลาสเป็นคลาสเดียวกันทั้งหมดคือคลาส X และ Z ตามลำดับ ดังนั้น เราจะทำการตัดระเบียบเหล่านี้ ออกไป เพราะสามารถใช้ช่วงดังกล่าวเพียงช่วงเดียวก็สามารถจำแนกคลาสของฐานข้อมูลชุดนี้ได้ ดังนั้น จะเหลือระเบียบเพื่อใช้พิจารณาในการหารูปแบบช่วงของคุณลักษณะถัดไปเพียง 9 ระเบียบเท่านั้น คือระเบียบที่ 4 ถึง 12

จากข้างต้น จะสังเกตเห็นว่าคุณลักษณะที่ได้รับการแบ่งช่วงก่อน จะส่งผลต่อการหารูปแบบช่วงของคุณลักษณะถัดไป ซึ่งจะทำให้ลำดับในการแบ่งช่วงข้อมูลของแต่ละคุณลักษณะ ส่งผลต่อรูปแบบช่วงสุดท้ายที่จะได้ ดังนั้นอัลกอริทึม DCR จึงทำการเรียงลำดับก่อนหลังของ

คุณลักษณะที่จะทำการแบ่งช่วง โดยจะทำการคำนวณค่า เกนสารสนเทศ (information gain) ของแต่ละคุณลักษณะเทียบกับคลาส และทำการหารูปแบบช่วงของแต่ละคุณลักษณะนี้ ตามลำดับค่า เกนสารสนเทศจากมากไปหาน้อย โดยค่าเกนสารสนเทศ เป็นค่าที่ใช้ในอัลกอริทึม C4.5 ซึ่งเป็นอัลกอริทึมในการค้นหาโมเดลจำแนกประเภทข้อมูล แบบโครงสร้างต้นไม้ (decision trees) เพื่อตัดสินใจว่าคุณลักษณะใดเหมาะที่จะเป็นตัวจำแนกคลาสของข้อมูล ดังที่ได้กล่าวมาแล้วในขั้นต้น ซึ่งถ้าคุณลักษณะใดมีค่าเกนสารสนเทศสูง จะมีความหมายว่า คุณลักษณะนั้นมีความสับสนของคลาสน้อยที่สุด หรือมีการแปรผันตามกันของค่าของข้อมูลกับคลาสมากที่สุดนั่นเอง ทั้งนี้เพื่อเป็นการค้นหารูปแบบช่วงข้อมูลของคุณลักษณะที่เหมาะสมจะเป็นตัวจำแนกคลาสมาก่อนนั่นเอง

เนื่องจากกระบวนการในการตัดสินใจของอัลกอริทึม DCR นี้ จะส่งผลให้ความสัมพันธ์ระหว่างค่าของข้อมูลกับคลาสเปลี่ยนไป ดังนั้นอัลกอริทึมนี้จะทำการคำนวณค่าเกนสารสนเทศของคุณลักษณะที่เหลือใหม่ทุกครั้งหลังจากที่ทำการตัดสินใจเสร็จ เพื่อหาคุณลักษณะที่เหมาะสมจะเป็นตัวจำแนกคลาสมากที่สุดต่อไป

4. อัลกอริทึม DCR

การหารูปแบบช่วงที่ดีที่สุดนั้น สามารถทำได้โดยการคำนวณหาค่า *DCR* ของแต่ละรูปแบบช่วงที่เป็นไปได้ทั้งหมด แล้วเลือกรูปแบบช่วงที่ให้ค่า *DCR* สูงที่สุด (แบบ global) แต่ในการหารูปแบบช่วงของข้อมูลด้วยวิธีนี้ จะใช้เวลาในการทำงานที่สูงมาก ดังนั้นในการทดลอง เราจะใช้ greedy method เช่นเดียวกับอัลกอริทึม CAIM ในการค้นหารูปแบบช่วง (แบบ local) คือใช้วิธีการค่อยๆ เพิ่มจำนวนช่วงขึ้นทีละช่วง โดยในแต่ละช่วงที่เพิ่มขึ้นมาจะเป็นช่วงที่ให้ค่า *DCR* สูงที่สุด ซึ่งจะทำให้ใช้เวลาในการทำงานไม่มากนัก และยังได้รูปแบบช่วงของข้อมูลที่มีประสิทธิภาพค่อนข้างดีอีกด้วย ดังจะเห็นได้จากผลการวิจัยในหัวข้อถัดไป

อัลกอริทึม DCR ประกอบไปด้วย 3 ขั้นตอนหลักๆ คือ

- 4.1 จัดลำดับก่อนหลังของคุณลักษณะที่จะทำการแบ่งช่วง
- 4.2 หารูปแบบช่วงของข้อมูลในแต่ละคุณลักษณะ ตามลำดับที่ได้จัดไว้
- 4.3 ทำการปรับปรุงฐานข้อมูลใหม่โดยกระบวนการตัดสินใจ

โดยมีรหัสเทียม (psudocode) ของอัลกอริทึม DCR ดังภาพที่ 13

Input: Training data set DB ประกอบไปด้วย continuous attributes F_i และ class attribute C โดยมีคลาสทั้งหมด s คลาส

1. *for each* F_i
2. $E_i = \text{information_gain}(F_i, C)$;
3. $\text{arrange_order_desc}(F, E)$;
4. $db = DB$;
5. *for each* f_i // f_i = feature F_i of current training data set db
6. $d_{\min} = \min(F_i)$;
7. $d_{\max} = \max(F_i)$;
8. $D_i = \{[d_{\min}, d_{\max}] \}$
9. *if* ($db \neq \emptyset$) *then*
10. $k = 1$;
11. $MaxDCR = 0$;
12. $EB = \text{essential_boundary_set}(f_i)$;
13. Repeat
14. $DCR = \text{compute_max_dcr_boundary}(f_i, D_i, EB)$;
15. *If* ($DCR > MaxDCR$) *or* ($k < s$) *then*
16. Update D_i with an inner boundary that has the highest value of DCR
17. $MaxDCR = DCR$;
18. $k = k + 1$;
19. Until ($DCR \leq MaxDCR$) *and* ($k \geq s$)
20. $db = \text{reducing_transaction}(db, D_i)$;
21. $D = D \cup \{D_i\}$

Output: Set of all discretization scheme D

ภาพที่ 13 รหัสเทียมของอัลกอริทึม DCR

อัลกอริทึมนี้จะเริ่มต้นโดยการจัดลำดับก่อนหลังของแต่ละคุณลักษณะสำหรับการแบ่งช่วงข้อมูลตามค่า เกนสารสนเทศ จากมากไปน้อย (ขั้นตอนที่ 1-3) จากนั้น เราจะทำการค้นหารูปแบบช่วงของข้อมูลในแต่ละคุณลักษณะตามลำดับที่จัดไว้ (ขั้นตอนที่ 6-19) ซึ่งเริ่มจากการให้รูปแบบช่วงเริ่มต้นเป็นช่วงๆเดียว ที่ครอบคลุมข้อมูลที่เป็นไปได้ทั้งหมดของคุณลักษณะดังกล่าว (ขั้นตอนที่ 8) จากนั้น จะทำการหาเซต EB (Essential Boundary Set) ซึ่งเป็นเซตของจุดตัดที่เป็นไปได้ทั้งหมดที่แบ่งระหว่าง 2 ช่วงที่เป็นคนละคลาสกัน (ขั้นตอนที่ 12) โดยมีวิธีในการหาเซตดังกล่าวดังภาพที่ 11 จะเห็นว่าค่า 13.5 เป็นจุดที่แบ่งช่วงระหว่างระยะเบี่ยงที่ 1 และ 2 ซึ่งมีคลาสเป็นคลาสเดียวกันคือคลาส X ดังนั้นจะไม่นำจุดตัดนี้มาพิจารณา แต่ค่า 21.5 เป็นจุดที่แบ่งช่วงระหว่างระยะเบี่ยงที่ 3 และ 4 ซึ่งจะเห็นว่า ใน 2 ระยะเบี่ยงนี้ไม่ได้เป็นคลาสเดียวกันทั้งหมด ดังนั้นจึงนำจุดตัดนี้มาพิจารณาในการค้นหารูปแบบช่วงของข้อมูลในคุณลักษณะนั้นด้วย ซึ่งจะทำให้ได้เซต EB ของคุณลักษณะดังกล่าว คือ $\{21.5, 39.5, 46, 57.5, 61, 65.5, 69.5, 72\}$ หลังจากนั้น จะทำการหาจุดตัดในเซต EB ที่ให้ค่า DCR สูงที่สุด เพื่อมาเป็นจุดแบ่งช่วงจุดต่อไป ซึ่งอัลกอริทึม DCR จะมีวิธีการค้นหารูปแบบช่วงของแต่ละคุณลักษณะเช่นนี้ไปเรื่อยๆ จนกระทั่ง (1) ค่า DCR ที่สูงที่สุดของรูปแบบช่วงใหม่จะลดลงกว่าค่า DCR ที่สูงที่สุดเดิม และ (2) จำนวนช่วงของคุณลักษณะดังกล่าวมากกว่า จำนวนคลาสที่เป็นไปได้ทั้งหมดของข้อมูลที่พิจารณา (ขั้นตอนที่ 19) เพื่อให้มั่นใจว่าจะได้รูปแบบช่วงที่มีประสิทธิภาพมากขึ้น และสามารถนำไปใช้จำแนกคลาสที่มีได้ เช่น ถ้าเราจะทำการจำแนกคลาสซึ่งมีทั้งหมด 3 คลาสด้วยคุณลักษณะเดียวแล้ว เราก็จะต้องมีช่วงของข้อมูลทั้งหมดอย่างน้อย 3 ช่วงเช่นกัน

ในขั้นตอนสุดท้าย หลังการค้นหารูปแบบช่วงของแต่ละคุณลักษณะเสร็จ จะทำการปรับปรุงฐานข้อมูลใหม่ โดยกระบวนการตัดระยะเบี่ยงที่มีค่าอยู่ในช่วงซึ่งคลาสของทุกระเบี่ยงในช่วงนั้นเป็นคลาสเดียวกันทั้งหมด (ขั้นตอนที่ 20) ดังขั้นตอนในภาพที่ 12 เป็นต้น

5. ตัวอย่างการทำงานของอัลกอริทึม DCR

ตารางที่ 10 ตัวอย่างฐานข้อมูลที่ใช้ในการค้นหารูปแบบช่วง

Transaction ID	Attribute 1	Attribute 2	Attribute 3	Class
1	17	49	33	Z
2	19	48	21	Y
3	21	50	50	Y
4	21	53	19	X
5	22	65	49	Y
6	35	70	55	Y
7	33	89	76	Z
8	42	48	80	Z
9	40	63	33	Y
10	22	72	21	X
11	23	80	10	X
12	20	73	9	X
13	19	65	43	Y
14	25	90	95	Z
15	29	73	21	Y

จากตัวอย่างฐานข้อมูลในตารางที่ 10 อัลกอริทึม DCR จะเริ่มจากการเรียงลำดับคุณลักษณะที่จะนำมาพิจารณา โดยวัดจากค่าเกนสารสนเทศ (Information gain) ซึ่งจากตัวอย่างฐานข้อมูลนี้ จะได้ค่าเกนสารสนเทศของคุณลักษณะที่ 1, 2 และ 3 เป็น 0.78, 1.023 และ 1.53 ตามลำดับ จะเห็นว่าคุณลักษณะที่ 3 มีค่าเกนสารสนเทศสูงที่สุด ดังนั้น อัลกอริทึม DCR จะเริ่มทำการหารูปแบบช่วงของข้อมูลดังกล่าวเป็นลำดับแรก

ในขั้นตอนการหารูปแบบช่วง จะเริ่มจากการเรียงลำดับค่าของข้อมูลจากน้อยไปหามากดังภาพที่ 14 จากนั้นจะทำการหาเซต EB ของคุณลักษณะดังกล่าว (ภาพที่ 15) จะเห็นว่าเริ่มแรกมีเซต

ของจุดที่ต้องพิจารณาทั้งหมด (B) ถึง 11 จุด แต่หลังจากทำการตัดจุดที่ไม่ควรนำมาพิจารณาแล้วจะเหลือจุดสำคัญที่ต้องพิจารณา (EB) เพียง 4 จุดเท่านั้น



Transaction ID	Attribute 3	Class	Transaction ID	Attribute 3	Class
1	33	Z	12	9	X
2	21	Y	11	10	X
3	50	Y	4	19	X
4	19	X	2	21	Y
5	49	Y	10	21	X
6	55	Y	15	21	Y
7	76	Z	1	33	Z
8	80	Z	9	33	Y
9	33	Y	13	43	Y
10	21	X	5	49	Y
11	10	X	3	50	Y
12	9	X	6	55	Y
13	43	Y	7	76	Z
14	95	Z	8	80	Z
15	21	Y	14	95	Z

ภาพที่ 14 การเรียงลำดับค่าของข้อมูลที่เป็นค่าต่อเนื่อง

Transaction ID	Attribute 3	Class
12	9	X
11	10	X
4	19	X
2	21	Y
10	21	X
15	21	Y
1	33	Z
9	33	Y
13	43	Y
5	49	Y
3	50	Y
6	55	Y
7	76	Z
8	80	Z
14	95	Z

↩ 9.5 ✕

↩ 14.5 ✕

↩ 20

↩ 27

↩ 38

↩ 46 ✕

↩ 49.5 ✕

↩ 52.5 ✕

↩ 65.5

↩ 78 ✕

↩ 87.5 ✕

Set EB

20,

27,

38,

65.5

ภาพที่ 15 การหาเซตของจุดตัดสำคัญที่ต้องพิจารณา (EB) ของอัลกอริทึม DCR

อัลกอริทึม DCR จะให้รูปแบบช่วงเริ่มต้นเป็น $D = \{[d_0, d_n]\}$ หรือจากตัวอย่างนี้ก็คือ $D = \{[9, 95]\}$ และจะคำนวณค่า DCR ของจุดตัดที่ละจุดที่อยู่ในเซต EB และยังไม่อยู่ในเซต D จากนั้นจะทำการคัดเลือกจุดตัดที่ดีที่สุด ซึ่งเป็นจุดที่ให้ค่า DCR สูงที่สุดจุดแรกมาเป็นจุดแบ่งช่วงจริง โดยจากตัวอย่างนี้เราจะคำนวณค่า DCR ของจุด 20, 27, 38, และ 65.5 ได้เป็น 4.25, 3.944, 3.41 และ 4.25 ตามลำดับ ดังนั้นเราจึงทำการเลือกจุด 20 ซึ่งเป็นจุดแบ่งช่วงที่ให้ค่า DCR สูงที่สุด (ภาพที่ 16) มาเป็นจุดแบ่งช่วงจริง จึงมีผลทำให้รูปแบบช่วงใหม่เป็น $D = \{[9, 20], (20, 95]\}$ และเหลือจุดตัดที่ต้องพิจารณาในเซต EB อีก 3 จุดคือ 27, 38, และ 65.5 เป็นต้น

Class	Interval		Class Total
	$[9, 20]$	$(20, 95]$	
<i>X</i>	3	1	4
<i>Y</i>	0	7	7
<i>Z</i>	0	4	4
Interval Total	3	12	15

$$DCR = \frac{\left(\frac{3^2 + 0^2 + 0^2}{3}\right) + \left(\frac{1^2 + 7^2 + 4^2}{12}\right)}{2}$$

ภาพที่ 16 ตัวอย่างการคำนวณหาค่า DCR เมื่อรูปแบบช่วงเป็น $\{[9, 20], (20, 95]\}$

ลักษณะการทำงานของอัลกอริทึม DCR จะเป็นเช่นนี้ไปเรื่อยๆ โดยในแต่ละรอบการทำงาน จะมีจำนวนช่วงเพิ่มขึ้นมา 1 ช่วง และจะหยุดการทำงานก็ต่อเมื่อค่า DCR ที่คำนวณได้ของรูปแบบช่วงขณะนั้นมีค่าลดลงจากรอบที่ผ่านมา โดยที่มีจำนวนช่วงไม่น้อยกว่าจำนวนคลาส ซึ่งในที่นี้คือ 3 ช่วง

ในขั้นตอนสุดท้ายจะเป็นขั้นตอนของการตัดระเบียบ ซึ่งจากผลลัพธ์ที่ได้หลังจากทำการค้นหารูปแบบช่วงของคุณลักษณะที่ 3 (ตารางที่ 11) จะได้ผลลัพธ์ที่มีจำนวนช่วงเป็น 5 ช่วง และมีอยู่ 3 ช่วง คือช่วงที่ 1, 4 และ 5 ที่คลาสของของมูลในแต่ละช่วงนั้นเป็นคลาสเดียวกันทั้งหมด เราจึงทำการตัดระเบียบที่มีค่าอยู่ในช่วง 3 ช่วงนี้ จึงส่งผลให้เหลืออีกเพียง 5 ระเบียบ คือระเบียบที่ 1, 3, 5, 6, 9 และ 13 ที่จะใช้ในการค้นหารูปแบบช่วงของคุณลักษณะถัดไป

ตารางที่ 11 ผลลัพธ์ที่ได้เมื่อทำการแบ่งช่วงข้อมูลของคุณลักษณะที่ 3 ด้วยอัลกอริทึม DCR

Transaction ID	Attribute 3	Class
12	9	X
11	10	X
4	19	X
2	21	Y
10	21	X
15	21	Y
1	33	Z
9	33	Y
13	43	Y
5	49	Y
3	50	Y
6	55	Y
7	76	Z
8	80	Z
14	95	Z

6. สรุปข้อแตกต่างระหว่างอัลกอริทึม DCR กับอัลกอริทึม CAIM

จากที่ได้กล่าวมานั้น จะสังเกตเห็นว่า ขั้นตอนในการค้นหารูปแบบช่วงของข้อมูลในแต่ละคุณลักษณะของอัลกอริทึม DCR นั้น จะคล้ายกับขั้นตอนในการทำงานของอัลกอริทึม CAIM แต่จะมีส่วนที่แตกต่างกันหลักๆ คือ

6.1 มีการเพิ่มส่วนของการเรียงลำดับคุณลักษณะที่จะนำมาหารูปแบบช่วง

6.2 มีการคัดเลือกจุดตัดที่จะนำมาพิจารณา ขณะที่อัลกอริทึม CAIM จะพิจารณาจุดตัดทุกจุดที่เป็นไปได้

6.3 มีการปรับปรุงสูตรในการหาความสัมพันธ์ระหว่างรูปแบบช่วงของข้อมูลในแต่ละคุณลักษณะ กับ คลาส ให้มีประสิทธิภาพเพิ่มขึ้น

6.4 เพิ่มกระบวนการตัดระเบียบของฐานข้อมูล เพื่อลดจำนวนช่วงของผลลัพธ์ที่ได้ และช่วยลดเวลาในการทำงาน

ส่วนความแตกต่างระหว่างอัลกอริทึม DCR กับอัลกอริทึมอื่นๆ ที่เกี่ยวข้อง จะเป็นไปดังตารางที่ 12 ซึ่งเป็นตารางเปรียบเทียบคุณลักษณะต่างๆ ของแต่ละอัลกอริทึม

ตารางที่ 12 ตารางเปรียบเทียบคุณลักษณะของอัลกอริทึม DCR กับอัลกอริทึมอื่นๆ

อัลกอริทึม	จำนวนช่วงของผลลัพธ์	ประเภทของอัลกอริทึม	ลักษณะการทำงาน	การใช้ความสัมพันธ์ระหว่างคุณลักษณะ	การคัดเลือกจุดตัด	การเรียงลำดับคุณลักษณะ	การลดจำนวนช่วง
Equal Width	ผู้กำหนด	unsupervised	top-down	X	X	X	X
Equal Frequency	ผู้กำหนด	unsupervised	top-down	X	X	X	X
ChiMerge	อัตโนมัติ	supervised	bottom-up	X	X	X	X
CAIM	อัตโนมัติ	supervised	top-down	X	X	X	X
DCR	อัตโนมัติ	supervised	top-down	✓	✓	✓	✓

ผลและวิจารณ์

ผล

1. วิธีวัดผลการทดลอง

ในหัวข้อนี้จะกล่าวถึงผลการวิจัยของอัลกอริทึม DCR เมื่อนำไปเปรียบเทียบกับอัลกอริทึมในการหารูปแบบช่วงอีก 5 อัลกอริทึม ซึ่งแบ่งออกเป็น อัลกอริทึมแบบไม่มีผู้สอน (unsupervised) 2 อัลกอริทึมคือ Equal Width (EW) และ Equal Frequency (EF) โดยทั้ง 2 อัลกอริทึมนี้จะต้องมีการกำหนดค่าของจำนวนช่วงที่ต้องการ ซึ่งในที่นี้ เราจะกำหนดจำนวนช่วงโดยคำนวณจาก heuristic formula ซึ่งเป็นสูตรที่ใช้ในการค้นหาจำนวนช่วงของข้อมูล ดังที่ใช้ในการวัดผลของอัลกอริทึม CAIM และได้ทำการเปรียบเทียบกับอัลกอริทึมแบบมีผู้สอน (supervised) อีก 3 อัลกอริทึมคือ CAIM ซึ่งจากผลการทดลองของ CAIM นั้น เป็นอัลกอริทึมที่ให้ประสิทธิภาพของรูปแบบช่วงที่ดีที่สุด เมื่อเทียบกับอัลกอริทึมแบบ top-down ด้วยกัน, อัลกอริทึม ChiMerge ซึ่งเป็นอัลกอริทึมต้นแบบของอัลกอริทึมแบบ bottom-up อื่นๆ โดยอัลกอริทึมนี้จะให้ผู้ใช้ระบุค่า significance level ซึ่งในที่นี้ เรากำหนดให้เป็น 5% ดังที่ใช้ในการวัดผลของ ChiMerge และการวัดผลในงานวิจัยอื่นๆ และสุดท้ายคืออัลกอริทึมในการแบ่งช่วงโดยใช้ค่า entropy ของ WEKA ซึ่งเป็น open-source ทางด้านค่าใดมโนหนึ่งที่ได้รับความนิยมสูงสุดตัวหนึ่งในปัจจุบัน

ในการเปรียบเทียบผลการวิจัยแต่ละด้านนั้น เพื่อให้เห็นถึงผลของการเปลี่ยนแปลงในแต่ละขั้นตอนของงานวิจัยนี้ อัลกอริทึม DCR จึงแบ่งออกเป็น 3 แบบด้วยกัน คือ

- 1.1 DCR1 เป็นการเปลี่ยนแปลงเฉพาะในส่วนสูตรของการคำนวณค่าความสัมพันธ์ระหว่างรูปแบบช่วงกับคลาส ของแต่ละคุณลักษณะ
- 1.2 DCR2 เป็นการเปลี่ยนแปลงในส่วนสูตรของการคำนวณ และเปลี่ยนการหาเซตของจุดตัดที่เป็นไปได้ทั้งหมด (B) เป็นเซตของจุดตัดสำคัญที่ต้องพิจารณา (EB)
- 1.3 DCR3 เป็นอัลกอริทึม DCR ที่ได้นำเสนอทั้งหมด คือ เปลี่ยนแปลงในส่วนสูตรของการคำนวณ เปลี่ยนการหาเซตของจุดตัด และเพิ่มขั้นตอนการลดจำนวนช่วงของข้อมูลในแต่ละคุณลักษณะ

ตารางที่ 13 รายละเอียดของฐานข้อมูลแต่ละชุดจาก UCI Repository of Machine Learning Database

คุณสมบัติ	ฐานข้อมูล					
	iris	ion	thy	sat	wav	hea
จำนวนคลาส	3	2	3	6	3	2
จำนวนข้อมูล (ระเบียน)	150	351	7200	6435	3600	270
จำนวนคุณลักษณะ	4	34	21	36	21	13
จำนวนคุณลักษณะ ที่เป็นค่าต่อเนื่อง	4	32	6	36	21	6

ฐานข้อมูลที่นำมาใช้ในการวัดผลนั้น จะใช้ฐานข้อมูลจริง 6 ชุดจาก UCI Repository of Machine Learning Database (Blake and Merz, 1998) คือ Iris dataset (iris), Ionosphere dataset (ion), Ann-Thyroid dataset (thy), SatImage dataset (sat), Waveform dataset (wav) และ Heart Disease dataset (hea) ซึ่งเป็นฐานข้อมูล 6 ชุดจาก 8 ชุดที่ใช้ในการวัดผลของอัลกอริทึม CAIM ที่งานวิจัยนี้ได้ทำการหามาได้ โดยมีรายละเอียดของฐานข้อมูลแต่ละชุด ดังตารางที่ 13

2. ผลการทดลอง

การวัดผลนั้น จะทำการเปรียบเทียบในด้าน จำนวนช่วงของข้อมูลที่ได้จากการแบ่งช่วง, เวลาในการค้นหารูปแบบช่วง และ ความแม่นยำของรูปแบบช่วงเมื่อนำไปใช้ในการจำแนกคลาส ซึ่งจะแสดงให้เห็นถึงผลลัพธ์ที่ได้ในแต่ละด้านของแต่ละอัลกอริทึมในฐานข้อมูลทั้ง 6 ชุด โดยค่าที่เป็นตัวหนาจะแสดงถึงอัลกอริทึมที่ดีที่สุดในแต่ละฐานข้อมูล และค่า Score จะเป็นการนำผลลัพธ์ที่ได้นั้นหารด้วยผลลัพธ์ที่มากที่สุดในแต่ละฐานข้อมูล

ในด้านความแม่นยำของรูปแบบช่วงนั้น งานวิจัยนี้ได้นำรูปแบบช่วงที่ได้จากอัลกอริทึมในการแบ่งช่วงต่างๆ มาเป็นข้อมูลนำเข้า (input) ให้กับอัลกอริทึม C4.5 ของโปรแกรม WEKA ซึ่งเป็นอัลกอริทึมสำหรับการจำแนกประเภท แบบโครงสร้างต้นไม้ ที่ได้รับความนิยมสูงที่สุดตัวหนึ่ง ด้วยวิธี 10-fold cross validation โดยมีผลการวิจัยโดยละเอียดดังนี้

ตารางที่ 14 เปรียบเทียบจำนวนช่วงที่ได้จากการแบ่งช่วง

Discretization Algorithm	UCI Repository of Machine Learning Database												Average Score
	Iris		ion		thy		sat		wav		hea		
	Interval	Score	Interval	Score	Interval	Score	Interval	Score	Interval	Score	Interval	Score	
EW	16	1.00	640	1.00	126	1.00	252	0.34	630	0.79	54	1.00	0.854
EF	16	1.00	640	1.00	126	1.00	252	0.34	630	0.79	53	0.98	0.851
WEKA	14	0.88	147	0.23	37	0.29	512	0.68	100	0.12	10	0.19	0.398
ChiMerge	15	0.94	398	0.62	117	0.93	752	1.00	801	1.00	33	0.61	0.850
CAIM	12	0.75	64	0.10	18	0.14	216	0.29	63	0.08	12	0.22	0.263
DCR1	12	0.75	64	0.10	18	0.14	216	0.29	63	0.08	12	0.22	0.263
DCR2	12	0.75	64	0.10	18	0.14	216	0.29	63	0.08	12	0.22	0.263
DCR3	9	0.56	64	0.10	18	0.14	154	0.20	62	0.08	12	0.22	0.218

ตารางที่ 15 เปรียบเทียบเวลาที่ใช้ในการค้นหารูปแบบช่วงของข้อมูล (วินาที)

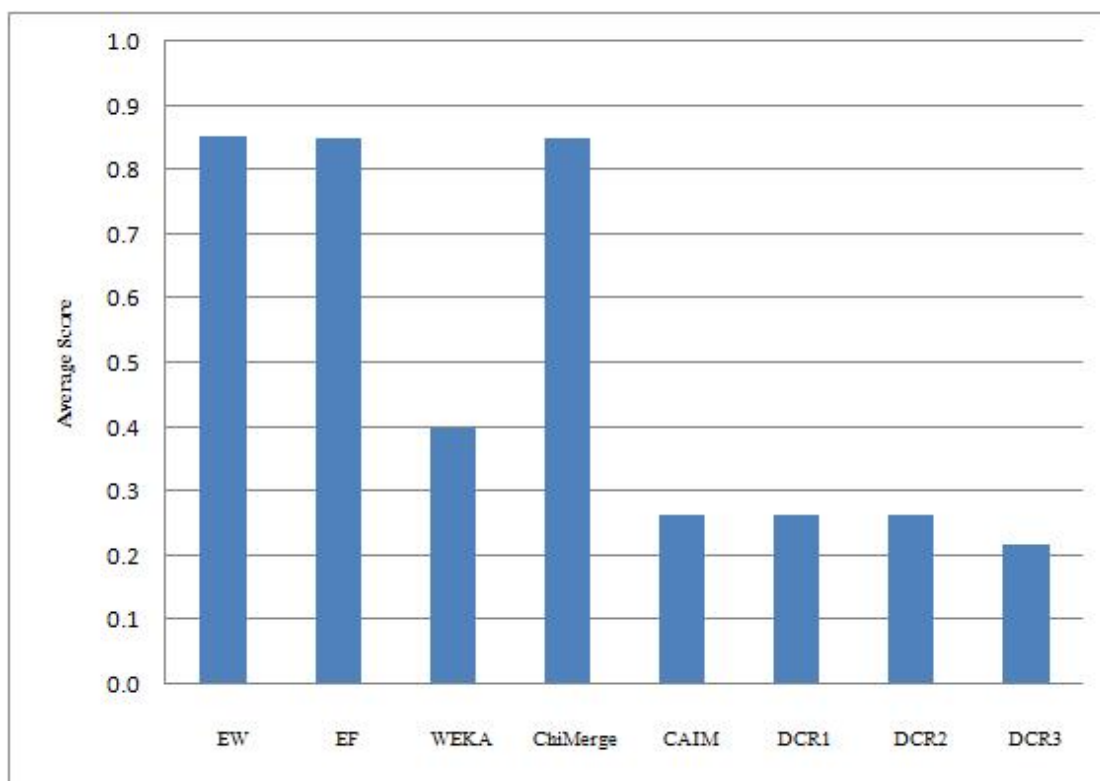
Discretization Algorithm	UCI Repository of Machine Learning Database												Average Score
	iris		ion		thy		sat		wav		hea		
	Time	Score	Time	Score	Time	Score	Time	Score	Time	Score	Time	Score	
EW	0.031	0.01	1.406	0.05	122.578	0.39	445.547	0.64	94.172	0.24	0.141	0.04	0.228
EF	0.032	0.01	1.297	0.05	122.937	0.39	435.046	0.62	94.188	0.24	0.156	0.04	0.225
ChiMerge	3.266	1.00	28.625	1.00	314.359	1.00	646.547	0.92	212.000	0.54	3.703	1.00	0.911
CAIM	1.735	0.53	24.171	0.84	246.500	0.78	678.641	0.97	389.047	1.00	2.766	0.75	0.813
DCR1	1.688	0.52	21.750	0.76	252.281	0.80	700.094	1.00	371.547	0.96	3.078	0.83	0.811
DCR2	0.860	0.26	17.953	0.63	218.563	0.70	595.109	0.85	344.765	0.89	2.125	0.57	0.649
DCR3	0.625	0.19	9.000	0.31	273.547	0.87	591.875	0.85	270.734	0.70	2.375	0.64	0.593

ตารางที่ 16 เปรียบเทียบความแม่นยำของรูปแบบช่วงที่ได้เมื่อนำไปใช้ในการจำแนกคลาส

Discretization Algorithm	UCI Repository of Machine Learning Database												Average Score
	iris		ion		thy		sat		wav		hea		
	Accuracy	Score	Accuracy	Score	Accuracy	Score	Accuracy	Score	Accuracy	Score	Accuracy	Score	
EW	94.667	0.97	90.883	0.96	94.764	0.95	86.371	1.00	67.167	0.85	73.333	0.90	0.939
EF	94.667	0.97	90.598	0.96	98.319	0.99	84.880	0.98	66.944	0.85	74.074	0.90	0.943
Built-in C5.0	95.302	0.98	90.857	0.96	99.708	1.00	85.934	0.99	75.854	0.96	78.810	0.96	0.977
WEKA	96.000	0.99	89.744	0.95	99.361	1.00	84.880	0.98	76.222	0.97	81.852	1.00	0.981
ChiMerge	97.333	1.00	92.023	0.98	98.528	0.99	83.388	0.96	71.611	0.91	76.296	0.93	0.962
CAIM	94.000	0.97	91.453	0.97	92.639	0.93	85.843	0.99	77.000	0.98	77.407	0.95	0.963
DCR1	94.000	0.97	92.308	0.98	92.611	0.93	86.434	1.00	77.556	0.98	78.519	0.96	0.970
DCR2	94.000	0.97	92.308	0.98	96.556	0.97	86.434	1.00	77.556	0.98	78.519	0.96	0.976
DCR3	94.667	0.97	94.302	1.00	96.556	0.97	85.952	0.99	78.778	1.00	78.519	0.96	0.982

วิจารณ์

1. จำนวนช่วงที่ได้จากการแบ่งช่วง

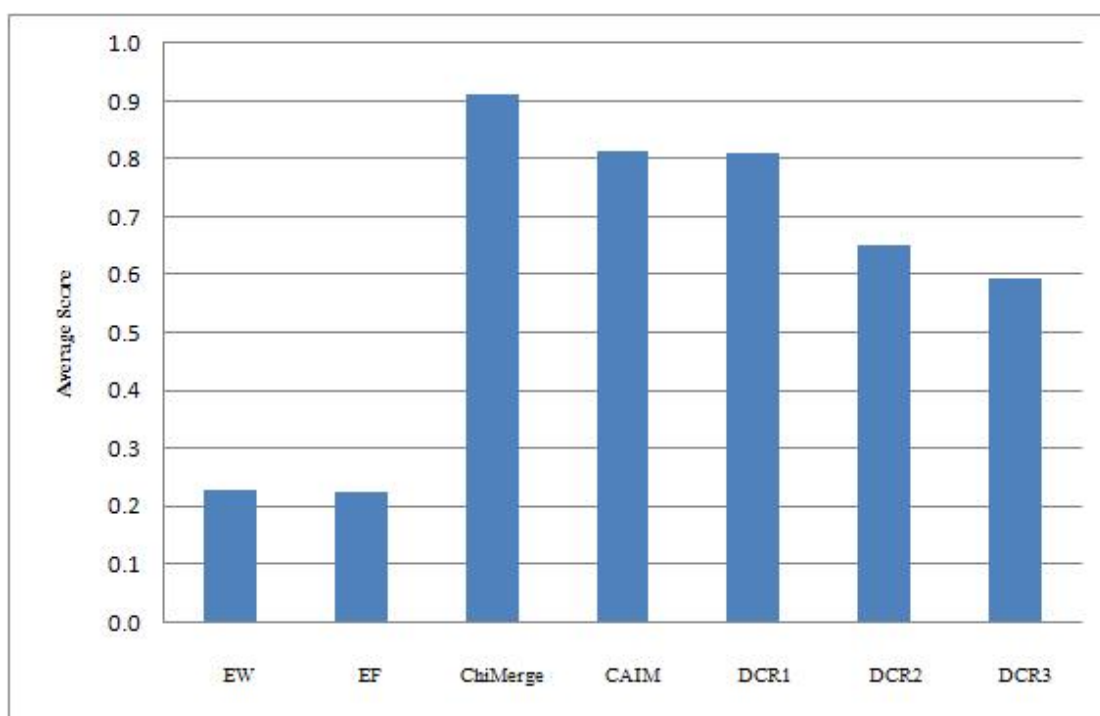


ภาพที่ 17 เปรียบเทียบ Average score ของจำนวนช่วงที่ได้จากการแบ่งช่วง

เนื่องจาก DCR1 เป็นการปรับปรุงอัลกอริทึม CAIM ในส่วนสูตรของการหาค่าความสัมพันธ์ระหว่างรูปแบบช่วงกับคลาสของแต่ละคุณลักษณะ ซึ่งมีจุดประสงค์ในการหารูปแบบช่วงที่มีประสิทธิภาพเพิ่มขึ้น แต่จากตารางที่ 14 แม้ว่าอัลกอริทึม CAIM และ DCR1 จะให้จำนวนช่วงที่เท่ากันในทุกฐานข้อมูล แต่ในข้อเท็จจริงแล้วผลลัพธ์ที่ได้นั้นมีรูปแบบช่วงที่ต่างกัน ดังนั้นจึงจะเห็นได้จากผลของความแม่นยำในการจำแนกประเภท เมื่อนำรูปแบบช่วงที่ได้ไปเป็นข้อมูลนำเข้าของอัลกอริทึม C4.5 (ตารางที่ 16) และเนื่องจาก DCR2 เป็นการปรับปรุง DCR1 ในส่วนของการลดจำนวนจุดตัดที่จะนำมาพิจารณา เพื่อลดเวลาในการทำงานเท่านั้น รูปแบบช่วงที่ได้จึงไม่มีความแตกต่างกัน แต่ใช้เวลาในการที่ลดลง ดังตารางที่ 15

จากภาพที่ 17 จะเห็นว่า โดยเฉลี่ยอัลกอริทึม DCR3 สามารถหารูปแบบช่วงที่มีจำนวนช่วงรวมได้น้อยที่สุด รองลงมาจะเป็น DCR1, DCR2 และ CAIM ซึ่งได้จำนวนช่วงรวมที่เท่ากันในทุกฐานข้อมูล จากนั้นจะเป็น supervised discretization algorithm ของโปรแกรม WEKA, อัลกอริทึม ChiMerge, Equal Frequency และ Equal Width ตามลำดับ ซึ่งการที่ได้จำนวนช่วงรวมที่น้อยกว่านั้น มีผลดีคือ จะช่วยลดปริมาณข้อมูลที่จะนำไปใช้ในการจำแนกคลาสได้ ทำให้ใช้เวลาในการสร้างโมเดลจำแนกประเภทข้อมูลที่ลดลง อีกทั้งยังทำให้เข้าใจความหมายของคุณลักษณะที่เป็นค่าต่อเนื่องต่างๆ ได้ง่ายขึ้นอีกด้วย

2. เวลาที่ใช้ในการค้นหารูปแบบช่วงของข้อมูล

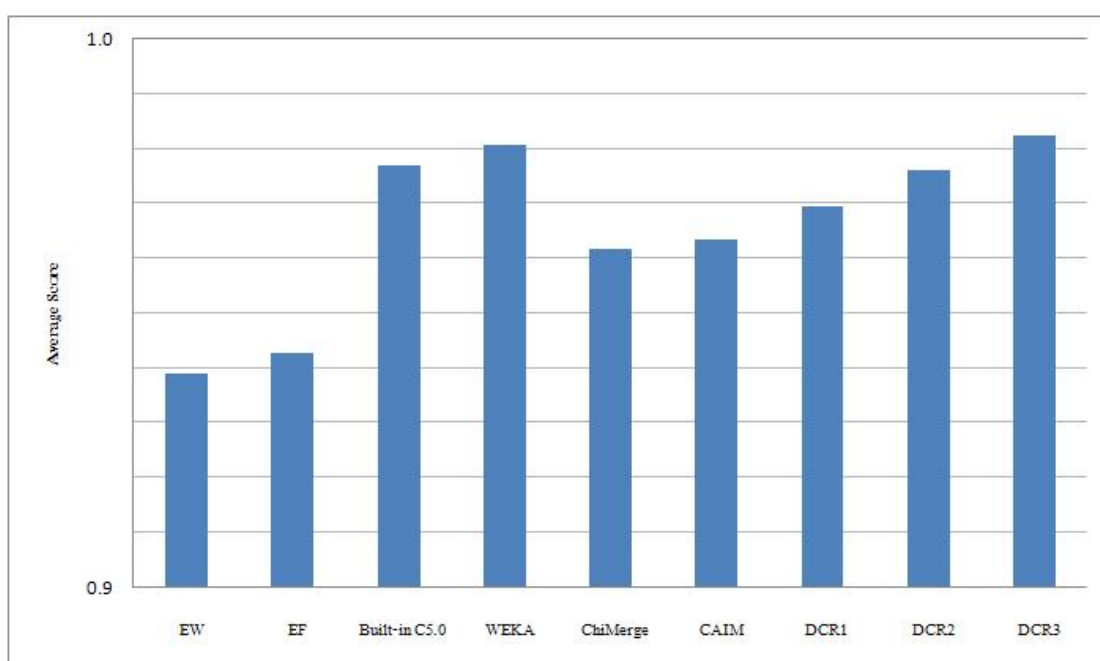


ภาพที่ 18 เปรียบเทียบ Average score ของเวลาที่ใช้ในการค้นหารูปแบบช่วงของข้อมูล

ในการเปรียบเทียบด้านเวลาที่ใช้ในการค้นหารูปแบบช่วงของข้อมูลในหัวข้อนี้ ไม่ได้นำอัลกอริทึมการแบ่งช่วงของ WEKA มาเปรียบเทียบกับ เนื่องจากความแตกต่างทางด้านภาษาที่ใช้ในการเขียน จึงไม่สามารถนำมาเปรียบเทียบกันได้

จากภาพที่ 18 จะพบว่าอัลกอริทึมที่เป็นแบบไม่มีผู้สอน ทั้ง 2 อัลกอริทึม จะใช้เวลาในการทำงานน้อยที่สุด เพราะไม่มีการนำความสัมพันธ์ระหว่างคลาส กับ คุณลักษณะต่างๆ มาพิจารณา จึงทำให้ได้ความแม่นยำในการจำแนกประเภทที่ไม่สูงมากนัก แต่สำหรับอัลกอริทึมแบบมีผู้สอน ทั้ง 5 อัลกอริทึม นั้น อัลกอริทึม ChiMerge จะใช้เวลาในการทำงานเฉลี่ยมากที่สุด รองลงมาจะเป็น อัลกอริทึม CAIM และ DCR1 ซึ่งใช้เวลาในการทำงานเฉลี่ยพอกัน ตามด้วยอัลกอริทึม DCR2 และอัลกอริทึม DCR3 ซึ่งใช้เวลาในการทำงานน้อยที่สุดตามลำดับ

3. ความแม่นยำของรูปแบบช่วงที่ได้



ภาพที่ 19 เปรียบเทียบ Average score ของความแม่นยำของรูปแบบช่วงที่ได้เมื่อนำไปใช้ในการจำแนกคลาส

เนื่องจากอัลกอริทึม C4.5 นั้น ได้รวมอัลกอริทึมในการแบ่งช่วงไว้ภายในแล้ว จึงสามารถรองรับข้อมูลที่เป็นค่าต่อเนื่องได้ ดังนั้น ผลการวิจัยในส่วนนี้ จึงได้นำผลของความแม่นยำที่ได้จากอัลกอริทึม C4.5 เมื่อใช้ข้อมูลนำเข้าที่เป็นค่าต่อเนื่องโดยตรง (Built-in C4.5) มาร่วมพิจารณาด้วย

จากผลการวิจัยในตารางที่ 16 และภาพที่ 19 จะพบว่า โดยเฉลี่ยแล้ว อัลกอริทึม DCR3 เป็นอัลกอริทึมที่ได้รูปแบบช่วงของข้อมูลที่ทำให้ความแม่นยำในการจำแนกคลาสูงที่สุด แม้ว่าจะมี

จำนวนช่วงที่น้อยกว่าอัลกอริทึมอื่นๆ รองลงมาจะเป็นอัลกอริทึมในการค้นหารูปแบบช่วงของ WEKA ซึ่งมีความแม่นยำที่ใกล้เคียงกับอัลกอริทึมที่ได้นำเสนอ

สรุปและข้อเสนอแนะ

สรุป

งานวิจัยนี้ได้นำเสนอ อัลกอริทึม DCR ซึ่งเป็นอัลกอริทึมในการแปลงคุณลักษณะที่เป็นค่าต่อเนื่องให้เป็นค่าไม่ต่อเนื่อง โดยจัดเป็นอัลกอริทึมแบบมีผู้สอน ซึ่งจากหัวข้อที่ผ่านมาจะเห็นว่า อัลกอริทึม DCR เป็นอัลกอริทึมที่ได้จำนวนช่วงรวมของคุณลักษณะที่เป็นค่าต่อเนื่องน้อยที่สุด โดยให้ความแม่นยำเฉลี่ยในการนำไปใช้จำแนกคลาสของข้อมูลสูงที่สุด และเมื่อไม่พิจารณาอัลกอริทึมแบบไม่มีผู้สอนแล้ว อัลกอริทึม DCR จะใช้เวลาในการทำงานโดยเฉลี่ยน้อยที่สุด เมื่อเทียบกับ อัลกอริทึมที่ใช้แบ่งช่วงข้อมูลแบบมีผู้สอนด้วยกัน โดยใช้ฐานข้อมูลมาตรฐานทั้งหมด 6 ชุด

ข้อเสนอแนะ

เนื่องจากอัลกอริทึม DCR ยังมีข้อบกพร่องอยู่ด้วยกันหลักๆ 4 ด้านคือ

1. อัลกอริทึม DCR ยังไม่ได้้นำความสัมพันธ์ระหว่างคลาสกับคุณลักษณะที่เป็นค่าไม่ต่อเนื่อง มาร่วมพิจารณาในการค้นหารูปแบบช่วงด้วย ซึ่งถ้าเป็นฐานข้อมูลแบบผสมที่มีคุณลักษณะที่เป็นค่าไม่ต่อเนื่องอยู่เป็นจำนวนมากแล้ว อาจทำให้รูปแบบช่วงที่ได้ไม่มีประสิทธิภาพมากนัก ดังนั้นถ้านำคุณลักษณะที่เป็นค่าไม่ต่อเนื่องมาร่วมพิจารณาด้วยแล้ว อาจช่วยในการลดจำนวนช่วงของผลลัพธ์ หรือเป็นการตัดคุณลักษณะที่ไม่จำเป็นอื่นๆออกไปได้อีกด้วย

2. อัลกอริทึม DCR จะใช้ค่าเกนสารสนเทศ (information gain) ในขั้นตอนในการเรียงลำดับคุณลักษณะที่จะทำการแบ่งช่วง ซึ่งอาจให้ผลที่ไม่ดีมากนัก ถ้าเป็นข้อมูลแบบ imbalance data คือ ฐานข้อมูลที่มีจำนวนคลาสแต่ละคลาสด่างกันมากๆ ดังนั้นอาจเปลี่ยนวิธีการเรียงลำดับคุณลักษณะใหม่ หรือ ปรับปรุงสูตรของการคำนวณใหม่ให้เหมาะสมกับลักษณะข้อมูลประเภทนี้ (Pham *et al.*, 2008)

3. สูตรในการคำนวณหาความสัมพันธ์ระหว่างรูปแบบช่วงกับคลาสของแต่ละคุณลักษณะ ยังขึ้นกับจำนวนช่วงขณะนั้นมากเกินไป ซึ่งจะทำให้ค่าของ DCR ลดลงอย่างรวดเร็ว เมื่อมีจำนวนช่วงเพิ่มขึ้น จึงอาจทำให้ได้รูปแบบช่วงที่ได้น้อยกว่าที่ควรจะเป็น ดังนั้น จึงควร

ปรับปรุงในส่วนสูตรของการคำนวณใหม่ให้เหมาะสม (Tsai *et al.*, 2008) หรืออาจใช้การคำนวณค่าที่ไม่ขึ้นกับจำนวนช่วงแทน เช่นค่า F-Measure เป็นต้น

4. เกณฑ์ในการหยุดของอัลกอริทึม DCR นั้นคือการที่ค่า DCR ในรอบปัจจุบันมีค่าที่ลดลงจากรอบก่อนหน้า โดยขณะนั้นต้องมีจำนวนช่วงอย่างน้อยเท่ากับจำนวนคลาส ดังนั้น จากการที่ค่า DCR ลดลงอย่างรวดเร็วเมื่อมีจำนวนช่วงเพิ่มขึ้น จึงอาจทำให้ได้รูปแบบช่วงที่ได้น้อยกว่าที่ควรจะเป็นเช่นกัน อาจแก้ไขได้โดยการปล่อยให้มีการลดลงของค่า DCR ระดับหนึ่ง จึงหยุดการทำงาน เป็นต้น

เอกสารและสิ่งอ้างอิง

- กฤษณะ ไวยมัย. 2550. เอกสารคำสอนวิชา 214554 คลังข้อมูลและการทำเหมืองข้อมูล. ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์.
- ชลนิศา สารระ. 2550. การจำแนกกลุ่มสถานภาพการสำเร็จการศึกษาโดยแบบจำลองต้นไม้ตัดสินใจ. วิทยานิพนธ์ปริญญาโท, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.
- ปรัชญา ผ่องอักษร, ธนาวิทย์ รักธรรมานนท์ และ กฤษณะ ไวยมัย. 2551. การค้นหารูปแบบช่วงของข้อมูลโดยใช้คลาสเพื่อลดปริมาณช่วงของรูปแบบที่ได้, น. 552-559. ใน **National Computer Science and Engineering Conference ครั้งที่ 12.**
- วีระพล หาญโชติช่วง. 2549. ระบบการจัดกลุ่มและทำนายข้อมูลโดยใช้กฎความสัมพันธ์. วิทยานิพนธ์ปริญญาโท, มหาวิทยาลัยเกษตรศาสตร์.
- Catlett, J. 1991. On changing continuous attributes into ordered discrete attributes, pp. 164-178. **Proc. European Working Session on Learning.**
- Blake, C. and C. Merz. 1998. **UCI repository of machine learning database** Department of Information and Computer Science, University of California, Irvine, Irvine, CA.
Available Source: <http://www.cs.uci.edu/~mlearn/MLRepository.html>
- Ching, J.Y., A.K.C. Wong and K.C.C. Chan. 1995. Class-Dependent Discretization for Inductive Learning from Continuous and Mixed Mode Data, pp. 641-651. **IEEE Trans. Pattern Analysis and Machine Intelligence.**
- Cios, K.J. and L. Kurgan. 2001. Hybrid Inductive Machine Learning: An Overview of CLIP Algorithms, pp. 276-322. In L.C. Jain and J. Kacprzyk, eds. **New Learning Paradigms in Soft Computing.**

- Clark, P. and T. Niblett. 1989. The CN2 Algorithm, pp. 261-283. **Machine Learning**.
- Cost, S. and S. Salzberg. 1993. A Weighted Nearest Neighbor Algorithm for Learning with Symbolic Features, pp. 57-78. **Machine Learning**.
- Dougherty, J., R. Kohavi and M. Sahami. 1995. Supervised and unsupervised discretization of continuous features, pp. 194-202. **Proceeding of Twelfth International Conference on Machine Learning**.
- Fayyad, U.M. and K.B. Irani. 1992. On the handling of continuous-valued attributes in decision tree generation, pp. 87-102. **Machine Learning**.
- Fayyad, U.M. and K.B. Irani. 1993. Multi-interval discretization of continuous-valued attributes for classification learning, pp. 1022-1027. **Proceeding of Thirteenth International Conference on Artificial Intelligence**.
- Ho, K.M. and P.D. Scott. 1997. Zeta: A Global Method for Discretization of Continuous Variables, pp. 191-194. **KDD97: 3rd International Conference of Knowledge Discovery and Data Mining**. Newport Beach, CA.
- Holte, R.C. 1993. Very Simple Classification Rules Perform Well on Most Commonly Used Datasets, pp. 63-90. **Machine Learning**.
- Ian, H. Witten. and F. Eibe. 2005. Data Mining : Practical Machine Learning Tools and Techniques with Java Implementations. 2nd ed. California: Morgan Kaufmann.
- Kaufman, K.A. and R.S. Michalski. 1999. Learning from Inconsistent and Noisy Data: The AQ18 Approach. **Methodologies for Intelligent Systems**.

- Kerber, R. 1992. Chimerge: Discretization of Numeric Attributes, pp. 123-128. **Proc. AAAI-92, Ninth National Conference Artificial Intelligence.**
- Kurgan, L. and K.J. Cios. 2004. CAIM Discretization Algorithm, pp. 145-153. **IEEE Trans. Knowledge And Data Engineering.**
- Lee, C.I., C.J. Tsai, Y.R. Yang and W.P. Yang. 2007. A top-down and greedy method for discretization of continuous attributes. **Proceedings of Fourth International Conference on Fuzzy Systems and Knowledge Discovery.** Haikou, China.
- Liu, H., F. Hussain, C.L. Tan and M. Dash. 2002. Discretization: an enabling technique, pp. 393-423. **Journal of Data Mining and Knowledge Discovery.**
- Liu, H. and R. Setiono. 1995. Chi2: Feature Selection and Discretization of Numeric Attributes, pp. 388-391. **IEEE Computer Society.**
- Nguyen, H.S. and S.H. Nguyen. 1998. Discretization Methods in Data Mining, pp. 451-482. *In* L. Polkowski and A. Skowron, eds. **Rough Sets in Knowledge Discovery.** Physica.
- Paterson, A. and T.B. Niblett. 1987. *In* ACLS manual. **Intelligent Terminals, Ltd.** Edinburgh.
- Pfahring, B. 1995. Compression-based discretization of continuous attributes, pp. 456-463. **Proceeding of Twelfth International Conference on Machine Learning.**
- Pham N.K., T.N. Do, P. Lenca and S. Lallich. 2008. Using Local Node Information in Decision Trees: Coupling a Local Decision Rule with an Off-Centered Entropy, pp. 117-123. *In* R. Stahlbock, S.F. Crone and S. Lessmann ,eds. **The International Conference on Data Mining.** CSREA Press, Las Vegas, Nevada, USA.

- Pongaksorn, P., T. Rakthanmanon and K. Waiyamai. 2009. DCR: Discretization using class information to reduce number of intervals. **Quality issues, measures of interestingness and evaluation of data mining models (QIMIE'09)**.
- Quinlan, J.R. 1993. C4.5: Programs for machine learning. **Morgan Kaufmann**.
- Tay, F. and L. Shen. 2002. A modified chi2 algorithm for discretization, pp.666-670. **IEEE Transactions on Knowledge and Data Engineering**.
- Tsai C.J., C.I. Lee and W.P. Yang. 2008. A discretization algorithm based on Class-Attribute Contingency Coefficient, pp. 714-731. **Information Sciences 178**.
- Wang, K., S. Zhou and Y. He. 2000. Growing decision tree on support-less association rules. **Proceedings of the 6th International Conference on Knowledge Discovery and Data Mining (KDD'00)**. Boston, MA.
- Wong, A.K.C. and D.K.Y. Chiu. 1987. Synthesizing Statistical Knowledge from Incomplete Mixed-Mode Data, pp. 796-805. **IEEE Trans. Pattern Analysis and Machine Intelligence**.

ประวัติการศึกษา และการทำงาน

ชื่อ –นามสกุล	นายปรัชญา ผ่องอักษร
วัน เดือน ปี ที่เกิด	วันที่ 22 ธันวาคม 2525
สถานที่เกิด	กรุงเทพมหานคร
ประวัติการศึกษา	วศ.บ. (วิศวกรรมไฟฟ้า) คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ (พ.ศ.2548)
ตำแหน่งหน้าที่การงานปัจจุบัน	
สถานที่ทำงานปัจจุบัน	
ผลงานดีเด่นและรางวัลทางวิชาการ	
ทุนการศึกษาที่ได้รับ	