การเรียนรู้ร่วมกันสำหรับปัญหาการจำแนกข้อมูลไม่สมดุล

นางสาวภาสพิชญ์ ชูใจ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2557

ENSEMBLE LEARNING FOR IMBALANCED DATA CLASSIFICATION PROBLEM

Pasapitch Chujai

A Thesis Submitted in Partial Fulfillment of the Requirements forthe

Degree of Doctorof Philosophy in Computer Engineering

Suranaree University of Technology

Academic Year 2014

การเรียนรู้ร่วมกันสำหรับปัญหาการจำแนกข้อมูลไม่สมดุล

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

_____

(รศ.ดร.กิตติศักดิ์เกิดประสพ)
ประธานกรรมการ

_____

(รศ.ดร.นิตยา เกิดประสพ)
กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

_____

(ผศ.ดร.ศุภกฤษฎิ์นิวัฒนากูล)
กรรมการ

_____

(ผศ. ดร.สายสุนีย์จับโจร)
กรรมการ

_____

(ดร.ขุนเสกเสกขุนทด)
กรรมการ

_____

(ศ.ดร.ชูกิจลิมปิจำนงค์)
รองอธิการบดีฝ่ายวิชาการและนวัตกรรม

_____

(รศ.ร.อ.ดร.กนต์ธรชำนิประศาสน์)
คณบดีสำนักวิชาวิศวกรรมศาสตร์

ภาสพิชญ์ชูใจ : การเรียนรู้ร่วมกันสำหรับปัญหาการจำแนกข้อมูลไม่สมดุล (ENSEMBLE LEARNING FOR IMBALANCED DATA CLASSIFICATION PROBLEM)อาจารย์ที่ปรึกษา:รองศาสตราจารย์ ดร.นิตยาเกิดประสพ, 130 หน้า.

ข้อมูลไม่สมดุลเป็นข้อมูลที่สามารถพบเจอได้จริงในชีวิตประจำวัน เช่น ข้อมูลการวินิจฉัยโรคที่พบได้ยากทางด้านการแพทย์ เมื่อนำข้อมูลเหล่านี้มาใช้งานทางด้านการเรียนรู้ของเครื่องจักรและการทำเหมืองข้อมูลจะส่งผลกระทบต่อการเรียนรู้ของอัลกอริทึม เนื่องจากข้อมูลที่ใช้ในการเรียนรู้มีและเป็นกลุ่มที่ให้ความสนใจมีจำนวนข้อมูลที่น้อยมากเมื่อเทียบกับกลุ่มอื่น ๆ ที่เหลืออัลกอริทึมทางด้านการเรียนรู้ของเครื่องจักรนั้นสามารถทำงานได้ดีในกรณีที่ข้อมูลสมดุล สำหรับข้อมูลไม่สมดุลนั้นขอบเขตของการตัดสินใจของอัลกอริทึมการเรียนรู้ของเครื่องจักรนั้นจะมีความเอนเอียงไปทางกลุ่มข้อมูลส่วนมากส่งผลให้การจัดกลุ่มของข้อมูลส่วนน้อยมีแนวโน้มที่จะได้รับการจัดกลุ่มที่ผิดประเภทดังนั้นงานวิจัยนี้จึงนำเสนออัลกอริทึมใหม่ชื่อว่า EnsDTV (Ensemble_Learning_with_DecisionTree_Visualization) เพื่อแก้ปัญหาการจำแนกประเภทข้อมูลไม่สมดุลที่ข้อมูลอาจจะมีอัตราความไม่สมดุลของกลุ่มข้อมูลสูงและมีอัตราการซ้อนทับกันของกลุ่มข้อมูลที่แตกต่างกันด้วยการนำวิธีการเรียนรู้ร่วมกันแบบการใช้การตัดสินใจร่วมกันทั้งแบ็กกิง (Bagging)และบูสต์ติง (Boosting)มาทำการสร้างโมเดล ชดเชยการทำนายข้อมูลผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่าย (Cost-Sensitive Learning)ด้วยการนำค่าจากตารางค่าใช้จ่ายมาใช้ในขั้นตอนการเรียนรู้และปรับพารามิเตอร์ของการเรียนรู้ร่วมกัน ใช้โครงสร้างต้นไม้ตัดสินใจ (Decision Tree) เป็นเครื่องมือในการจำแนกข้อมูลพร้อมทั้งปรับลดจำนวนต้นไม้ตัดสินใจด้วยวิธีการสร้างมโนภาพหรือวิชวลไลเซชัน (Visualization) และการเตรียมข้อมูลให้เหมาะสมด้วยการลดการใช้พื้นที่ร่วมกันให้เบาบางลง ผลที่ได้ปรากฏว่า เมื่อนำวิธีการที่นำเสนอมาทำงานกับชุดข้อมูลที่มีการลดอัตราการซ้อนทับกันของกลุ่มข้อมูลที่เหมาะสมแล้วพบว่าสามารถนำมาใช้แก้ปัญหาการจำแนกประเภทข้อมูลไม่สมดุลที่มีอัตราความไม่สมดุลที่สูงและมีอัตราการซ้อนทับที่แตกต่างกันได้อย่างมีประสิทธิภาพ โดยเฉพาะโมเดลที่มีการเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบบูสต์ติงนั้นจะให้ประสิทธิภาพในการจำแนกประเภทข้อมูลกลุ่มส่วนน้อยได้ดีกว่าโมเดลที่มีการเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบแบ็กกิง ในขณะที่แบ็กกิงนั้นจะไม่สามารถทำงานได้เมื่อข้อมูลมีอัตราความไม่สมดุลที่สูงและมีอัตราการซ้อนทับที่ต่ำ

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ลายมือชื่อนักศึกษา

ปีการศึกษา2557 ลายมือชื่ออาจารย์ที่ปรึกษา

PASAPITCH CHUJAI : ENSEMBLE LEARNING FOR IMBALANCED DATA CLASSIFICATION PROBLEM.THESIS ADVISOR: ASSOC. PROF.NITTAYA KERDPRASOP, Ph.D., 130 PP.

ENSEMBLE LEARNING/ DECISITION TREE / IMBALANCED DATA/ COST-SENSITIVE LEARNING/ VISUALIZATION

Imbalanced data area kind of data that can be found in real life, such as rare case in medical diagnosis. When used in machine learning and data mining,these data will affect the learning performance of algorithms.This is due to the amount of instances in the group of interest is much smaller than the other groups. In the field of machine learning, when data are balanced, a learning algorithm can be applied efficiently in terms ofoverall classification accuracy. For unbalanced data, the boundary of decision of most learning algorithms tend to bias toward the majority class andthe classification in the minority class will be misclassified. Therefore, we present a new technique called EnsDTV (Ensemble_Learning_with_DecisionTree_Visualization) for dealing with imbalanced classification problem: high imbalanced ratio and different overlapped ratio.To solve this problem,we apply the ensemble learningusing both bagging and boosting techniques to build models. We compensate themisclassification with cost sensitive learning and then use the value from cost matrix in the learning process to adjust the parameters of the ensemble learning. We adopt decision tree algorithm for data classification and reduce the number of decision trees by visualization.We prepared

optimal imbalanced dataset by reducing an overlapped region. The results showed that the proposed method work with datasets

thatreduction of the overlapped region then sends the EnsDTV method can solve the imbalanced data classification problem efficiently which high imbalance ratio and different overlapped ratio. Especially the ensemble learning using boosting techniques will enhance the classification minority classbetter than bagging. While the ensemble learning using bagging techniques cannot work in both case of high imbalance ratio and low overlapped ratio.

School of Computer Engineering Student's Signature

Academic Year2014 Advisor's Signature

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณ บุคคล และกลุ่มบุคคลต่างๆที่ได้กรุณาให้คำปรึกษา แนะนำ ช่วยเหลืออย่างดียิ่ง ทั้งในด้านวิชาการ และด้านการดำเนินงานวิจัย ดังต่อไปนี้

รองศาสตราจารย์ ดร.นิตยาเกิดประสพ อาจารย์ที่ปรึกษาวิทยานิพนธ์ และรองศาสตราจารย์ ดร.กิตติศักดิ์เกิดประสพ ที่ให้คำปรึกษาในการทำงานวิจัย การจัดรูปแบบ และช่วยตรวจทานความถูกต้องของวิทยานิพนธ์

คุณกัลญาพับโพธิ์ เลขานุการสาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่ให้ความช่วยเหลือในการประสานงานด้านเอกสารระหว่างศึกษาคุณกิตติพงศ์ชมบุญที่ช่วยตรวจทานความถูกต้องของวิทยานิพนธ์และนักศึกษามหาบัณฑิตสาขาวิชาวิศวกรรมคอมพิวเตอร์ทุกท่านที่ให้คำปรึกษา

สำนักงานคณะกรรมการอุดมศึกษาที่ช่วยสนับสนุนทุนการศึกษาทุนวิจัยและค่าใช้จ่าย

นอกจากนี้ขอขอบคุณครูอาจารย์ทั้งในอดีตและปัจจุบันที่ให้ความรู้แก่ผู้วิจัยจนประสบความสำเร็จในชีวิต

ท้ายที่สุดขอกราบขอบพระคุณ บิดา มารดา ที่ให้กำเนิด อบรม เลี้ยงดูและส่งเสริมการศึกษาเป็นอย่างดีทำให้ผู้วิจัยมีความรู้ ความสามารถ มีจิตใจที่เข้มแข็ง รวมทั้งเป็นกำลังใจแก่ผู้วิจัยจนทำให้ผู้วิจัยประสบความสำเร็จในชีวิต

ภาสพิชญ์ชูใจ

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

ตารางที่ **หน้า**

สารบัญตาราง (ต่อ)

ตารางที่ **หน้า**

สารบัญรูป

รูปที่ **หน้า**

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

บทที่ 1
บทนำ

สำหรับบทนี้จะกล่าวถึงปัญหาและที่มาของการวิจัย วัตถุประสงค์ ขอบเขต และประโยชน์ที่ได้รับจากการวิจัย

1.1 ความสำคัญและที่มาของปัญหาการวิจัย

เทคโนโลยีสารสนเทศในปัจจุบันถือได้ว่าเป็นตัวขับเคลื่อนที่สำคัญที่จะช่วยให้ผู้ใช้สามารถเก็บรวบรวมข้อมูล ประมวลผลเพื่อนำไปใช้ แก้ไขปรับปรุงให้ทันสมัยค้นหาข้อมูลตามความต้องการ และวิเคราะห์ข้อมูลซึ่งการวิเคราะห์ข้อมูลนั้นสามารถทำได้ง่ายสะดวก และรวดเร็ว แต่การวิเคราะห์ข้อมูลสารสนเทศจากฐานข้อมูลเดียวนั้นบางครั้งความรู้ที่ได้อาจจะไม่เพียงพอต่อความต้องการ จึงจำเป็นจะต้องวิเคราะห์ข้อมูลที่ได้มาจากการรวบรวมข้อมูลหลาย ๆ ฐานข้อมูลเข้าด้วยกันเพื่อให้มีประสิทธิภาพและประโยชน์สูงสุด ข้อมูลจากหลาย ๆ ฐานข้อมูลนี้เรียกว่าคลังข้อมูล (Data Warehouse) ซึ่งในการวิเคราะห์ข้อมูลเหล่านี้จำเป็นจะต้องใช้เครื่องมือทางด้านการทำเหมืองข้อมูล (Data Mining) เข้ามาช่วยเพื่อพยากรณ์ความรู้ที่เรียนรู้มาจากข้อมูลที่มีอยู่ให้ทราบถึงสิ่งที่จะเกิดขึ้นในอนาคต และสามารถค้นพบความรู้ที่ซ่อนอยู่ในข้อมูลได้

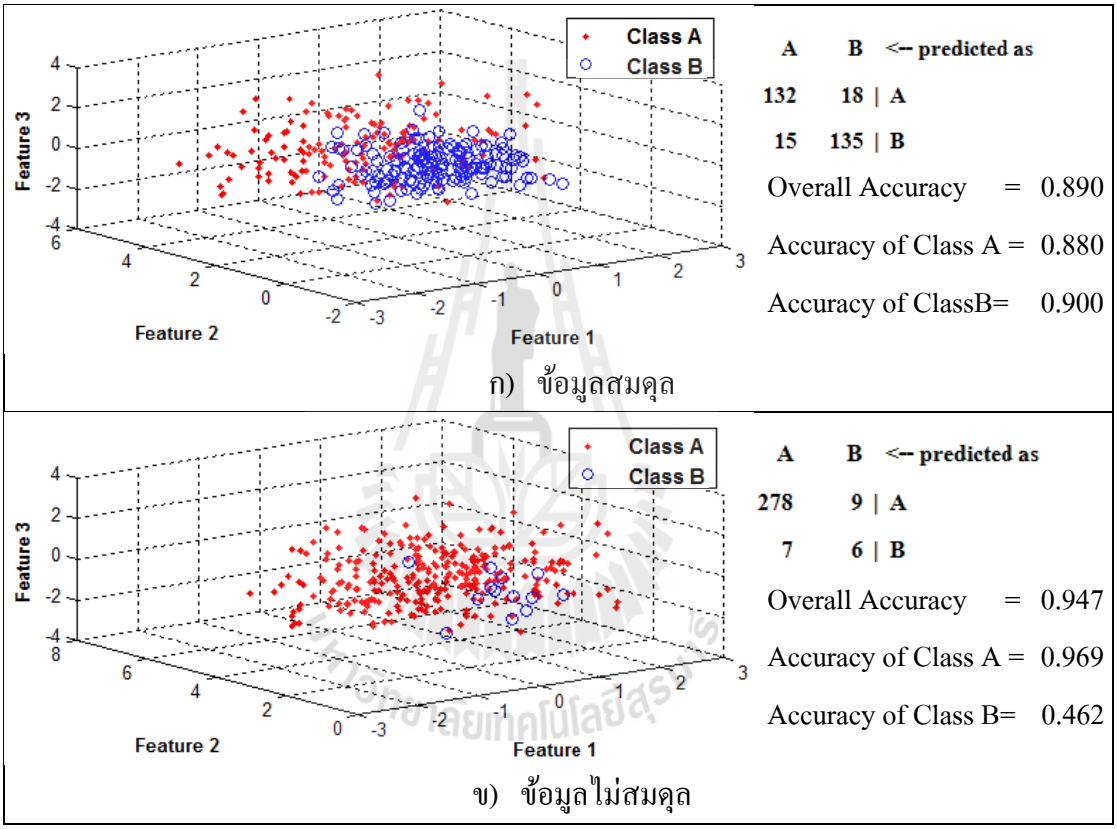
การทำเหมืองข้อมูล (Han and Kamber, 2006) คือกระบวนการขุดค้นความรู้จากข้อมูลที่มีขนาดใหญ่เพื่อหารูปแบบและความสัมพันธ์ที่ซ่อนอยู่ในข้อมูลเหล่านั้นด้วยวิธีการทางคณิตศาสตร์ สถิติ หรือคอมพิวเตอร์ การทำเหมืองข้อมูลนั้นมีหลายประเภทขึ้นอยู่กับวัตถุประสงค์ที่จะนำไปใช้งานเช่น การจำแนกประเภทข้อมูล (Data Classification) การหาความสัมพันธ์ของข้อมูล (Association Mining) และการจัดกลุ่มข้อมูล (Data Clustering) เป็นต้น ปัจจุบันการทำเหมืองข้อมูลได้รับความนิยมเกือบทุกวงการซึ่งจะเห็นได้จากการแก้ปัญหาทางด้านวิทยาศาสตร์และอุตสาหกรรมที่นำการจำแนกประเภทข้อมูลเข้ามาแก้ปัญหาด้วยการนำโมเดลหรือกฎที่ได้จากการจำแนกประเภทข้อมูล(Classification Rule) ไปทำการพยากรณ์ข้อมูลที่ไม่เคยเห็นหรือทำนายข้อมูลที่ยังไม่ทราบประเภท ซึ่งผลลัพธ์ที่ได้คือโมเดลที่มีค่าความถูกต้องสูง สำหรับเทคนิคที่นิยมนำมาใช้ในการจำแนกประเภทข้อมูลนั้นมีหลายเทคนิค เช่น การใช้โครงข่ายประสาทเทียม (Artificial Neural Network: ANN) ซึ่งเป็นเทคนิคที่มีพื้นฐานมาจากการจำลองการทำงานของมนุษย์ด้วยการทำให้คอมพิวเตอร์สามารถเรียนรู้ได้เหมือนกับที่มนุษย์เรียนรู้ การใช้เทคนิควิธีนาอีฟเบย์ (Naïve Bayes) ซึ่งเป็นการจำแนกประเภทข้อมูลโดยอาศัยค่าความน่าจะเป็นของข้อมูลเรียนรู้ หรือการใช้

ต้นไม้ตัดสินใจ(Decision Tree) ซึ่งเป็นเทคนิคการจำแนกประเภทข้อมูลให้อยู่ในลักษณะคล้ายต้นไม้จริงกลับหัวที่มีโหนดรากอยู่บนสุดและโหนดใบอยู่ล่างสุด โดยที่คุณลักษณะ (Attribute) ซึ่งเป็นตัวเลือกทดสอบอยู่ที่โหนดของต้นไม้ ค่าที่เป็นไปได้ของคุณลักษณะ (Attributes Value) จะอยู่ที่กิ่ง และคลาสที่กำหนดไว้จะอยู่ที่โหนดใบเป็นต้น ซึ่งเทคนิคการเรียนรู้เหล่านี้จะมีประสิทธิภาพและความแม่นยำของการจำแนกประเภทข้อมูลโดยรวม(Accuracy) ที่สูงเมื่อนำไปใช้ในการค้นหารูปแบบของข้อมูลที่มีลักษณะสมดุล (Balanced Data) ซึ่งมีจำนวนข้อมูลตัวอย่างของแต่ละคลาสเป้าหมายที่ใกล้เคียงกัน แต่เทคนิคเหล่านี้จะประสบกับปัญหาการเรียนรู้ก็ต่อเมื่อข้อมูลที่นำมาใช้ในการจำแนกประเภทนั้นมีลักษณะที่ไม่สมดุล(Imbalanced Data) (Chawla, 2005) ซึ่งปัญหานี้เป็นประเด็นปัญหาที่สำคัญและได้รับการกล่าวถึงในงานวิจัยของ Chawla et al. (2004)He and Garcia (2009) และ Sun et al. (2009)สำหรับข้อมูลไม่สมดุลนั้นสามารถพบเห็นได้ทั่วไป เช่น ข้อมูลการวินิจฉัยทางด้านการแพทย์ที่มีข้อมูลผู้ป่วยด้วยโรคร้ายแรงน้อยกว่าข้อมูลของผู้ที่มีสุขภาพดีเป็นจำนวนมาก ข้อมูลของบัตรเครดิตที่มีข้อมูลลูกค้าผิดปกติน้อยกว่าลูกค้าที่ปกติ ข้อมูลการตรวจจับผู้บุกรุกของเครือข่ายข้อมูลซึ่งมีข้อมูลของผู้บุกรุกในจำนวนที่น้อยมากเมื่อเทียบกับข้อมูลของผู้ที่ไม่บุกรุกเครือข่าย เป็นต้น ซึ่งการเกิดความไม่สมดุลของข้อมูลนั้นอาจจะมาจากหลายสาเหตุด้วยกัน เช่น ข้อมูลไม่สมดุลที่เกิดจากธรรมชาติของข้อมูลเองหรือข้อมูลไม่สมดุลอาจจะเกิดจากข้อจำกัดในการจัดเก็บ เช่น ค่าใช้จ่ายที่สูงมาก อันตรายที่เกิดจากการรวบรวมข้อมูล เป็นต้น

ข้อมูลที่มีลักษณะไม่สมดุลนั้นจะมีจำนวนข้อมูลตัวอย่างของแต่ละคลาสเป้าหมายที่แตกต่างกันมาก(Chawla et al., 2002) เช่น มีข้อมูลตัวอย่างจำนวน 10 ตัวอย่าง ข้อมูลนี้แบ่งออกเป็น 2 กลุ่ม คือ กลุ่ม A และ B โดยกลุ่ม A มีจำนวนข้อมูล 9 ตัวอย่าง และกลุ่ม B มี 1 ตัวอย่าง ซึ่งข้อมูลที่อยู่ในกลุ่มA ซึ่งเป็นกลุ่มที่มีจำนวนข้อมูลตัวอย่างมากจะถูกเรียกว่า คลาสส่วนมาก (Majority Class) และข้อมูลที่อยู่ในกลุ่มB ซึ่งเป็นกลุ่มที่มีจำนวนข้อมูลตัวอย่างน้อยจะถูกเรียกว่า คลาสส่วนน้อย (Minority Class) เมื่อนำข้อมูลชุดนี้ไปทำการจำแนกประเภทข้อมูลด้วยอัลกอริทึมที่เป็นมาตรฐานของการเรียนรู้ของเครื่องจักร จะส่งผลให้ขอบเขตของการตัดสินใจที่เป็นที่ยอมรับของอัลกอริทึมนั้นมีความเอนเอียง (Biased) ไปทางกลุ่มข้อมูลที่มีจำนวนข้อมูลตัวอย่างมาก และกลุ่มข้อมูลที่มีจำนวนข้อมูลตัวอย่างน้อยก็จะมีแนวโน้มสูงที่จะได้รับการจัดกลุ่มที่ผิดประเภท (Misclassified) โดยผลที่ได้จากการจำแนกประเภทข้อมูลด้วยอัลกอริทึมที่เป็นมาตรฐานของการเรียนรู้ของเครื่องจักรของข้อมูลชุดนี้ คือ ไม่สามารถจำแนกประเภทข้อมูลที่อยู่ในคลาสส่วนน้อยได้ แต่ในขณะเดียวกันจะสามารถจำแนกประเภทข้อมูลที่อยู่ในคลาสส่วนมากได้อย่างถูกต้องและมีประสิทธิภาพ

ตัวอย่างรูปที่ 1.1 เป็นการเปรียบเทียบให้เห็นว่า เมื่อมีการนำข้อมูลสมดุลและข้อมูลไม่สมดุลซึ่งมีลักษณะการกระจายตัวดังรูปที่ 1.1 ก) และรูปที่ 1.1 ข) ตามลำดับ มาทำการเรียนรู้แบบ

จำแนกประเภทข้อมูลด้วยอัลกอริทึมต้นไม้ตัดสินใจแล้วประสิทธิภาพของโมเดลที่ได้จากชุดข้อมูลทั้งสองจะมีประสิทธิภาพที่ดีเท่าเทียมกันหรือไม่ ซึ่งจำนวนข้อมูลตัวอย่างของทั้งสองชุดนี้จะมีจำนวนที่เท่ากัน คือ 300 ตัวอย่าง และมีคลาสเป้าหมาย 2 คลาสคือ คลาส A และคลาส Bข้อมูลสมดุลจะมีจำนวนข้อมูลทั้งสองคลาสเท่ากัน ในขณะที่ข้อมูลไม่สมดุลนั้นมีจำนวนข้อมูลที่อยู่ในคลาส Aซึ่งเป็นคลาสส่วนมาก 287ตัวอย่าง และข้อมูลที่เหลือ 13ตัวอย่างเป็นข้อมูลที่อยู่ในคลาส B ซึ่งเป็นคลาสส่วนน้อย



รูปที่ 1.1 แสดงการเปรียบเทียบการจำแนกประเภทข้อมูลด้วยอัลกอริทึมต้นไม้ตัดสินใจด้วยชุดข้อมูลที่สมดุลและไม่สมดุล

ผลที่ได้จากการเรียนรู้แบบจำแนกประเภทข้อมูลด้วยอัลกอริทึมต้นไม้ตัดสินใจพบว่าโมเดลที่ได้จากการเรียนรู้จะมีประสิทธิภาพที่แตกต่างกันซึ่งแสดงประสิทธิภาพในการจำแนกประเภทข้อมูลด้วยเมตริกซ์วัดประสิทธิภาพ (Confusion Matrix) แสดงโมเดลของข้อมูลสมดุลและข้อมูลไม่สมดุลดังรูปที่ 1.1 ก) และรูปที่ 1.1 ข) ตามลำดับ โดยที่ข้อมูลสมดุลนั้นความแม่นยำโดยรวมของการจำแนกประเภทจะอยู่ในระดับสูงคือ 0.890 และความแม่นยำของการจำแนกประเภทข้อมูลทั้งคลาสส่วนมากและคลาสส่วนน้อยนั้นจะอยู่ในระดับสูงด้วยกันทั้งคู่ (0.880 และ

0.900 ตามลำดับ) ในขณะที่ข้อมูลไม่สมดุลนั้นความแม่นยำโดยรวมของการจำแนกประเภทจะอยู่ในระดับสูงเช่นเดียวกับข้อมูลสมดุล คือ 0.947 แต่ความแม่นยำของการจำแนกประเภทของแต่ละคลาสนั้นจะแตกต่างกัน โดยการจำแนกประเภทข้อมูลของกลุ่มที่อยู่ในคลาสส่วนมากจะอยู่ในระดับสูง คือ 0.969 ในขณะที่ความแม่นยำในการจำแนกประเภทข้อมูลของกลุ่มที่อยู่ในคลาสส่วนน้อยจะอยู่ในระดับต่ำ คือ 0.462

จากประสิทธิภาพของโมเดลที่ได้นั้นแสดงให้เห็นว่า เมื่อนำข้อมูลที่มีลักษณะไม่สมดุลไปเรียนรู้ด้วยอัลกอริทึมมาตรฐานของการจำแนกประเภทข้อมูลแล้วจะส่งผลให้กลุ่มข้อมูลที่มีจำนวนมากมีค่าความถูกต้องและเกิดประสิทธิภาพสูงกว่ากลุ่มข้อมูลที่มีจำนวนน้อย ทำให้โมเดลที่ได้จากการจำแนกประเภทข้อมูลนั้นไม่เป็นกลาง

จากปัญหาการจำแนกประเภทข้อมูลไม่สมดุลที่กล่าวมาข้างต้นแล้วนั้นมีนักวิจัยจำนวนมากได้ให้ความสนใจและเสนอวิธีการต่าง ๆ เพื่อนำมาใช้ในการแก้ปัญหานี้โดยให้ความสำคัญกับข้อมูลที่อยู่ในคลาสส่วนน้อยให้มีการจำแนกประเภทข้อมูลถูกต้องแม่นยำมากขึ้นซึ่งงานวิจัยเหล่านั้นได้พยายามแก้ปัญหาทั้งในส่วนของขั้นตอนก่อนการประมวลผลที่มีการปรับจำนวนข้อมูลให้สมดุลด้วยการสุ่มข้อมูล (Resampling) ซึ่งมีทั้งการเพิ่มหรือลดจำนวนข้อมูลตัวอย่างหรือแบบผสมผสานทั้งสองวิธีเข้าด้วยกัน การแก้ปัญหาในขั้นตอนการประมวลผลโดยการปรับปรุงพารามิเตอร์ที่นำมาใช้สำหรับอัลกอริทึมทางด้านการจำแนกประเภทข้อมูล หรือแบบผสมผสานระหว่างการปรับให้ข้อมูลสมดุลและการปรับพารามิเตอร์ของอัลกอริทึมซึ่งจะเห็นได้จากงานวิจัยของBownandMues (2012)ที่ได้ลดจำนวนข้อมูลที่อยู่ในคลาสส่วนมากให้มีจำนวนที่ใกล้เคียงกับจำนวนข้อมูลที่อยู่ในคลาสส่วนน้อยด้วยเทคนิควิธีการสุ่มลด (Undersampling Method) โดยใช้ข้อมูลจริงของการประเมินเครดิต (Credit Scoring Data Sets) จำนวน 5 ชุดข้อมูลด้วยกันงานวิจัยของ Cateni et al. (2014) ที่เพิ่มจำนวนข้อมูลที่อยู่ในคลาสส่วนน้อยให้มีจำนวนที่ใกล้เคียงกับจำนวนข้อมูลที่อยู่ในคลาสส่วนมากด้วยเทคนิควิธีการสุ่มเกิน (Oversampling Method) และลดจำนวนข้อมูลที่อยู่ในคลาสส่วนมากให้มีจำนวนที่ใกล้เคียงกับจำนวนข้อมูลที่อยู่ในคลาสส่วนน้อยด้วยเทคนิควิธีการสุ่มลด(Undersampling Method) มาทำงานร่วมกันกับชุดข้อมูลจริงจากแหล่งข้อมูลมาตรฐาน Keel งานวิจัยของ Lopez et al. (2012) ที่ได้นำเสนอวิธีการเรียนรู้แบบมีค่าใช้จ่าย (Cost Sensitive Leaning) ซึ่งเป็นวิธีการชดเชยเมื่อมีการทำนายผิดกลุ่มร่วมกับการเพิ่มจำนวนข้อมูลตัวอย่างของคลาสส่วนน้อยด้วยเทคนิค SMOTE และ SMOTE + ENN และลดจำนวนข้อมูลตัวอย่างของคลาสส่วนมากด้วยวิธีการสุ่มลด เป็นต้น ซึ่งงานวิจัยเหล่านี้เมื่อได้ข้อมูลที่สมดุลแล้วก็จะนำข้อมูลเหล่านั้นไปทำการจำแนกประเภทข้อมูลด้วยอัลกอริทึมที่เหมาะสม เช่น ต้นไม้ตัดสินใจ ซัพพอร์ตเวกเตอร์แมชชีน นิวรอลเน็ตเวอร์กแบบไม่มีผู้สอน (Labeled Self-Organizing Map)และการจำแนกประเภทแบบเบย์ เป็นต้นซึ่งการทำงานของอัลกอริทึมเหล่านี้จะอยู่

ในลักษณะของโมเดลเดียว (Single Model) มีนักวิจัยจำนวนหนึ่งที่ให้ความเห็นว่าการจำแนกประเภทข้อมูลด้วยโมเดลเดียวนั้นถึงแม้ว่าจะมีประสิทธิภาพที่สูงแต่จะมีปัญหาในการกำหนดกลุ่มของข้อมูลที่ใช้ในการเรียนรู้ที่คงที่แน่นอนมากเกินไป ส่งผลให้เกิดความเอนเอียงได้ ดังนั้นจึงได้นำเสนอวิธีการเรียนรู้ร่วมกัน (Ensemble Method) เข้ามาช่วยในการตัดสินใจแทนการจำแนกประเภทข้อมูลด้วยโมเดลเดียว ซึ่งจะเห็นได้จากงานวิจัยของGalar et al. (2013) ที่ได้นำเทคนิควิธีการเรียนรู้ร่วมกัน (Ensemble Learning) ด้วยการนำเสนอเทคนิคEUSBoostซึ่งเป็นการนำเทคนิคการสุ่มลดมาทำงานร่วมกับเทคนิคบูสต์ติง (Boosting)โดยใช้อัลกอริทึม C4.5เป็นตัวจำแนกประเภทข้อมูลและงานวิจัยของWu et al. (2014) ที่นำเทคนิควิธีการเรียนรู้ร่วมกันด้วยอัลกอริทึมกลุ่มต้นไม้ตัดสินใจ (Random Forest) มาแก้ปัญหาการจำแนกข้อมูลข้อความที่ไม่สมดุลโดยการนำเสนออัลกอริทึมใหม่ที่ชื่อว่า FORESTEXTER เป็นต้น สำหรับกรณีของข้อมูลไม่สมดุลที่มีมิติสูง (High Dimensional) หรือมีจำนวนคุณลักษณะที่มากนั้นก่อนที่จะนำไปทำการจำแนกประเภทข้อมูลจะมีการทำในส่วนของการคัดเลือกคุณลักษณะ (Feature Selection) ซึ่งจะเห็นได้จากงานวิจัยของ Yin et al. (2013) ที่ได้นำเสนอเทคนิควิธีเกี่ยวกับการคัดเลือกคุณลักษณะร่วมกับการจำแนกประเภทข้อมูลด้วยอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน ต้นไม้ตัดสินใจ และการจำแนกประเภทแบบเบย์

จากที่กล่าวมาข้างต้นผู้วิจัยจึงเล็งเห็นความสำคัญที่จะแก้ปัญหาการจำแนกประเภทข้อมูลที่ไม่สมดุลให้มีความถูกต้องและมีประสิทธิภาพสูงขึ้น ซึ่งผู้วิจัยมีแนวคิดที่จะทำการปรับปรุงกระบวนการทำงานของการจำแนกประเภทข้อมูลไม่สมดุลจากชุดข้อมูลที่มีอัตราความไม่สมดุลของข้อมูลและการซ้อนทับของกลุ่มข้อมูลที่แตกต่างกัน โดยให้ความสำคัญกับกลุ่มข้อมูลจำนวนน้อยมากกว่ากลุ่มข้อมูลจำนวนมากแต่ในขณะเดียวกันประสิทธิภาพของการจำแนกประเภทข้อมูลของกลุ่มข้อมูลจำนวนมากก็จะยังคงอยู่ด้วยวิธีการเรียนรู้ร่วมกันแบบการใช้การตัดสินใจร่วมกันทั้งแบ็กกิง (Bagging)และบูสต์ติงชดเชยความผิดพลาดของการจำแนกประเภทข้อมูลผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่าย และทำการปรับค่าพารามิเตอร์ของตัวแบบการเรียนรู้ร่วมกันด้วยค่าที่ได้จากการสร้างตารางค่าใช้จ่าย (Cost Matrix)งานวิจัยนี้จะใช้อัลกอริทึมต้นไม้ตัดสินใจเป็นเครื่องมือในการจำแนกประเภทข้อมูลพร้อมทั้งหาจำนวนต้นไม้ตัดสินใจที่เหมาะสมด้วยการปรับลดจำนวนต้นไม้ตัดสินใจด้วยวิธีการสร้างมโนภาพหรือวิชวลไลเซชัน (Visualization)

1.2 วัตถุประสงค์ของการวิจัย

ในการดำเนินการวิจัยนี้มีวัตถุประสงค์ คือ

1) เพื่อศึกษาวิธีการต่าง ๆ ที่สามารถช่วยเพิ่มความถูกต้องในการค้นหาโมเดลการจำแนกประเภทข้อมูลไม่สมดุลโดยที่ข้อมูลอาจจะมีอัตราความไม่สมดุล (Imbalanced Ratio) ของกลุ่มข้อมูลและมีอัตราการซ้อนทับกันของกลุ่มข้อมูล (Overlapping Ratio)ที่แตกต่างกัน
2) เพื่อสร้างอัลกอริทึมที่สามารถช่วยในการค้นหาโมเดลจำแนกประเภทข้อมูลที่สามารถทำงานได้ดีกับข้อมูลที่มีอัตราความไม่สมดุลและอัตราการซ้อนทับที่แตกต่างกัน
3) เพื่อทดสอบประสิทธิภาพของโมเดลที่ได้จากอัลกอริทึมที่คิดค้นด้วยชุดข้อมูลที่ได้จากการสังเคราะห์จากโปรแกรมและข้อมูลจริงจากแหล่งข้อมูลมาตรฐาน

1.3 ขอบเขตของการวิจัย

ในการดำเนินงานได้กำหนดขอบเขตของการวิจัยไว้ดังนี้

1) ในการทำนายผิดพลาดนั้นจะชดเชยด้วยค่าใช้จ่ายจากตารางค่าใช้จ่ายซึ่งได้จากการการเรียนรู้แบบมีค่าใช้จ่าย
2) ข้อมูลที่ใช้ในการทดสอบเป็นข้อมูลชนิดตัวเลขที่ได้จากแหล่งข้อมูลมาตรฐานคือ KeelData Sets (http://www.keel.es) และจากการสังเคราะห์ด้วยโปรแกรมMATLAB R2013b
3) ข้อมูลจริงจากแหล่งข้อมูลมาตรฐาน คือKeel Data Sets จะต้องเป็นข้อมูลที่ไม่สมดุลและมีอัตราส่วนความไม่สมดุล (Imbalanced Ratio) ที่ระดับต่าง ๆ
4) มาตรวัดประสิทธิภาพของโมเดลประกอบด้วย Accuracy, Precision, Sensitivity, Specificity, F-measure, Geometric Mean (G-Mean), AUC: Area under the ROC Curve และ Total Misclassification Costs

1.4 ประโยชน์ที่ได้รับ

ผลสำเร็จของการวิจัยนี้ก่อให้เกิดประโยชน์ดังนี้

1) สามารถพัฒนาอัลกอริทึมที่ค้นพบโมเดลหรือรูปแบบของข้อมูลไม่สมดุลจากข้อมูลที่มีอัตราความไม่สมดุลของกลุ่มข้อมูลและอัตราการซ้อนทับกันของกลุ่มข้อมูลที่แตกต่างกัน
2) สามารถนำอัลกอริทึมที่พัฒนาไปเป็นแนวทางในการพัฒนาวิธีการค้นหารูปแบบของข้อมูลไม่สมดุลเฉพาะด้านได้ต่อไปในอนาคต

บทที่ 2
ปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

บทนี้จะกล่าวถึงปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้องกับงานวิจัยวิทยานิพนธ์นี้ โดยเนื้อหาจะประกอบด้วยข้อมูลไม่สมดุลวิธีการเรียนรู้ร่วมกัน เทคนิคการจำแนกประเภทข้อมูล การเรียนรู้ต้นไม้ตัดสินใจ การเรียนรู้แบบมีค่าใช้จ่ายการสุ่มเลือกตัวอย่างแบบชั้นภูมิการวัดระยะทางแบบยุคลิดเครื่องมือวัดประสิทธิภาพ (Performance Measurement Tools) และงานวิจัยที่เกี่ยวข้อง

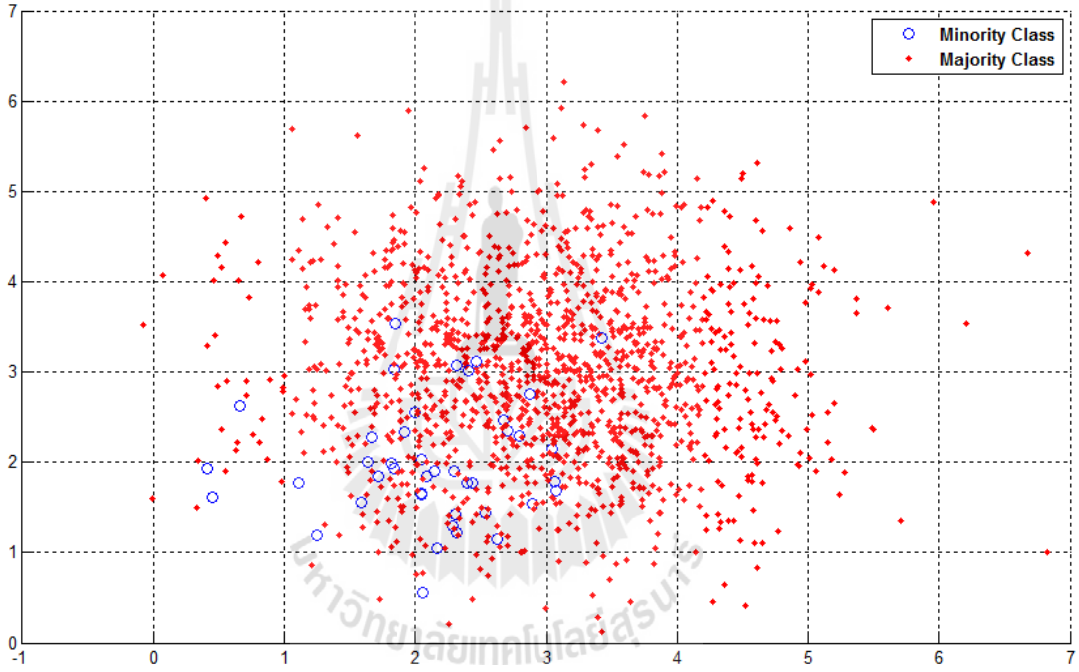
2.1 ข้อมูลไม่สมดุล

ลักษณะโดยทั่วไปของข้อมูลไม่สมดุล คือ ข้อมูลที่มีจำนวนข้อมูลของกลุ่มหนึ่งมากกว่าจำนวนข้อมูลของกลุ่มที่เหลือเป็นจำนวนมาก (Chawla et al., 2002;Chawla et al., 2004) ซึ่งข้อมูลไม่สมดุลนี้จะส่งผลกระทบต่อการจำแนกประเภทข้อมูลทำให้ไม่สามารถจำแนกประเภทข้อมูลของกลุ่มที่มีจำนวนข้อมูลน้อยได้ถูกต้องแม่นยำ ในขณะเดียวกันจะสามารถจำแนกประเภทข้อมูลของกลุ่มที่มีจำนวนมากได้อย่างแม่นยำ โดยทั่วไปข้อมูลกลุ่มที่มีจำนวนมากจะถูกเรียกว่า คลาสส่วนมาก(Majority Class หรือ Negative Class) และข้อมูลกลุ่มที่มีจำนวนน้อยจะถูกเรียกว่า คลาสส่วนน้อย(Minority Classหรือ Positive Class) (Farquad and Bose, 2012; Gao et al., 2011) ซึ่งข้อมูลที่อยู่ในคลาสส่วนน้อยจะเป็นข้อมูลที่งานวิจัยนี้ให้ความสำคัญมากกว่าข้อมูลที่อยู่ในคลาสส่วนมาก

ข้อมูลไม่สมดุลนั้นสามารถพบเห็นได้ทั่วไป ซึ่งสาเหตุของการเกิดความไม่สมดุลนั้นอาจจะมาจากหลายสาเหตุ เช่น ข้อมูลไม่สมดุลที่เกิดจากธรรมชาติของข้อมูลเองซึ่งสามารถพบเจอได้ในข้อมูลการวินิจฉัยทางด้านการแพทย์ที่มีข้อมูลผู้ป่วยด้วยโรคร้ายแรงน้อยกว่าข้อมูลของผู้ที่มีสุขภาพดีเป็นจำนวนมาก ข้อมูลของบัตรเครดิตที่มีข้อมูลลูกค้าปกติมากกว่าลูกค้าที่ผิดปกติ ข้อมูลการตรวจจับผู้บุกรุกของเครือข่ายข้อมูล หรือข้อมูลไม่สมดุลอาจจะเกิดจากข้อจำกัดในการจัดเก็บ เช่น ค่าใช้จ่ายที่สูงมาก อันตรายที่เกิดจากการรวบรวมข้อมูลเป็นต้น

ตัวอย่างข้อมูลไม่สมดุล เช่น มีข้อมูลจำนวน 1,500 ตัวอย่าง แบ่งออกเป็น 2 กลุ่ม กลุ่มที่หนึ่งมี 1,458 ตัวอย่าง กลุ่มที่สองมี 42 ตัวอย่าง ลักษณะการกระจายตัวของข้อมูลแสดงดังรูปที่ 2.1 โดยที่สัญลักษณ์วงกลมสีน้ำเงิน หรือ‘O’ แสดงข้อมูลคลาสส่วนน้อยและสัญลักษณ์บวกสีแดง หรือ ‘+’ แสดงข้อมูลคลาสส่วนมากเมื่อนำข้อมูลชุดนี้ไปเรียนรู้ด้วยโมเดลของการจำแนกประเภท

ข้อมูล ผลลัพธ์ที่ได้พบว่า ความถูกต้องของการจำแนกประเภทข้อมูลมีความเอนเอียง นั่นคือ สามารถจำแนกประเภทข้อมูลกลุ่มที่เป็นข้อมูลที่อยู่ในคลาสส่วนมากได้อย่างถูกต้องแม่นยำ ในขณะเดียวกันข้อมูลที่อยู่ในกลุ่มที่เป็นข้อมูลที่อยู่ในคลาสส่วนน้อยจะไม่สามารถจำแนกประเภทข้อมูลได้หรือจำแนกประเภทข้อมูลได้น้อย ทั้งนี้เนื่องจากในขั้นตอนการเรียนรู้ของโมเดลนั้นจะให้ความสำคัญกับข้อมูลที่อยู่ในคลาสส่วนมากเมื่อนำข้อมูลที่ไม่เคยผ่านขั้นตอนการเรียนรู้เข้าไปทดสอบ ความน่าจะเป็นของการจำแนกประเภทข้อมูลก็จะเกิดความเอนเอียงไปยังกลุ่มของคลาสส่วนมากส่งผลให้ข้อมูลกลุ่มที่เป็นข้อมูลที่อยู่ในคลาสส่วนน้อยเกิดการจำแนกประเภทผิดกลุ่ม



รูปที่ 2.1แสดงตัวอย่างการกระจายตัวของข้อมูลไม่สมดุล

2.1.1ลักษณะของปัญหาข้อมูลไม่สมดุล (Class Imbalanced Problems)

ปัญหาการจำแนกประเภทข้อมูลไม่สมดุลนั้นเป็นปัญหาที่ได้รับความสนใจจากนักวิจัยเป็นจำนวนมาก ปัญหานี้จะเกิดขึ้นเมื่อจำนวนข้อมูลตัวอย่างของกลุ่มหนึ่งมีมากกว่ากลุ่มที่เหลือเป็นจำนวนมาก ซึ่งอัลกอริทึมที่มีอยู่นั้นจะทำงานได้อย่างมีประสิทธิภาพก็ต่อเมื่อข้อมูลมีความสมดุล เมื่อไรก็ตามที่มีข้อมูลไม่สมดุลเกิดขึ้นการเรียนรู้ของอัลกอริทึมทั่วไปก็จะมีความเอนเอียงไปทางด้านข้อมูลคลาสส่วนมาก ทำให้เกิดการทำนายผิดพลาดในข้อมูลคลาสส่วนน้อย

ลักษณะของข้อมูลไม่สมดุลที่มีอิทธิพลต่อปัญหาการจำแนกประเภทข้อมูลนั้น (Phung et al., 2009; He and Ghodsi, 2010) สามารถแบ่งออกได้เป็น 3 กลุ่มด้วยกัน ดังรายละเอียดต่อไปนี้

2.1.1.1 ความไม่สมดุลของการกระจายของกลุ่มข้อมูลหรืออัตราความไม่สมดุล (Imbalance in Class Distribution or Imbalanced Ratio)

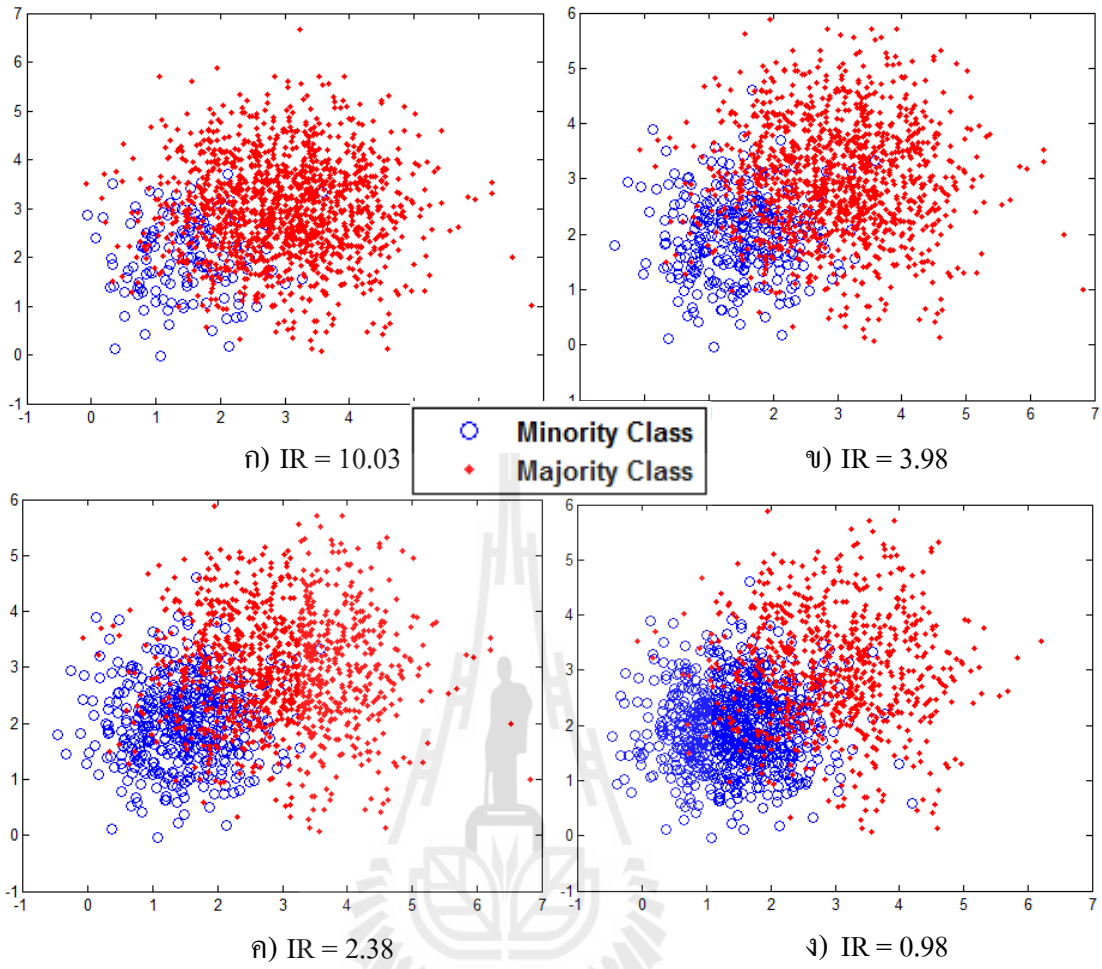
การจำแนกประเภทข้อมูลไม่สมดุลนั้น นักวิจัยได้แบ่งกลุ่มของข้อมูลออกเป็น 2 กลุ่ม คือ คลาสส่วนน้อยซึ่งเป็นกลุ่มที่มีจำนวนข้อมูลน้อย และอีกกลุ่มคือ คลาสส่วนมากซึ่งข้อมูลที่อยู่ในกลุ่มนี้จะมีจำนวนตัวอย่างที่มากกว่าข้อมูลที่อยู่ในคลาสส่วนน้อยเป็นจำนวนมากระดับของความไม่สมดุล (Imbalanced Degree) นั้นสามารถแสดงด้วยอัตราส่วนระหว่างจำนวนข้อมูลของคลาสส่วนมากและจำนวนข้อมูลของคลาสส่วนน้อย(Orriols-Puig et al., 2009; Villar et al., 2011)ดังสมการที่ 2.1

$$Imbalance\ Ratio\ (IR) = \frac{n_{majority}}{n_{minority}} \tag{2.1}$$

โดยที่$n_{majority}$คือจำนวนข้อมูลของคลาสส่วนมาก

$n_{minority}$คือจำนวนข้อมูลของคลาสส่วนน้อย

จากตัวอย่างข้อมูลสังเคราะห์ที่มีลักษณะการกระจายของข้อมูลแบบปกติซึ่งมีจำนวนข้อมูล1,500 ตัวอย่าง และมีคลาสเป้าหมายสองกลุ่มโดยมีอัตราความไม่สมดุลระหว่างข้อมูลของคลาสส่วนมากและข้อมูลของคลาสส่วนน้อยที่แตกต่างกัน ซึ่งสัญลักษณ์วงกลมสีน้ำเงินหรือ ‘O’ แสดงข้อมูลคลาสส่วนน้อย และสัญลักษณ์บวกสีแดง หรือ ‘+’ แสดงข้อมูลคลาสส่วนมาก แสดงดังรูปที่ 2.2



ก) IR = 10.03 ข) IR = 3.98

ค) IR = 2.38 ง) IR = 0.98

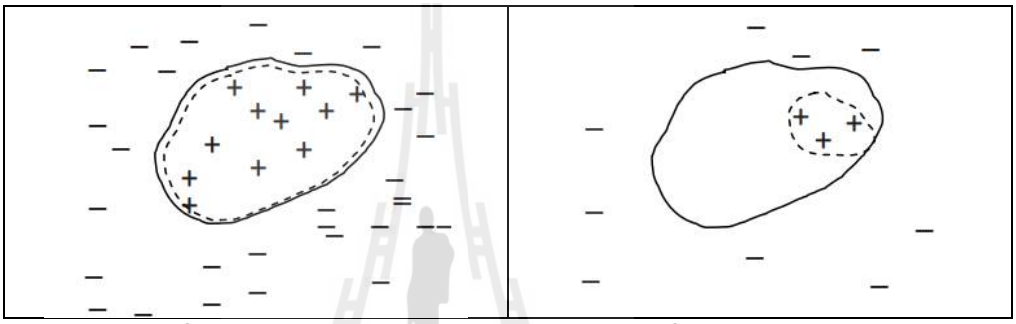
รูปที่ 2.2 แสดงตัวอย่างข้อมูลที่มีอัตราความไม่สมดุลที่แตกต่างกัน

จากรูปที่ 2.2 ซึ่งแสดงลักษณะของข้อมูลที่มีอัตราความไม่สมดุลที่แตกต่างกัน โดยที่ค่า IR จะแสดงถึงอัตราส่วนความไม่สมดุลของกลุ่มข้อมูล ในกรณีที่ค่า IR = 0.98จะหมายถึงข้อมูลนั้นมีลักษณะสมดุลดังรูปที่ 2.2 ง) และในกรณีที่ค่า IR มีค่าสูง จะหมายถึง ข้อมูลชุดนั้นจะมีอัตราความไม่สมดุลที่สูงดังรูปที่ 2.2 ก)

2.1.1.2 การขาดข้อมูล (Lack of Data)

ปัญหาหนึ่งของการเรียนรู้ข้อมูลไม่สมดุลคือ ปัญหาที่เกี่ยวข้องกับการขาดข้อมูลซึ่งจะเกิดขึ้นเมื่อข้อมูลตัวอย่างมีจำนวนน้อยมาก (Phunget al., 2009) ในการจำแนกประเภทข้อมูลนั้นขนาดของข้อมูลตัวอย่างเป็นปัจจัยหนึ่งที่สำคัญต่อการเรียนรู้เพื่อนำไปสู่การสร้างตัวจำแนกประเภทข้อมูลที่ดี การขาดข้อมูลตัวอย่างจะส่งผลกระทบต่อการค้นหารูปแบบของโมเดลของกลุ่มข้อมูลซึ่งทำให้การค้นหารูปแบบของโมเดลเป็นไปได้ยากจากรูปที่ 2.3 แสดงปัญหาที่จะ

สามารถเกิดขึ้นได้จากการขาดข้อมูล โดยรูปที่ 2.3 ก) เส้นประจะหมายถึงขอบเขตการตัดสินใจซึ่งจะได้มาเมื่อมีข้อมูลที่มากพอสำหรับการเรียนรู้ เช่นเดียวกันกับรูปที่ 2.3 ข) ที่แสดงผลลัพธ์ที่ได้จากการเรียนรู้ด้วยข้อมูลที่มีจำนวนน้อยเมื่อข้อมูลมีจำนวนมากพอจะทำให้ขอบเขตการตัดสินใจใกล้เคียงกับขอบเขตการตัดสินใจที่แท้จริงที่เป็นเส้นทึบ แต่ถ้าข้อมูลมีจำนวนน้อยมาก จะทำให้ขอบเขตการตัดสินใจห่างจากขอบเขตการตัดสินใจที่แท้จริงทำให้เกิดความผิดพลาดในการตัดสินใจซึ่งตรงกับความจริงที่ว่าเมื่อขนาดของข้อมูลที่ใช้สำหรับเรียนรู้มีขนาดที่มากพอจะทำให้ความผิดพลาดที่เกิดจากการเรียนรู้ของข้อมูลไม่สมดุลลดลง



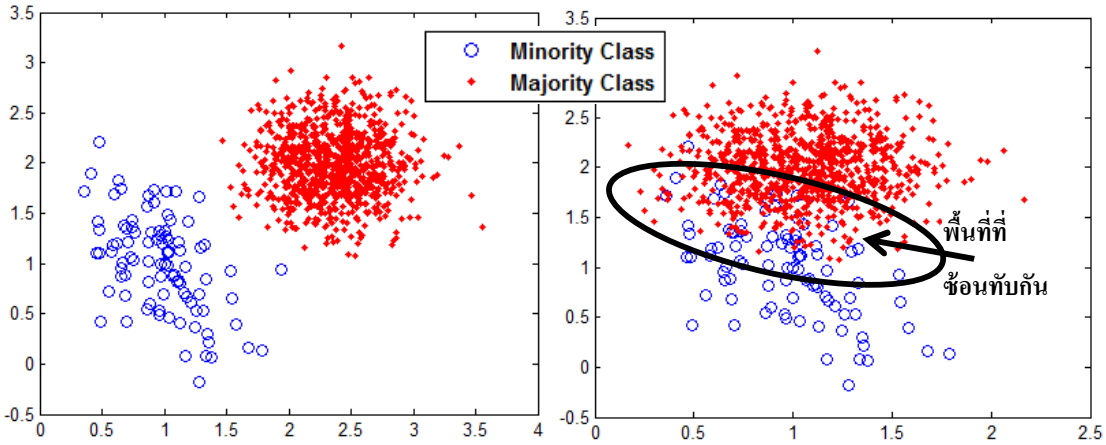
ก)ข้อมูลที่เพียงพอต่อการเรียนรู้ ข)ข้อมูลที่ไม่เพียงพอต่อการเรียนรู้

รูปที่ 2.3 แสดงผลกระทบของการขาดข้อมูลในปัญหาข้อมูลไม่สมดุล(Phunget al., 2009)

2.1.1.3 อัตราการซ้อนทับของกลุ่มข้อมูล (Overlapping Ratio between Classes)

การซ้อนทับกันของข้อมูล หมายถึง ตัวอย่างข้อมูลของทั้งสองคลาสมีการใช้พื้นที่ร่วมกันการซ้อนทับกันของข้อมูลนั้นเป็นปัญหาหนึ่งที่ยากต่อการจำแนกประเภทข้อมูล และเมื่อข้อมูลที่มีการซ้อนทับกันเกิดร่วมกับข้อมูลไม่สมดุลก็จะส่งผลให้การจำแนกประเภทมีความซับซ้อนมากยิ่งขึ้น (Xing et al., 2010)

ลักษณะของข้อมูลไม่สมดุลที่ไม่มีการซ้อนทับและซ้อนทับกันของข้อมูลแสดงดังรูปที่ 2.4ก) และ 2.4 ข) ตามลำดับ ซึ่งตัวอย่างข้อมูลของคลาสส่วนน้อยแสดงด้วยเครื่องหมายวงกลมสีน้ำเงิน ‘O’ และตัวอย่างคลาสส่วนมากแสดงด้วยเครื่องหมายบวกสีแดง ‘+’



ก) ข้อมูลไม่มีการซ้อนทับ ข) ข้อมูลมีการซ้อนทับ

รูปที่ 2.4แสดงตัวอย่างการซ้อนทับและไม่ซ้อนทับกันของข้อมูลไม่สมดุล

สำหรับเครื่องมือที่ใช้ในการวัดระดับการซ้อนทับของกลุ่มข้อมูลนั้นมีหลายตัววัด หนึ่งในเครื่องมือที่ได้รับความนิยม คือ TheFisher's Discriminant Ratio (F1) (Luengo et al., 2009) ซึ่งอัตราการซ้อนทับกันของกลุ่มข้อมูลจะพิจารณาจากค่าเฉลี่ย (μ) และค่าความแปรปรวน (σ^2) ของแต่ละกลุ่มข้อมูลดังนั้นค่า F1 ของแต่ละมิติแสดงดังสมการที่ 2.2

$$f_i = \frac{(\mu_{minority} - \mu_{majority})_i^2}{(\sigma^2_{minority} + \sigma^2_{majority})_i} \tag{2.2}$$

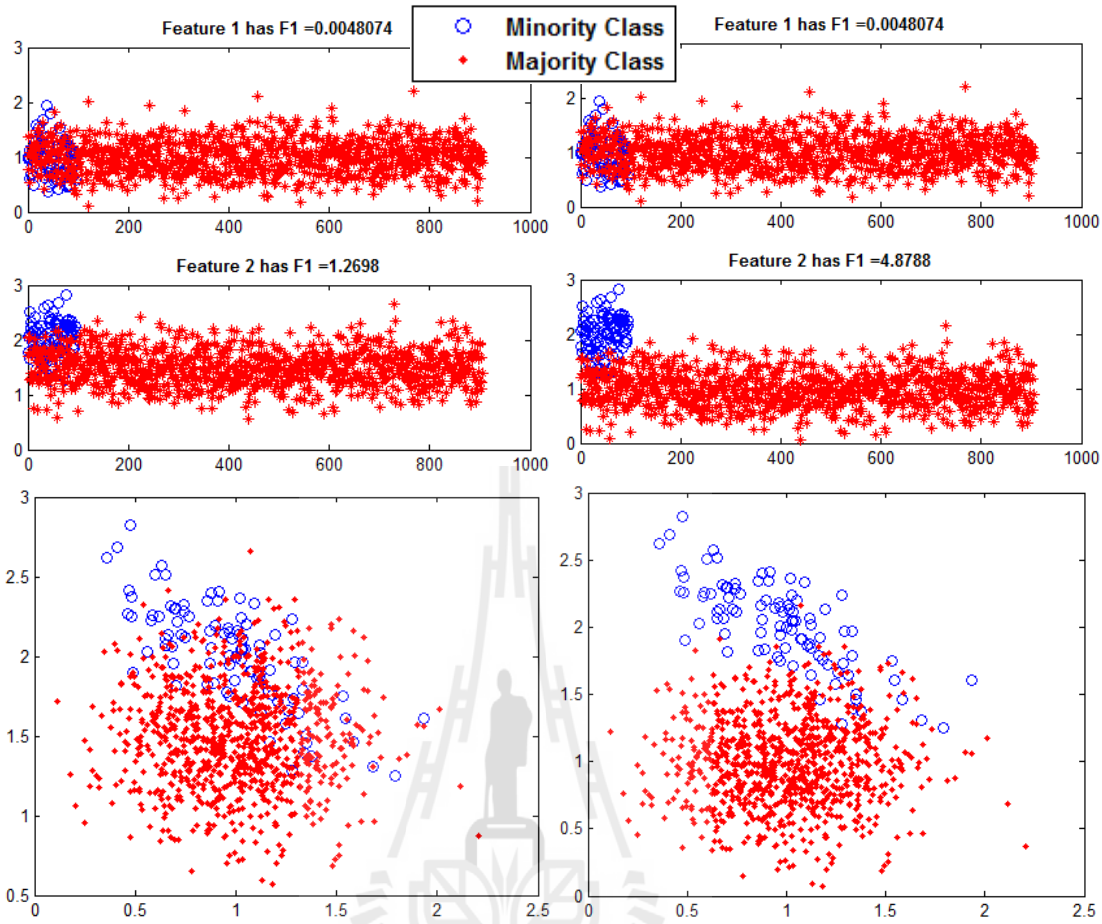
โดยที่

f_iคือFisher's Discriminant Ratio ของมิติที่ i

$\mu_{minority}, \mu_{majority}$คือ ค่าเฉลี่ยของข้อมูลของคลาสส่วนน้อย และค่าเฉลี่ยของข้อมูลของคลาสส่วนมากของมิติที่ iตามลำดับ

$\sigma^2_{minority}, \sigma^2_{majority}$ คือ ค่าความแปรปรวนของข้อมูลของคลาสส่วนน้อย และค่าความแปรปรวนของข้อมูลของคลาสส่วนมากของมิติที่ iตามลำดับ

ค่ามากที่สุดของ F1 เมื่อพิจารณาทุกมิติหรือMaximum Fisher's Discriminant Ratio (maxF) นั้นจะแสดงให้เห็นถึงอัตราการซ้อนทับของกลุ่มข้อมูล ดังนั้นถ้าmaxF มีค่าน้อยจะแสดงถึงอัตราการซ้อนทับของกลุ่มข้อมูลของชุดข้อมูลที่สูง และในทางกลับกันถ้าmaxFมีค่ามากจะแสดงให้เห็นถึงอัตราการซ้อนทับกันของกลุ่มข้อมูลของชุดข้อมูลที่ต่ำ



ก) ข้อมูลชุดที่ 1 มี maxF = 1.27 ข) ข้อมูลชุดที่ 2 มี maxF = 4.88

รูปที่ 2.5แสดงตัวอย่างข้อมูลไม่สมดุลที่มีอัตราการซ้อนทับกันที่แตกต่างกัน

จากรูปที่ 2.5 ซึ่งแสดงการซ้อนทับกันของกลุ่มข้อมูลไม่สมดุล เมื่อพิจารณาค่า maxF จะพบว่า รูป 2.5 ก) มีค่า maxF ที่ 1.27 ซึ่งต่ำกว่ารูปที่ 2.5 ข) ที่มีค่า maxF เป็น 4.88 ดังนั้นข้อมูลทั้งสองกลุ่มของรูป 2.5 ก) จะมีการใช้พื้นที่ร่วมกันมากกว่ารูปที่ 2.5 ข)

2.1.2วิธีการแก้ปัญหาข้อมูลไม่สมดุล

ปัญหาข้อมูลไม่สมดุลเป็นปัญหาที่นักวิจัยให้ความสนใจเป็นอย่างมาก ซึ่งนักวิจัยเหล่านั้นได้นำเสนอเทคนิควิธีการต่าง ๆ เพื่อนำมาใช้สำหรับแก้ปัญหานี้(López et al., 2012)วิธีการที่นำเสนอนั้นได้ทำการแบ่งออกเป็น 3ระดับ คือ การแก้ปัญหาข้อมูลไม่สมดุลที่ระดับข้อมูล (Data Level Solutions) การแก้ปัญหาข้อมูลไม่สมดุลที่ระดับขั้นตอนวิธีการ (Algorithmic LevelSolutions) และการแก้ปัญหาข้อมูลไม่สมดุลด้วยการเรียนรู้แบบมีค่าใช้จ่ายซึ่งทั้งสามระดับนี้มีเป้าหมาย

เดียวกัน คือ เพื่อเพิ่มประสิทธิภาพและความแม่นยำในการจำแนกประเภทข้อมูลของทั้งสองคลาส สำหรับงานวิจัยนี้จะให้ความสำคัญกับคลาสส่วนน้อยมากกว่าคลาสส่วนมาก โดยมีรายละเอียดดังนี้

2.1.2.1 การแก้ปัญหาระดับข้อมูล

เป็นการแก้ปัญหาในขั้นตอนก่อนการประมวลผล (Preprocessing Stage) ซึ่งจะเกี่ยวข้องกับข้อมูลโดยตรง โดยจะปรับข้อมูลที่ไม่สมดุลให้กลายเป็นข้อมูลสมดุลด้วยเทคนิคการสุ่มเลือกข้อมูล (DataSampling Technique)ซึ่งเทคนิคการสุ่มเลือกข้อมูลที่ได้รับความนิยมจะแบ่งออกเป็น 3 กลุ่ม คือ

1) วิธีสุ่มเกิน เป็นการเพิ่มจำนวนข้อมูลที่อยู่ในคลาสส่วนน้อยให้มีจำนวนใกล้เคียงหรือเท่ากับจำนวนข้อมูลที่อยู่ในคลาสส่วนมาก ซึ่งการเพิ่มข้อมูลนั้นจะเพิ่มโดยการสุ่มเลือกจากข้อมูลเดิม หรือสร้างขึ้นมาใหม่จากตัวอย่างที่มีอยู่ วิธีการสุ่มเกินที่ได้รับความนิยม เช่น Synthetic Minority Oversampling TEchnique(SMOTE)(Chawla et al., 2002)Borderline-SMOTE (Han et al., 2005)เป็นต้น
2) วิธีสุ่มลด เป็นการลดจำนวนข้อมูลที่อยู่ในคลาสส่วนมากให้มีจำนวนใกล้เคียงหรือเท่ากับจำนวนข้อมูลที่อยู่ในคลาสส่วนน้อยวิธีการสุ่มลดที่รู้จักกันดี เช่น Wilson’s edited nearest neighbor (ENN) (Wilson,1972) ซึ่งจะเอาตัวอย่างที่อยู่ในคลาสส่วนมากออกโดยจะเอาออก 2 ใน 3 ของเพื่อนบ้านที่ใกล้เคียง และ The One-Sided Selection (OSS) เป็นต้น
3) วิธีผสมผสาน (HybridMethods)เป็นวิธีการที่นำเทคนิควิธีสุ่มเกิน และวิธีสุ่มลดมาทำงานร่วมกัน เช่น การนำเทคนิค SMOTE มาใช้ร่วมกับเทคนิค ENN กลายเป็นเทคนิค SMOTE-ENN (Batista et al., 2004b) หรือ นำเทคนิค SMOTE มาใช้ร่วมกับเทคนิคTomekLinks กลายเป็น SMOTE+TomekLinks (Batista et al., 2004a)เป็นต้น

นักวิจัยได้นำเทคนิคการสุ่มเลือกข้อมูลมาใช้เพื่อปรับข้อมูลให้มีความสมดุล หลังจากนั้นจะนำข้อมูลที่สมดุลนั้นไปทำงานร่วมกับเทคนิควิธีอื่น ๆ ซึ่งจะเห็นได้จากงานวิจัยของ Qian et al. (2014) ที่ได้ทำการเพิ่มข้อมูลที่อยู่ในคลาสส่วนน้อยด้วยวิธีสุ่มเกิน และลดข้อมูลที่อยู่ในคลาสส่วนมากด้วยวิธีการสุ่มลด โดยที่อัตราการสุ่มเลือกจะกำหนดด้วยอัตราส่วนของจำนวนข้อมูลกลุ่มน้อยและจำนวนข้อมูลกลุ่มมาก หลังจากนั้นจะนำข้อมูลที่มีความสมดุลไปทำการการจำแนกประเภทข้อมูลด้วยวิธีการเรียนรู้ร่วมกันแบบแบ็กกิง และงานวิจัยของ Dubey et al. (2014) ซึ่งนำเทคนิคการสุ่มเลือกข้อมูลทั้งแบบสุ่มเกิน สุ่มลด และวิธีผสมผสานมาใช้งานร่วมกับเทคนิคการ

เรียนรู้ร่วมกันและการคัดเลือกคุณลักษณะ (Feature Selection) ด้วยชุดข้อมูลโรคอัลไซเมอร์ (Alzheimer's Disease) เป็นต้น

2.1.2.2 การแก้ปัญหาระดับขั้นตอนวิธีการ

เป็นการแก้ปัญหาโดยการปรับการเรียนรู้ของอัลกอริทึมมาตรฐานสำหรับการจำแนกประเภทข้อมูลที่มีอยู่เดิมให้สามารถเรียนรู้ข้อมูลไม่สมดุลโดยให้มีการเอนเอียงไปทางข้อมูลของคลาสกลุ่มน้อย

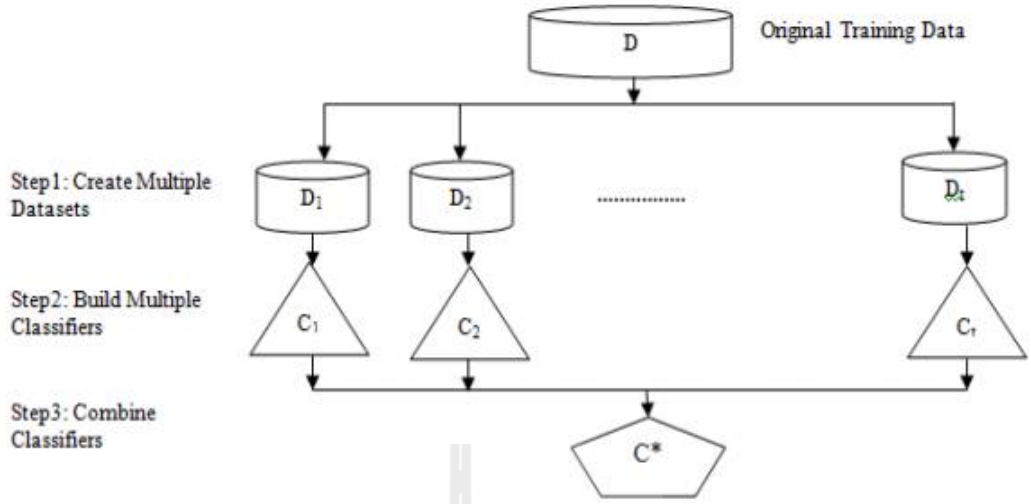
2.1.2.3 การแก้ปัญหาด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่าย

เป็นวิธีการแก้ปัญหาที่นำทั้งการแก้ปัญหาที่ระดับข้อมูล และระดับอัลกอริทึมมาทำงานร่วมกัน โดยที่ระดับข้อมูลจะทำการเพิ่มค่าใช้จ่าย (Cost) ที่พิเศษสำหรับกรณีที่มีการจำแนกประเภทผิดพลาด และที่ระดับอัลกอริทึมจะทำการปรับการเรียนรู้ของอัลกอริทึมมาตรฐานให้สอดคล้องกับการจำแนกประเภทข้อมูลผิดพลาด

2.2 วิธีการเรียนรู้ร่วมกัน

การจำแนกประเภทข้อมูลด้วยโมเดลเดียวนั้นถึงแม้จะให้ประสิทธิภาพในการจำแนกที่แม่นยำแต่ก็ประสบปัญหาในส่วนของการกำหนดกลุ่มของข้อมูลที่ใช้ในการเรียนรู้ และพารามิเตอร์ที่แน่นอนตายตัวมากเกินไปซึ่งส่งผลให้เกิดความเอนเอียงขึ้นได้ ดังนั้นวิธีการหนึ่งที่ได้รับความนิยมในการนำมาใช้เพื่อลดค่าความเอนเอียงที่เกิดขึ้น คือ การนำวิธีการเรียนรู้ร่วมกันเข้ามาช่วยในการตัดสินใจ ซึ่งประสิทธิภาพของวิธีการเรียนรู้ร่วมกันนี้จะขึ้นอยู่กับความหลากหลายและความแม่นยำของตัวแทนที่นำมาใช้ในการจำแนกประเภทข้อมูล

ในการเรียนรู้ของโมเดลที่นำวิธีการเรียนรู้ร่วมกันมาใช้ในการจำแนกประเภทข้อมูลนั้น (Polikar, 2006) จะมีการนำตัวจำแนกประเภทข้อมูลหลาย ๆ ตัว (Multiple Classifiers) เข้ามาเรียนรู้ด้วยชุดข้อมูลเริ่มต้น (Original Data Set) ผลที่ได้จากการทำนายจะนำมารวมกันเพื่อนำไปจำแนกประเภทข้อมูลที่ไม่เคยเห็นมาก่อน สำหรับงานวิจัยนี้จะใช้ตัวจำแนกประเภทชนิดเดียวกัน ลักษณะการทำงานพื้นฐานของวิธีการเรียนรู้ร่วมกันนั้นแสดงดังรูปที่ 2.6



รูปที่ 2.6แสดงลักษณะการทำงานพื้นฐานของวิธีการเรียนรู้ร่วมกัน

(Thalor andPatil, 2014)

วิธีการเรียนรู้ร่วมกันแบ่งออกเป็น 3ประเภท (Skurichina and Duin , 2002) ดังนี้

2.2.1 วิธีการบูสต์ติง

วิธีการบูสต์ติง(Freund andSchapire, 1996) เป็นการสร้างโมเดลจำแนกประเภทข้อมูลหลายโมเดล ซึ่งแต่ละโมเดลจะมีค่าถ่วงน้ำหนัก(Weight) เพิ่มเข้ามา โดยค่าถ่วงน้ำหนักนี้ได้มาจากความแม่นยำของการเรียนรู้บนชุดข้อมูล สำหรับคำตอบสุดท้ายของการทำงานด้วยวิธีการบูสต์ติงนั้น จะใช้วิธีการโหวตแบบถ่วงน้ำหนักแล้วกำหนดกลุ่มให้ข้อมูลใหม่ด้วยผลโหวตที่มากที่สุด(Majority Voting) ตัวอย่างอัลกอริทึมที่ได้รับความนิยม เช่น AdaBoost

วิธีการบูสต์ติงได้รับการปรับปรุงเพื่อนำมาใช้สำหรับแก้ไขปัญหากลุ่มข้อมูลมีจำนวนน้อย ตัวอย่างอัลกอริทึมของวิธีการบูสต์ติงที่นำมาใช้เพื่อแก้ไขปัญหาข้อมูลไม่สมดุล เช่น SMOTEBoost (Chawla et al., 2002) DataBoostp-IM และ Cost-Sensitive Boosting

ขั้นตอนการทำงานของวิธีการบูสต์ติงด้วยอัลกอริทึม AdaBoost มีดังต่อไปนี้

1. กำหนดน้ำหนักให้กับข้อมูลทุกตัวด้วยค่า $w_i = 1/N \ \ (i = 1..N)$ // *N*คือ จำนวนข้อมูล
2. สำหรับการเรียนรู้ รอบที่ m โดยที่ m เริ่มต้นที่ 1 ถึง M // M คือ จำนวนรอบทั้งหมด
   a. สุ่มเลือกข้อมูลแบบใส่คืน (Bootstrap with Replacement) สำหรับ S_m
   b. ทำการเรียนรู้ด้วยตัวจำแนกประเภทข้อมูล C_mกับชุดข้อมูล S_mด้วยน้ำหนักปัจจุบัน
   c. คำนวณค่าความผิดพลาดของน้ำหนัก $err_m = \frac{\sum w_i \text{ของข้อมูลที่ทำนายผิดคลาส} e_i}{\sum_i^N w_i}$
   d. คำนวณน้ำหนักของตัวจำแนกประเภทข้อมูล $\alpha_m = \frac{1}{2}\log\left(\frac{1-err_m}{err_m}\right)$
   e. สำหรับข้อมูลทุกตัวที่ทำนายถูกคลาส $e_i : w_i \leftarrow w_i e^{-\alpha_m}$
   f. สำหรับข้อมูลทุกตัวที่ทำนายผิดคลาส $e_i : w_i \leftarrow w_i e^{\alpha_m}$
   g. ทำการแปลงข้อมูล (Normalization)น้ำหนัก w_i โดยให้ผลรวมมีค่าเข้าใกล้ 1
3. สำหรับการทดสอบของแต่ละข้อมูล
   a. ทดสอบกับทุกตัวจำแนกข้อมูล C_m
   b. ทำนายคลาสเป้าหมายด้วยวิธีการโหวตแบบถ่วงน้ำหนักแล้วกำหนดกลุ่มให้ข้อมูลใหม่ด้วยผลโหวตที่มากที่สุด

2.2.2 วิธีการแบ็กกิง

วิธีการแบ็กกิง(Breiman, 1996) เป็นการสร้างโมเดลจำแนกประเภทหลายโมเดล และแต่ละโมเดลจะเรียนรู้ด้วยชุดข้อมูลที่แตกต่างกัน แต่จะใช้อัลกอริทึมเดียวกันในการสร้างโมเดล ซึ่งอัลกอริทึมแบ็กกิงจะช่วยปรับปรุงประสิทธิภาพของการจำแนกประเภทข้อมูลส่งผลให้ทำนายได้แม่นยำมากขึ้น เมื่อทำงานเสร็จสิ้นวิธีการแบ็กกิงจะนับผลโหวตซึ่งได้มาจากโมเดลทั้งหมด แล้วกำหนดกลุ่มให้ข้อมูลใหม่ด้วยผลโหวตที่มากที่สุดสำหรับวิธีการแบ็กกิงที่นำมาใช้ในการแก้ปัญหาข้อมูลนั้นจะเป็นการนำวิธีการนี้มาทำงานร่วมกับการสุ่มข้อมูลเช่น underBagging, overBagging และ SMOTEBagging เป็นต้น

ขั้นตอนการทำงานของวิธีการแบ็กกิงมีดังต่อไปนี้

1. สำหรับการเรียนรู้ รอบที่ m โดยที่ m เริ่มต้นที่ 1 ถึง M // M คือ จำนวนรอบทั้งหมด
    a. สุ่มเลือกข้อมูลแบบใส่คืน (Bootstrap with Replacement) สำหรับ S_m
    b. ทำการเรียนรู้ด้วยตัวจำแนกข้อมูล C_mกับชุดข้อมูล S_m
2. สำหรับการทดสอบของแต่ละข้อมูล
    a. ทดสอบกับทุกตัวจำแนกข้อมูล C_m
    b. ทำนายคลาสเป้าหมายด้วยผลโหวตที่มากที่สุด

2.2.3 วิธีการสุ่มเลือกสับสเปซ (Random Subspace Method)

วิธีการสุ่มเลือกสับสเปซ หรือเรียกอีกอย่างหนึ่งว่า Attribute Bagging (Bryll, 2003) เป็นตัวจำแนกประเภทข้อมูลที่มีการเรียนรู้ร่วมกัน โดยจะประกอบด้วยตัวจำแนกประเภทข้อมูลที่มากมายและการทำนายคลาสที่ได้จะขึ้นอยู่กับตัวจำแนกประเภทข้อมูลเหล่านี้ วิธีการสุ่มเลือกสับสเปซมีลักษณะทั่วไปเป็นอัลกอริทึมกลุ่มของต้นไม้ตัดสินใจ (Random Forest)ในขณะที่กลุ่มของต้นไม้ตัดสินใจจะประกอบด้วยอัลกอริทึมต้นไม้ตัดสินใจ วิธีการสุ่มเลือกสับสเปซจะถูกนำไปใช้ในการจำแนกประเภทข้อมูลเชิงเส้น ซัพพอร์ตเวกเตอร์แมชชีน การหาเพื่อนบ้านที่ใกล้เคียง (Nearest Neighbors) และการจำแนกประเภทข้อมูลชนิดอื่น ๆ ซึ่งเทคนิคนี้จะมีการนำไปประยุกต์ใช้กับการจำแนกประเภทข้อมูลที่มีคลาสเป้าหมายเพียงคลาสเดียว (One Class Classifiers) (Cheplyginaand Tax, 2011;Nanni, 2006)

ขั้นตอนการทำงานของวิธีการสุ่มเลือกสับสเปซ มีดังต่อไปนี้

1. สำหรับการเรียนรู้ รอบที่ m โดยที่ m เริ่มต้นที่ 1 ถึง M // M คือ จำนวนรอบทั้งหมด
    a. กำหนดให้ N เป็นจำนวนของข้อมูลฝึกสอน และ D เป็นจำนวนของคุณลักษณะ (Attributes) ของข้อมูลฝึกสอน
    b. กำหนดให้ L เป็นจำนวนของตัวจำแนกประเภทข้อมูล (Classifiers) ที่นำมาใช้สำหรับการเรียนรู้ร่วมกัน
    c. กำหนดให้ I เป็นตัวจำแนกประเภทข้อมูลแต่ละตัวที่มีข้อมูลเข้าเป็น d_I โดยที่ d_I < D และจำนวน d_I ของตัวจำแนกประเภทข้อมูลแต่ละตัวที่น้อยสุด คือ 1
    d. สร้างข้อมูลฝึกสอนสำหรับตัวจำแนกประเภทข้อมูลแต่ละตัวโดยการสุ่มเลือก

d_I จาก D แบบไม่ใส่คืนเมื่อสร้างเสร็จแล้วก็ทำการฝึกสอนตัวจำแนกประเภทข้อมูล

2. สำหรับการทดสอบของแต่ละข้อมูล
   a. ทดสอบกับทุกตัวจำแนกข้อมูล C_m
   b. ทำนายคลาสเป้าหมายด้วยผลโหวตที่มากที่สุดหรือจากความน่าจะเป็นที่เกิดขึ้นภายหลัง (Posterior Probabilities)

2.3 การจำแนกประเภทข้อมูล

การจำแนกประเภทข้อมูล(Han and Kamber, 2006) เป็นเทคนิคหนึ่งของงานทางด้านเหมืองข้อมูลซึ่งจะทำการจำแนกประเภทหรือคลาสจากข้อมูลขนาดใหญ่ที่มีหลายคลาส ในการจำแนกประเภทข้อมูลนั้นจะทำการจำแนกประเภทข้อมูลออกอย่างชัดเจนด้วยการสร้างโมเดลจำแนกประเภทข้อมูลให้อยู่ในกลุ่มที่กำหนดมา โดยจะทำการสร้างกฎขึ้นมาเพื่อช่วยในการตัดสินใจจากข้อมูลที่มีอยู่ เพื่อนำไปใช้ในการจำแนกประเภทหรือทำนายกลุ่มของข้อมูลใหม่ซึ่งเป็นข้อมูลที่ไม่เคยเรียนรู้หรือข้อมูลที่จะเกิดขึ้นในอนาคต (Unseen Data)

ตัวอย่างเช่น การดูคุณสมบัติของบุคคลที่จะก่อหนี้ดีหรือหนี้เสีย เพื่อนำมาใช้ในการพิจารณาอนุมัติสินเชื่อ ซึ่งแต่ละแถวของข้อมูล (Record) จะประกอบด้วยเงินเดือน อายุ ประวัติการชำระสินเชื่อและการชำระบัตรเครดิต(ชำระปกติ ผิดนัดชำระ) ของแต่ละบุคคลจากสถาบันการเงินแหล่งต่าง ๆ แล้วนำข้อมูลที่มีอยู่เหล่านั้นมาใช้วิเคราะห์คุณสมบัติว่าควรพิจารณาอนุมัติสินเชื่อให้กับลูกค้ารายใหม่หรือไม่

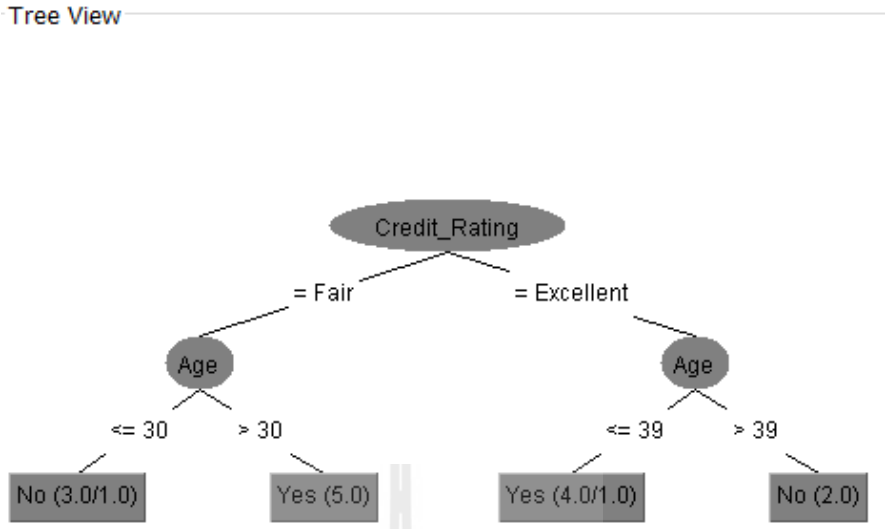
2.4 การเรียนรู้ต้นไม้ตัดสินใจ

การเรียนรู้ต้นไม้ตัดสินใจ (Quinlan, 1986) เป็นการเรียนรู้ที่มีการแทนความรู้ด้วยรูปแบบของต้นไม้ตัดสินใจซึ่งจะมีการนำไปใช้จำแนกประเภทของข้อมูลลักษณะโครงสร้างจะคล้ายต้นไม้กลับหัวที่มีรากอยู่ด้านบนและใบอยู่ด้านล่างสุด ภายในต้นไม้จะประกอบด้วยโหนด (Node) เส้นเชื่อมโยง และใบ โหนดที่อยู่บนสุดของต้นไม้จะเรียกว่า โหนดราก (Root Node) ซึ่งโหนดเหล่านั้นจะแสดงให้เห็นถึงการตัดสินใจบนข้อมูลด้วยคุณลักษณะต่าง ๆ เส้นเชื่อมโยงจะแสดงการเชื่อมโยงจากโหนดหนึ่งไปยังโหนดหนึ่ง หรือจากโหนดไปยังใบ โดยจำนวนเส้นเชื่อมโยงจะเท่ากับจำนวนค่าที่เป็นไปได้ทั้งหมดของคุณลักษณะที่เป็นโหนด ซึ่งค่าที่เป็นไปได้จะปรากฏบนเส้นเชื่อมโยงและใบแสดงถึงกลุ่มของข้อมูล (Class) หรือผลลัพธ์ที่ได้จากการทำนาย

ตัวอย่างข้อมูลสำหรับใช้ในการทำนายว่าลูกค้าที่มีคุณลักษณะเช่นไรที่มีความสนใจจะซื้อคอมพิวเตอร์ รายละเอียดข้อมูลดังตารางที่ 2.1 โดยมีคอลัมน์ Buys เป็นคุณลักษณะที่เป็นจุดมุ่งหมายในการทำการจำแนกประเภทข้อมูล และคอลัมน์ที่เหลือจะเป็นคอลัมน์ที่ใช้ประกอบในการค้นหารูปแบบของลูกค้าที่จะซื้อคอมพิวเตอร์ (Buys = “Yes”)นำข้อมูลเหล่านี้ไปหารูปแบบ (Pattern) ที่สามารถบอกลักษณะของลูกค้าที่สนใจจะซื้อคอมพิวเตอร์จากนั้นนำรูปแบบที่ได้ไปใช้กับลูกค้ารายอื่น ๆ เพื่อทำนายว่า ลูกค้ารายนั้นมีความสนใจจะซื้อคอมพิวเตอร์หรือไม่ การจำแนกประเภทข้อมูลด้วยอัลกอริทึมต้นไม้ตัดสินใจ รูปแบบที่ได้แสดงดังรูปที่ 2.7โดยโหนดที่แสดงด้วยรูปวงรีจะหมายถึงการทดสอบค่าที่ไปเป็นไปได้ของคุณลักษณะนั้น ๆ เส้นเชื่อมโยงซึ่งเป็นเส้นตรงจะบอกค่าของคุณลักษณะ และใบที่แทนด้วยรูปสี่เหลี่ยมจะแสดงการจำแนกของข้อมูล ซึ่งเป็นผลลัพธ์ว่าจะซื้อ (Yes) หรือไม่ซื้อคอมพิวเตอร์ (No)

ตารางที่ 2.1 แสดงตัวอย่างข้อมูลลูกค้าที่มีความสนใจจะซื้อคอมพิวเตอร์

Age	Income	Credit_Rating	Buys
30	High	Fair	No
30	High	Excellent	No
39	High	Fair	Yes
42	Medium	Fair	Yes
42	Low	Fair	Yes
42	Low	Excellent	No
39	Low	Excellent	Yes
30	Medium	Fair	No
30	Low	Fair	Yes
42	Medium	Fair	Yes
30	Medium	Excellent	Yes
39	Medium	Excellent	Yes
39	High	Fair	Yes
42	Medium	Excellent	No



รูปที่ 2.7แสดงตัวอย่างต้นไม้ตัดสินใจที่ใช้ตัดสินใจซื้อคอมพิวเตอร์

จากรูป 2.6สามารถแปลงให้อยู่ในรูปของกฎ IF_THEN rules ได้ดังนี้

IF (Credit_Rating = Fair) and (Age <= 30) THEN (Buys = No)

IF (Credit_Rating = Fair) and (Age > 30) THEN (Buys = Yes)

IF (Credit_Rating = Excellent) and (Age <= 39) THEN (Buys = Yes)

IF (Credit_Rating = Excellent) and (Age > 39) THEN (Buys = No)

2.5 การเรียนรู้แบบมีค่าใช้จ่าย

ในงานหลาย ๆ ด้าน เช่น การวินิจฉัยทางด้านการแพทย์ การตรวจสอบการทุจริต การป้องกันการบุกรุกของเครือข่าย หรือการจัดการความเสี่ยง งานเหล่านี้ล้วนแล้วแต่ให้ความสำคัญกับข้อมูลของเหตุการณ์ที่เกิดขึ้นเพียงน้อยครั้ง เมื่อพิจารณาจำนวนข้อมูลของงานเหล่านี้จะพบว่าจำนวนข้อมูลของเหตุการณ์ที่ให้ความสำคัญจะมีจำนวนข้อมูลที่น้อยมากเมื่อเทียบกับจำนวนข้อมูลของเหตุการณ์ที่ไม่ให้ความสำคัญ เมื่อมีการนำข้อมูลเหล่านี้ไปทำการจำแนกประเภทข้อมูล ผลลัพธ์ที่ได้จะส่งผลให้ข้อมูลของเหตุการณ์ที่ให้ความสำคัญเกิดการทำนายผิดพลาดในขณะที่ข้อมูลของเหตุการณ์ที่ไม่ให้ความสำคัญจะมีการจำแนกประเภทข้อมูลที่ถูกต้อง ทั้งนี้เนื่องจากอัลกอริทึมที่นำมาใช้ในการเรียนรู้โดยทั่วไปนั้นจะกำหนดค่าความผิดพลาดที่เกิดจากการทำนายผิดให้มีค่าเท่ากันและจะปฏิเสธความแตกต่างระหว่างประเภทของความผิดพลาด (Type of Error) ดังนั้นเพื่อแก้ปัญหาการจำแนกประเภทข้อมูลไม่สมดุลนี้จึงได้มีการนำเทคนิคการเรียนรู้แบบมีค่าใช้จ่ายเข้ามาใช้แทนการปรับความสมดุลของข้อมูลไม่สมดุลด้วยวิธีการสุ่มเลือกข้อมูลโดยจะทำการสร้างตาราง

ค่าใช้จ่าย (Cost Matrix) ที่มีค่าใช้จ่ายที่แตกต่างกัน ซึ่งค่าใช้จ่ายในที่นี้จะหมายถึง ค่าใช้จ่ายที่เกิดจากการจำแนกผิดกลุ่ม (Misclassification Cost) โดยจะนำค่าใช้จ่ายไปพิจารณาในขั้นตอนของการสร้างโมเดลและทำการสร้างตัวจำแนกข้อมูลโดยให้มีค่าใช้จ่ายที่ต่ำที่สุด

กำหนดให้ *C(i,j)*หมายถึงค่าใช้จ่ายของการประมาณค่าตัวอย่างจากคลาส *i*ไปยังคลาส *j* สำหรับปัญหาการจำแนกประเภทข้อมูล 2 กลุ่มแสดงดังตารางที่ 2.2โดยที่

*C(0,0)*และ *C(1,1)*จะหมายถึง ค่าใช้จ่ายที่เกิดจากการทำนายที่ถูกกลุ่มของคลาสส่วนมากและคลาสส่วนน้อยตามลำดับ จะกำหนดให้เป็น 0

*C(0,1)*หมายถึง ค่าใช้จ่ายที่เกิดจากการทำนายผิดคลาส คือ ข้อมูลเป็นคลาสส่วนมาก แต่ทำนายเป็นคลาสส่วนน้อย จะกำหนดให้เป็น 1

*C(1,0)*หมายถึง ค่าใช้จ่ายที่เกิดจากการทำนายผิดคลาส คือ ข้อมูลเป็นคลาสส่วนน้อย แต่ทำนายเป็นคลาสส่วนมากจะกำหนดให้เป็นค่าใด ๆ ที่มากกว่า *C(0,1)*หรือ $C_{minority} > C_{majority}$

ตารางที่ 2.2 แสดงตารางค่าใช้จ่ายสำหรับการจำแนกประเภทข้อมูล 2 กลุ่ม

	Predicted	
Actual	Majority Class	Minority Class
Majority Class	$C(0,0) = 0$	$C(0,1) = C_{majority} = 1$
Minority Class	$C(1,0) = C_{minority}$	$C(1,1) = 0$

2.6 การสุ่มเลือกตัวอย่างแบบชั้นภูมิ(Stratified Random Sampling)

การสุ่มตัวอย่างแบบชั้นภูมิเป็นเทคนิคหนึ่งของการสุ่มเลือกข้อมูลตัวอย่างที่ใช้ความน่าจะเป็น (Probability Sampling) โดยที่ข้อมูลประชากรทุกตัวมีโอกาสเท่าเทียมกันที่จะเป็นตัวแทนที่ดีของข้อมูลตัวอย่างที่จะนำไปใช้ในงานวิจัยสำหรับการสุ่มตัวอย่างแบบชั้นภูมินั้นจะทำการสุ่มเลือกข้อมูลตัวอย่างจากข้อมูลประชากรทั้งหมดซึ่งเป็นข้อมูลที่มีลักษณะความแตกต่างระหว่างหน่วยสุ่มที่ชัดเจนสามารถที่จะจำแนกออกเป็นชั้นภูมิ (Stratum) ได้ ซึ่งข้อมูลตัวอย่างที่ถูกจำแนกออกเป็นชั้นภูมินั้นจะมีลักษณะที่แตกต่างกันแต่ข้อมูลตัวอย่างที่อยู่ในชั้นภูมิเดียวกันจะมีลักษณะที่คล้ายกัน ในการสุ่มเลือกนั้นจะทำการสุ่มเลือกข้อมูลตัวอย่างจากแต่ละชั้นภูมิตามสัดส่วน (Proportional Allocation) ซึ่งชั้นภูมิใดมีจำนวนข้อมูลมากก็จะมีโอกาสได้รับการสุ่มเลือกมากข้อมูลตัวอย่างที่ได้จากการสุ่มเลือกแบบชั้นภูมินั้นจะเป็นข้อมูลตัวอย่างที่ครอบคลุมและครบถ้วน

2.7 การวัดระยะทางแบบยุคลิด (Euclidean Distance)

ระยะทางแบบยุคลิดเป็นเครื่องมือวัดระยะทางปกติระหว่างจุดสองpและ qในแนวเส้นตรง ถ้า $p = (p_1, p_2, \dots, p_n)$และ$q = (q_1, q_2, \dots, p_n)$ในระบบพิกัดคาร์ทีเซียนเป็นจุดสองจุดบนปริภูมิยุคลิดที่มีn มิติ ดังนั้นระยะทางระหว่างจุดสองจุด pและ qสามารถคำนวณได้ดังสมการ 2.3

$$d(p,q) = \sqrt{(p-q)(p-q)^T} \tag{2.3}$$

สำหรับชุดข้อมูลสังเคราะห์สองมิติที่มีค่าความแปรปรวน ($\sum$) และค่ากลาง (μ) ดังต่อไปนี้

$$\sum = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix} \qquad \mu = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

ข้อมูลที่ได้ คือ จุด A เป็น (1.16,0.89) จุด B เป็น (0.63, -0.29) และจุด C เป็น (0.08, 1.51)

เมื่อนำตัววัดระยะทางแบบยุคลิดมาทำการหาระยะทางระหว่างจุด A และ B $(d(A,B))$และหาระยะทางระหว่างจุด A และ C $(d(A,C))$ด้วยสมการ 2.3 ผลลัพธ์ที่ได้คือ

$$d(A,B) = 1.29 \text{ และ } d(A,C) = 1.25$$

2.8 เครื่องมือวัดประสิทธิภาพ (Performance Measurement Tool)

สำหรับเครื่องมือที่ใช้วัดประสิทธิภาพของวิธีการจำแนกข้อมูลนั้นมีหลากหลายวิธี ซึ่งแต่ละวิธีจะแสดงให้เห็นถึงประสิทธิภาพของโมเดลสำหรับการวัดค่าความถูกต้องในการจำแนกข้อมูลโดยรวมของทุกคลาสในโมเดลนั้นไม่สามารถนำมาใช้ในการหาประสิทธิภาพที่แท้จริงของโมเดลที่มีข้อมูลไม่สมดุลได้ ดังนั้นการเลือกเครื่องมือวัดประสิทธิภาพจึงเป็นปัจจัยที่สำคัญอย่างหนึ่งของการจำแนกข้อมูลไม่สมดุล

เมตริกซ์วัดประสิทธิภาพ (Confusion Matrix)แสดงผลสรุปการประเมินความสามารถในการจำแนกข้อมูลจากการทดสอบด้วยชุดทดสอบ โดยมีรูปแบบดังตารางที่2.3

ตารางที่ 2.3 แสดงเมตริกซ์วัดประสิทธิภาพสำหรับการจำแนกประเภทข้อมูล 2 กลุ่ม

	Positive Prediction	Negative Prediction
Actual Positive Class	True Positive (TP)	False Negative (FN)
Actual Negative Class	False Positive (FP)	True Negative (TN)

จากตารางที่ 2.3แถวของเมตริกซ์จะแสดงจำนวนของตัวอย่างจริงของแต่ละคลาส และคอลัมน์จะแสดงจำนวนที่ทำนายได้ของแต่ละคลาส โดยจะแบ่งออกเป็น 4 กรณี ดังนี้

ค่า TP หรือ True Positive คือ จำนวนข้อมูลที่อยู่ในคลาส Positive แล้วโมเดลทำนายได้ถูกต้องว่าเป็นคลาส Positive

ค่า FN หรือ False Negative คือ จำนวนข้อมูลที่อยู่ในคลาส Positive แล้วโมเดลทำนายผิดว่าเป็นคลาส Negative

ค่า FP หรือ False Positiveคือ จำนวนข้อมูลที่อยู่ในคลาส Negative แล้วโมเดลทำนายผิดว่าเป็นคลาส Positive

ค่า TNหรือ True Negativeคือ จำนวนข้อมูลที่อยู่ในคลาส Negativeแล้วโมเดลทำนายได้ถูกต้องว่าเป็นคลาส Negative

จากตารางที่ 2.3 สามารถแสดงตัวอย่างผลลัพธ์ที่ได้จากการจำแนกประเภทข้อมูลไม่สมดุล ซึ่งมี 2 กลุ่มโดยมีข้อมูลทั้งหมด 165 ตัวอย่าง เป็นข้อมูลกลุ่มส่วนน้อย 15 ตัวอย่าง และข้อมูลกลุ่มส่วนมาก 150 ตัวอย่าง รายละเอียดแสดงดังตารางที่ 2.4

ตารางที่ 2.4 แสดงตัวอย่างผลลัพธ์ที่ได้จากการจำแนกประเภทข้อมูลไม่สมดุล 2 กลุ่ม

	Positive Prediction	Negative Prediction
Actual Positive Class	13	2
Actual Negative Class	16	134

2.8.1Accuracy

เป็นการประเมินประสิทธิภาพการจำแนกประเภทข้อมูลโดยรวมทุกคลาสของโมเดล ดังสมการ 2.4

$$Accuracy = \frac{TP + TN}{(TP + FN + TN + FP)} \tag{2.4}$$

จากตารางที่ 2.4 สามารถคำนวณประสิทธิภาพการจำแนกประเภทข้อมูลโดยรวมทุกคลาสได้ดังนี้

$$Accuracy = \frac{13 + 134}{(13 + 2 + 134 + 16)} = 0.89$$

2.8.2 Precision

เป็นการวัดความแม่นยำของการทำนายข้อมูลที่อยู่ในคลาส Positive (Minority Class) โดยหาจากอัตราส่วนของการทำนายข้อมูลที่อยู่ในคลาส Positiveได้ถูกต้องเทียบกับจำนวนข้อมูลที่ทำนายว่าเป็นคลาส Positive ทั้งหมดดังสมการ 2.5

$$Precision = \frac{TP}{(TP + FP)} \tag{2.5}$$

จากตารางที่ 2.4 สามารถคำนวณความแม่นยำของการทำนายข้อมูลที่อยู่ในคลาส Positive ได้ดังนี้

$$Precision = \frac{13}{(13 + 16)} = 0.45$$

2.8.3 Sensitivity

ค่า Sensitive บางครั้งจะเรียกว่า True Positive rate (TPrate) หรือ Recall จะเป็นการวัดความสามารถในการค้นหาข้อมูลที่อยู่ในคลาส Positive โดยหาจากอัตราส่วนของการทำนายข้อมูลที่อยู่ในคลาส Positiveได้ถูกต้องเทียบกับข้อมูลจริงทั้งหมดของคลาส Positiveดังสมการ 2.6

$$Recall = Sensitivity = TPrate = \frac{TP}{(TP + FN)} \tag{2.6}$$

จากตารางที่ 2.4 สามารถคำนวณหาความสามารถของการค้นหาข้อมูลที่อยู่ในคลาส Positive ได้ดังนี้

$$Recall = Sensitivity = TPrate = \frac{13}{(13 + 2)} = 0.87$$

2.8.4 Specificity

ค่า Specificity บางครั้งจะเรียกว่า True Negative rate (TNrate) จะเป็นการวัดความแม่นยำของการทำนายข้อมูลที่อยู่ในคลาส Negative โดยหาจากอัตราส่วนของการทำนายข้อมูลที่อยู่ในคลาส Negative ได้ถูกต้องเทียบกับข้อมูลจริงทั้งหมดของคลาส Negative ดังสมการ 2.7

$$Specificity = TNrate = \frac{TN}{(TN + FP)} \tag{2.7}$$

จากตารางที่ 2.4 สามารถคำนวณหาความสามารถของการค้นหาข้อมูลที่อยู่ในคลาส Negative ได้ดังนี้

$$Specificity = TNrate = \frac{134}{(134 + 16)} = 0.89$$

2.8.5 F-measure

เป็นการวัดความแม่นยำโดยดูจากผลเฉลี่ยของ Precision และ Recall ดังสมการที่ 2.8

$$F - measure = \frac{(2 * Precision * Recall)}{(Precision + Recall)} \quad (2.8)$$

จากตารางที่ 2.4 สามารถคำนวณหาความแม่นยำโดยดูจากผลเฉลี่ยของ Precision และ Recall ได้ดังนี้

$$F - measure = \frac{(2 * 0.45 * 0.87)}{(0.45 + 0.87)} = 0.59$$

2.8.6 Geometric Mean (G-Mean)

G-Mean ถูกนำเสนอโดย Kubateta.l. (1988) เป็นการวัดความแม่นยำด้วยค่าเฉลี่ยเรขาคณิต ซึ่งจะทำการวัดความแม่นยำของแต่ละกลุ่มแยกจากกัน ตัววัดนี้นิยมนำมาใช้เมื่อประสิทธิภาพของแต่ละกลุ่มมีความสัมพันธ์กัน และมีแนวโน้มว่าจะมีค่าสูงขึ้นไปพร้อม ๆ กันซึ่ง G-Mean จะหลีกเลี่ยงปัญหา Over Fitting สำหรับกลุ่มข้อมูล Negative การคำนวณหา G-Mean ดังสมการที่ 2.9

$$G - Mean = \sqrt[n]{Sensitivity * Specificity} \quad (2.9)$$

โดยที่nคือ จำนวนกลุ่มข้อมูล

จากตารางที่ 2.4 สามารถคำนวณหา G-Mean ได้ดังนี้

$$G - Mean = \sqrt[2]{0.87 * 0.89} = 0.88$$

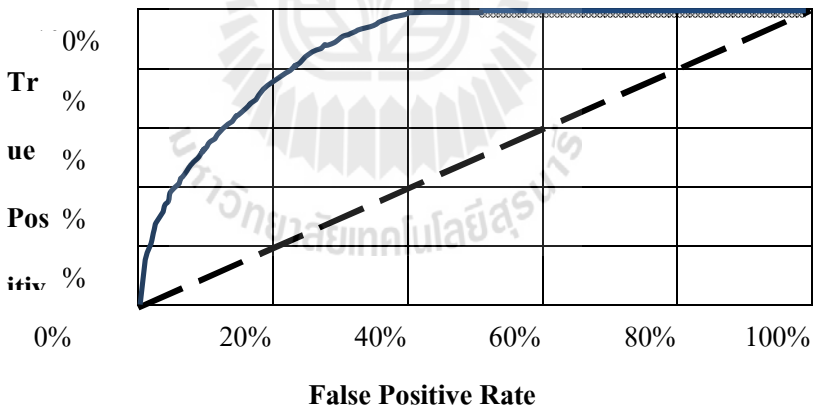
2.8.7AUC: Area under the ROC Curve

AUC คือ พื้นที่ใต้เส้นโค้ง ROC (Receiver Operating Characteristic curve) โดยที่เส้นโค้งROC จะเป็นเส้นกราฟที่พล็อตระหว่างค่า Sensitivityซึ่งเป็นค่าที่ทำนายได้ถูกต้องของการเกิดเหตุการณ์ที่สนใจซึ่งจะแทนด้วยแกน y และค่า 1- Specificity หรือค่าที่ทำนายผิดพลาดของการเกิดเหตุการณ์ที่สนใจซึ่งแทนด้วยแกน x ดังรูปที่ 2.8โดยที่ AUC จะแสดงให้เห็นถึงความสามารถในการจำแนกกลุ่มของเหตุการณ์ที่สนใจออกจากกลุ่มของเหตุการณ์ที่ไม่สนใจ ค่า AUC สามารถคำนวณหาได้ดังสมการที่ 2.10

$$AUC = \frac{1 + TPrate - FPrate}{2} \tag{2.10}$$

จากตารางที่ 2.4 สามารถคำนวณหา AUCได้ดังนี้

$$AUC = \frac{1 + 0.87 - 0.11}{2} = 0.88$$



รูปที่ 2.8แสดงพื้นที่ใต้เส้นโค้ง ROC

2.8.8 Total Misclassification Costs

เป็นตัววัดประสิทธิภาพของการจำแนกข้อมูลโดยการพิจารณาจากค่าความผิดพลาดของการจำแนกประเภทของข้อมูลที่อยู่ในคลาส Positive และ Negative ซึ่งสามารถหาค่าได้ดังสมการที่ 2.11

$$Total\ Misclassification\ Costs = FPrate + FNrate \quad (2.11)$$

จากตารางที่ 2.4 สามารถคำนวณหาTotal Misclassification Costs ได้ดังนี้

$$Total\ Misclassification\ Costs = 0.11 + 0.13 = 0.24$$

2.9 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับการค้นหาข้อมูลที่ไม่สมดุลนั้นมีนักวิจัยจำนวนมากนำเสนอเทคนิควิธีการต่าง ๆ เพื่อให้สามารถค้นหาข้อมูลที่ไม่สมดุลได้อย่างมีประสิทธิภาพ ซึ่งผู้วิจัยได้ศึกษางานวิจัยที่ได้รับการตีพิมพ์เหล่านั้นและสรุปได้ดังนี้

Krawczyk et al. (2014) ได้ทำการศึกษาเกี่ยวกับการจำแนกข้อมูลไม่สมดุลโดยใช้วิธีการเรียนรู้ร่วมกันด้วยอัลกอริทึมต้นไม้ตัดสินใจ ร่วมกับเทคนิควิธีการเรียนรู้แบบมีค่าใช้จ่าย ซึ่งในงานวิจัยนี้ได้นำเสนออัลกอริทึมสำหรับการปรับปรุง (Pruning) วิธีการเรียนรู้ร่วมกันด้วยอัลกอริทึมต้นไม้ตัดสินใจ และกำหนดพารามิเตอร์สำหรับตารางค่าใช้จ่ายของ$C_{minority}$ด้วยค่าที่ได้จากการวิเคราะห์พื้นที่ใต้กราฟROC และกำหนดค่าคงที่ $C_{majority}$เท่ากับหนึ่ง ทำการทดลองกับ 6 ชุดข้อมูลจาก Keel ที่มีอัตราส่วนความไม่สมดุลที่หลากหลาย และใช้มาตรวัด 2 ชนิด ได้แก่ Sensitivity และ Specificity โดยเปรียบเทียบอัลกอริทึมที่พัฒนากับอัลกอริทึม SingleCTTree, MCS, SMOTEBagging, SMOTEBoost, IIvotes และ EasyEnsemble ผลการทดสอบปรากฏว่า อัลกอริทึมที่นำเสนอให้ประสิทธิภาพที่ดีกว่าอัลกอริทึมอื่นสำหรับบางชุดข้อมูลเท่านั้น

Liao et al.(2014)ได้ทำการศึกษาเกี่ยวกับการใช้วิธีการเรียนรู้ร่วมกันสำหรับการจำแนกข้อมูลไม่สมดุลที่มีเพียงสองคลาสโดยการนำอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine :SVM) มาใช้ในขั้นตอนก่อนการประมวลผลเพื่อให้ได้ข้อมูลที่สมดุลหลังจากนั้นทำการคัดเลือกคุณลักษณะ (Feature Selection) และนำคุณลักษณะที่ได้ทำการคัดเลือกเข้าสู่กระบวนการการเรียนรู้ร่วมกันด้วยอัลกอริทึมโครงข่ายประสาทเทียมแบบแพร่ย้อนกลับ (Back-Propagation Neural Network : BPNN) และนำผลลัพธ์ที่ได้จากการการเรียนรู้ร่วมกันไปสร้างองค์ความรู้ใหม่ด้วยทฤษฎีของราฟเซ็ต (Rough Set Theory)งานวิจัยนี้ได้ทำการทดลองกับข้อมูลของบริษัทที่จดทะเบียนในตลาดหลักทรัพย์ตั้งแต่ปี 2005 ถึงปี 2011ซึ่งประกอบด้วยข้อมูลของ 63 บริษัทที่เกิดวิกฤตทางการเงิน และ 2680 บริษัทที่ไม่เกิดวิกฤตทางการเงิน และเพื่อความน่าเชื่อถือของโมเดลที่นำเสนอผู้วิจัยได้นำโมเดลไปทดสอบกับข้อมูลทางด้านการเงินอื่น ๆ ผลที่ได้จากการทดลองกับหลายสถานการณ์ปรากฏว่า โมเดลที่นำเสนอให้ประสิทธิภาพในการจำแนกข้อมูลและมีความแม่นยำมากกว่าวิธีการอื่น ๆ

López et al. (2012) ได้ทำการศึกษาเกี่ยวกับประสิทธิภาพของการจำแนกข้อมูลไม่สมดุลด้วยวิธีการที่ใช้ในระดับข้อมูล คือ เทคนิค SMOTE และ SMOTE+ENN และวิธีการที่ใช้ในระดับอัลกอริทึมโดยมุ่งเน้นไปที่เทคนิคการเรียนรู้แบบมีค่าใช้จ่าย พร้อมทั้งเปรียบเทียบเทคนิคเหล่านี้กับวิธีการผสมผสานของเทคนิคที่ใช้ในการแก้ปัญหาข้อมูลไม่สมดุลทั้งสองระดับงานวิจัยนี้ได้ทำการทดลองกับ66 ชุดข้อมูลจาก Keel ซึ่งข้อมูลมีลักษณะการซ้อนทับกันของกลุ่มข้อมูลและข้อมูลมีการกระจายของข้อมูลฝึกสอนและข้อมูลทดสอบที่แตกต่างกัน(Dataset Shift)สำหรับอัลกอริทึมที่ใช้ในงานวิจัยนี้ประกอบด้วย C4.5, SVM, FH-GBML (Fuzzy Hybrid Genetic Based Machine Learning rule Generation Algorithm) และ k-NN (K-Nearest Neighbor Algorithm)ทดสอบประสิทธิภาพของโมเดลด้วย 5 Fold Cross Validation และวัดประสิทธิภาพของการจำแนกข้อมูลด้วยการวิเคราะห์พื้นที่ใต้กราฟ ROC ผลการทดสอบพบว่า เทคนิคต่าง ๆ ที่ได้นำมาทดสอบกับข้อมูลไม่สมดุลนั้นได้ปรับปรุงประสิทธิภาพของการจำแนกข้อมูลได้ตามที่คาดหวังไว้

Brown andChristophe (2012) ได้นำเสนอเทคนิคการจำแนกกับข้อมูลที่ไม่สมดุลและเพิ่มจำนวนคลาสที่มีน้อยด้วยเทคนิคการลดแบบสุ่มกับ 5 ชุดข้อมูลสินเชื่อและวัดประสิทธิภาพด้วยพื้นที่ใต้กราฟโดยการหาค่าเฉลี่ยว่ามีคู่ใดบ้างที่มีแตกต่างกัน (Post Hoc Tests) ด้วยวิธี FriedmanTest และ NemenyiTest ผลที่ได้ปรากฏว่า กลุ่มของต้นไม้ตัดสินใจ (Random Forest) และ Gradient Boosting สามารถจำแนกข้อมูลสินเชื่อได้อย่างมีประสิทธิภาพ ในขณะเดียวกันถ้าข้อมูลไม่สมดุลมีขนาดของข้อมูลที่ใหญ่ขึ้นจะพบว่า อัลกอริทึมC4.5 QDA และ k-NN ไม่สามารถจำแนกข้อมูลได้อย่างมีประสิทธิภาพ

Cateni et al. (2014) ได้นำเสนอวิธีการสำหรับแก้ปัญข้อมูลที่ไม่สมดุลที่มีสองคลาสโดยการนำสองเทคนิควิธีของการสุ่มเลือกข้อมูล คือวิธีสุ่มเกินและวิธีสุ่มลดมาทำงานร่วมกัน เรียกวิธีนี้ว่า SUNDO โดยจะทำงานร่วมกับสี่โมเดลสำหรับการจำแนกข้อมูล คือ ซัพพอร์ตเวกเตอร์แมชชีน ต้นไม้ตัดสินใจ Self-Organizing Map (SOM) และ Bayesian Classifiers งานวิจัยนี้ได้ทำการทดลองกับสี่ชุดข้อมูลซึ่งประกอบด้วยชุดข้อมูลที่สังเคราะห์เอง ชุดข้อมูลมะเร็งเต้านมจาก UCI (Wisconsin) และสองชุดข้อมูลจากอุตสาหกรรมโลหะในการทดลองได้ทำการแบ่งชุดข้อมูลที่ไม่สมดุลออกเป็น 75% สำหรับชุดข้อมูลฝึกสอน และ 25% สำหรับชุดข้อมูลทดสอบ ผลที่ได้ปรากฏว่า วิธีที่นำเสนอสามารถทำการจำแนกข้อมูลที่ไม่สมดุลได้อย่างมีประสิทธิภาพ

Galar et al.(2013) ได้นำเสนอเทคนิควิธีการเรียนรู้ร่วมกันที่ชื่อว่า EUSBoostเพื่อนำมาใช้แก้ปัญข้อมูลที่ไม่สมดุล โดยใช้อัลกอริทึม C4.5 ร่วมกับเทคนิควิธีการสุ่มลดและวิธีการบูสต์ติงโดยได้ทำการทดสอบประสิทธิภาพกับข้อมูลจาก Keelผลที่ได้ปรากฏว่า เทคนิคดังกล่าวให้ประสิทธิภาพที่ดี

จากการศึกษางานวิจัยที่เกี่ยวข้องกับการข้อมูลไม่สมดุล พบว่า งานวิจัยส่วนใหญ่จะนำอัลกอริทึมที่มีอยู่แล้วมาทำการปรับปรุงเพื่อเพิ่มประสิทธิภาพในการจำแนกข้อมูลไม่สมดุล และทดสอบด้วยข้อมูลทั้งที่สังเคราะห์เองและจากแหล่งข้อมูลที่น่าเชื่อถือ สำหรับการวัดประสิทธิภาพในการจำแนกข้อมูลไม่สมดุลโดยส่วนใหญ่จะเน้นไปที่การวัดประสิทธิภาพโดยรวม เช่น ค่าความถูกต้อง ค่า F-measure ในงานวิจัยของวิทยานิพนธ์ฉบับนี้เป็นการพัฒนาวิธีการจำแนกข้อมูลไม่สมดุลด้วยวิธีการเรียนรู้ร่วมกันโดยใช้อัลกอริทึมที่หลากหลายร่วมกับการกำหนดค่าใช้จ่ายในกรณีที่มีการทำนายผิดพลาดซึ่งเป็นค่าใช้จ่ายจากวิธีการการเรียนรู้แบบมีค่าใช้จ่าย สำหรับการวัดประสิทธิภาพของการจำแนกข้อมูลจะเน้นไปที่มาตรวัดโดยคำนึงถึงข้อมูลที่อยู่ในคลาสส่วนน้อยเป็นหลัก โดยงานวิจัยนี้ได้ทำการเปรียบเทียบกับงานวิจัยอื่น ๆ ซึ่งสามารถสรุปได้ดังตารางที่ 2.5

ตารางที่ 2.5สรุปการเปรียบเทียบงานวิจัยที่เกี่ยวข้องกับเทคนิคการจำแนกสำหรับข้อมูลไม่สมดุล

	ก	ข	ค	ง	จ	ฉ	ช	ฌ	ญ*
อัลกอริทึมที่เกี่ยวข้อง									
Ensemble Learning	✓				✓	✓		✓	✓
Bagging		✓							✓
Boosting						✓			✓
Random Subspace Learning	✓								
Over-Sampling		✓	✓	✓	✓				
Under-Sampling		✓	✓	✓	✓				
Feature Selection					✓		✓	✓	
Cost-Sensitive Learning	✓			✓					✓
Decision Tree			✓	✓		✓			✓
Random Forest					✓			✓	
K-Nearest Neighbors		✓		✓					
Support Vector Machine			✓	✓	✓			✓	
Labeled Self-Organizing Map			✓						
Bayesian Classifiers			✓						
Genetic Algorithm	✓								
การลดอัตราการใช้พื้นที่ร่วมกัน									
Euclidean Distance									✓

ตารางที่ 2.5สรุปการเปรียบเทียบงานวิจัยที่เกี่ยวข้องกับเทคนิคการจำแนกสำหรับข้อมูลไม่สมดุล (ต่อ)

	ก	ข	ค	ง	จ	ฉ	ช	ฌ	ญ*
การหาจำนวนต้นไม้ตัดสินใจที่เหมาะสม									
Visualization									✓
มาตรวัดที่ใช้									
F-measure		✓					✓		✓
Sensitivity	✓		✓		✓				✓
Specificity	✓		✓		✓				✓
G-Mean		✓							✓
Total Misclassification Costs									✓
AUC-ROC	✓			✓	✓		✓	✓	✓
Kappa-AUC						✓			
Precision		✓	✓						✓
Accuracy			✓		✓				✓
IDX			✓						
วัตถุประสงค์ของการวิจัย									
เพื่อทดสอบประสิทธิภาพของโมเดล	✓	✓	✓	✓	✓	✓	✓	✓	✓
เพื่อทดสอบความถูกต้อง	✓	✓	✓	✓	✓	✓	✓	✓	✓
เพื่อเสนอแนวคิดใหม่	✓	✓	✓	✓	✓	✓	✓	✓	✓

ก หมายถึง งานวิจัยของ BartoszKrawczyk, Michal Wozniak, and Gerald Schaefer (2014)

ข หมายถึง งานวิจัยของ Yun Qian,Yanchun Liang, Mu Li, GuoxiangFeng, and Xiaohu Shi (2014)

ค หมายถึง งานวิจัยของ Silvia Cateni, ValentinaColla, and Marco Vannucc (2014)

ง หมายถึง งานวิจัยของ Victoria López, Alberto Fernández, Jose G. Moreno-Torres, and Francisco Herrera (2012)

จ หมายถึง งานวิจัยของ RashmiDubey, JiayuZhou,YalinWang, Paul M. Thompson, and Jieping Ye (2014)

ฉ หมายถึง งานวิจัยของ MikelGalar, Alberto Fernández, EdurneBarrenechea, and Francisco Herrera (2013)

ช หมายถึง งานวิจัยของ Liuzhi Yin, Yong Ge, Keli Xiao, Xuehua Wang, and XiaojunQuan (2013)

ฌหมายถึง งานวิจัยของQingyao Wu, Yunming Ye, Haijun Zhang, Michael K. Ng, and Shen-Shyang Ho(2014)

ญ*หมายถึง งานวิจัยเรื่อง การเรียนรู้ร่วมกันสำหรับปัญหาการจำแนกข้อมูลไม่สมดุล (งานวิจัยของวิทยานิพนธ์ฉบับนี้)

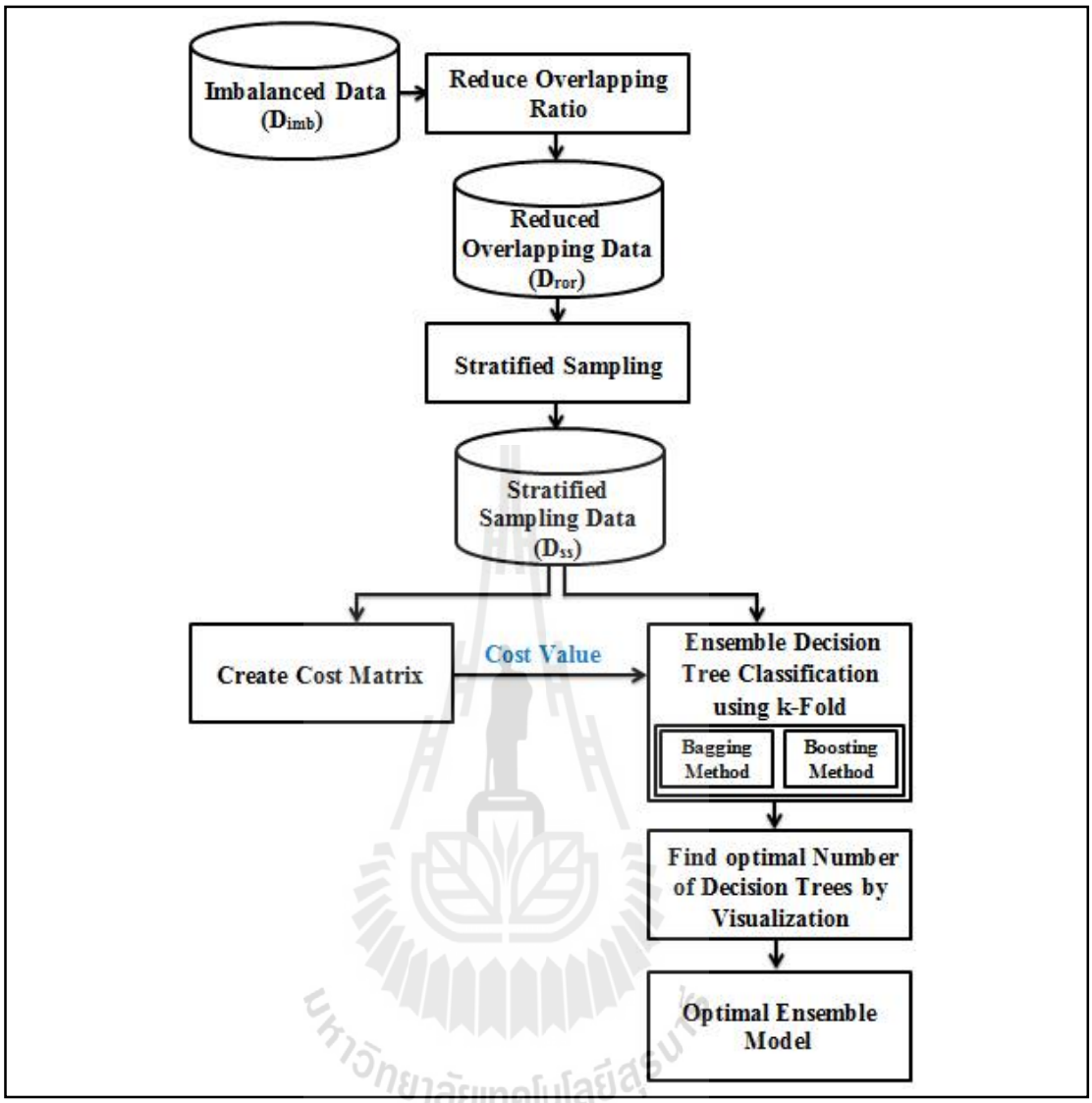
บทที่ 3
วิธีดำเนินการวิจัย

ในส่วนของบทที่ 3 นี้จะกล่าวถึงกรอบแนวคิดของการวิจัย และขั้นตอนการออกแบบอัลกอริทึมที่ใช้ในการค้นหาโมเดลจำแนกประเภทข้อมูลที่สามารถทำงานได้ดีกับข้อมูลที่มีอัตราความไม่สมดุลและอัตราการซ้อนทับที่แตกต่างกันด้วยวิธีการเรียนรู้ร่วมกันและชดเชยการทำนายผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่ายรายละเอียดมีดังต่อไปนี้

3.1 กรอบแนวคิดของการวิจัย

กรอบแนวคิดของการวิจัยวิทยานิพนธ์นี้คือ การปรับปรุงกระบวนการทำงานของการค้นหาโมเดลจำแนกประเภทข้อมูลที่สามารถทำงานได้ดีกับข้อมูลที่มีอัตราความไม่สมดุลและอัตราการซ้อนทับที่แตกต่างกันโดยจะนำวิธีการเรียนรู้ร่วมกันแบบการใช้การตัดสินใจร่วมกันทั้งแบ็กกิงและบูสต์ติงมาทำการสร้างโมเดล และชดเชยการจำแนกผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่ายซึ่งจะนำค่าที่ได้จากการสร้างตารางค่าใช้จ่ายมาใช้ในการปรับค่าพารามิเตอร์ของการเรียนรู้ร่วมกันและใช้โครงสร้างต้นไม้ตัดสินใจเป็นตัวจำแนกประเภทข้อมูลพร้อมทั้งหาจำนวนต้นไม้ตัดสินใจที่เหมาะสมด้วยวิธีการมโนภาพหรือวิชวลไลเซชัน(Visualization)

กรอบแนวคิดสำหรับการวิจัยที่ต้องการค้นหาโมเดลจำแนกประเภทข้อมูลที่สามารถทำงานได้ดีกับข้อมูลที่มีอัตราความไม่สมดุลและอัตราการซ้อนทับที่แตกต่างกันด้วยวิธีการเรียนรู้ร่วมกันนั้น แสดงดังรูปที่ 3.1



รูปที่ 3.1 แสดงกรอบแนวคิดของโมเดลการเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุล

จากรูปที่ 3.1 ซึ่งแสดงกรอบแนวคิดในการค้นหาโมเดลการเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลที่สามารถทำงานได้ดีกับข้อมูลที่มีอัตราความไม่สมดุลและอัตราการซ้อนทับที่แตกต่างกันและชดเชยการทำนายผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่ายนั้น มีรายละเอียดการทำงานดังต่อไปนี้

1) นำชุดข้อมูลไม่สมดุลซึ่งเก็บไว้ในแฟ้มข้อมูลD_{imb}มาทำการลดอัตราการซ้อนทับระหว่างกลุ่มของข้อมูล บันทึกผลที่ได้ไว้ในแฟ้มข้อมูลD_{ror}
2) สุ่มเลือกข้อมูล(Stratified Sampling) จากแฟ้มข้อมูลD_{ror}มาจำนวนหนึ่งบันทึกผลที่ได้ไว้ในแฟ้มข้อมูลD_{ss}

3) สร้างข้อมูลตารางค่าใช้จ่าย ซึ่งค่าใช้จ่ายสำหรับข้อมูลคลาสส่วนน้อยนั้นจะสร้างจากอัตราความไม่สมดุลของข้อมูลD_{ss}และค่าใช้จ่ายที่เหลือจะได้จากการกำหนดค่าคงที่ นำข้อมูลค่าใช้จ่ายที่ได้ไปใช้ในขั้นตอนที่ 4 ซึ่งเป็นการเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุล

4) นำข้อมูลที่อยู่ในแฟ้มข้อมูลD_{ss}มาสร้างโมเดลสำหรับเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุลด้วยโครงสร้างต้นไม้ตัดสินใจในการเรียนรู้ร่วมกันนั้นจะใช้การตัดสินใจร่วมกันทั้งแบบแบ็กกิงและบูสต์ติงพร้อมทั้งชดเชยการทำนายผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่ายซึ่งได้จากขั้นตอนที่ 3ในขั้นตอนนี้จะกำหนดค่าเริ่มต้นของจำนวนต้นไม้ตัดสินใจที่ 200 และทดสอบโมเดลด้วยการทดสอบไขว้ข้าม (k-fold Cross Validation) กำหนดจำนวนรอบของการทดสอบที่k = 5

5) ค้นหาจำนวนต้นไม้ตัดสินใจที่เหมาะสมสำหรับโมเดลการเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุลซึ่งขั้นตอนนี้ประกอบด้วย2 ขั้นตอนสำคัญ คือ

   5.1) ทำการปรับลดจำนวนต้นไม้ตัดสินใจเพื่อหาจำนวนต้นไม้ตัดสินใจที่เหมาะสมสำหรับโมเดลการเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุลด้วยวิธีการมโนภาพหรือวิชวลไลเซชัน ซึ่งจะแสดงรูปภาพเกี่ยวกับค่าความผิดพลาดที่เกิดจากการจำแนกประเภทผิดคลาสของข้อมูลทดสอบ(Test Classification Error) ณ ตำแหน่งของจำนวนต้นไม้ตัดสินใจที่ใช้ในการเรียนรู้ร่วมกัน (Number of Trees)ด้วยการเลือกจำนวนต้นไม้ตัดสินใจที่มีการจำแนกประเภทผิดคลาสน้อยที่สุดจำนวน 10 อันดับแรก(Top 10) เพื่อนำมาใช้ในการสร้างโมเดลการเรียนรู้ร่วมกัน

   5.2) ทดสอบประสิทธิภาพของโมเดลที่ได้ด้วยมาตรวัดที่ได้กล่าวไว้แล้วในหัวข้อ 2.8 ซึ่งประกอบด้วย Accuracy, Precision, Sensitivity, Specificity, F-measure, G-Mean, AUC และ Total Misclassification Costs

6) คัดเลือกโมเดลที่ได้จากขั้นตอนที่ 5 เพื่อให้ได้โมเดลที่มีประสิทธิภาพสูงที่สุด

3.2 การออกแบบอัลกอริทึม

จากกรอบแนวคิดในหัวข้อ 3.1 ผู้วิจัยได้นำมาพัฒนาอัลกอริทึมซึ่งในงานวิจัยวิทยานิพนธ์นี้เรียกว่า EnsDTV(Ensemble_Learning_with_DecisionTree_Visualization)อัลกอริทึมEnsDTV นี้จะสามารถช่วยในการค้นหาโมเดลจำแนกประเภทข้อมูลที่สามารถทำงานได้ดีกับข้อมูลที่มีอัตราความไม่สมดุลและอัตราการซ้อนทับที่แตกต่างกัน โมเดลที่ได้จากอัลกอริทึมนี้จะถูกนำไปทดสอบประสิทธิภาพด้วยชุดข้อมูลสังเคราะห์จากโปรแกรมและข้อมูลจากแหล่งข้อมูลมาตรฐานและวัด

ประสิทธิภาพของโมเดลที่ได้ด้วยมาตรวัดประสิทธิภาพต่าง ๆ ที่นำเสนอไปในหัวข้อ 2.8 ขั้นตอนทำงานโดยรวมของอัลกอริทึมEnsDTV แสดงดังรูปที่3.2

Algorithm EnsDTV

Input:

D_{imb} (imbalanced dataset)

U (set of parameter for create ensemble model)//พารามิเตอร์เริ่มต้นสำหรับสร้างโมเดล

N(จำนวนรอบการทำงานของโปรแกรมขึ้นอยู่กับจำนวนครั้งในการเลือกอัตราไม่สมดุล)

M(จำนวนรอบของการสร้างโมเดลขึ้นอยู่กับจำนวนของอัลกอริทึมที่ใช้)

Output:

optEnsM (optimal ensemble model)

1. D_{ror}= Reducing_Overlapping_Ratio(D_{imb})
2. Split data of D_{ror} into $D_{minority}$ and $D_{majority}$
3. Samplingdata of $D_{minority}$ then stored at D_{ssMin}
4. Repeat for N iterations
5. Samplingdata of $D_{majority}$ then stored at D_{ssMaj}
6. Combine data of $D_{ssMin}$and D_{ssMaj}then stored at D_{ss}
7. Count the members of each group of D_{ss}then stored at $N_{minority}$ and $N_{majority}$
8. costVal= Create_Cost_Matrix ($N_{minority}$, $N_{majority}$)
9. Repeat for M iterations
10. [optDT,errV]= Create_Classification_Ensemble_Model(D_{ss}, costVal,U)
11. ListEnsModel= Choose_Optimal_DecisionTree(D_{ss}, costVal,U,[optDT,errV])
12. Choose the optimal ensemble model from ListEnsModel, stored in optEnsM
13. Show optimal ensemble modelof imbalance ratio
14. Return optEnsM
15. End ofEnsDTV

รูปที่ 3.2 แสดงขั้นตอนการทำงานโดยรวมของอัลกอริทึมEnsDTV

จากรูปที่ 3.2 ซึ่งเป็นขั้นตอนการทำงานโดยรวมของอัลกอริทึมEnsDTVซึ่งขั้นตอนเริ่มต้นก่อนที่จะนำอัลกอริทึมนี้ไปใช้งานนั้นจะมีการกำหนดค่าเริ่มต้นต่าง ๆ เก็บไว้ที่ตัวแปร รายละเอียดมีดังต่อไปนี้

1. ตัวแปร U เป็นเซ็ตของตัวแปรสำหรับเก็บค่าเริ่มต้นของข้อมูลต่าง ๆ เพื่อนำไปใช้ในขั้นตอนการสร้างโมเดลซึ่งประกอบด้วยตัวแปรต่าง ๆ ดังนี้
   - nAlgo ใช้สำหรับเก็บข้อมูลรายละเอียดของอัลกอริทึมที่นำมาใช้สำหรับเรียนรู้ร่วมกัน สำหรับวิทยานิพนธ์นี้จะใช้ทั้งหมด 5 อัลกอริทึมด้วยกัน ประกอบด้วย {‘AdaboostM1’, ‘Bag’, ‘TotalBoost’, ‘LogitBoost’, ‘RUSBoost’}
   - TopK เป็นตัวแปรสำหรับจำนวนครั้งของการหาจำนวนต้นไม้ตัดสินใจที่เหมาะสมสำหรับสร้างโมเดลการเรียนรู้ร่วมกันซึ่งจะเป็นต้นไม้ตัดสินใจที่มีการจำแนกประเภทผิดคลาสน้อยที่สุด ในวิทยานิพนธ์นี้จะกำหนดที่ 10 อันดับแรก
   - numDTเป็นตัวแปรสำหรับกำหนดค่าเริ่มต้นของจำนวนต้นไม้ตัดสินใจที่เหมาะสมสำหรับสร้างโมเดลการเรียนรู้ร่วมกัน โดยกำหนดที่ 200 ต้นไม้ตัดสินใจ
   - nClassify เป็นตัวแปรสำหรับกำหนดชนิดของการจำแนกประเภทข้อมูล ซึ่งในงานวิจัยวิทยานิพนธ์นี้จะกำหนดชนิดของการจำแนกประเภทข้อมูลเป็นต้นไม้ตัดสินใจ โดยจะกำหนดค่าnClassifyเป็น ‘Decision Tree’
   - kFold เป็นตัวแปรสำหรับกำหนดจำนวน Fold ของการทดสอบโมเดลแบบไขว้ข้าม โดยจะกำหนดจำนวน Fold เท่ากับ 5
2. ตัวแปร N เป็นตัวแปรที่ใช้สำหรับควบคุมการทำงานวนรอบของอัลกอริทึม EnsDTV ซึ่งจำนวนรอบนั้นจะขึ้นอยู่กับจำนวนครั้งของการเลือกอัตราไม่สมดุลในงานวิทยานิพนธ์นี้ได้ทำการทดลองเพื่อทดสอบประสิทธิภาพของโมเดลด้วยอัตราความไม่สมดุลที่แตกต่างกัน คือ 1:5 1:10 1:15 1:20 1:25 1:30 1:35 1:40 1:45 และ 1:50 ดังนั้น N จะมีค่าเท่ากับ 10
3. ตัวแปร Mเป็นตัวแปรที่ใช้สำหรับควบคุมการทำงานวนรอบของการสร้างโมเดล ซึ่งจำนวนรอบนั้นจะขึ้นอยู่กับจำนวนของอัลกอริทึมที่ใช้ซึ่งเก็บไว้ที่ตัวแปร nAlgo ซึ่งในที่นี้จะมีทั้งหมด 5 อัลกอริทึม ดังนั้น M จะมีค่าเท่ากับ 5

การทำงานของอัลกอริทึม EnsDTVแต่ละขั้นตอน มีรายละเอียดดังต่อไปนี้

3.2.1 การลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล

การทำงานของอัลกอริทึม EnsDTV นั้นจะเริ่มต้นจากการนำชุดข้อมูลไม่สมดุล (D_{imb})ซึ่งเป็นข้อมูลชนิดตัวเลข (Numerical Data) และมีหนึ่งคุณลักษณะที่เป็นคลาสลาเบล

ข้อมูลไม่สมดุลจะมีรูปแบบข้อมูลดังต่อไปนี้

$$D_{imb} = \{x_1, x_2, x_3, \dots, x_n, y\}$$

โดยที่ $x_1, x_2, x_3, \dots, x_n$แสดงถึงลักษณะของข้อมูลตัวอย่างตั้งแต่มิติที่ 1 ถึง มิติที่ n

yแสดงถึงลักษณะของคลาสลาเบล

ตัวอย่างของข้อมูลไม่สมดุลซึ่งเป็นข้อมูลสังเคราะห์ที่มี 3มิติ2 คลาส จำนวน 1,000 ตัวอย่าง โดยมีข้อมูล44 ตัวอย่างเป็นข้อมูลคลาสส่วนน้อยหรือคลาส Positive และ 956 ตัวอย่างเป็นข้อมูลคลาสส่วนมากหรือคลาส Negative ข้อมูลบางส่วนของ D_{imb}แสดงดังตารางที่ 3.1

ตารางที่ 3.1 แสดงข้อมูลไม่สมดุลบางส่วนจากแฟ้มข้อมูลD_{imb}

TransactionID	x_1	x_2	x_3	y
1	4.1650	0.6268	0.0751	negative
2	3.0591	1.7971	0.2641	negative
3	3.8717	-1.4462	-0.7012	negative
4	4.2460	-0.6390	0.5774	negative
5	2.6400	-0.1356	-1.3493	negative
6	1.7296	0.9846	-0.0449	negative
7	2.2011	-0.7652	0.8617	negative
8	2.4207	1.6033	-0.7196	positive
9	-0.0562	0.5135	0.3967	positive
…	…	…	…	…
1000	0.6225	-0.2738	-0.3229	negative

การลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลจะเป็นการกำจัดข้อมูลที่อยู่ในกลุ่มของคลาสส่วนมากที่อยู่ใกล้ชิดกับข้อมูลที่อยู่ในกลุ่มของคลาสส่วนน้อยภายใต้ค่าระยะทางที่กำหนดออก ในขณะเดียวกันจะเก็บข้อมูลที่อยู่ในกลุ่มของคลาสส่วนน้อยไว้กระบวนการทำงานแสดงดังรูปที่ 3.3

AlgorithmReducing_Overlapping_Ratio

Input:

D_{imb}(imbalanced dataset)

Output:

D_{ror}(imbalanced dataset with reduce overlapping ratio)

1. Input distance ratio into distRatio
2. Split data of D_{imb} into $D_{minority}$and $D_{majority}$
3. For I = 1 to Size of $D_{majority}$Do
4. For J = 1 to Size of $D_{minority}$Do
5. $D[I, J] = \sqrt[2]{(D_{minority}[J] - D_{majority}[I]) * (D_{minority}[J] - D_{majority}[I])^T}$
6. If D ≤ distRatio Then
7. Delete data of $D_{majority}[I]$
8. Combine data of $D_{minority}$ and the rest of $D_{majority}$ then stored into D_{ror}
9. Return D_{ror}
10. End of Reducing_Overlapping_Ratio

รูปที่ 3.3 แสดงขั้นตอนการทำงานของกระบวนการ Reducing_Overlapping_Ratio

จากรูปที่ 3.3เป็นการลดอัตราการซ้อนทับระหว่างกลุ่มของข้อมูลไม่สมดุลที่อยู่ในแฟ้มข้อมูล D_{imb}จนกระทั่งได้ข้อมูลที่มีการลดอัตราการซ้อนทับระหว่างกลุ่มของข้อมูลซึ่งจัดเก็บไว้ที่แฟ้มข้อมูล D_{ror}นั้นจะมีขั้นตอนการทำงานที่เริ่มต้นจากการกำหนดระยะห่างระหว่างข้อมูลทั้งสองกลุ่มด้วยค่าคงที่ จากนั้นแบ่งข้อมูลไม่สมดุลD_{imb}ออกเป็นสองกลุ่ม คือ กลุ่มของข้อมูลคลาสส่วนน้อย($D_{minority}$) และกลุ่มของข้อมูลคลาสส่วนมาก ($D_{majority}$) แล้วทำการคำนวณหาระยะทางระหว่างข้อมูลทั้งสองกลุ่ม โดยจะเริ่มต้นคำนวณหาระยะทางของข้อมูลคู่แรกระหว่างข้อมูลตัวแรกของคลาสส่วนมากกับข้อมูลตัวแรกของคลาสส่วนน้อยเก็บข้อมูลระยะทางที่คำนวณได้ไว้ที่ตัวแปร Dจากนั้นคำนวณหาระยะทางของข้อมูลคู่ถัดไป และทำการคำนวณหาระยะทาง

อย่างนี้ไปจนกระทั่งถึงข้อมูลคู่สุดท้ายระหว่างข้อมูลตัวแรกของคลาสส่วนมากกับข้อมูลตัวแรกของคลาสส่วนน้อย ขั้นตอนถัดไปจะทำการตรวจสอบว่าระยะทางที่คำนวณได้มีค่าน้อยกว่าหรือเท่ากับค่าระยะทางที่กำหนดไว้หรือไม่ ถ้าเป็นจริงจะทำการกำจัดข้อมูลตัวแรกของคลาสส่วนมากทิ้ง และจะทำขั้นตอนการคำนวณหาระยะทางนี้ซ้ำไปจนกระทั่งถึงข้อมูลตัวสุดท้ายของของคลาสส่วนมาก เมื่อทำการคำนวณหาระยะทางเสร็จสิ้นแล้วจะได้ข้อมูลของคลาสส่วนมากที่ไม่มีพื้นที่ซ้อนทับกับข้อมูลของคลาสส่วนน้อย นำข้อมูลชุดนี้ไปรวมเข้ากับข้อมูลของคลาสส่วนน้อยแล้วเก็บไว้ในแฟ้มข้อมูลD_{ror}

ตัวอย่าง การลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล

เริ่มต้นจากการกำหนดระยะทางระหว่างข้อมูลทั้งสองกลุ่มด้วยค่าคงที่ ในที่นี้ได้กำหนดระยะห่างระหว่างข้อมูลdistRatio เท่ากับ 1.25จากนั้นนำข้อมูลไม่สมดุล D_{imb} จากตารางที่ 3.1มาทำการแบ่งออกเป็น 2 กลุ่ม คือ กลุ่มของข้อมูลคลาสส่วนน้อย $D_{minority}$และกลุ่มของข้อมูลคลาสส่วนมาก $D_{majority}$ จะได้ข้อมูลใหม่ดังตารางที่ 3.2และ 3.3

ตารางที่ 3.2 แสดงตัวอย่างข้อมูลของ$D_{minority}$

Transaction ID	x_1	x_2	x_3
8	2.4207	1.6033	-0.7196
9	-0.0562	0.5135	0.3967

ตารางที่ 3.3 แสดงตัวอย่างข้อมูลของ$D_{majority}$

Transaction ID	x_1	x_2	x_3
1	4.1650	0.6268	0.0751
2	3.0591	1.7971	0.2641
3	3.8717	-1.4462	-0.7012
4	4.2460	-0.6390	0.5774
5	2.6400	-0.1356	-1.3493
6	1.7296	0.9846	-0.0449
7	2.2011	-0.7652	0.8617
1000	0.6225	-0.2738	-0.3229

ขั้นตอนถัดไปทำการคำนวณระยะทางระหว่างข้อมูล $D_{minority}$และ $D_{majority}$ผลที่ได้ดังตารางที่ 3.4

ตารางที่ 3.4แสดงระยะทางระหว่างข้อมูลที่อยู่ในกลุ่มของ$D_{minority}$และ $D_{majority}$

ข้อมูล$D_{majority}$	ข้อมูล$D_{minority}$	
Transaction ID	Transaction ID = 8	Transaction ID = 9
1	2.151171	4.2349
~~2~~	***1.188539***	3.3720
3	3.377155	4.5248
4	3.168861	4.4576
5	1.862384	3.2771
~~6~~	***1.147027***	1.8989
7	2.856304	2.6356
1000	2.629452	1.2643

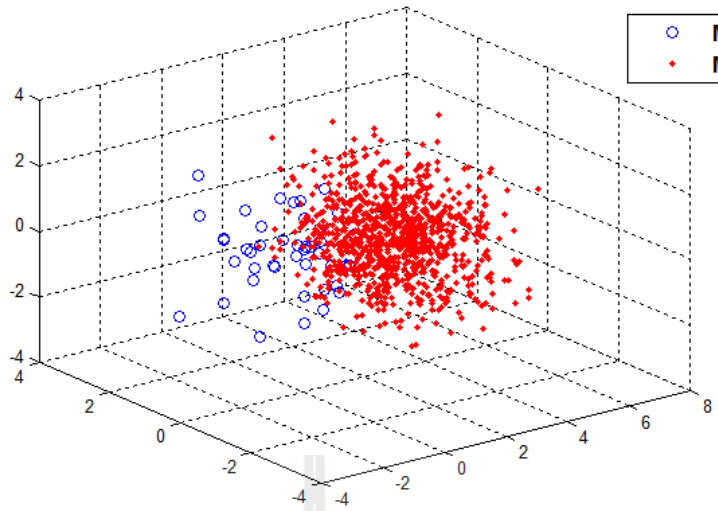
จากตารางที่ 3.4 มีข้อมูลเพียง 2 ตัวอย่างเท่านั้นคือ *Transaction ID2* และ *Transaction ID6* ที่มีระยะทางใกล้กับ *Transaction ID8*คือ 1.188539 และ 1.147027 ตามลำดับ ซึ่งมีระยะทางน้อยกว่าค่าระยะทางที่กำหนดคือ 1.25 ดังนั้นจะทำการกำจัดข้อมูล *Transaction ID2* และ *Transaction ID6* ออก สำหรับข้อมูลอื่น ๆ ที่เหลือจะมีระยะทางที่มากกว่าค่าระยะทางที่กำหนดก็จะเก็บไว้

ขั้นตอนสุดท้ายนำข้อมูลของกลุ่ม $D_{minority}$มารวมเข้ากับข้อมูลที่เหลือของกลุ่ม $D_{majority}$และเก็บไว้ที่ D_{ror}

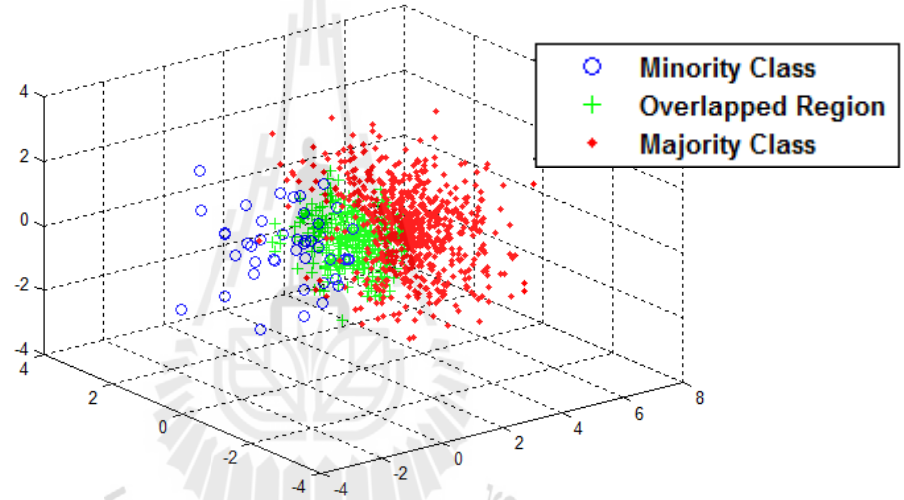
จากตัวอย่างของข้อมูลสังเคราะห์ไม่สมดุลนั้น เมื่อทำการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลแล้วนั้น ข้อมูลคลาสส่วนมากซึ่งมีจำนวนตัวอย่างทั้งหมด956 ตัวอย่าง ปรากฏว่าเมื่อได้มีการคำนวณหาระยะทางระหว่างข้อมูลทั้งสองกลุ่มแล้ว และมีระยะทางระหว่างกลุ่มข้อมูลที่น้อยกว่าระยะทางที่กำหนด ส่งผลให้ข้อมูลคลาสส่วนมากถูกกำจัดออกไปเป็นจำนวน312 ตัวอย่าง จะได้ข้อมูลคลาสส่วนมากที่เหลือชุดใหม่จำนวน644 ตัวอย่าง ดังนั้นข้อมูลไม่สมดุลที่ผ่านการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลซึ่งเก็บไว้ที่ D_{ror}จะมีจำนวนข้อมูลรวมทั้งหมด 688 ตัวอย่าง ซึ่งแบ่งเป็นข้อมูลคลาสส่วนน้อย 44 ตัวอย่างและข้อมูลคลาสส่วนมาก 644 ตัวอย่าง

ลักษณะของข้อมูลสังเคราะห์ไม่สมดุลที่ผ่านการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลโดยภาพรวมแสดงดังรูปที่ 3.4

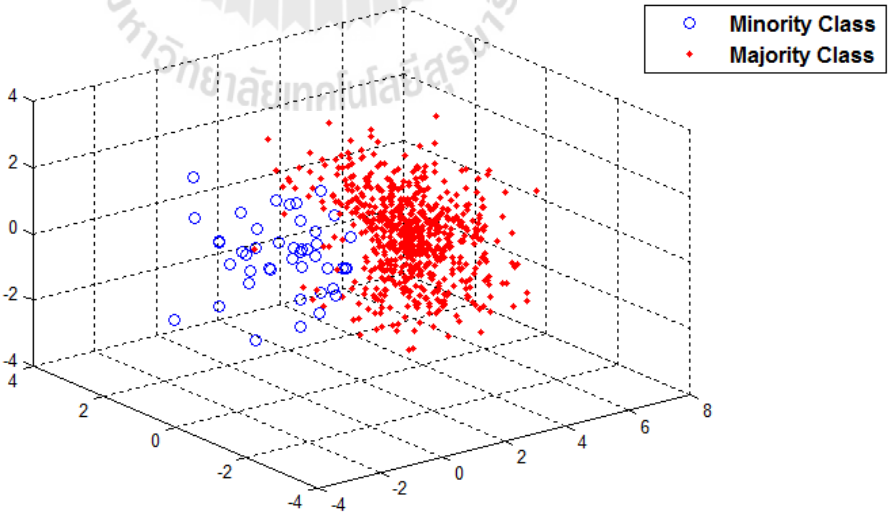
ก)



ข)



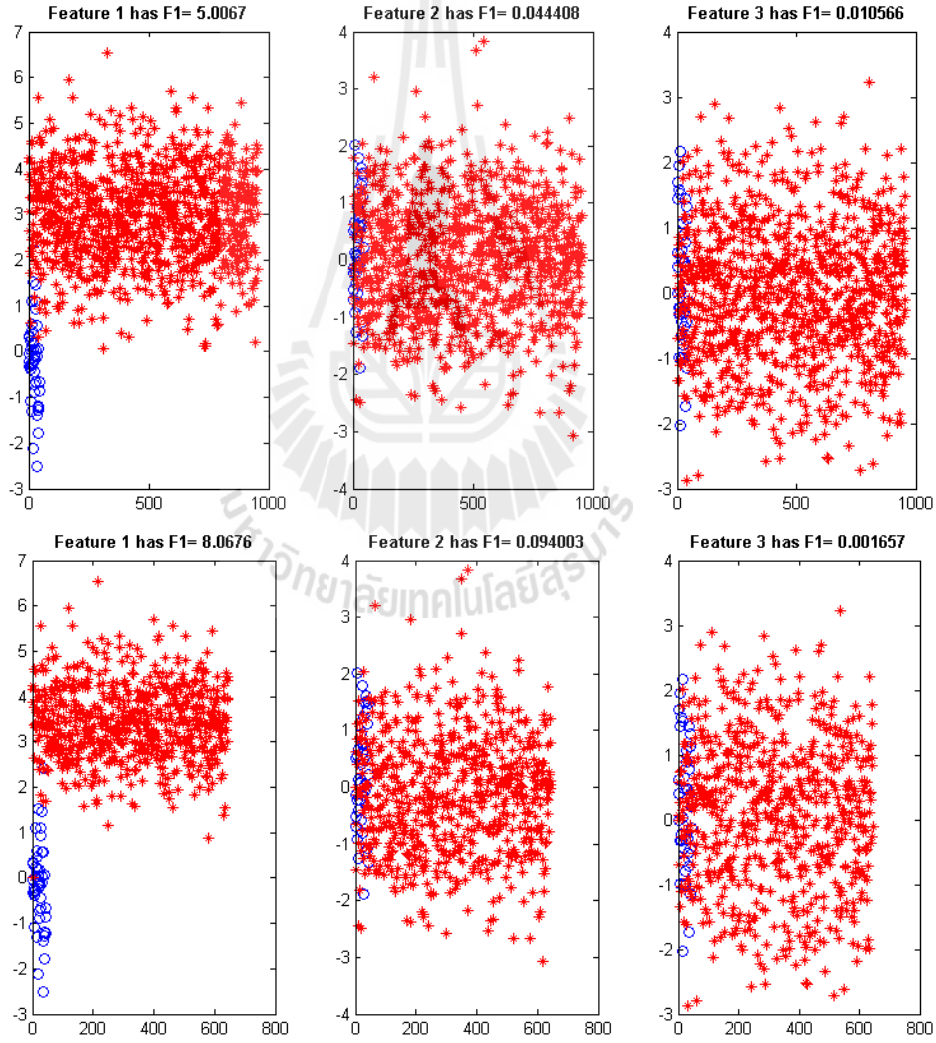
ค)



รูปที่ 3.4แสดงลักษณะข้อมูลสังเคราะห์ไม่สมดุลก) ข้อมูลดั้งเดิม ข) ข้อมูลที่มีการใช้พื้นที่ร่วมกัน ค) ข้อมูลที่ผ่านการการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล

จากรูปที่ 3.4ข้อมูลดั้งเดิมซึ่งเป็นข้อมูลไม่สมดุล (รูป ก) นั้นจะมีข้อมูลกลุ่ม Majority ซึ่งแทนด้วยสัญลักษณ์ดอกจันสีแดง และข้อมูลกลุ่ม Minority ซึ่งแทนด้วยสัญลักษณ์วงกลมสีน้ำเงิน เมื่อนำข้อมูลชุดนี้มาหาพื้นที่ที่มีการใช้ร่วมกันระหว่างสองคลาส จะปรากฏว่ามีข้อมูลกลุ่ม Majority ซึ่งแทนด้วยเครื่องหมาย “+” สีเขียว (รูป ข) ใช้พื้นที่ร่วมกันกับข้อมูลกลุ่ม Minority ภายใต้ระยะห่างระหว่างกลุ่มที่กำหนด ดังนั้นจะทำการกำจัดข้อมูลกลุ่ม Majority เฉพาะพื้นที่ที่แสดงด้วยสีเขียว ในที่สุดจะได้ข้อมูลไม่สมดุลที่มีการใช้พื้นที่ร่วมกันอย่างเบาบาง (รูป ค)

เมื่อนำข้อมูลไม่สมดุลแต่ละมิติมาทำการหาอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลด้วยตัววัด Fisher’s Discriminant Ratio (F1)ผลที่ได้แสดงดังรูปที่ 3.5



รูปที่ 3.5แสดงลักษณะข้อมูลสังเคราะห์ไม่สมดุลของแต่ละมิติก่อน (บน) และหลัง (ล่าง) การลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล

จากรูปที่ 3.5ค่า F1 ที่มีค่าต่ำนั้นจะหมายถึง การซ้อนทับกันระหว่างกลุ่มข้อมูลมีอัตราที่สูง จากรูปพบว่า อัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลของแต่ละมิตินั้นจะลดลง เช่น รูปบนของมิติที่ 1ถึงมิติที่ 3 มีค่า F1 เท่ากับ 5.0067 0.0444408 และ 0.010566 ตามลำดับ เมื่อผ่านกระบวนการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลของมิติที่ 1 ถึงมิติที่ 3 (รูปล่าง) มีค่า F1 เท่ากับ 8.0676 0.094003 และ 0.001657 ตามลำดับเมื่อพิจารณาค่า maxF จะพบว่าข้อมูลสังเคราะห์ไม่สมดุลก่อนการลดอัตราการซ้อนทับจะมีค่า maxF อยู่ที่ 5.0067 และหลังการลดอัตราการซ้อนทับจะมีค่า maxF อยู่ที่ 8.0676จะเห็นได้ว่าค่าของmaxF มีค่าเพิ่มขึ้น ซึ่งแสดงให้เห็นถึงการซ้อนทับกันของข้อมูลสองกลุ่มนั้นเบาบางลง

3.2.2การสุ่มเลือกข้อมูล

การสุ่มเลือกข้อมูลนั้นจะสุ่มเลือกจากชุดข้อมูลไม่สมดุลที่ผ่านการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลแล้ว โดยจะทำการสุ่มเลือกข้อมูลกลุ่มส่วนน้อยและข้อมูลกลุ่มส่วนมากแยกจากกัน ในงานวิจัยนี้จะสุ่มเลือกข้อมูลกลุ่มส่วนน้อยเพียงครั้งเดียว และสุ่มเลือกข้อมูลกลุ่มส่วนมากด้วยจำนวนข้อมูลที่แตกต่างกันจากนั้นนำข้อมูลกลุ่มส่วนน้อยและข้อมูลกลุ่มส่วนมากของการสุ่มเลือกแต่ละครั้งมารวมกัน ข้อมูลที่ได้นี้จะนำไปใช้ในขั้นตอนของการสร้างโมเดลการเรียนรู้ร่วมกัน

ตัวอย่าง การสุ่มเลือกข้อมูล

ทำการสุ่มเลือกข้อมูลกลุ่ม $D_{minority}$ จำนวน20ตัวอย่างเก็บไว้ที่ข้อมูล D_{ssMin} และทำการสุ่มเลือกข้อมูลกลุ่ม$D_{majority}$ จำนวน200ตัวอย่างเก็บไว้ที่ข้อมูล D_{ssMaj} นำข้อมูลทั้งสองกลุ่มมารวมเข้าด้วยกันจะได้ข้อมูลไม่สมดุลชุดใหม่(D_{ss}) ซึ่งมีจำนวนข้อมูล220ตัวอย่าง

ครั้งถัดไปของการทำงานจะทำการสุ่มเลือกเฉพาะข้อมูลกลุ่ม $D_{majority}$ เท่านั้น ส่วนข้อมูลกลุ่ม $D_{minority}$ จะใช้ข้อมูลชุดเดิม นำข้อมูลทั้งสองชุดมารวมกันแล้วนำไปสร้างโมเดลจะทำเช่นนี้ไปจนกว่าจะสิ้นสุดการทำงานของโปรแกรม

3.2.3 การสร้างตารางค่าใช้จ่าย

การสร้างตารางค่าใช้จ่ายสำหรับชดเชยการจำแนกผิดประเภทนั้น มีขั้นตอนการสร้างตารางดังรูปที่ 3.6

AlgorithmCreate_Cost_Matrix

Input:

$N_{minority}$(number of imbalanced data in minority group)

$N_{majority}$(number of imbalanced data in majority group)

Output:

costValue(cost matrix of imbalanced classification)

1. $C_{minority} = roundup\left(\frac{N_{majority}}{N_{minority}}\right)$

2. Set costValue $= \begin{bmatrix} C(0,0) & C_{majority} \\ C_{minority} & C(1,1) \end{bmatrix}$

3. Return costValue

4. End of Create_Cost_Matrix

รูปที่ 3.6แสดงขั้นตอนการทำงานของกระบวนการCreate_Cost_Matrix

จากรูปที่ 3.6 แสดงขั้นตอนการสร้างตารางค่าใช้จ่าย ในกระบวนการทำงานนี้จะนำจำนวนข้อมูลไม่สมดุลของทั้งสองกลุ่มซึ่งเก็บไว้ที่ $N_{minority}$ และ $N_{majority}$ มาทำการคำนวณหาอัตราความไม่สมดุล จากนั้นนำค่าอัตราความไม่สมดุลที่คำนวณได้มากำหนดให้เป็นค่าใช้จ่ายของข้อมูลกลุ่มน้อย ($C_{minority}$) สำหรับค่าใช้จ่ายอื่น ๆ นั้นจะกำหนดเป็นค่าคงที่ ดังนี้

สำหรับกรณีที่มีการจำแนกผิดประเภทของข้อมูลMajorityหรือ $C_{majority}$ จะมีค่าใช้จ่ายเท่ากับ *1*

สำหรับกรณีที่มีการจำแนกประเภทได้ถูกต้องนั้นจะไม่มีค่าใช้จ่ายดังนั้นจะกำหนด *C(0,0)*และ *C(1,1)*เท่ากับ *0*

ตัวอย่าง การสร้างตารางค่าใช้จ่าย

จากจำนวนข้อมูลตัวอย่างข้อมูลไม่สมดุลสามารถนำมาสร้างค่าใช้จ่ายของการจำแนกผิดประเภทของข้อมูล Minority หรือ $C_{minority}$หรือ *C(1,0)* ได้ดังนี้

$$C_{minority} = \frac{200}{20} = 10$$

ผลที่ได้จากการสร้างตารางค่าใช้จ่ายแสดงดังตารางที่ 3.5

ตารางที่ 3.5 แสดงตารางค่าใช้จ่ายสำหรับการจำแนกประเภทข้อมูล2 กลุ่ม

	Predicted	
Actual	Majority Class (0)	Minority Class (1)
Majority Class (0)	$C(0,0) = 0$	$C(0,1) = C_{majority} = 1$
Minority Class (1)	$C(1,0) = C_{minority} = 10$	$C(1,1) = 0$

ดังนั้นค่าใช้จ่ายสำหรับการจำแนกประเภทข้อมูล2 กลุ่ม ซึ่งเก็บไว้ที่ตัวแปร costValue คือ

$$costVaule = \begin{bmatrix} 0 & 1 \\ 10 & 0 \end{bmatrix}$$

3.2.4การสร้างโมเดล

การสร้างโมเดลสำหรับเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุลดังรูปที่ 3.7

AlgorithmCreate_Classification_Ensemble_Model

Input :

D_{ss}(imbalanced data from sampling step) //ข้อมูลไม่สมดุลที่ได้จากการสุ่มเลือก

costValue(cost matrix of imbalanced classification)

U(set of parameter for create ensemble model)

Output:

[optDT,errV](set of misclassification with stored number of Decision Treeand error value)

1. Create classification ensemble model then stored the result of model at EnsModel
2. Keep an error of misclassifications from EnsModel into set of misclassification[optDT,errV]
3. Show misclassification error by generate visualization
4. Return [optDT,errV]
5. End of Create_Classification_Ensemble_Model

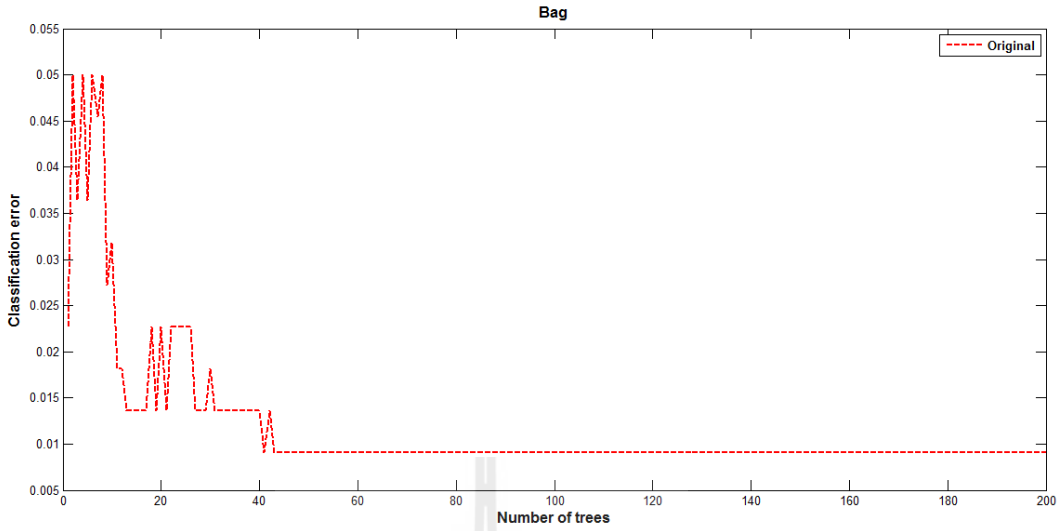
รูปที่ 3.7แสดงขั้นตอนการทำงานของกระบวนการCreate_Classification_Ensemble_Model

จากรูปที่ 3.7แสดงกระบวนการสร้างโมเดลการทำงานร่วมกันด้วยการนำข้อมูลไม่สมดุลที่อยู่ในแฟ้มข้อมูล D_{ss}มาทำการสร้างโมเดล ในขั้นตอนนี้จะเป็นการเรียนรู้ร่วมกันแบบการใช้การตัดสินใจร่วมกันด้วยอัลกอริทึมแบบแบ็กกิงและบูสต์ติง ซึ่งอัลกอริทึมที่นำมาใช้นั้นจะเก็บไว้ในตัวแปรที่ชื่อ nAlgo โดยมีตัวจำแนกประเภทข้อมูลเป็นต้นไม้ตัดสินใจ(nClassify) จากนั้นกำหนดจำนวนเริ่มต้นของต้นไม้ตัดสินใจ (numDT) ชดเชยการทำนายผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่าย (costValue) และทดสอบประสิทธิภาพของโมเดลด้วยการทดสอบแบบไขว้ข้าม (kFold)ในขณะที่มีการสร้างโมเดล EnsModel นั้นจะมีความผิดพลาดเกิดขึ้น ซึ่งความผิดพลาดนั้นจะเกิดขึ้นจากการจำแนกข้อมูลผิดประเภท ดังนั้นจะทำการเก็บข้อมูลของจำนวนของต้นไม้ตัดสินใจที่ส่งผลให้เกิดการจำแนกผิดประเภทและค่าความผิดพลาดไว้ที่ตัวแปร optDT และ errV ตามลำดับ จากนั้นทำการมโนภาพหรือวิชวลไลเซชั่นเพื่อแสดงความผิดพลาดที่เกิดจากการจำแนกประเภทผิดคลาสของข้อมูลทดสอบ ณ ตำแหน่งของจำนวนต้นไม้ตัดสินใจ นำจำนวนต้นไม้ตัดสินใจและค่าความผิดพลาดจากการจำแนกประเภทซึ่งเก็บไว้ที่ตัวแปร [optDT,errV] ไปใช้สำหรับสร้างโมเดลการทำงานร่วมกันอีกครั้ง

ตัวอย่าง การสร้างโมเดล

นำข้อมูลไม่สมดุล D_{ss}ซึ่งมีข้อมูลจำนวน 220 ตัวอย่าง โดยแบ่งเป็นข้อมูลกลุ่ม Majority จำนวน 200 ตัวอย่าง และข้อมูลกลุ่ม Minority จำนวน 20 ตัวอย่างมาสร้างโมเดลการเรียนรู้ร่วมกันแบบการใช้การตัดสินใจร่วมกันด้วยอัลกอริทึมแบบแบ็กกิงและบูสต์ติง ซึ่งอัลกอริทึมที่จะนำมาใช้นั้นมีทั้งหมด 5 อัลกอริทึมด้วยกัน คือ AdaboostM1, Bag,TotalBoost, LogitBoost และ RUSBoostในตัวอย่างนี้จะสร้างโมเดลด้วยอัลกอริทึม Bagจากนั้นกำหนดตัวจำแนกประเภทข้อมูลเป็นต้นไม้ตัดสินใจกำหนดค่าใช้จ่ายสำหรับทำนายผิดกลุ่มด้วยค่า *costValue*ที่ได้จากขั้นตอนก่อนหน้าจำนวนต้นไม้ตัดสินใจด้วย numDT ซึ่งเริ่มต้นที่ 200 ต้นไม้ตัดสินใจและทดสอบประสิทธิภาพของโมเดลด้วยการทดสอบแบบไขว้ข้ามที่ k เท่ากับ5

ในขั้นตอนของการสร้างโมเดล EnsModel นั้นจะใช้การจำแนกประเภทด้วยต้นไม้ตัดสินใจซึ่งจะใช้ตั้งแต่จำนวน 1 ต้นถึงจำนวน200ต้น ส่งผลให้มีความผิดพลาดในการจำแนกประเภทข้อมูลเกิดขึ้น และจะทำการจัดเก็บข้อมูลความผิดพลาดของการจำแนกประเภท ณ ตำแหน่งต้นไม้ตัดสินใจไว้ที่ตัวแปร errV และจำนวนต้นไม้ตัดสินใจที่ตัวแปร optDTจากนั้นทำการมโนภาพหรือวิชวลไลเซชั่นเพื่อแสดงให้เห็นถึงความผิดพลาดที่เกิดจากการจำแนกประเภทผิดคลาสของข้อมูลทดสอบ ณ ตำแหน่งของจำนวนต้นไม้ตัดสินใจ ผลของการมโนภาพหรือวิชวลไลเซชั่นจะแสดงดังรูปที่ 3.8



รูปที่ 3.8แสดงความผิดพลาดของการจำแนกประเภทข้อมูลด้วยต้นไม้ตัดสินใจ ตั้งแต่จำนวน 1 ต้นถึง 200 ต้น

3.2.5การเลือกจำนวนต้นไม้ตัดสินใจที่เหมาะสม

ในการสร้างโมเดลการเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุลด้วยโครงสร้างต้นไม้ตัดสินใจนั้น จำนวนของต้นไม้ตัดสินใจจะมีผลต่อเวลาที่ใช้ในการสร้างโมเดล ถ้ากำหนดจำนวนต้นไม้ตัดสินใจที่มากเกินไปจะส่งผลให้เวลาที่ใช้ในการสร้างโมเดลมากขึ้นด้วย

ขั้นตอนของการหาจำนวนต้นไม้ตัดสินใจที่เหมาะสม แสดงดังรูปที่ 3.9ซึ่งกระบวนการเลือกจำนวนต้นไม้ตัดสินใจที่เหมาะสมสำหรับการสร้างโมเดลการเรียนรู้ร่วมกันเริ่มต้นด้วยการนำข้อมูลความผิดพลาดของการจำแนกประเภทมาจัดเรียงลำดับตามค่าความผิดพลาดจากน้อยไปมาก แล้วทำการเลือกจำนวนต้นไม้ตัดสินใจที่มีค่าความผิดพลาดต่ำสุดTopK อันดับ ซึ่งในงานวิจัยนี้จะเลือกจำนวนต้นไม้ตัดสินใจที่มีค่าความผิดพลาดในการจำแนกประเภทข้อมูลต่ำที่สุดจำนวน 10 อันดับแรก จากนั้นทำการสร้างโมเดลการเรียนรู้ด้วยจำนวนต้นไม้ตัดสินใจที่เลือก ซึ่งการสร้างโมเดลในขั้นตอนนี้จะทำการวนรอบสร้างโมเดลการเรียนรู้อย่างน้อย 10 รอบ เพื่อให้ได้มาซึ่งจำนวนต้นไม้ตัดสินใจที่เหมาะสมและมีค่าความผิดพลาดในการจำแนกประเภทน้อยที่สุด สำหรับค่าพารามิเตอร์อื่น ๆ นั้นจะเป็นค่าเดิมที่ใช้ในขั้นตอนก่อนหน้า

AlgorithmChoose_Optimal_DecisionTree

Input:

D_{ss} (imbalanced data from sampling step) //ข้อมูลไม่สมดุลที่ได้จากการสุ่มเลือก

costValue(cost matrix of imbalanced classification)

U(set of parameter for create ensemble model)

[optDT,errV](set of misclassification with stored number of Decision Tree and error value)

Output:

ListEnsModel(set of performance ensemble model)

1. Reorder set of misclassification by errVascending //เรียงลำดับข้อมูลของชุดข้อมูลความผิดพลาดจากการจำแนกประเภทตามลำดับของค่าความผิดพลาดจากน้อยไปมาก
2. Choose TopK of optDT with minimum error //เลือกจำนวนต้นไม้ตัดสินใจที่มีค่าความผิดพลาดต่ำสุด10 อันดับแรก
3. For I = 1 to TopK Do
4. สร้างโมเดลการเรียนรู้ร่วมกันด้วยจำนวนต้นไม้ตัดสินใจที่เก็บไว้ที่ optDT[I]
5. ทำการวัดประสิทธิภาพของโมเดลที่ได้ด้วยมาตรวัดต่าง ๆ
6. เก็บข้อมูลและประสิทธิภาพของโมเดลที่ได้ไว้ที่ ListEnsModel
7. End For
8. Return ListEnsModel
9. End of Choose_Optimal_DecisionTree

รูปที่ 3.9แสดงขั้นตอนการทำงานของกระบวนการChoose_Optimal_DecisionTree

ในแต่ละรอบของการสร้างโมเดลนั้นจะมีการทำงานในส่วนของการมโนภาพหรือวิชวลไลเซชันด้วยการแสดงรูปภาพเกี่ยวกับค่าความผิดพลาดที่เกิดจากการจำแนกประเภทผิดคลาสของข้อมูลทดสอบ ณ ตำแหน่งของจำนวนต้นไม้ตัดสินใจและวัดประสิทธิภาพของโมเดลที่ได้ด้วยมาตรวัดต่าง ๆ

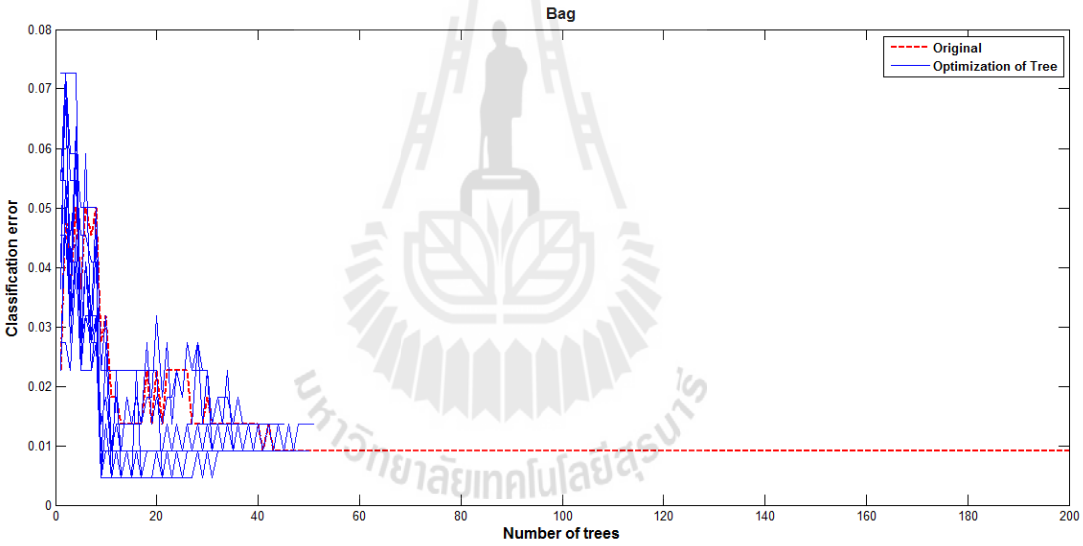
ผลที่ได้ของขั้นตอนนี้คือประสิทธิภาพของโมเดล ณ ตำแหน่งของจำนวนต้นไม้ตัดสินใจที่เลือกซึ่งเก็บไว้ที่ ListEnsModelเมื่อทำงานครบ 10 รอบจะทำการส่งข้อมูลของโมเดลพร้อมทั้งประสิทธิภาพเพื่อนำไปใช้ในขั้นตอนการคัดเลือกโมเดลที่มีประสิทธิภาพที่ดีที่สุด

ตัวอย่างการเลือกจำนวนต้นไม้ตัดสินใจที่เหมาะสม

จากรูปที่ 3.8จะเห็นว่าจำนวนต้นไม้ที่เหมาะสมที่มีค่าความผิดพลาดของการจำแนกประเภทข้อมูลต่ำที่สุดจะมีค่าประมาณ 0.009ซึ่งจะอยู่ที่การใช้จำนวนต้นไม้ตัดสินใจ 41 ต้นไม้ตัดสินใจ และช่วงของการใช้จำนวนต้นไม้ตัดสินใจตั้งแต่43ถึง 200 ต้นไม้ตัดสินใจ

ดังนั้นได้ทำการนำข้อมูลความผิดพลาดของการจำแนกประเภทซึ่งจัดเก็บไว้ที่ [optDT,errV]มาจัดเรียงลำดับตามค่าความผิดพลาดจากน้อยไปมากแล้วเลือกจำนวนต้นไม้ตัดสินใจที่มีค่าความผิดพลาดในการจำแนกประเภทต่ำสุด 10 อันดับแรก ผลที่ได้ของจำนวนต้นไม้ตัดสินใจที่เหมาะสม 10 อันดับแรกประกอบด้วย 414344454647484950และ51ต้นไม้ตัดสินใจ

ผลที่ได้จากการนำจำนวนต้นไม้ตัดสินใจที่เหมาะสมไปสร้างโมเดลเรียนรู้ร่วมกันสามารถแสดงผลการสร้างมโนภาพได้ดังรูปที่ 3.10



รูปที่ 3.10แสดงความผิดพลาดของการจำแนกประเภทข้อมูลด้วยจำนวนต้นไม้ตัดสินใจที่เหมาะสม

สำหรับโมเดลสำหรับเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุลด้วยต้นไม้ตัดสินใจที่เหมาะสมนั้นจะนำไปวัดประสิทธิภาพด้วยมาตรวัดที่ได้กล่าวไว้แล้วในหัวข้อ 2.8

จากตัวอย่างที่กล่าวมาแล้วตั้งแต่ต้นนั้นเป็นการสร้างโมเดลสำหรับเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุลด้วยต้นไม้ตัดสินใจที่เหมาะสมซึ่งมีการเรียนรู้ร่วมกันด้วยอัลกอริทึมแบบBagนำโมเดลที่ได้มาทำการวัดประสิทธิภาพด้วยมาตรวัดต่าง ๆ ผลที่ได้แสดงดังตารางที่ 3.6

ตารางที่3.6แสดงประสิทธิภาพของโมเดลการเรียนรู้ร่วมกันแบบ Bag ด้วยมาตรวัดต่าง ๆ

เรียงลำดับตามการรันโปรแกรม

ลำดับการรันโปรแกรม	จำนวนต้นไม้ตัดสินใจ	มาตรวัดประสิทธิภาพ							
		Accuracy	Precision	Sensitivity	Specificity	F-measure	G-mean	AUC	Total Misclassification Costs
1	41	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
2	43	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
3	44	0.986	1.000	0.850	1.000	0.919	0.922	0.925	0.150
4	45	0.986	1.000	0.850	1.000	0.919	0.922	0.925	0.150
5	46	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
6	47	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
7	48	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
8	49	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
9	50	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
10	51	0.986	1.000	0.850	1.000	0.919	0.922	0.925	0.150

3.2.6การเลือกโมเดลที่มีประสิทธิภาพ

เมื่อทำการสร้างโมเดลการเรียนรู้ร่วมกันของการจำแนกประเภทข้อมูลไม่สมดุลครบทุกอัลกอริทึมแล้ว ขั้นตอนสุดท้ายคือ การคัดเลือกโมเดลที่มีประสิทธิภาพเพื่อนำมาใช้สำหรับจำแนกประเภทข้อมูลไม่สมดุลด้วยการพิจารณาจากค่า Total Misclassification Costs ซึ่งเป็นค่าที่ได้จากผลรวมความผิดพลาดของการจำแนกประเภทของข้อมูลกลุ่ม Minority หรือ FPRateและข้อมูลกลุ่ม Majority หรือ FNRate โดยจะเลือกโมเดลที่มีค่า Total Misclassification Costs ต่ำที่สุด ในกรณีที่มีมากกว่า 1 ค่าจะนำข้อมูลของจำนวนต้นไม้ตัดสินใจมาทำการพิจารณาร่วมด้วยซึ่งหมายความว่า จะคัดเลือกโมเดลเพื่อให้ได้มาซึ่งโมเดลที่มีประสิทธิภาพด้วยการพิจารณา Total Misclassification Costs ที่มีค่าต่ำที่สุด และใช้จำนวนต้นไม้ตัดสินใจที่น้อยที่สุดด้วยเช่นกัน ผลที่ได้คือโมเดลที่มีความผิดพลาดในการจำแนกน้อยที่สุดซึ่งใช้จำนวนต้นไม้ตัดสินใจที่เหมาะสมที่สุด นำโมเดลที่ได้นี้ไปแสดงผล

ตัวอย่างการเลือกโมเดลที่มีประสิทธิภาพ

จากตารางที่ 3.6นำข้อมูลประสิทธิภาพของโมเดลมาทำการเรียงลำดับตามค่าความผิดพลาดTotal Misclassification Costs และจำนวนต้นไม้ตัดสินใจจากน้อยไปมาก ผลที่ได้ดังตารางที่ 3.7

ตารางที่ 3.7แสดงประสิทธิภาพของโมเดลการเรียนรู้ร่วมกันแบบ Bag ด้วยมาตรวัดต่าง ๆ เรียงลำดับตามค่าความผิดพลาดและจำนวนต้นไม้ตัดสินใจจากน้อยไปมาก

ลำดับการเรียนรู้	จำนวนต้นไม้ตัดสินใจ	มาตรวัดประสิทธิภาพ							
		Accuracy	Precision	Sensitivity	Specificity	F-measure	G-mean	AUC	Total Misclassification Costs
1	***41***	**0.991**	1.000	0.900	1.000	0.947	0.949	0.950	***0.100***
2	43	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
3	46	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
4	47	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
5	48	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
6	49	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
7	50	0.991	1.000	0.900	1.000	0.947	0.949	0.950	0.100
8	44	0.986	1.000	0.850	1.000	0.919	0.922	0.925	0.150
9	45	0.986	1.000	0.850	1.000	0.919	0.922	0.925	0.150
10	51	0.986	1.000	0.850	1.000	0.919	0.922	0.925	0.150

จากตารางที่ 3.7 จะเห็นว่าความผิดพลาดในการจำแนกประเภทข้อมูลที่ต่ำที่สุด คือ 0.100นั้นมีมากกว่าหนึ่งโมเดล แต่ในงานวิจัยนี้จะเลือกโมเดลที่มีค่าความผิดพลาดในการจำแนกประเภทข้อมูลที่ต่ำที่สุดและใช้จำนวนต้นไม้ตัดสินใจที่น้อยที่สุดด้วยเพื่อลดเวลาในการสร้างโมเดล ดังนั้นจะเลือกโมเดลที่สร้างด้วยต้นไม้ตัดสินใจจำนวน 41 ต้น

บทที่ 4
การทดสอบและอภิปรายผล

ในส่วนของบทที่ 4 นี้จะเป็นการทดสอบและอภิปรายผลของการใช้อัลกอริทึม EnsDTV เพื่อจำแนกประเภทข้อมูลไม่สมดุลซึ่งข้อมูลนั้นจะมีอัตราความไม่สมดุลของกลุ่มข้อมูลและมีอัตราการซ้อนทับกันของกลุ่มข้อมูลที่แตกต่างกันสำหรับเนื้อหาในบทนี้จะประกอบด้วย การเตรียมข้อมูลสำหรับการทดสอบ การออกแบบวิธีการทดสอบ ผลการทดสอบประสิทธิภาพ และอภิปรายผล

4.1 การเตรียมข้อมูลสำหรับการทดสอบ

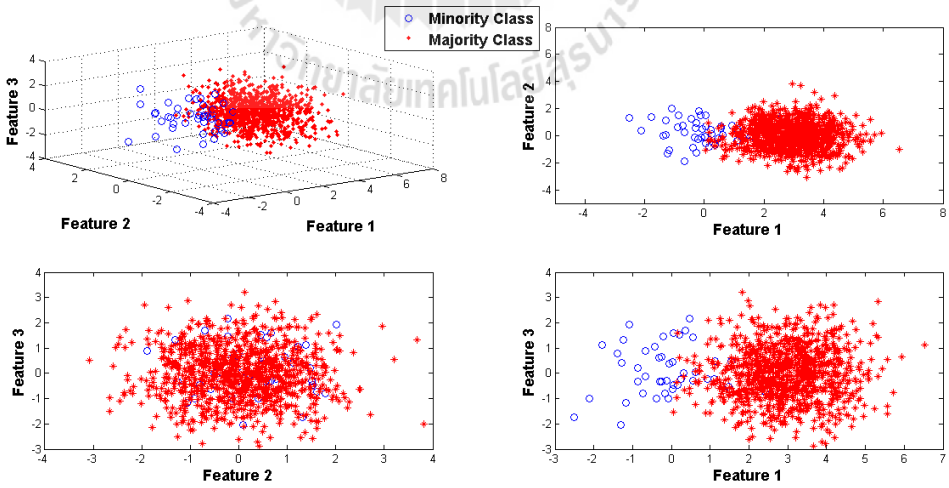
งานวิจัยของวิทยานิพนธ์ฉบับนี้ได้ทำการวัดประสิทธิภาพของโมเดลด้วยชุดข้อมูลชนิดตัวเลขที่ได้จากการสังเคราะห์จากโปรแกรมและชุดข้อมูลจากแหล่งข้อมูลมาตรฐาน โดยมีรายละเอียดดังต่อไปนี้

4.1.1 ชุดข้อมูลสังเคราะห์จากโปรแกรม

สำหรับชุดข้อมูลสังเคราะห์ที่ใช้ในงานวิจัยของวิทยานิพนธ์ฉบับนี้จะทำการสังเคราะห์ขึ้นมาด้วยโปรแกรม MATLAB R2013bลักษณะของข้อมูลที่สังเคราะห์ขึ้นมานั้นจะเป็นข้อมูลชนิดตัวเลขเท่านั้น มีการกระจายตัวของข้อมูลเป็นแบบมาตรฐาน มีการซ้อนทับกันของข้อมูลมีอัตราความน่าจะเป็นของการเกิดคลาสส่วนน้อยและคลาสส่วนมากที่แตกต่างกัน จำนวนคุณลักษณะ (Attributes) เท่ากับ 3 และมีคลาสเป้าหมายสองกลุ่มรายละเอียดชุดข้อมูลสังเคราะห์จากโปรแกรมแสดงดังตารางที่ 4.1และลักษณะการกระจายตัวของชุดข้อมูลสังเคราะห์แสดงดังรูปที่ 4.1

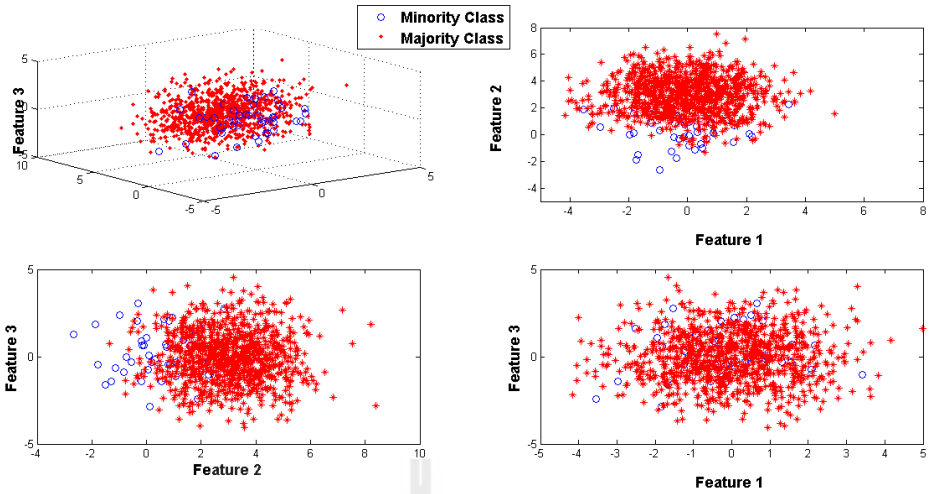
ตารางที่ 4.1แสดงลักษณะของชุดข้อมูลสังเคราะห์จากโปรแกรม

ชุดข้อมูล	ค่า Mean	ค่าความแปรปรวน		จำนวนตัวอย่าง			IR	maxF
		กลุ่ม Majority	กลุ่ม Minority	กลุ่ม Majority	กลุ่ม Minority	รวม		
Data1	[0 0 0; 3 0 0]	[1 0 0; 0 1 0; 0 0 1]	[1 0 0; 0 1 0; 0 0 1]	956	44	1000	21.72	5.01
Data2	[0 0 0; 0 30]	[2 0 0; 0 2 0; 0 0 2]	[2 0 0; 0 2 0; 0 0 2]	956	44	1000	21.72	1.85
Data3	[0 0 0; 0 30]	[0.2 0 0; 0 0.2 0; 0 0 0.2]	2 0 0; 0 2 0; 0 0 2]	956	44	1000	21.72	3.81
Data4	[0 0 0; 3 3 0]	[1 0.50.5; 0.510.5; 0.50.51]	[1 0.50.5; 0.510.5; 0.50.51]	956	44	1000	21.72	5.01

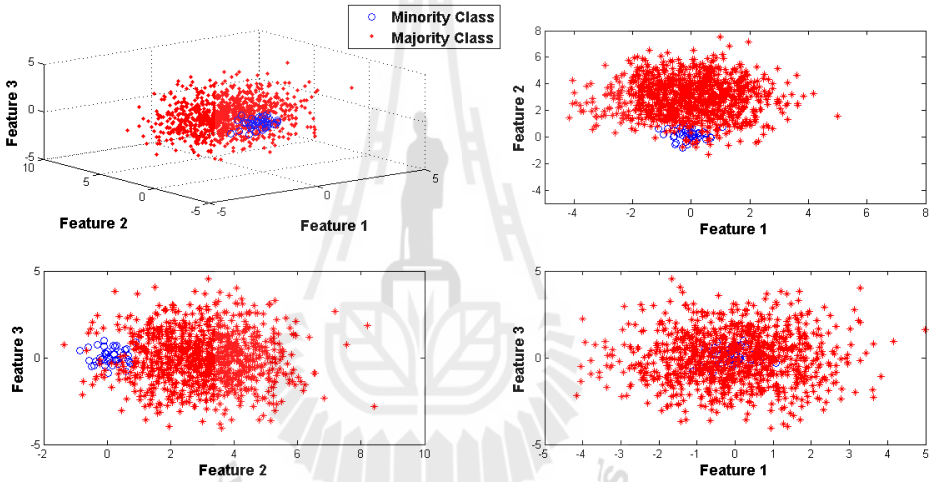


ก) ชุดข้อมูลData1

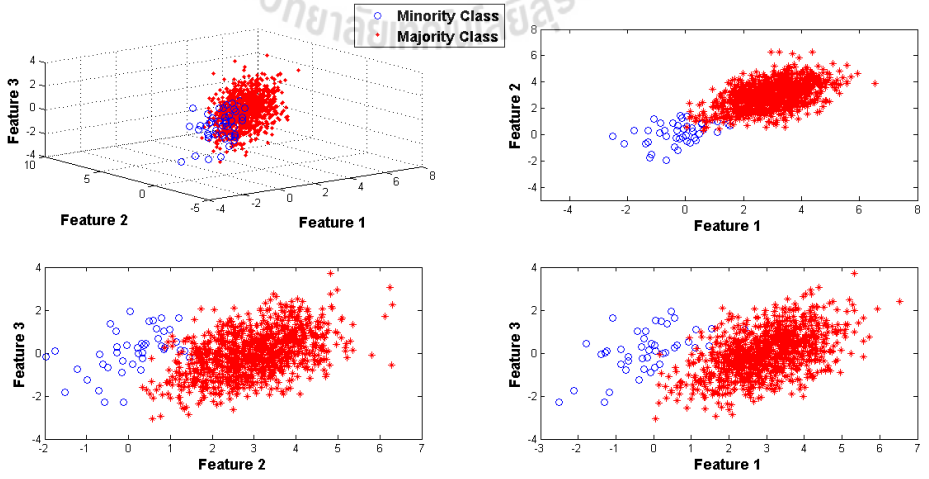
รูปที่ 4.1 แสดงลักษณะการกระจายตัวของชุดข้อมูลสังเคราะห์ตั้งแต่ Data1 ถึง Data4



ข) ชุดข้อมูล Data2



ค) ชุดข้อมูลData3



ง) ชุดข้อมูล Data4

รูปที่ 4.1 แสดงลักษณะการกระจายตัวของชุดข้อมูลสังเคราะห์ตั้งแต่Data1 ถึง Data4(ต่อ)

4.1.2 ชุดข้อมูลจากแหล่งข้อมูลมาตรฐาน

สำหรับชุดข้อมูลชุดที่สองที่ใช้ในวิทยานิพนธ์นี้จะนำมาจากแหล่งข้อมูลมาตรฐาน KeelRepository (http://www.keel.es/datasets.php)จำนวน 9 ชุดข้อมูล รายละเอียดดังตารางที่ 4.2

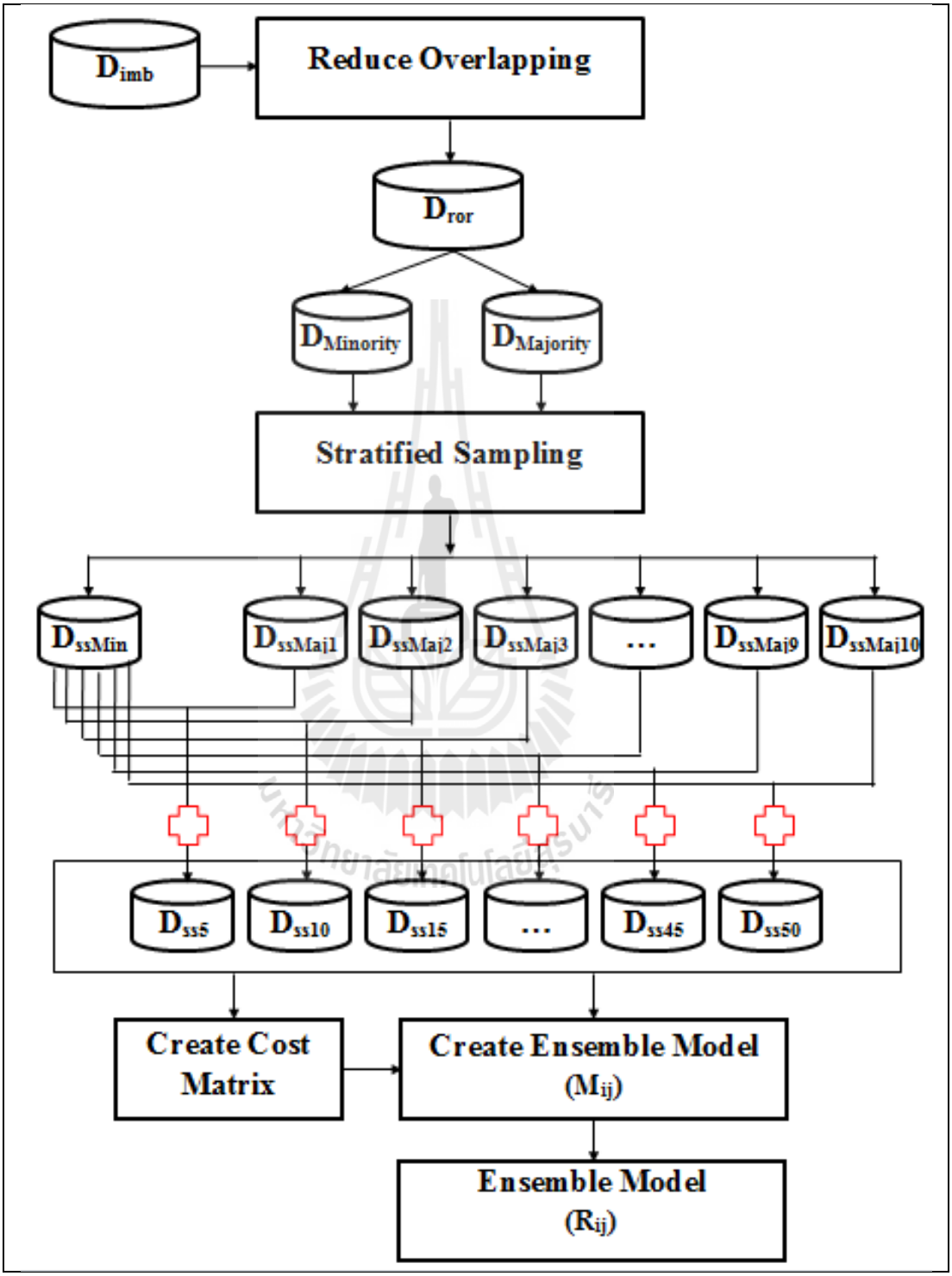
ตารางที่ 4.2แสดงลักษณะชุดข้อมูลจากแหล่งข้อมูลมาตรฐานKeel

ชุดข้อมูล	ลักษณะข้อมูล (R/I/N)*	จำนวนตัวอย่าง			IR	maxF
		ทั้งหมด	กลุ่ม Majority	กลุ่ม Minority		
ecoli4	7 (7/0/0)	336	316	20	15.8	3.17
glass-6	9 (9/0/0)	214	185	29	6.38	2.35
new-thyroid1	5 (4/1/0)	215	180	35	5.14	3.50
page-blocks	10 (4/6/0)	5472	4913	559	8.79	0.51
pima	8 (8/0/0)	768	500	268	1.87	0.57
segment	18 (18/0/0)	2308	1979	329	6.02	1.80
shuttle	9 (0/9/0)	1829	1706	123	13.87	12.96
vehicle	18 (0/18/0)	846	628	218	2.88	0.38
yeast	8 (8/0/0)	1484	1321	163	8.10	2.74
หมายเหตุ * R/I/N คือ ชนิดของข้อมูล Real/Integer/Nominal						

4.2 การออกแบบวิธีการทดสอบ

สำหรับการออกแบบวิธีการทดสอบอัลกอริทึม EnsDTV นั้นจะออกแบบการทดสอบโดยการนำข้อมูลไม่สมดุลที่ผ่านกระบวนการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล(D_{ror}) มาทำการแบ่งข้อมูลไม่สมดุลออกเป็นสองกลุ่ม คือข้อมูลกลุ่มส่วนน้อย($D_{minorty}$) และข้อมูลกลุ่มส่วนมาก ($D_{majority}$) จากนั้นทำการสุ่มเลือกข้อมูลกลุ่มน้อยจำนวนหนึ่งเพียงครั้งเดียว (D_{ssMin}) และสุ่มเลือกข้อมูลกลุ่มส่วนมากทั้งหมด 10 ครั้งด้วยจำนวนข้อมูลที่แตกต่างกัน ($D_{ssMaj1}, D_{ssMaj2}, \dots, D_{ssMaj10}$) นำข้อมูลกลุ่มส่วนน้อยและข้อมูลกลุ่มส่วนมากมารวมกันจะได้ข้อมูลไม่สมดุลที่มีอัตราความไม่สมดุลทั้ง 10 ระดับ คือ 1:5 1:10 1:15 1:20 1:25 1:301:351:40 1:45 และ 1:50 ($D_{ss5}, D_{ss10}, \dots, D_{ss50}$) จากนั้นนำข้อมูลเหล่านี้มาสร้างโมเดลแบบเรียนรู้ร่วมกันร่วมกับเทคนิคการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลและเทคนิคการชดเชยการทำนายผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่ายพร้อมทั้งหาจำนวนต้นไม้ตัดสินใจที่เหมาะสม และทดสอบโมเดลด้วยการทดสอบไขว้ข้าม(M_{ij})ผลที่ได้คือโมเดลที่มีประสิทธิภาพ(R_{ij})

ขั้นตอนการออกแบบการทดลองของงานวิจัยของวิทยานิพนธ์ฉบับนี้แสดงดังรูปที่ 4.2



รูปที่ 4.2แสดงขั้นตอนการออกแบบการทดลองอัลกอริทึม EnsDTV

จากรูปที่ 4.2 โมเดลแบบเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันทั้งแบ็กกิงและบูสต์ติงและชดเชยการทำนายผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่ายนั้นจะมีโมเดลทั้งหมด 50 โมเดลด้วยกัน และผลที่ได้คือโมเดลที่มีประสิทธิภาพ ซึ่งแทนโมเดลด้วย M_{ij}และโมเดลที่มีประสิทธิภาพด้วย R_{ij} โดยที่

iหมายถึง อัลกอริทึมของการตัดสินใจร่วมกัน ซึ่ง

i =1 คือ AdaBoostM1

i =2 คือ Bag

i =3 คือ LogitBoost

i =4 คือ TotalBoost

i =5 คือ RUSBoost

jหมายถึง อัตราความไม่สมดุลซึ่ง

j =1 คือ 5

j =2คือ 10

j =3 คือ 15

j =4 คือ 20

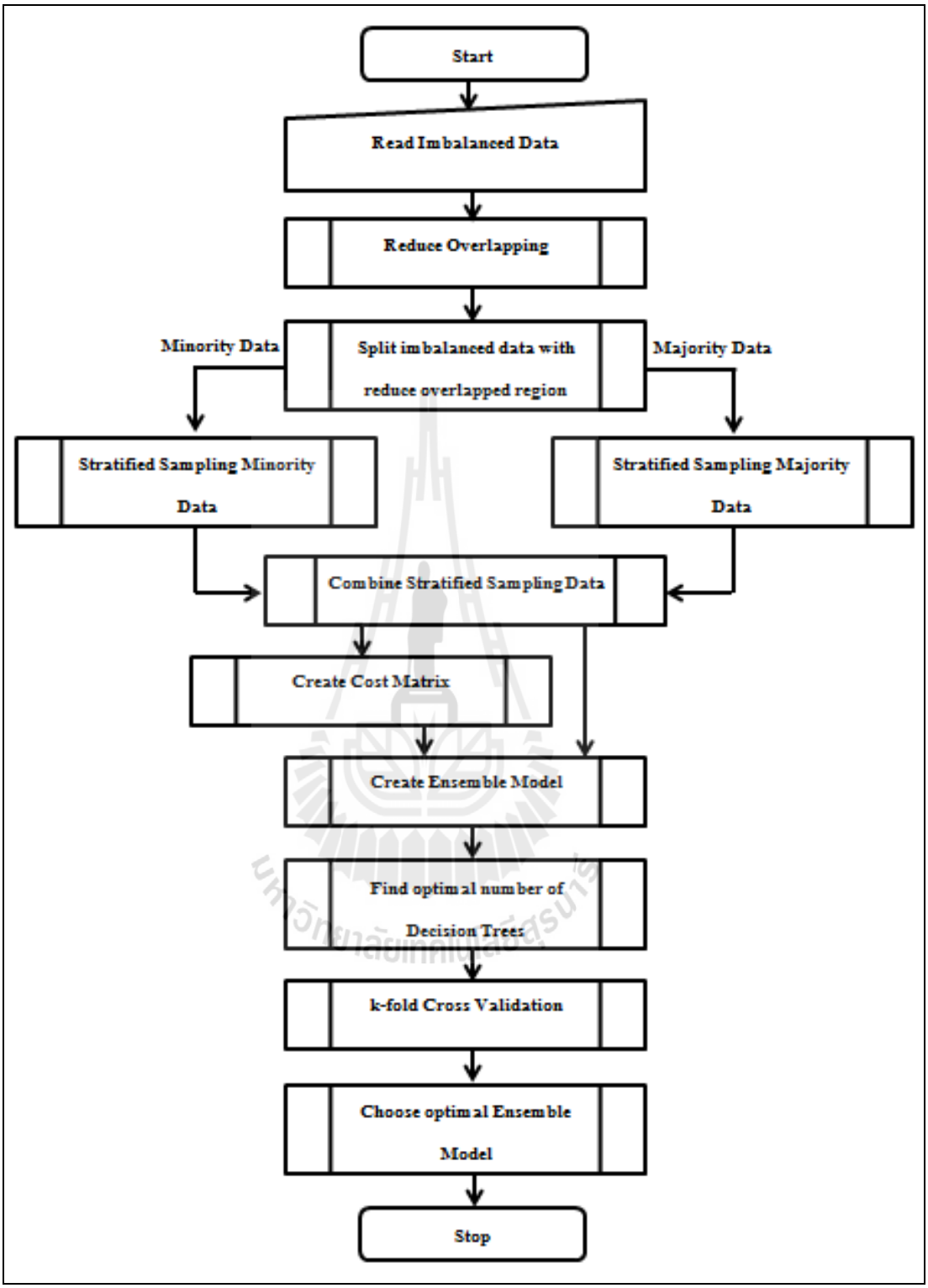
j =5 คือ 25

⋮

j =10 คือ 50

ดังนั้น M_{15}หมายถึง โมเดลแบบเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบ AdaBoostM1 ที่มีอัตราความไม่สมดุลของข้อมูลเท่ากับ 25 และผลที่ได้คือโมเดลที่มีประสิทธิภาพ R_{15}ซึ่งหมายถึง โมเดลที่มีประสิทธิภาพซึ่งทำงานด้วยอัลกอริทึม AdaBoostM1 ที่มีอัตราความไม่สมดุลของข้อมูลเท่ากับ 25

จากขั้นตอนการออกแบบการทดลองอัลกอริทึม EnsDTV สามารถแสดงเป็นผังงาน (Flow Chart) ได้ดังรูปที่ 4.3



รูปที่ 4.3ผังงานแสดงขั้นตอนการทำงานของอัลกอริทึม EnsDTV

4.3 ผลการทดสอบประสิทธิภาพ

จากหัวข้อ 4.2 ซึ่งเป็นการออกแบบการทดสอบนั้น ในงานวิจัยของวิทยานิพนธ์ฉบับนี้ได้นำชุดข้อมูลชนิดตัวเลขที่ได้จากการสังเคราะห์จากโปรแกรมและชุดข้อมูลจากแหล่งข้อมูลมาตรฐานมาทำการทดสอบ ซึ่งคุณลักษณะของชุดข้อมูลนั้นได้กล่าวไปแล้วข้างต้นในหัวข้อ 4.1

สำหรับข้อมูลไม่สมดุลนั้นจะเป็นข้อมูลที่ผ่านกระบวนการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล และจำนวนข้อมูลไม่สมดุลที่ใช้ในการทดสอบนั้นจะได้จากการสุ่มเลือกที่อัตราความไม่สมดุลที่แตกต่างกัน โดยที่จำนวนข้อมูลตัวอย่างของกลุ่มส่วนน้อยนั้นจะทำการสุ่มเลือกเพียงครั้งเดียวและใช้ข้อมูลนี้กับทุกโปรเซส

รายละเอียดการทดสอบประสิทธิภาพจะแบ่งออกเป็น 2 กรณี ดังต่อไปนี้

4.3.1 การทดสอบประสิทธิภาพชุดข้อมูลสังเคราะห์จากโปรแกรม

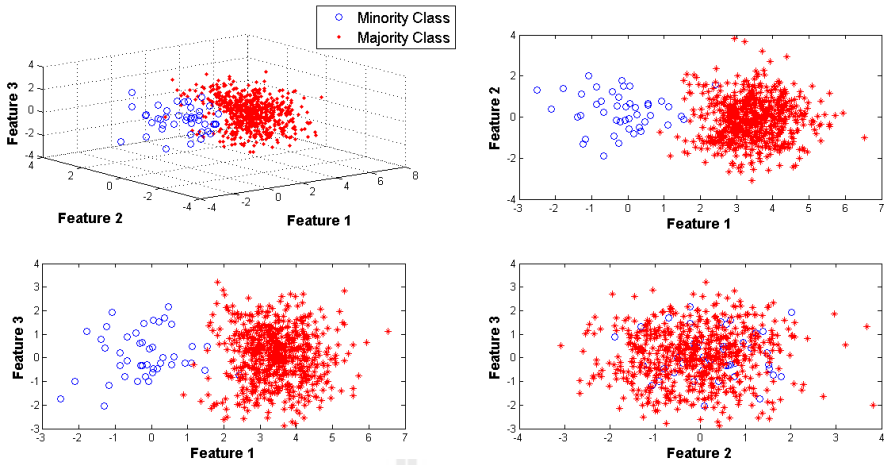
ผลการทดสอบประสิทธิภาพของการทำงานของอัลกอริทึม EnsDTV ด้วยชุดข้อมูลสังเคราะห์จากโปรแกรมซึ่งสร้างด้วยการกำหนดค่ากลางค่าความแปรปรวนของแต่ละกลุ่มและจำนวนข้อมูลตามรายละเอียดของตารางที่ 4.1 แล้วนั้นสามารถนำข้อมูลที่สังเคราะห์ขึ้นมานั้นมาใช้สำหรับทดสอบประสิทธิภาพของอัลกอริทึม EnsDTV และสามารถอธิบายผลการทดลองของแต่ละขั้นตอนได้ดังต่อไปนี้

4.3.1.1 ผลการทดลองของการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล

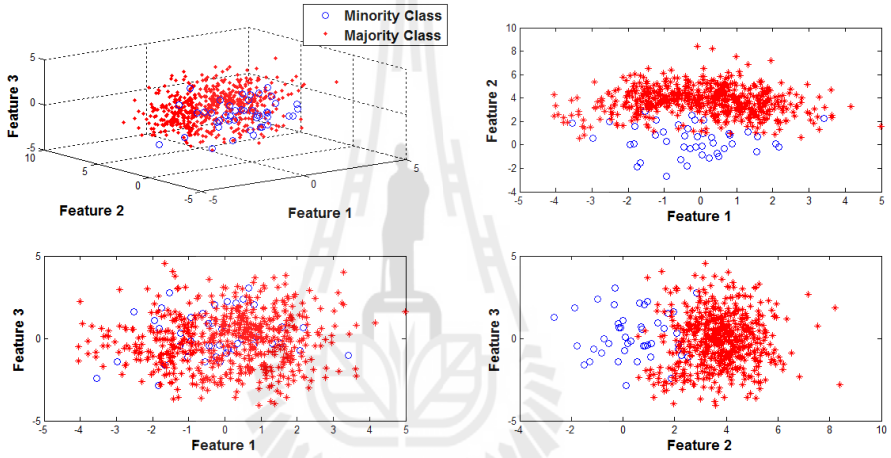
ผลการทดลองของการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลของชุดข้อมูลสังเคราะห์ ผลที่ได้แสดงดังตารางที่ 4.3และรูปที่ 4.4ตามลำดับ

ตารางที่ 4.3 แสดงรายละเอียดของชุดข้อมูลสังเคราะห์จากโปรแกรมที่ผ่านกระบวนการลดการซ้อนทับกันระหว่างข้อมูล

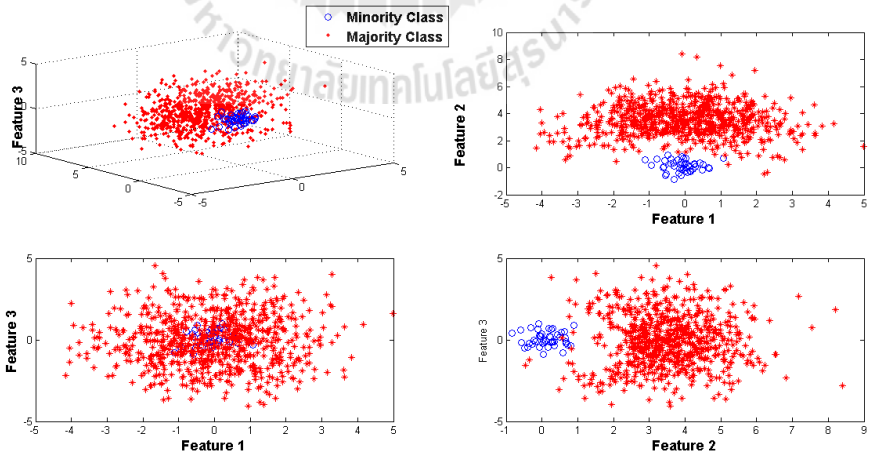
ชุดข้อมูล	ระยะห่างระหว่างสองกลุ่มข้อมูล	จำนวนตัวอย่าง			IR	maxF
		ทั้งหมด	กลุ่ม Majority	กลุ่ม Minority		
Data1	1.25	688	644	44	14.64	8.18
Data2	1.25	631	587	44	13.34	3.69
Data3	1.75	780	736	44	16.73	7.22
Data4	1.40	765	721	44	16.39	5.19



ก) ข้อมูลData1

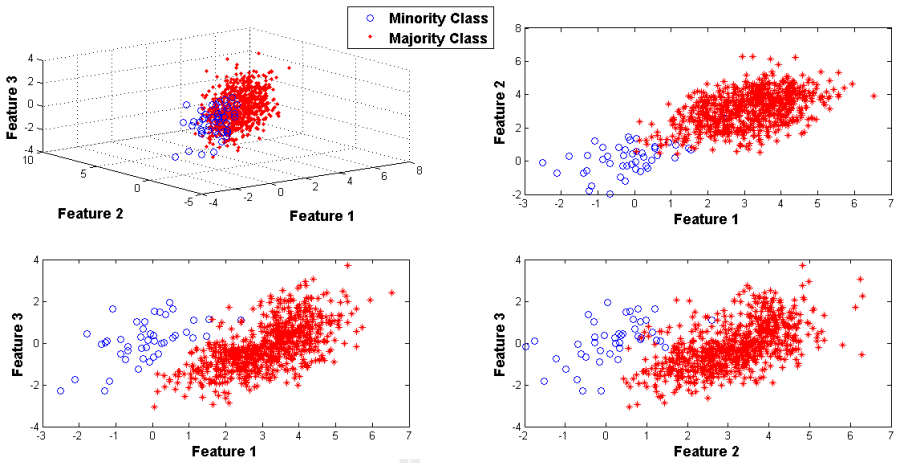


ข) ข้อมูล Data2



ค) ข้อมูลData3

รูปที่ 4.4 แสดงลักษณะการกระจายตัวของชุดข้อมูลสังเคราะห์ที่ผ่านกระบวนการลดการซ้อนทับกันระหว่างข้อมูลตั้งแต่ชุดข้อมูล Data1 ถึงชุดข้อมูล Data4



ง) ข้อมูลData4

รูปที่ 4.4 แสดงลักษณะการกระจายตัวของชุดข้อมูลสังเคราะห์ที่ผ่านกระบวนการลดการซ้อนทับกันระหว่างข้อมูลตั้งแต่ชุดข้อมูล Data1 ถึงชุดข้อมูล Data4(ต่อ)

ผลที่ได้จากตารางที่ 4.3 จะพบว่าอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล ซึ่งพิจารณาจากค่า maxF นั้นจะมีค่าเพิ่มขึ้นส่งผลให้ข้อมูลแต่ละชุดมีการใช้พื้นที่ร่วมกันน้อยลง

4.3.1.2 ผลการทดลองของการสุ่มเลือกข้อมูล

นำข้อมูลจากตารางที่ 4.3 ไปทำการสุ่มเลือกข้อมูลด้วยอัตราความไม่สมดุลที่แตกต่างกัน ผลการทดลองที่ได้แสดงดังตารางที่ 4.4

ตารางที่ 4.4 แสดงรายละเอียดของชุดข้อมูลสังเคราะห์จากโปรแกรมสำหรับการทดสอบที่มีIR ตั้งแต่ 1:5 ถึง 1:50

ชุดข้อมูล	จำนวนตัวอย่าง									
	กลุ่ม Minority	กลุ่ม Majorityตามอัตราความไม่สมดุล (IR)								
		5	10	15	20	25	30	35	40	45	50
Data1	20	100	200	300	400	500	600	700	800	900	1000
Data2	19	95	190	285	380	475	570	665	760	855	950
Data3	24	120	240	360	480	600	720	840	960	1080	1200
Data4	24	120	240	360	480	600	720	840	960	1080	1200

4.3.1.3 ผลการทดลองของการสร้างตารางค่าใช้จ่าย

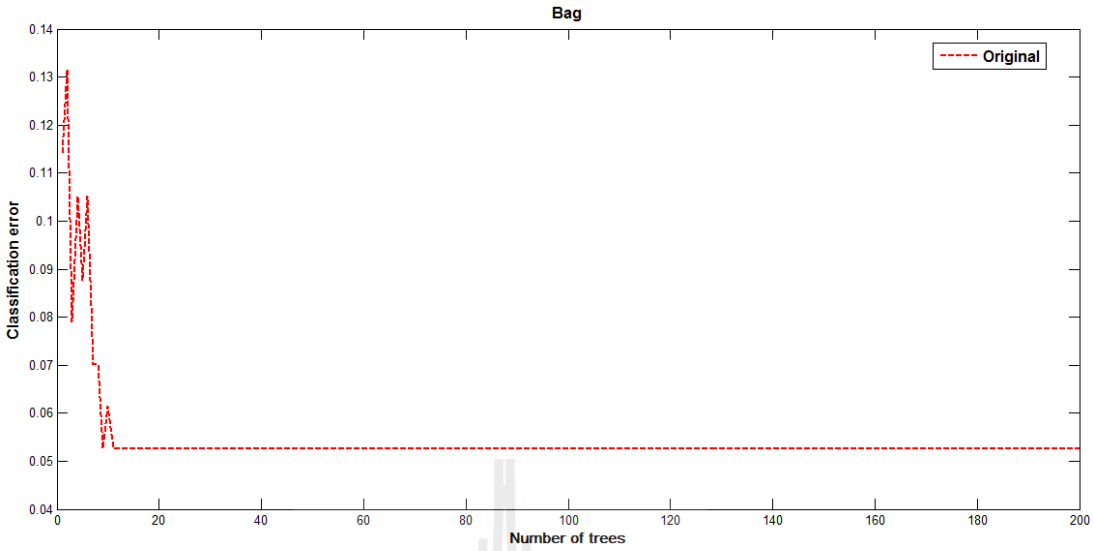
นำจำนวนข้อมูลจากตารางที่ 4.4 ไปทำการสร้างตารางค่าใช้จ่าย ผลการทดลองที่ได้แสดงดังตารางที่ 4.5

ตารางที่ 4.5 แสดงรายละเอียดของตารางค่าใช้จ่ายสำหรับทดลองอัลกอริทึม EnsDTV

	IR = 5	**IR = 10**	**IR = 15**	**IR = 20**	**IR = 25**
CostValue	$\begin{bmatrix} 0 & 1 \\ 5 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 10 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 15 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 20 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 25 & 0 \end{bmatrix}$
	IR = 30	**IR = 35**	**IR = 40**	**IR = 45**	**IR = 50**
CostValue	$\begin{bmatrix} 0 & 1 \\ 30 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 35 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 40 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 45 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 50 & 0 \end{bmatrix}$

4.3.1.4ผลการทดลองของการสร้างโมเดล

นำข้อมูลสังเคราะห์ที่สุ่มเลือกไว้แล้วตามอัตราความไม่สมดุลของตารางที่ 4.4 และค่า CostValue จากตารางที่ 4.5 มาทำการสร้างโมเดล ตัวอย่างเช่น นำชุดข้อมูล Data2มาทำการสร้างโมเดลที่มีอัตราความไม่สมดุล 1:5 ดังนั้นจำนวนข้อมูลตัวอย่างทั้งหมดที่นำมาใช้ในการสร้างโมเดล คือ 114 ตัวอย่าง แบ่งเป็นข้อมูลกลุ่มส่วนน้อย 19ตัวอย่าง และข้อมูลกลุ่มส่วนมาก 95 และใช้ค่า CostValue ที่มี IR=5 นำข้อมูลเหล่านี้มาทำการเรียนรู้ด้วยอัลกอริทึม Bag การสร้างโมเดลครั้งแรกนั้นจะกำหนดจำนวนต้นไม้ตัดสินใจที่ 200 ต้นไม้ตัดสินใจ ซึ่งการเรียนรู้ด้วยจำนวนต้นไม้ตัดสินใจที่ไม่เหมาะสมนั้นจะส่งผลให้ใช้เวลาในการเรียนรู้ที่มาก ดังนั้นเพื่อลดเวลาที่ใช้ในการเรียนรู้และหาจำนวนต้นไม้ที่เหมาะสมกับข้อมูลชุดนี้ งานวิจัยนี้จึงได้ทำการมโนภาพเพื่อแสดงให้เห็นถึงความผิดพลาดของการจำแนกประเภทข้อมูลทดสอบ ณ ตำแหน่งของจำนวนต้นไม้ตัดสินใจ ผลการทดลองที่ได้แสดงดังรูปที่ 4.5

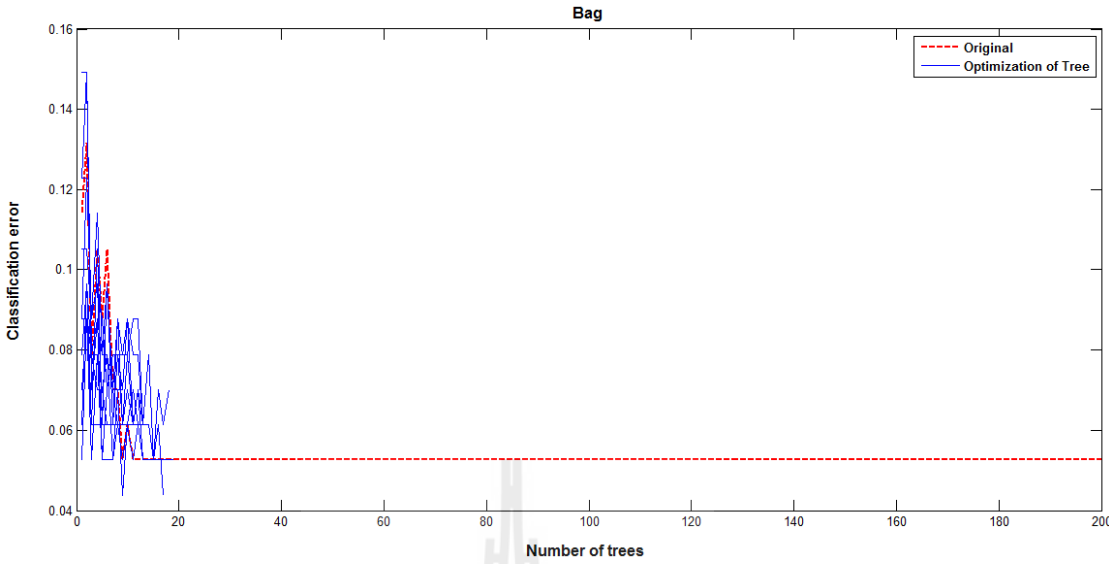


รูปที่ 4.5แสดงความผิดพลาดของการจำแนกประเภทข้อมูลด้วยต้นไม้ตัดสินใจตั้งแต่จำนวน 1 ต้น ถึง 200 ต้น ด้วยชุดข้อมูล Data2 ที่อัตราความไม่สมดุล 1:5

4.3.1.5 ผลการทดลองของการเลือกจำนวนต้นไม้ตัดสินใจที่เหมาะสม

จากรูปที่ 4.5จะทำการเลือกจำนวนต้นไม้ตัดสินใจที่มีค่าการจำแนกผิดกลุ่มต่ำสุด 10 อันดับแรกมาทำการสร้างโมเดลการเรียนรู้อีกครั้ง ซึ่งการเรียนรู้ด้วยจำนวนต้นไม้ตัดสินใจที่มีค่าการจำแนกผิดกลุ่มต่ำสุดนี้จะส่งผลให้ได้จำนวนต้นไม้ตัดสินใจที่เหมาะสมกับชุดข้อมูลนั้น ๆ ซึ่งข้อมูลจำนวนต้นไม้ตัดสินใจที่มีค่าการจำแนกผิดกลุ่มที่ต่ำสุด 10 อันดับแรก คือ 9, 11, 12, 13, 14, 15, 16, 17, 18 และ 19 ซึ่งมีค่าความผิดพลาดของการจำแนกประเภทอยู่ที่ 0.053

ผลการทดลองที่ได้จากการนำจำนวนต้นไม้ตัดสินใจที่เหมาะสมไปสร้างโมเดลเรียนรู้ร่วมกันด้วยอัลกอริทึม Bag สามารถแสดงผลการสร้างมโนภาพด้วยรูปที่ 4.6



รูปที่ 4.6แสดงความผิดพลาดของการจำแนกประเภทข้อมูลด้วยต้นไม้ตัดสินใจที่เหมาะสมของชุดข้อมูล Data2 ที่อัตราความไม่สมดุล 1:5

สำหรับชุดข้อมูล Data2 ที่อัตราความไม่สมดุล 1:5 นั้นเมื่อนำไปเรียนรู้ร่วมกันด้วยอัลกอริทึม Bag ด้วยจำนวนต้นไม้ตัดสินใจที่เหมาะสมแล้ว นำโมเดลที่ได้ไปทำการวัดประสิทธิภาพด้วยมาตรวัดต่าง ๆ ผลการทดลองที่ได้แสดงดังตารางที่ 4.6

ตารางที่ 4.6 แสดงประสิทธิภาพของโมเดลการเรียนรู้ร่วมกันแบบ Bag ด้วยมาตรวัดต่าง ๆ ของชุดข้อมูล Data2 ที่อัตราความไม่สมดุล 1:5เรียงลำดับตามการรันโปรแกรม

ลำดับการรันโปรแกรม	จำนวนต้นไม้ตัดสินใจ	มาตรวัดประสิทธิภาพ							
		Accuracy	Precision	Sensitivity	Specificity	F-measure	G-mean	AUC	Total Misclassification Costs
1	9	0.930	1.000	0.579	1.000	0.733	0.761	0.789	0.421
2	11	0.939	1.000	0.632	1.000	0.774	0.795	0.816	0.368
3	12	0.939	1.000	0.632	1.000	0.774	0.795	0.816	0.368
4	13	0.939	1.000	0.632	1.000	0.774	0.795	0.816	0.368
5	14	0.939	1.000	0.632	1.000	0.774	0.795	0.816	0.368

ตารางที่ 4.6 แสดงประสิทธิภาพของโมเดลการเรียนรู้ร่วมกันแบบ Bag ด้วยมาตรวัดต่าง ๆ ของชุดข้อมูล Data2 ที่อัตราความไม่สมดุล 1:5 เรียงลำดับตามการรันโปรแกรม(ต่อ)

ลำดับการรันโปรแกรม	จำนวนต้นไม้ตัดสินใจ	มาตรวัดประสิทธิภาพ							
		Accuracy	Precision	Sensitivity	Specificity	F-measure	G-mean	AUC	Total Misclassification Costs
6	15	0.947	1.000	0.684	1.000	0.813	0.827	0.842	0.316
7	16	0.947	1.000	0.684	1.000	0.813	0.827	0.842	0.316
8	17	0.956	1.000	0.737	1.000	0.848	0.858	0.868	0.263
9	18	0.930	0.923	0.632	0.989	0.750	0.791	0.811	0.379
10	19	0.947	1.000	0.684	1.000	0.813	0.827	0.842	0.316

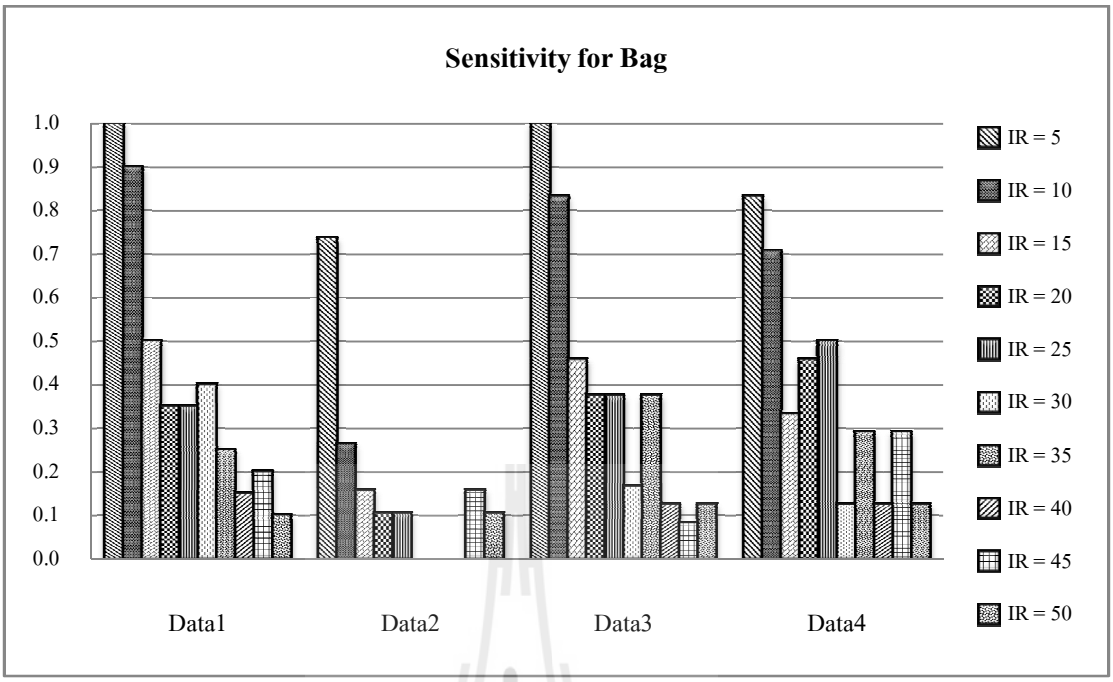
4.3.1.6 ผลการทดลองของการเลือกโมเดลที่มีประสิทธิภาพ

ขั้นตอนสุดท้ายของการทดลอง คือ การเลือกโมเดลการเรียนรู้ร่วมกันแบบ Bag ที่มีประสิทธิภาพสำหรับชุดข้อมูล Data2 ที่อัตราความไม่สมดุล 1:5 สำหรับขั้นตอนนี้จะนำข้อมูลจากตารางที่ 4.6 มาทำการเรียงลำดับตามค่า Total Misclassification Cost และจำนวนต้นไม้ตัดสินใจจากน้อยไปมาก แล้วทำการพิจารณาเลือกโมเดลที่มีค่า Total Misclassification Cost ที่ต่ำที่สุด ซึ่งโมเดลที่ได้จะเป็นโมเดลที่มีประสิทธิภาพเนื่องจากเป็นโมเดลที่มีประสิทธิภาพในการจำแนกประเภทข้อมูลที่สูงและใช้เวลาในการเรียนรู้ที่น้อยที่สุดด้วยจำนวนต้นไม้ตัดสินใจที่เหมาะสม ซึ่งโมเดลที่เหมาะสม คือ โมเดลที่ใช้จำนวนต้นไม้ตัดสินใจทั้งหมด 17 ต้น โดยมีประสิทธิภาพในการจำแนกประเภทแสดงดังตารางที่ 4.7

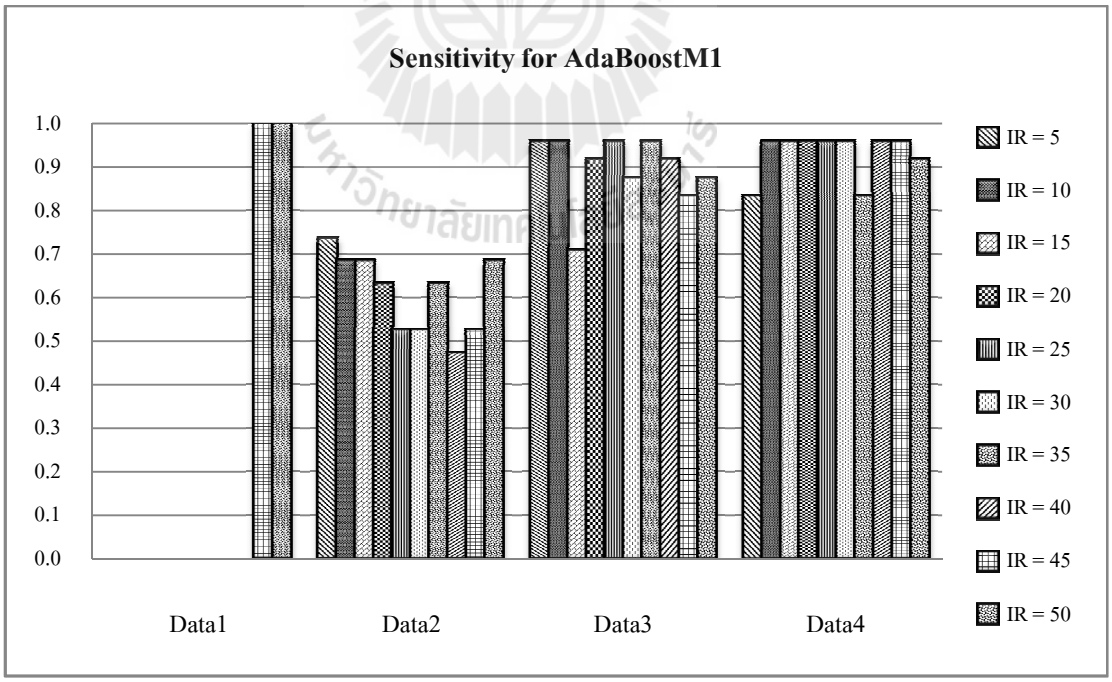
ตารางที่ 4.7 แสดงประสิทธิภาพของโมเดลการเรียนรู้ร่วมกันแบบ Bag ด้วยมาตรวัดต่าง ๆ ของชุดข้อมูล Data2 ที่อัตราความไม่สมดุล 1:5 เรียงลำดับตามค่าความผิดพลาดและจำนวนต้นไม้ตัดสินใจจากน้อยไปมาก

จำนวนการเรียนรู้ร่วมกัน	จำนวนต้นไม้ตัดสินใจ	มาตรวัดประสิทธิภาพ							
		Accuracy	Precision	Sensitivity	Specificity	F-measure	G-mean	AUC	Total Misclassification Costs
8	***17***	***0.956***	***1.000***	***0.737***	***1.000***	***0.848***	***0.858***	***0.868***	***0.263***
6	15	0.947	1.000	0.684	1.000	0.813	0.827	0.842	0.316
7	16	0.947	1.000	0.684	1.000	0.813	0.827	0.842	0.316
10	19	0.947	1.000	0.684	1.000	0.813	0.827	0.842	0.316
2	11	0.939	1.000	0.632	1.000	0.774	0.795	0.816	0.368
3	12	0.939	1.000	0.632	1.000	0.774	0.795	0.816	0.368
4	13	0.939	1.000	0.632	1.000	0.774	0.795	0.816	0.368
5	14	0.939	1.000	0.632	1.000	0.774	0.795	0.816	0.368
9	18	0.930	0.923	0.632	0.989	0.750	0.791	0.811	0.379
1	9	0.930	1.000	0.579	1.000	0.733	0.761	0.789	0.421

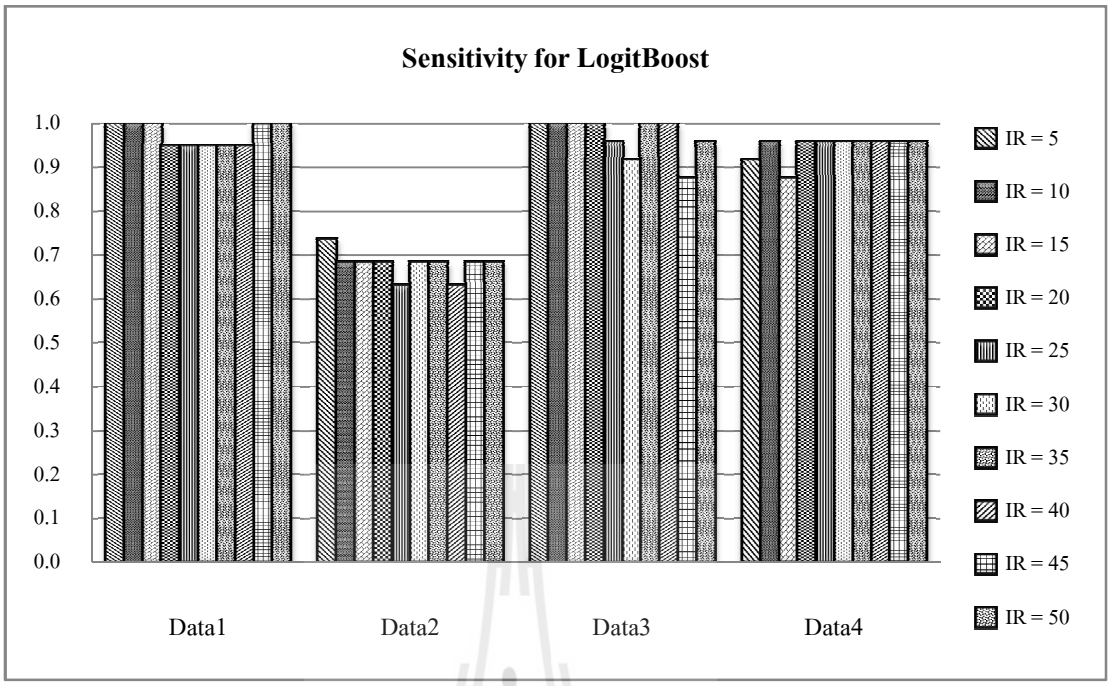
สำหรับผลการทดลองของข้อมูลสังเคราะห์อื่น ๆ ที่ยังไม่ได้กล่าวถึงนั้นก็จะทำการทดลองเช่นเดียวกับชุดข้อมูล Data2 ที่อัตราความไม่สมดุล 1:5 ผลการทดลองโดยรายละเอียดนั้นจะไม่แสดงรายละเอียด แต่จะแสดงเฉพาะผลการทดสอบประสิทธิภาพของการจำแนกประเภทข้อมูลสำหรับข้อมูลกลุ่มน้อยของแต่ละโมเดลที่อัตราความไม่สมดุลแตกต่างกันด้วยการสร้างเป็นกราฟเปรียบเทียบประสิทธิภาพ ซึ่งแสดงได้ดังรูปที่ 4.7 ถึง 4.11 ดังต่อไปนี้



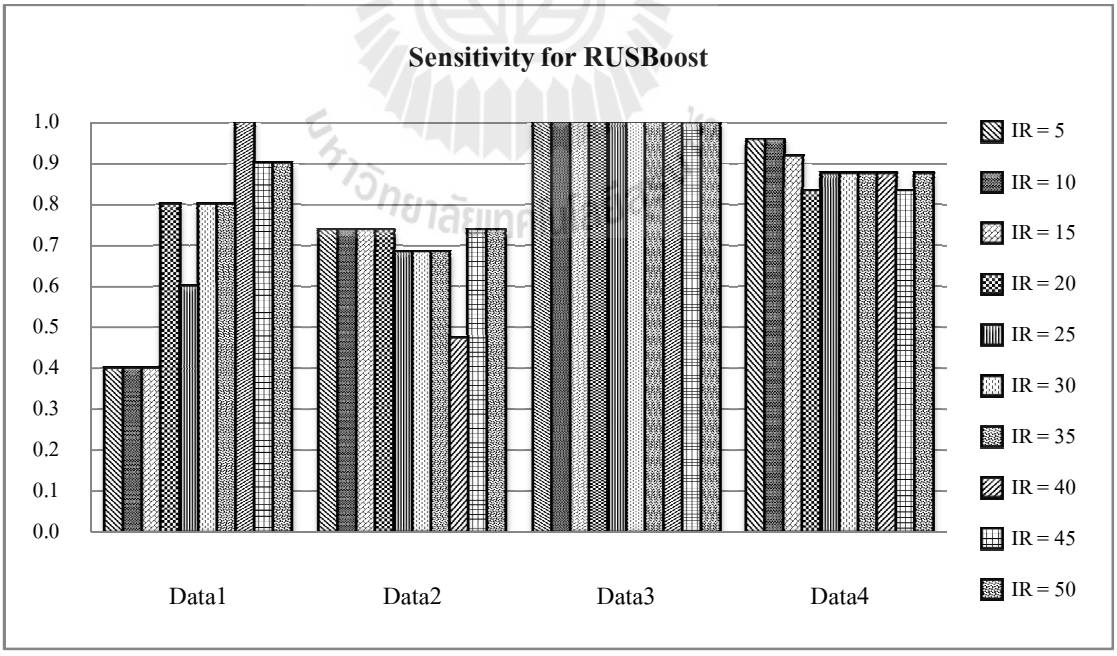
รูปที่ 4.7แสดงประสิทธิภาพของการจำแนกประเภทข้อมูลกลุ่มน้อยของชุดข้อมูลสังเคราะห์ด้วยโมเดลการเรียนรู้ร่วมกันแบบใช้การตัดสินใจร่วมกันแบบ Bag



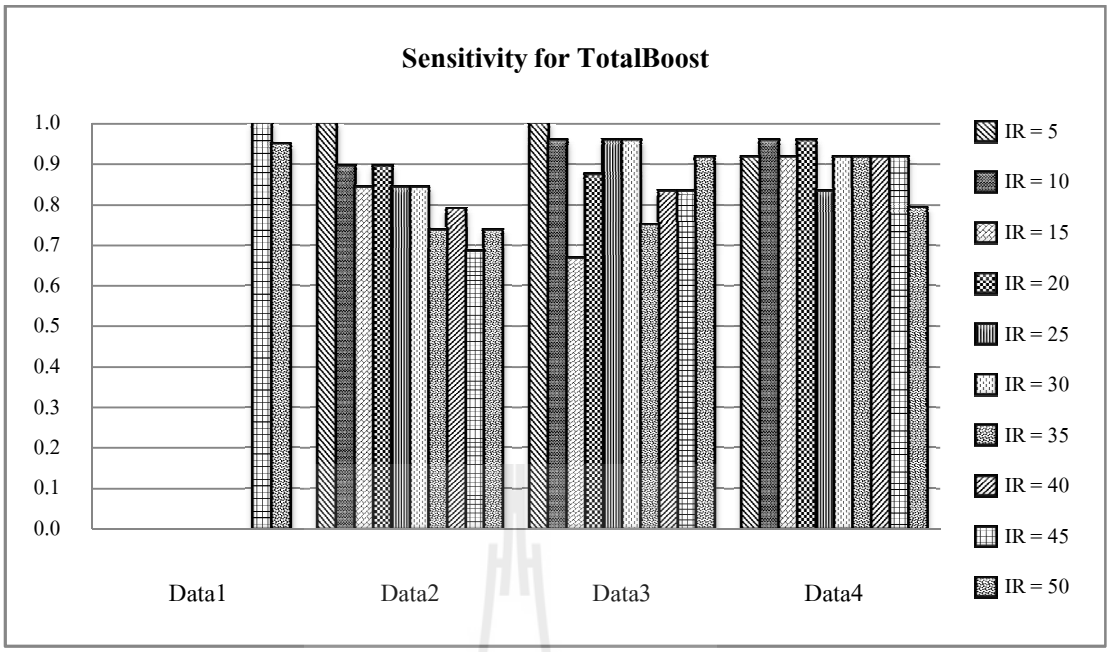
รูปที่ 4.8แสดงประสิทธิภาพของการจำแนกประเภทข้อมูลกลุ่มน้อยของชุดข้อมูลสังเคราะห์ด้วยโมเดลการเรียนรู้ร่วมกันแบบใช้การตัดสินใจร่วมกันแบบ AdaBoostM1



รูปที่ 4.9แสดงประสิทธิภาพของการจำแนกประเภทข้อมูลกลุ่มน้อยของชุดข้อมูลสังเคราะห์ด้วยโมเดลการเรียนรู้ร่วมกันแบบใช้การตัดสินใจร่วมกันแบบ LogitBoost



รูปที่ 4.10แสดงประสิทธิภาพของการจำแนกประเภทข้อมูลกลุ่มน้อยของชุดข้อมูลสังเคราะห์ด้วยโมเดลการเรียนรู้ร่วมกันแบบใช้การตัดสินใจร่วมกันแบบ RUSBoost



รูปที่ 4.11แสดงประสิทธิภาพของการจำแนกประเภทข้อมูลกลุ่มน้อยของชุดข้อมูลสังเคราะห์ด้วยโมเดลการเรียนรู้ร่วมกันแบบใช้การตัดสินใจร่วมกันแบบ TotalBoost

4.3.2 การทดสอบประสิทธิภาพชุดข้อมูลจากแหล่งข้อมูลมาตรฐาน

ผลการทดสอบประสิทธิภาพของการทำงานของอัลกอริทึม EnsDTV ด้วยชุดข้อมูลจากแหล่งข้อมูลมาตรฐานซึ่งแสดงข้อมูลตามรายละเอียดของตารางที่ 4.2 แล้วนั้น สามารถอธิบายผลการทดลองของแต่ละขั้นตอนได้ดังต่อไปนี้

4.3.2.1 ผลการทดลองของการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล

ผลการทดลองของการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลของชุดข้อมูลจากแหล่งข้อมูลมาตรฐาน ผลที่ได้แสดงดังตารางที่ 4.8

ตารางที่ 4.8 แสดงรายละเอียดของชุดข้อมูลจากแหล่งข้อมูลมาตรฐานที่ผ่านกระบวนการลดการซ้อนทับกันระหว่างข้อมูล

ชุดข้อมูล	ระยะห่างระหว่างกลุ่มข้อมูล	จำนวนตัวอย่าง			IR	maxF
		ทั้งหมด	กลุ่ม Majority	กลุ่ม Minority		
ecoli4	0.40	220	200	20	10.00	3.02
glass-6	0.55	204	175	29	6.03	3.38
new-thyroid1	4.50	182	147	35	4.20	3.96
page-blocks	13.00	5098	4539	559	8.12	0.54
pima	12.00	654	386	268	1.44	0.62
segment	30.00	1819	1490	329	4.53	2.57
shuttle	75.00	1437	1314	123	10.68	13.89
vehicle	20.00	799	581	218	2.67	0.39
yeast	0.10	1261	1098	163	6.74	3.15

ผลที่ได้จากตารางที่ 4.8 จะพบว่าอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูล ซึ่งพิจารณาจากค่า maxF นั้นจะมีค่าเพิ่มขึ้นส่งผลให้ข้อมูลแต่ละชุดมีการใช้พื้นที่ร่วมกันน้อยลง

4.3.2.2 ผลการทดลองของการสุ่มเลือกข้อมูล

นำข้อมูลจากตารางที่ 4.8 ไปทำการสุ่มเลือกข้อมูลด้วยอัตราความไม่สมดุลที่แตกต่างกัน ผลการทดลองที่ได้แสดงดังตารางที่ 4.9 และตารางที่ 4.10

ตารางที่ 4.9 แสดงรายละเอียดของชุดข้อมูลจากแหล่งข้อมูลมาตรฐานสำหรับการทดสอบที่มี IR ตั้งแต่ 1:5 ถึง 1:25

ชุดข้อมูล	จำนวนตัวอย่างกลุ่ม Minority	จำนวนตัวอย่างกลุ่ม Majorityตามอัตราความไม่สมดุล (IR)				
		5	10	15	20	25
ecoli4	5	25	50	75	100	120
glass-6	6	25	60	90	120	149
new-thyroid1	5	25	50	75	100	125
page-blocks	35	175	350	525	700	875
pima	15	75	150	225	300	375

ตารางที่ 4.9 แสดงรายละเอียดของชุดข้อมูลจากแหล่งข้อมูลมาตรฐานสำหรับการทดสอบที่มี IR ตั้งแต่ 1:5 ถึง 1:25 (ต่อ)

ชุดข้อมูล	จำนวนตัวอย่างกลุ่ม Minority	จำนวนตัวอย่างกลุ่ม Majorityตามอัตราความไม่สมดุล (IR)				
		5	10	15	20	25
segment	25	125	250	375	500	625
shuttle	40	180	370	600	800	1000
vehicle	20	100	200	300	400	500
yeast	30	145	285	450	595	750

ตารางที่ 4.10 แสดงรายละเอียดของชุดข้อมูลจากแหล่งข้อมูลมาตรฐานสำหรับการทดสอบที่มี IRตั้งแต่ 1:30 ถึง 1:50

ชุดข้อมูล	จำนวนตัวอย่างกลุ่ม Minority	จำนวนตัวอย่างกลุ่ม Majorityตามอัตราความไม่สมดุล (IR)				
		30	35	40	45	50
ecoli4	5	150	175	200	225	250
glass-6	6	175	210	240	270	300
new-thyroid1	5	146	175	200	225	250
page-blocks	35	1050	1225	1400	1575	1750
pima	15	443	525	600	675	750
segment	25	750	875	1000	1125	1250
shuttle	40	1200	1400	1600	1800	2000
vehicle	20	581	700	800	900	1000
yeast	30	900	1050	1200	1350	1500

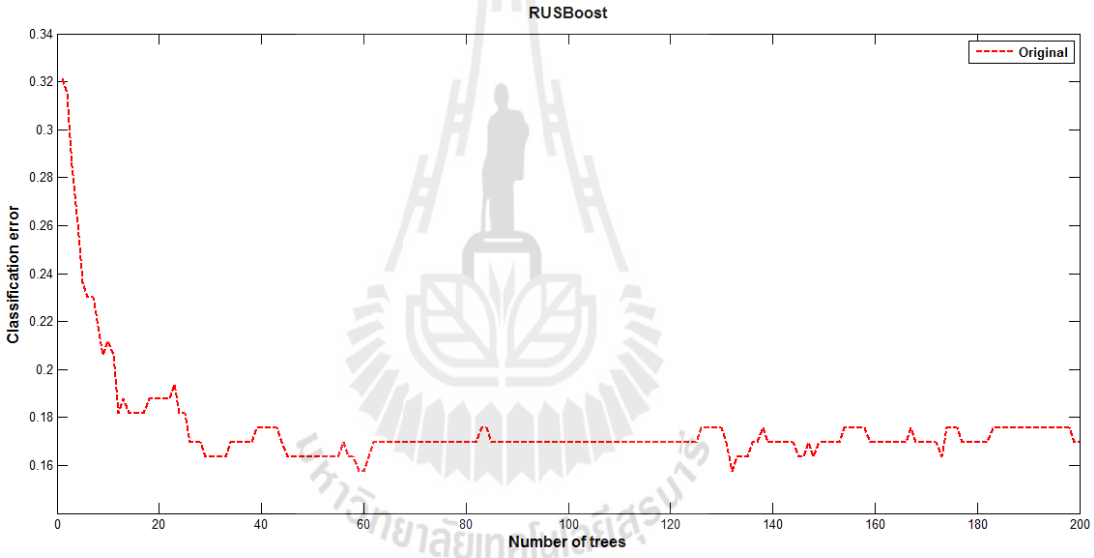
4.3.2.3 ผลการทดลองของการสร้างตารางค่าใช้จ่าย

นำจำนวนข้อมูลจากตารางที่ 4.9 และตารางที่ 4.10 ไปทำการสร้างตารางค่าใช้จ่าย ผลการทดลองที่ได้จะแสดงเช่นเดียวกับตารางที่ 4.5

4.3.2.4 ผลการทดลองของการสร้างโมเดล

นำชุดข้อมูลจากแหล่งข้อมูลมาตรฐานที่สุ่มเลือกไว้แล้วตามอัตราความไม่สมดุลของตารางที่ 4.9 และตารางที่ 4.10 และค่า CostValue จากตารางที่ 4.5 มาทำการสร้างโมเดล

ตัวอย่างเช่น นำชุดข้อมูลpimaมาทำการสร้างโมเดลที่มีอัตราความไม่สมดุล 1:10ดังนั้นจำนวนข้อมูลตัวอย่างทั้งหมดที่นำมาใช้ในการสร้างโมเดล คือ 165 ตัวอย่าง แบ่งเป็นข้อมูลกลุ่มส่วนน้อย 15ตัวอย่าง และข้อมูลกลุ่มส่วนมาก 150 และใช้ค่า CostValue ที่มี IR=10นำข้อมูลเหล่านี้มาทำการเรียนรู้ด้วยอัลกอริทึม RUSBoost การสร้างโมเดลครั้งแรกนั้นจะกำหนดจำนวนต้นไม้ตัดสินใจที่ 200 ต้นไม้ตัดสินใจ ซึ่งการเรียนรู้ด้วยจำนวนต้นไม้ตัดสินใจที่ไม่เหมาะสมนั้นจะส่งผลให้ใช้เวลาในการเรียนรู้ที่มาก ดังนั้นเพื่อลดเวลาที่ใช้ในการเรียนรู้และหาจำนวนต้นไม้ที่เหมาะสมกับข้อมูลชุดนี้ งานวิจัยนี้จึงได้ทำการมโนภาพเพื่อแสดงให้เห็นถึงความผิดพลาดของการจำแนกประเภทข้อมูลทดสอบ ณ ตำแหน่งของจำนวนต้นไม้ตัดสินใจ ผลการทดลองที่ได้แสดงดังรูปที่ 4.12

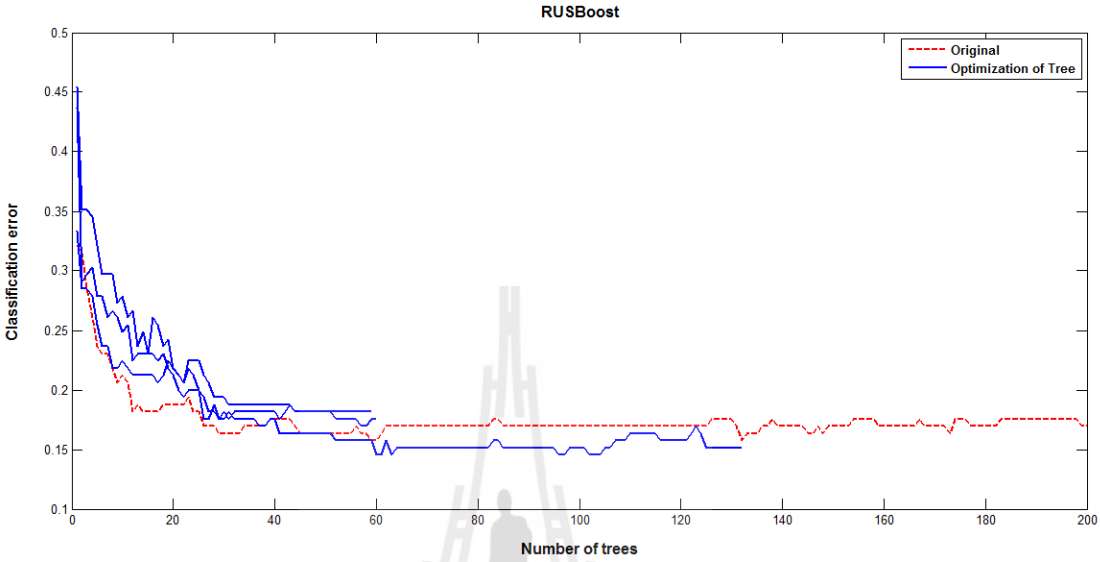


รูปที่ 4.12แสดงความผิดพลาดของการจำแนกประเภทข้อมูลด้วยต้นไม้ตัดสินใจตั้งแต่จำนวน 1 ต้นถึง 200 ต้น ด้วยชุดข้อมูลpimaที่อัตราความไม่สมดุล 1:10

4.3.2.5 ผลการทดลองของการเลือกจำนวนต้นไม้ตัดสินใจที่เหมาะสม

จากรูปที่ 4.12จะทำการเลือกจำนวนต้นไม้ตัดสินใจที่มีค่าการจำแนกผิดกลุ่มต่ำสุด 10 อันดับแรกมาทำการสร้างโมเดลการเรียนรู้อีกครั้ง ซึ่งการเรียนรู้ด้วยจำนวนต้นไม้ตัดสินใจที่มีค่าการจำแนกผิดกลุ่มต่ำสุดนี้จะส่งผลให้ได้จำนวนต้นไม้ตัดสินใจที่เหมาะสมกับชุดข้อมูลนั้น ๆ ซึ่งข้อมูลจำนวนต้นไม้ตัดสินใจที่มีค่าการจำแนกผิดกลุ่มที่ต่ำสุดมีเพียงสามอันดับเท่านั้น คือ 59, 60และ 132ซึ่งมีค่าความผิดพลาดของการจำแนกประเภทอยู่ที่ 0.158

ผลการทดลองที่ได้จากการนำจำนวนต้นไม้ตัดสินใจที่เหมาะสมไปสร้างโมเดลเรียนรู้ร่วมกันด้วยอัลกอริทึม RUSBoost สามารถแสดงผลการสร้างมโนภาพด้วยรูปที่ 4.13



รูปที่ 4.13แสดงความผิดพลาดของการจำแนกประเภทข้อมูลด้วยต้นไม้ตัดสินใจที่เหมาะสมของชุดข้อมูลpimaที่อัตราความไม่สมดุล 1:10

สำหรับชุดข้อมูล pimaที่อัตราความไม่สมดุล 1:10 นั้นเมื่อนำไปเรียนรู้ร่วมกันด้วยอัลกอริทึม RUSBoost ด้วยจำนวนต้นไม้ตัดสินใจที่เหมาะสมแล้ว นำโมเดลที่ได้ไปทำการวัดประสิทธิภาพด้วยมาตรวัดต่างๆ ผลการทดลองที่ได้แสดงดังตารางที่ 4.11

ตารางที่ 4.11 แสดงประสิทธิภาพของโมเดลการเรียนรู้ร่วมกันแบบ RUSBoost ด้วยมาตรวัดต่าง ๆ ของชุดข้อมูลpimaที่อัตราความไม่สมดุล 1:10เรียงลำดับตามการรันโปรแกรม

ลำดับการรันโปรแกรม	จำนวนต้นไม้ตัดสินใจ	มาตรวัดประสิทธิภาพ							
		Accuracy	Precision	Sensitivity	Specificity	F-measure	G-mean	AUC	Total Misclassification Costs
1	59	0.818	0.308	0.800	0.820	0.444	0.810	0.810	0.380
2	60	0.824	0.325	0.867	0.820	0.473	0.843	0.843	0.313
3	132	0.848	0.361	0.867	0.847	0.510	0.857	0.857	0.287

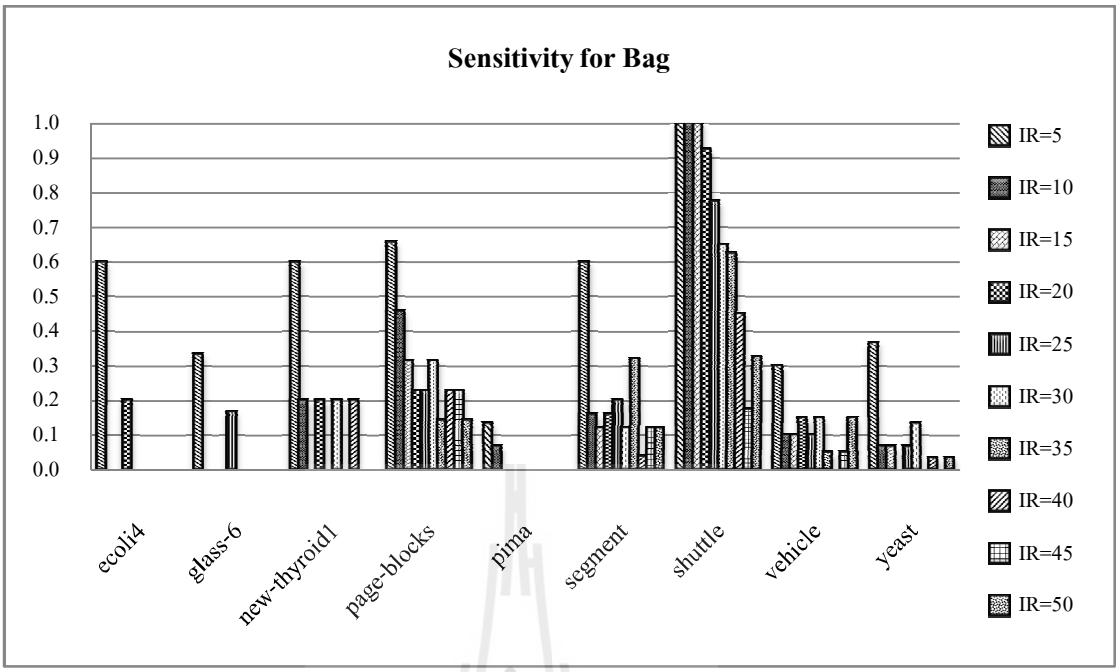
4.3.2.6 ผลการทดลองของการเลือกโมเดลที่มีประสิทธิภาพ

ขั้นตอนสุดท้ายของการทดลอง คือ การเลือกโมเดลการเรียนรู้ร่วมกันแบบ RUSBoost ที่มีประสิทธิภาพสำหรับชุดข้อมูล pima ที่อัตราความไม่สมดุล 1:10 สำหรับขั้นตอนนี้จะนำข้อมูลจากตารางที่ 4.11 มาทำการเรียงลำดับตามค่า Total Misclassification Cost และจำนวนต้นไม้ตัดสินใจจากน้อยไปมาก แล้วทำการพิจารณาเลือกโมเดลที่มีค่า Total Misclassification Cost ที่ต่ำที่สุด ซึ่งโมเดลที่ได้จะเป็นโมเดลที่มีประสิทธิภาพเนื่องจากเป็นโมเดลที่มีประสิทธิภาพในการจำแนกประเภทข้อมูลที่สูงและใช้เวลาในการเรียนรู้ที่น้อยที่สุดด้วยจำนวนต้นไม้ตัดสินใจที่เหมาะสม ซึ่งโมเดลที่เหมาะสม คือ โมเดลที่ใช้จำนวนต้นไม้ตัดสินใจทั้งหมด 132ต้น โดยมีประสิทธิภาพในการจำแนกประเภทแสดงดังตารางที่ 4.12

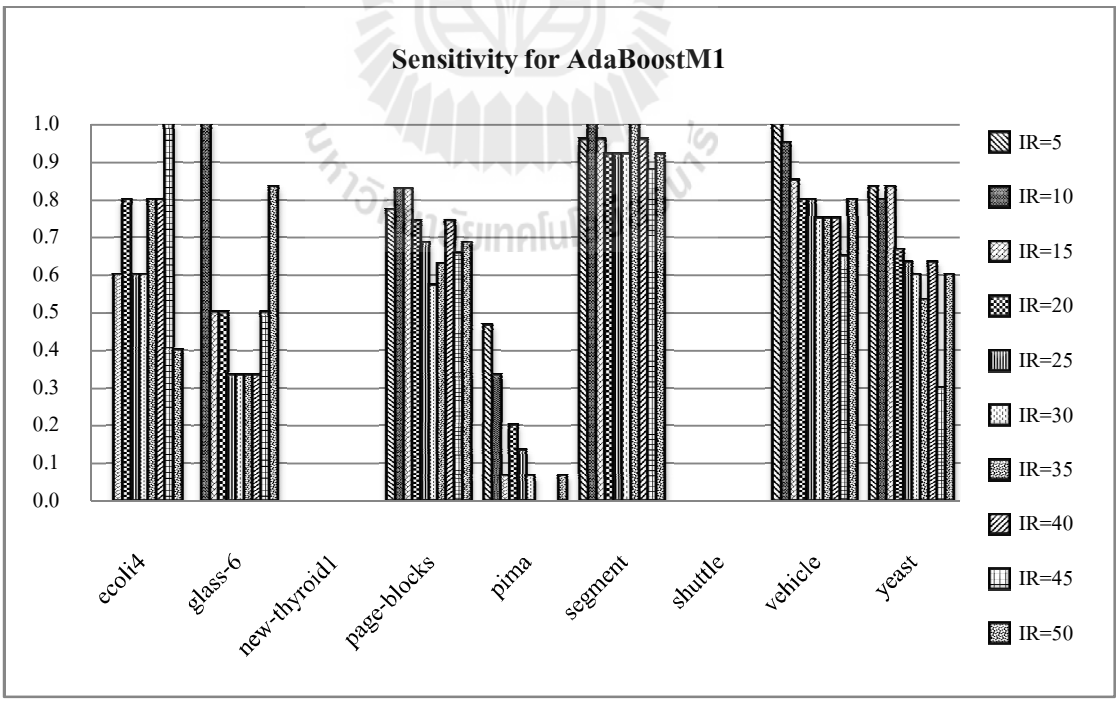
ตารางที่ 4.12 แสดงประสิทธิภาพของโมเดลการเรียนรู้ร่วมกันแบบ RUSBoost ด้วยมาตรวัดต่าง ๆ ของชุดข้อมูลpimaที่อัตราความไม่สมดุล 1:10เรียงลำดับตามค่าความผิดพลาดและจำนวนต้นไม้ตัดสินใจจากน้อยไปมาก

จำนวนการเรียนรู้ร่วมกัน	จำนวนต้นไม้ตัดสินใจ	มาตรวัดประสิทธิภาพ							
		Accuracy	Precision	Sensitivity	Specificity	F-measure	G-mean	AUC	Total Misclassification Costs
3	***132***	***0.848***	***0.361***	***0.867***	***0.847***	***0.510***	***0.857***	***0.857***	***0.287***
2	60	0.824	0.325	0.867	0.820	0.473	0.843	0.843	0.313
1	59	0.818	0.308	0.800	0.820	0.444	0.810	0.810	0.380

สำหรับผลการทดลองของข้อมูลจากแหล่งข้อมูลมาตรฐานอื่น ๆ ที่ยังไม่ได้กล่าวถึงนั้นก็จะทำการทดลองเช่นเดียวกับชุดข้อมูลpimaที่อัตราความไม่สมดุล 1:10ผลการทดลองโดยรายละเอียดนั้นจะไม่แสดงรายละเอียด แต่จะแสดงเฉพาะผลการทดสอบประสิทธิภาพของการจำแนกประเภทข้อมูลสำหรับข้อมูลกลุ่มน้อยของแต่ละโมเดลที่อัตราความไม่สมดุลแตกต่างกันด้วยการสร้างเป็นกราฟเปรียบเทียบประสิทธิภาพ ซึ่งแสดงได้ดังรูปที่ 4.14 ถึง 4.18 ดังต่อไปนี้



รูปที่ 4.14แสดงประสิทธิภาพของการจำแนกประเภทข้อมูลกลุ่มน้อยของชุดข้อมูลจากแหล่งข้อมูลมาตรฐานด้วยโมเดลการเรียนรู้ร่วมกันแบบใช้การตัดสินใจร่วมกันแบบ Bag



รูปที่ 4.15แสดงประสิทธิภาพของการจำแนกประเภทข้อมูลกลุ่มน้อยของชุดข้อมูลจากแหล่งข้อมูลมาตรฐานด้วยโมเดลการเรียนรู้ร่วมกันแบบใช้การตัดสินใจร่วมกันแบบ AdaBoostM1

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

ข้อมูลไม่สมดุลเป็นข้อมูลที่สามารถพบเจอได้จริงในชีวิตประจำวัน เช่น ข้อมูลการวินิจฉัยทางด้านการแพทย์ ข้อมูลบัตรเครดิตหรือสินเชื่อทางการเงิน เมื่อนำข้อมูลเหล่านี้มาใช้งานทางด้านการเรียนรู้ของเครื่องจักรและการทำเหมืองข้อมูลจะส่งผลกระทบต่อการเรียนรู้ของอัลกอริทึมมาตรฐานที่มีอยู่ เนื่องจากข้อมูลที่ใช้ในการเรียนรู้มีกลุ่มหนึ่งซึ่งเป็นกลุ่มที่ให้ความสนใจมีจำนวนข้อมูลที่น้อยมากเมื่อเทียบกับข้อมูลกลุ่มอื่น ๆ ที่เหลือซึ่งอัลกอริทึมมาตรฐานทางด้านการเรียนรู้ของเครื่องจักรนั้นสามารถทำงานได้ดีในกรณีที่ข้อมูลมีจำนวนข้อมูลของแต่ละกลุ่มที่สมดุล เนื่องจากเป้าหมายของอัลกอริทึมมาตรฐานเหล่านั้นคือ การเพิ่มประสิทธิภาพและมีความแม่นยำของการจัดกลุ่มโดยรวม สำหรับข้อมูลไม่สมดุลนั้นขอบเขตของการตัดสินใจที่เป็นที่ยอมรับของอัลกอริทึมที่เป็นมาตรฐานของการเรียนรู้ของเครื่องจักรนั้นจะมีความเอนเอียงไปทางกลุ่มข้อมูลส่วนมาก ส่งผลให้การจัดกลุ่มของข้อมูลส่วนน้อยมีแนวโน้มที่จะได้รับการจัดกลุ่มที่ผิดประเภท

ข้อมูลไม่สมดุลนอกจากจะพิจารณาจำนวนตัวอย่างของแต่ละกลุ่มที่แตกต่างกันมากแล้ว สิ่งหนึ่งที่จะต้องพิจารณาร่วมกับขนาดของตัวอย่างคือ ลักษณะข้อมูลของแต่ละกลุ่มที่มีลักษณะคล้ายกันมากซึ่งข้อมูลเหล่านี้จะมีการใช้พื้นที่ร่วมกันซึ่งข้อมูลที่มีลักษณะดังที่ได้กล่าวมาแล้วนี้จะส่งผลให้การค้นหาโมเดลหรือรูปแบบด้วยการจำแนกประเภทข้อมูลนั้นทำได้ยากยิ่งขึ้น ส่งผลให้ประสิทธิภาพของการจำแนกประเภทข้อมูลที่ได้นั้นไม่มีประสิทธิภาพ

ดังนั้นวัตถุประสงค์ของงานวิจัยวิทยานิพนธ์นี้คือ เพื่อศึกษาวิธีการต่าง ๆ ที่สามารถช่วยเพิ่มความถูกต้องในการค้นหาโมเดลการจำแนกประเภทข้อมูลไม่สมดุลโดยที่ข้อมูลอาจจะมีอัตราความไม่สมดุลของกลุ่มข้อมูลและมีอัตราการซ้อนทับกันของกลุ่มข้อมูลที่แตกต่างกัน ดังนั้นจึงได้ทำการออกแบบและสร้างอัลกอริทึมใหม่ที่ชื่อว่า EnsDTV ซึ่งอัลกอริทึมนี้สามารถช่วยในการค้นหาโมเดลจำแนกประเภทข้อมูลที่สามารถทำงานได้ดีกับข้อมูลที่มีอัตราความไม่สมดุลและอัตราการซ้อนทับที่แตกต่างกัน การทำงานของอัลกอริทึมนี้จะเริ่มจากการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลด้วยการลดข้อมูลกลุ่มส่วนมากซึ่งเป็นข้อมูลที่ใช้พื้นที่ร่วมกับข้อมูลกลุ่มส่วนน้อยภายใต้ระยะทางที่กำหนดออกเพียงบางส่วน โดยจะเก็บข้อมูลกลุ่มส่วนน้อยซึ่งเป็นกลุ่มที่ให้ความสนใจไว้ จากนั้นนำข้อมูลไม่สมดุลที่ผ่านกระบวนการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลนี้ไปทำการค้นหารูปแบบด้วยวิธีการเรียนรู้ร่วมกันแบบการใช้การตัดสินใจร่วมกันทั้งแบบแบ็กกิงและบูสต์ติง

ชดเชยการทำนายข้อมูลผิดกลุ่มด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่าย ซึ่งจะนำค่าที่ได้จากการสร้างตารางค่าใช้จ่ายมาใช้ในการปรับค่าพารามิเตอร์ของการเรียนรู้ร่วมกันและใช้โครงสร้างต้นไม้ตัดสินใจเป็นตัวจำแนกประเภทข้อมูลพร้อมทั้งหาจำนวนต้นไม้ตัดสินใจที่เหมาะสมด้วยวิธีการมโนภาพหรือวิชวลไลเซชัน

5.1 สรุปผลการวิจัย

จากผลการทดสอบประสิทธิภาพของอัลกอริทึม EnsDTV ด้วยชุดข้อมูลสังเคราะห์จากโปรแกรมและชุดข้อมูลจากแหล่งข้อมูลมาตรฐานนั้น สามารถสรุปผลการทดสอบได้ดังนี้

1. ชุดข้อมูลไม่สมดุลของทั้งสองแหล่งที่นำมาใช้ในการทดลองนี้จะผ่านกระบวนการลดอัตราการซ้อนทับกันระหว่างกลุ่มข้อมูลด้วยการกำจัดข้อมูลกลุ่มส่วนมากออกไปบางส่วนโดยชุดข้อมูล vehicle มีอัตราการซ้อนทับของกลุ่มข้อมูลน้อยที่สุด คือ 0.39 ผลการทดสอบปรากฏว่า โมเดลที่มีการเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบบูสต์ติงโดยเฉพาะอัลกอริทึม RUSBoost นั้นสามารถเรียนรู้ได้ทุกระดับของอัตราความไม่สมดุล รองลงมาคือ LogitBoost TotalBoost และ AdaBoostM1 ตามลำดับ ในขณะที่โมเดลที่มีการเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบแบ็กกิงนั้นไม่สามารถเรียนรู้ได้
2. โมเดลที่มีการเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบบูสต์ติงนั้นจะให้ค่าความถูกต้องโดยรวมและค่าความถูกต้องของการจำแนกประเภทข้อมูลกลุ่มส่วนน้อยได้ดีกว่าโมเดลที่มีการเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบแบ็กกิง ซึ่งจากผลการทดสอบของโมเดลจะพบว่าโมเดลที่มีการเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบบูสต์ติงซึ่งทำงานด้วยอัลกอริทึม LogitBoost นั้นจะสามารถจำแนกประเภทข้อมูลกลุ่มส่วนน้อยได้อย่างมีประสิทธิภาพที่สุด รองลงมา คือ RUSBoost ส่วน TotalBoost และ AdaBoostM1 นั้นจะให้ประสิทธิภาพของการจำแนกประเภทข้อมูลกลุ่มส่วนน้อยที่ใกล้เคียงกัน
3. โมเดลที่มีการเรียนรู้ร่วมกันด้วยการใช้การตัดสินใจร่วมกันแบบแบ็กกิงนั้นเหมาะสำหรับชุดข้อมูลที่มีการซ้อนทับกันของข้อมูลที่น้อย และชุดข้อมูลนั้นควรจะเป็นข้อมูลที่มีอัตราความไม่สมดุลที่ต่ำ ซึ่งจะเห็นได้ว่าสามารถจำแนกประเภทข้อมูลที่มีอัตราความไม่สมดุลไม่เกิน 5 ได้ดีกว่าระดับอื่น ๆ

จากผลการสรุปแสดงให้เห็นว่า อัลกอริทึม EnsDTV ที่นำเสนอขึ้นมานั้นสามารถนำมาใช้แก้ปัญหาการจำแนกประเภทข้อมูลไม่สมดุลที่มีอัตราความไม่สมดุลที่สูงและมีอัตราการซ้อนทับที่แตกต่างกันได้อย่างมีประสิทธิภาพ

5.2 ปัญหาและข้อเสนอแนะ

ในการแก้ไขปัญหาของการจำแนกประเภทข้อมูลที่มีอัตราความไม่สมดุลสูงร่วมกับมีการใช้พื้นที่ร่วมกันระหว่างกลุ่มนั้นเป็นปัญหาที่ไม่สามารถนำวิธีการเพียงวิธีการเดียวมาใช้ในการแก้ปัญหาได้ เมื่อต้องการเพิ่มประสิทธิภาพในการจำแนกประเภทข้อมูลที่มีลักษณะเช่นนี้ให้สามารถจำแนกประเภทข้อมูลกลุ่มส่วนน้อยได้อย่างมีประสิทธิภาพและในขณะเดียวกันก็สามารถจำแนกประเภทข้อมูลกลุ่มส่วนมากได้อย่างมีประสิทธิภาพเช่นเดียวกันนั้นเครื่องมือที่จะนำมาใช้จะต้องเหมาะกับลักษณะการกระจายตัวของชุดข้อมูล และสัดส่วนของจำนวนตัวอย่างของแต่ละกลุ่ม เนื่องจากปัจจัยเหล่านี้จะมีผลต่อประสิทธิภาพของการจำแนกประเภทข้อมูล

ดังนั้นสิ่งที่จะนำเสนอคือ ปรับปรุงอัลกอริทึมร่วมกับเทคนิคอื่น ๆ ที่สามารถแก้ปัญหาข้อมูลไม่สมดุลที่มีการใช้พื้นที่ร่วมกันเพื่อให้สามารถจำแนกประเภทข้อมูลไม่สมดุลได้อย่างมีประสิทธิภาพ

รายการอ้างอิง

Batista, G. E., Prati, R. C., andMonard, M. C. (2004b).A study of the behavior of several methods for balancing machine learning training data.**ACM SIGKDD Explorations Newsletter: Special Issue on Imbalanced Data Sets**, 6(1):20-29.

Batista, G. E., Monard, M. C., andBazzan, A. L. (2004a). Improving rule induction precision for automated annotation by balancing skewed data sets. **In Knowledge Exploration in Life Science Informatics**, Springer Berlin Heidelberg, 20-32.

Breiman, L. (1996). Bagging predictors.**Machine learning**, 24(2):123-140.

Brown, I., andMues, C. (2012). An experimental comparison of classification algorithms for imbalanced credit scoring data sets. **Expert Systems with Applications**, 39(3):3446-3453.

Bryll, R. (2003). Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets. **Pattern Recognition**, 20 (6): 1291–1302.

Cateni, S., Colla, V., andVannucci, M. (2014). A method for resampling imbalanced datasets in binary classification tasks for real-world problems. **Neurocomputing**, 135:32-41.

Chawla, N.V. (2005). Data Mining for Imbalanced Datasets: An Overview. **In Data Mining and Knowledge Discovery Handbook**, 853-867.

Chawla, N.V., Bowyer, K.W., Hall, L.O., and Kegelmeyer, W.P. (2002).SMOTE: synthetic minority over-sampling technique.**Journal of Artificial Intelligence Research**, 16:321–357.

Chawla, N.V., Lazarevic, A., Hall, L.O., and Bowyer,K.W. (2003).Smoteboost: Improving prediction ofthe minority class in boosting. **Lecture Notes in Artificial Intelligence 2838**,107–119.

Chawla, N.V., Japkowicz, N., andKotcz, A. (2004). Editorial: special issue on learning from imbalanced data sets. **ACM Sigkdd Explorations Newsletter**, 6(1): 1-6.

Cheplygina, V., and Tax, D.M.J. (2011).Pruned random subspace method for one-class classifiers.**In Multiple Classifier Systems**, Springer Berlin Heidelberg, 96-105.

Dubey,R., Zhou, J., Wang,Y., Thompson,P.M., Ye, J. (2014). Analysis of sampling techniques for imbalanced data: An n = 648 ADNI study. **NeuroImag,** 87:220–241.

Farquad, M. A. H., and Bose, I. (2012). Preprocessing unbalanced data using support vector machine. **Decision Support Systems**, 53(1):226-233.

Freund, Y., and Schapire, R.E. (1996).Experiments with a new boosting algorithm.**Proceedings 13th International Conference on Machine Learning**, 96:148-156.

Galar, M., Fernández, A., Barrenechea, E., and Herrera, F. (2013).Eusboost: enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling. **Pattern Recognition**, 46(12):3460-3471.

Gao, M., Hong, X., Chen, S., and Harris, C. J. (2012). Probability density function estimation based over-sampling for imbalanced two-class problems. **In Neural Networks (IJCNN), The 2012 International Joint Conference on IEEE**, 1-8.

Han, J., and Kamber, M. (2006). Data mining: concepts and techniques. Amsterdam; Boston: Elsevier.

Han, H., Wang, W. Y., and Mao, B. H. (2005). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. **In Advances in intelligent computing**, Springer Berlin Heidelberg, 878-887.

He, H. and Garcia, E.A. (2009).Learning from imbalanced data.**Knowledge and Data Engineering, IEEE Transactions on**, 21(9):1263-1284.

He, H., andGhodsi, A. (2010).Rare class classification by support vector machine.**In Pattern Recognition (ICPR), 2010 20th International Conference on IEEE**, 548-551.

Krawczyk, B., Wozniak, M., and Schaefer,G. (2014).Cost-sensitive decision tree ensembles for effective imbalanced classification.**Applied Soft Computing**, 14:554-562.

Kubat, M., Holte, R. C., and Matwin, S. (1998). Machine learning for the detection of oil spills in satellite radar images. **Machine learning**, 30(2-3):195-215.

Liao, J. J., Shih, C. H., Chen, T. F., and Hsu, M. F. (2014).An ensemble-based model for two-class imbalanced financial problem.**Economic Modelling**, 37, 175-183.

Liu, X., Wu,J., and Zhou, Z. (2009).Exploratory undersampling for class-imbalance learning.**IEEE Transactions on Systems Man and Cybernetics Part B: Cybernetics**, 39 (2)539–550.

López, V., Fernández, A., Moreno-Torres, J. G., and Herrera, F. (2012).Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics. **Expert Systems with Applications**, 39(7): 6585-6608.

Luengo, J., Fernández, A., & Herrera, F. (2009).Addressing data-complexity for imbalanced data-sets: A preliminary study on the use of preprocessing for c4. 5. **In Intelligent Systems Design and Applications, 2009. ISDA'09. Ninth International Conference on IEEE**, 523-528.

Nanni, L. (2006). Experimental comparison of one-class classifiers for online signature verification.Neurocomputing**69** (7): 869-873.

Orriols-Puig, A., &Bernadó-Mansilla, E. (2009).Evolutionary rule-based systems for imbalanced data sets.**Soft Computing**, 13(3):213-225.

Phung, S. L., Bouzerdoum, A., and Nguyen, G. H. (2009).Learning pattern classification tasks with imbalanced data sets.**In P . Yin (Eds.), Pattern recognition, Vukovar, Croatia: In-Teh**, 193-208.

Polikar , R. (2006). Ensemble Based Systems in Decision Making.**IEEE Circuits and Systems Magazine**, 6(3): 21-45.

Qian, Y., Liang, Y., Li, M., Feng, G., and Shi, X. (2014).A resampling ensemble algorithm for classification of imbalance problems.**Neurocomputing**, 143:57-67.

Quinlan, J. R. (1986). Induction of decision trees.**Machine learning**, 1(1):81-106.

Skurichina, M., and Duin, R.P.W. (2002). Bagging, Boosting and the Random Subspace Method for Linear Classifiers. **Pattern Analysis and Applications**, 5(2): 121–135.

Sun, Y., Wong, A. K., andKamel, M. S. (2009). Classification of imbalanced data: A review. **International Journal of Pattern Recognition and Artificial Intelligence**, 23(04):687-719.

Thalor, M. A., andPatil, S. T. (2014). Review of Ensemble Based Classification Algorithms for Nonstationary and Imbalanced Data.**IOSR Journal of Computer Engineering (IOSR-JCE)**, 16(6):103-107

Villar, P., Fernández, A., & Herrera, F. (2011).Studying the behavior of a multiobjective genetic algorithm to design fuzzy rule-based classification systems for imbalanced data-sets.**In**

Proceedings of the 2011 IEEE International Conference onFuzzy Systems, 1239-1246.

Wilson, D.L. (1972). Asymptotic properties of nearest neighbor rules using edited data. **IEEE Transactions on Systems, Man, and Cybernetics**,3:408–421.

Wu, Q., Ye, Y., Zhang, H., Ng, M. K., and Ho, S. S. (2014). ForesTexter: An efficient random forest algorithm for imbalanced text categorization. **Knowledge-Based Systems**, 67:105-116.

Xiong, H.,Wu, J., and Liu, L. (2010). Classification with Class Overlapping: A Systematic Study. **In Proceedings of the 2010 International Conference on E-Business Intelligence**, 491-497.

Yin, L., Ge, Y., Xiao, K., Wang, X., and Quan, X. (2013). Feature selection for high-dimensional imbalanced data. **Neurocomputing**, 105:3-11.

ภาคผนวก ก

รหัสต้นฉบับของโปรแกรม EnsDTV

```
%1.สร้างข้อมูลสังเคราะห์ด้วย MatLabR2013b
%ข้อมูลสังเคราะห์มี 3 มิติ 2 คลาส
closeall;
clearall;
randn('seed',0);
m=[0 0 1; 1 2 3]';          %ค่ากลางของข้อมูลกลุ่มที่ 1 และ2
S1 = [1 0 0; 0 1 0; 0 0 1]; % ค่าความแปรปรวนของข้อมูลกลุ่มที่ 1
S2 = [0.5 0 0; 0 0.5 0; 0 0 0.5];% ค่าความแปรปรวนของข้อมูลกลุ่มที่ 2
S(:,:,1)=S1;
S(:,:,2)=S2;
P=[0.1 0.9]';
N=300;
sed=0;
[X,y]=mixt_model(m,S,P,N,sed);%ต้องมี mixt_model.m
c1=find(y==1);%คลาส '1'
c2=find(y==2); %คลาส '2'
figure;
subplot(2,2,1);
scatter3(X(1,c1,:),X(2,c1,:),X(3,c1,:),'bo');
holdon;
scatter3(X(1,c2,:),X(2,c2,:),X(3,c2,:),'r*');
xlabel('Feature 1');
ylabel('Feature 2');
zlabel('Feature 3');
title('Synthetics datasets');
legend('Minority Class', 'Majority Class','Location','NE');

subplot(2,2,2);
plot(X(1,c1,:),X(2,c1,:),'bo');
holdon;
```

```
plot(X(1,c2,:),X(2,c2,:),'r*');
xlabel('Feature 1');
ylabel('Feature 2');

subplot(2,2,3);
plot(X(2,c1,:),X(3,c1,:),'bo');
holdon;
plot(X(2,c2,:),X(3,c2,:),'r*');
xlabel('Feature 2');
ylabel('Feature 3');

subplot(2,2,4);
plot(X(1,c1,:),X(3,c1,:),'bo');
holdon;
plot(X(1,c2,:),X(3,c2,:),'r*');
xlabel('Feature 1');
ylabel('Feature 3');
holdoff;

X1=X(:,c1,:);%ข้อมูลกลุ่มที่ 1
X2=X(:,c2,:); %ข้อมูลกลุ่มที่ 2
[NumOfFeatures,N]=size(X1);
[m_hat, S_hat]  =Gaussian_ML_estimate(X) %ต้องมี Gaussian_ML_estimate.m
[m_hat1, S_hat1]=Gaussian_ML_estimate(X1)
[m_hat2, S_hat2]=Gaussian_ML_estimate(X2)

% Fisher Ratio
fori=1:NumOfFeatures
FDR_value(i)=Fisher(X1(i,:),X2(i,:));%ต้องมี Fisher.m
end
```

```
maxF = max(FDR_value)  %ค่า overlapped ratio แสดงการซ้อนทับของข้อมูล


%.ในกรณีที่เป็นข้อมูลสังเคราะห์ สามารถนำX1 และ X2 ไปใช้ได้เลย
%ในกรณีที่เป็นข้อมูลจริงจาก Keel  จะต้องทำดังต่อไปนี้
% 1. Import ข้อมูลด้วย wizard โดย
% ข้อมูลเก็บไว้ที่ตัวแปร X  มีชนิดเป็นMatrix
% คลาสลาเบลเก็บไว้ที่ตัวแปร y มีชนิดเป็น Cell Array
%2.แบ่งข้อมูล X ออกเป็น 2 กลุ่มดังนี้
%X=X';
%y=y';
% คลาสลาเบลของข้อมูลนี้คือ positive และ negative
%c1=find(strcmp(y,'positive'));
%c2=find(strcmp(y,'negative'));
%X1=X(:,c1,:);
%X2=X(:,c2,:);
%.นำ X1 และ X2 ไปใช้ในขั้นตอนที่ 2


%ขั้นตอนที่2.Reduce Overlapped Regionด้วย MatLabR2013b
XX = X1';%ข้อมูลกลุ่มที่ 1positive
YY = X2';%ข้อมูลกลุ่มที่ 2negative
distRatio = 0.3;%ระยะห่างระหว่างข้อมูลสองกลุ่ม กำหนดเอง
n = size(YY,1);
m = size(XX,1);
newYY  = zeros(1,size(YY,2));
fori =1:n
   c =0;
for j = 1:m
      d = (XX(j,:) -YY(i,:)) *(XX(j,:) -YY(i,:))';
dd = sqrt(d);
```

```
dist(i,j) = dd;
ifdd<= distRatio
        c = c+1;
end
end
cn(i)=c;
if c==0
newYY = [newYY;YY(i,:)];
end
end
newYY = newYY(2:size(newYY,1),:); %ข้อมูลกลุ่มที่ 2 ซึ่งเป็น Majority Class ที่ผ่านการลด
%ข้อมูลที่มีการใช้พื้นที่ร่วมกันกับข้อมูล Minority Class
%บันทึกข้อมูล X1 และเพิ่ม 1 Column เพื่อเก็บ Class Label เป็นminority.csv ไว้ที่ drive C
%บันทึกข้อมูล newYYและเพิ่ม 1 Column เพื่อเก็บ Class Label เป็นmajority.csv ไว้ที่ drive C

%ขั้นตอนที่ 3 Stratified Sampling Using R
require(sampling)
dataMin<- read.csv("C:/minority.csv", header=FALSE)
dataMaj<- read.csv("C:/majority.csv", header=FALSE)
stratified = function(tmpD, size,method1) {
group = dim(tmpD)[2]
strat= strata(tmpD, stratanames = names(tmpD[group]),size = size, method = method1)
dsample = getdata(tmpD, strat)
}
ssSizeMin = 10  # จำนวนที่ต้องการสุ่มเลือกของกลุ่ม Minority
ssSizeMaj= 100  # จำนวนที่ต้องการสุ่มเลือกของกลุ่ม Majority
ssMethod = "srswor"
#srsworis sampling without replacement, srswr is sampling with replacement
ssMin=stratified(dataMin,ssSizeMin,ssMethod)
ssMaj=stratified(dataMaj,ssSizeMaj,ssMethod)
```

```
c=(dim(ssMaj)[2])-4
write.csv(ssMin[1:c], file="C:/ssMin.csv"
write.csv(ssMaj[1:c], file="C:/ssMaj.csv"

%ขั้นตอนที่ 4 Gen Ensemble Model
Import data of ssMin into Xp
Import data of ssMaj into Xn

cp = size(Xp,1);
strArray(1:cp) = java.lang.String('positive');
Yp = cell(strArray);

cn = size(Xn,1);
strArray(1:cn) = java.lang.String('negative');
Yn = cell(strArray);

XX = [Xp;Xn];
yy = [Yp;Yn];

weakLearner = 'Tree';
numLearner  = 200;
dataD       = tabulate(yy);

ifdataD{1,1}== 'negative'
i = ceil(dataD{1,2}/dataD{2,2});
ClassNames = {'negative"positive'};
cost.ClassificationCosts = [0,1;i,0];
else
i  = ceil(dataD{2,2}/dataD{1,2});
ClassNames = {'positive"negative'};
```

```
cost.ClassificationCosts = [0,i;1,0];
end

cost.ClassNames = ClassNames;

ensFuncList = {'AdaBoostM1''Bag''TotalBoost''LogitBoost''RUSBoost'};
lenEnsFuncList   = length(ensFuncList);

DD.AdaBoostM1 = [];
DD.Bag        = [];
DD.TotalBoost = [];
DD.LogitBoost = [];
DD.RUSBoost   = [];

for k=1:1:lenEnsFuncList
cn =1;
rng(0,'twister')
display('==================================')
ensFunc    = ensFuncList{k}
   model     =
fitensemble(XX,yy,ensFunc,numLearner,weakLearner,'Type','Classification','Cost',cost.Classific
ationCosts,'kfold',5);
   [pre,Sfit] = kfoldPredict(model);
conmat     = confusionmat(yy,pre,'Order',ClassNames) ;
   CM         =
table(conmat(:,1),conmat(:,2),'VariableNames',ClassNames,'RowNames',ClassNames);
fprintf('Comfusion Matrix\n');
disp(CM)
accuracy   = (sum(strcmp(pre,yy))/length(yy))*100;
ifdataD{1,1}== 'negative'
```

```
        TP = conmat(4);
        FP = conmat(3);
        FN = conmat(2);
        TN = conmat(1);
else
        TP = conmat(1);
        FP = conmat(2);
        FN = conmat(3);
        TN = conmat(4);
end

%Plot graph for check error
figure;
plot(kfoldLoss(model,'Mode','Cumulative'),'r--');
xlabel('Number of trees');
ylabel('Classification error');
title(ensFunc);
holdon;

%Find suitable number of learner
result          = kfoldLoss(model,'Mode','Cumulative');
mLearner        = find(result == min(result));
lenMinLearner   = length(mLearner);

%Find maximum 10 Rounds
iflenMinLearner> 10
lenMinLearner = 10;
end

for j=1:1:lenMinLearner
```

```
ddModel= [];
      D     =[];
rng(0,'twister')
numLearnerJ= mLearner(j)
      model     =
fitensemble(XX,yy,ensFunc,numLearnerJ,weakLearner,'Type','Classification','Cost',cost.Classifi
cationCosts,'kfold',5);
      [pre,Sfit] = kfoldPredict(model);
conmat     = confusionmat(yy,pre);
      CM        =
table(conmat(:,1),conmat(:,2),'VariableNames',ClassNames,'RowNames',ClassNames);
fprintf('Comfusion Matrix\n');
disp(CM)
accuracy   = (sum(strcmp(pre,yy))/length(yy))*100;
ifdataD{1,1}== 'negative'
          TP = conmat(4);
          FP = conmat(3);
          FN = conmat(2);
          TN = conmat(1);
else
          TP = conmat(1);
          FP = conmat(2);
          FN = conmat(3);
          TN = conmat(4);
end
D.Cminority(cn)  = i;
D.Cmajority(cn)  = 1;
D.numTree(cn)    = numLearnerJ;
D.TP(cn)         = TP;
D.FP(cn)         = FP;
```

```
D.FN(cn)         = FN;
D.TN(cn)         = TN;
D.Accuracy(cn)   = (TP+TN)/(TP+FN+FP+TN);
D.Precision(cn)  = TP/(TP+FP);
D.Sensitivity(cn)= TP/(TP+FN);%(conmat(1)/(conmat(1)+conmat(3)))*100
D.Specificity(cn)= TN/(TN+FP);%(conmat(4)/(conmat(2)+conmat(4)))*100
D.FPrate(cn)     = 1-D.Specificity(cn);
D.FNrate(cn)     = 1-D.Sensitivity(cn);
D.Fmeasure(cn)   = (2*D.Precision(cn)*D.Sensitivity(cn))/(D.Precision(cn)+D.Sensitivity(cn));
D.Gmean(cn)      = sqrt(D.Sensitivity(cn)*D.Specificity(cn));
D.AUC(cn)        = (1+D.Sensitivity(cn)-D.FPrate(cn))/2;
D.TotalMis(cn)   = (D.FPrate(cn)*D.Cminority(cn))+ (D.FNrate(cn)*D.Cmajority(cn));
D.TotalError(cn) = D.FPrate(cn)+ D.FNrate(cn);

%Plot graph for check error
plot(kfoldLoss(model,'Mode','Cumulative'),'b-');
legend('Original', 'Optimization of Tree','Location','NE');
holdon;
end
holdoff;
ddModel = [D.Cminority' D.Cmajority' D.numTree' D.TP' D.FP' D.FN' D.TN' D.Accuracy'
D.Precision' D.Sensitivity' D.Specificity' D.FPrate' D.FNrate' D.Fmeasure' D.Gmean' D.AUC'
D.TotalMis' D.TotalError'];
if k ==1
      DD.AdaBoostM1 = ddModel;
elseif k ==2
DD.Bag       = ddModel;
elseif k ==3
DD.TotalBoost = ddModel;
```

```
elseif k ==4
```

```
DD.LogitBoost = ddModel;
else
DD.RUSBoost   = ddModel;
end
cn = cn+1;
display('================================')
end
```

ภาคผนวก ข

บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่

รายชื่อบทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่

ภาสพิชญ์ ชูใจ นิตยา เกิดประสพ และกิตติศักดิ์ เกิดประสพ (2557). **กระบวนการเตรียมข้อมูลสำหรับข้อมูลไม่สมดุลเพื่อเพิ่มประสิทธิภาพในการจำแนกข้อมูล**. การประชุมวิชาการระดับชาติมหาวิทยาลัยเทคโนโลยีราชมงคล ครั้งที่ 6 (6thRMUTNC).

Pasapitch Chujai, KittipongChomboon, PongsakornTeerarassamee, Nittaya Kerdprasop and Kittisak Kerdprasop(2015). **Ensemble Learning For Imbalanced Data Classification Problem.** ICIAE'2015the 3rd International Conferenceon Industrial Application Engineering 2015,The Institute of Industrial Applications Engineers, Japan.28-31March2015.

กระบวนการเตรียมข้อมูลสำหรับข้อมูลที่ไม่สมดุลเพื่อเพิ่มประสิทธิภาพในการจำแนกข้อมูล

Pre-processing of Imbalanced Data for Improving Data Classification

ภาสพิชญ์ ชูใจ*, นิตยา เกิดประสพและกิตติศักดิ์ เกิดประสพ

Pasapitch Chujai*, Nittaya Kerdprasopand Kittisak Kerdprasop

บทคัดย่อ

การจำแนกข้อมูลเป็นเทคนิคหนึ่งของงานทางด้านเหมืองข้อมูลที่ได้รับความนิยมเนื่องจากมีประสิทธิภาพและความถูกต้องในการจำแนกที่สูง แต่เทคนิคนี้จะประสบกับปัญหาในการจำแนกข้อมูลในกรณีที่ชุดข้อมูลมีลักษณะไม่สมดุลซึ่งจะปรากฏเมื่อจำนวนข้อมูลของกลุ่มหนึ่งมีมากกว่าอีกกลุ่มหนึ่งเป็นจำนวนมากดังนั้นในงานวิจัยนี้จึงได้นำขั้นตอนการเตรียมข้อมูลด้วยการคัดเลือกคอลัมน์ที่เหมาะสมด้วยเทคนิค Principal Component Analysis (PCA) และทำให้ข้อมูลสมดุลด้วยการเพิ่มจำนวนข้อมูลในกลุ่มที่มีข้อมูลน้อยด้วยเทคนิค Synthetic Minority Over-sampling Technique (SMOTE) หรือลดจำนวนข้อมูลในกลุ่มที่มีข้อมูลจำนวนมากด้วยเทคนิค Resample จากนั้นสร้างแบบจำลองข้อมูลทั้งแบบเชิงเดี่ยวและแบบเชิงกลุ่มด้วยวิธีการโครงข่ายประสาทเทียม ซัพพอร์ตเวกเตอร์แมชชีน ต้นไม้ตัดสินใจ และกลุ่มของต้นไม้ตัดสินใจ ผลที่ได้ปรากฏว่าเมื่อทำการคัดเลือกคอลัมน์ที่เหมาะสมด้วยเทคนิค PCA เป็นอันดับแรก แล้วเพิ่มจำนวนข้อมูลในกลุ่มที่มีข้อมูลน้อยด้วยเทคนิคSMOTE จะสามารถจำแนกข้อมูลที่ไม่สมดุลได้อย่างแม่นยำโดยเฉพาะแบบจำลองข้อมูลแบบเชิงกลุ่มที่ใช้ตัวจำแนกกลุ่มของต้นไม้ตัดสินใจ (RandomForest)สามารถจำแนกชุดข้อมูลที่ไม่สมดุลได้อย่างมีประสิทธิภาพและแม่นยำ

คำสำคัญ: ข้อมูลที่ไม่สมดุล การจำแนกข้อมูล กระบวนการเตรียมข้อมูล แบบจำลองข้อมูลเชิงกลุ่ม

Abstract

Classification of data is the popular technique that has been widely used in the field of data mining due to its high efficiency and predictive accuracy. However, this technique still suffers from the problem of classifying imbalanced datain which the data in one group outnumbers the data in other groups.Therefore, this research proposes a pre-processing technique by applyingfeature selectionwith Principal Component Analysis (PCA) and then balancing data by increasing the number of data in minority class with Synthetic Minority Over-sampling Technique (SMOTE) or decreasing the number of data in majority class with resample method. The pre-processed data set was then used to build the classification model through both a single modeling and ensemble methods using artificial neural network, support vector machine, decision tree induction, and RandomForest algorithms. The experimental results showed that feature selectionwithPCA technique first and then increasing the minority class with SMOTE can be used to classify

imbalanced dataset precisely. In particular, the ensemble model obtained from the RandomForest classifier can be used to classify imbalanced data effectively and has precision.

Keywords:Imbalanced Data, Classifications, Pre-processing, Ensemble Data Model

1. บทนำ

การจำแนกข้อมูลเป็นเทคนิคหนึ่งที่นิยมนำมาประยุกต์ใช้งานจริงในหลาย ๆ ด้านด้วยกัน เนื่องจากเทคนิคนี้สามารถจำแนกข้อมูลได้อย่างถูกต้องและมีความแม่นยำสูง อย่างไรก็ตามเทคนิคก็ยังคงมีจุดด้อยเมื่อมีการนำไปใช้กับงานที่ข้อมูลมีลักษณะไม่สมดุล (Imbalanced Dataset) [1,2,3]ซึ่งจะมีจำนวนข้อมูลของกลุ่มหนึ่งมากกว่าจำนวนข้อมูลของอีกกลุ่มหนึ่งเป็นจำนวนมาก ซึ่งข้อมูลไม่สมดุลนั้นสามารถพบเห็นได้จริงในงานหลาย ๆ ด้าน เช่น ด้านการวินิจฉัยทางการแพทย์ที่ให้ความสนใจกับคลาสส่วนน้อย(Minority Class) ที่เป็นผู้ป่วยมากกว่าคลาสส่วนมาก(Majority Class) ที่เป็นผู้ที่มีสุขภาพดีโดยที่ข้อมูลของผู้ป่วยจะมีจำนวนน้อยกว่าข้อมูลของผู้ที่มีสุขภาพดีเป็นจำนวนมาก หรือข้อมูลทางด้านบัตรเครดิตที่มีข้อมูลลูกค้าปกติมากกว่าลูกค้าที่ผิดปกติ ซึ่งข้อมูลที่มีลักษณะไม่สมดุลเหล่านี้จะส่งผลให้ไม่สามารถจำแนกข้อมูล[4] ที่เกี่ยวข้องกับคลาสส่วนน้อยได้หรือถ้าได้ก็จะมีความแม่นยำที่น้อยมากในขณะที่สามารถจำแนกข้อมูลของคลาสส่วนมากได้อย่างถูกต้องและมีความแม่นยำที่สูง

จากปัญหาความไม่สมดุลของข้อมูลที่กล่าวมาแล้วนั้นได้มีนักวิจัยจำนวนมากได้นำเสนอวิธีการต่าง ๆ เพื่อแก้ปัญหานี้และสามารถนำมาใช้กับวิธีการจำแนกข้อมูล ส่งผลให้สามารถทำนายข้อมูลที่อยู่ในกลุ่มที่มีจำนวนน้อยมีความแม่นยำที่สูงขึ้น ซึ่งวิธีการต่าง ๆ นั้นจะเห็นได้จากงานวิจัยดังต่อไปนี้

I.Bownและ M. Christophe [5] ได้นำเสนอเทคนิคการจำแนกกับข้อมูลที่ไม่สมดุลและเพิ่มจำนวนคลาสที่มีน้อยด้วยเทคนิคการลดแบบสุ่มกับ5 ชุดข้อมูลสินเชื่อและวัดประสิทธิภาพด้วยพื้นที่ใต้กราฟ (Area Under Curve: AUC) โดยการหาค่าเฉลี่ยว่ามีคู่ใดบ้างที่มีแตกต่างกัน (post hoc tests) ด้วยวิธี FriedmanTest และ NemenyiTest ผลที่ได้ปรากฏว่ากลุ่มของต้นไม้ตัดสินใจ (Random Forest)และGradient Boosting สามารถจำแนกข้อมูลสินเชื่อได้อย่างมีประสิทธิภาพ ในขณะเดียวกันถ้าข้อมูลไม่สมดุลมีขนาดของข้อมูลที่ใหญ่ขึ้นจะพบว่า อัลกอริทึม C4.5, QDA และ k-NN ไม่สามารถจำแนกข้อมูลได้อย่างมีประสิทธิภาพ

V. Lopez และคณะ [6] ได้ดำเนินการในส่วนของขั้นตอนการเตรียมข้อมูลสำหรับแก้ปัญหาข้อมูลไม่สมดุลที่มีสองคลาสด้วยเทคนิคการเรียนรู้แบบมีค่าใช้จ่าย (Cost Sensitive Leaning) เทคนิคการสุ่มเกิน (Oversampling) ด้วยวิธี SMOTE และ SMOTE +ENNและเทคนิคการสุ่มลด (Undersampling)ซึ่งในงานวิจัยนี้จะเน้นไปที่การนำเทคนิคการสุ่มเกินและเทคนิคการสุ่มลดมาทำงานร่วมกับเทคนิควิธีการเรียนรู้แบบมีค่าใช้จ่าย ผลที่ได้จากการทดสอบด้วย 66 ชุดข้อมูลจาก Keel และวัดประสิทธิภาพด้วยพื้นที่ใต้กราฟปรากฏว่าวิธีที่นำเสนอสามารถแก้ปัญหาของข้อมูลไม่สมดุลได้อย่างไม่มีนัยสำคัญ

S. Cateniและคณะ [7] ได้นำเสนอวิธีการสำหรับแก้ปัญข้อมูลที่ไม่สมดุลที่มีสองคลาสโดยการนำสองเทคนิควิธีของการสุ่มเลือกข้อมูล (Resampling)คือ วิธีการสุ่มเกินและวิธีการสุ่มลด มาทำงานร่วมกัน เรียกวิธีนี้ว่า SUNDO โดยจะทำงานร่วมกับสี่โมเดลสำหรับการจำแนกข้อมูล คือ ซัพพอร์ตเวกเตอร์แมชชีน ต้นไม้ตัดสินใจ นิวรอนเน็ตเวอร์กแบบไม่มีผู้สอน(Self-Organizing Map: SOM)และการจำแนกข้อมูลแบบเบย์

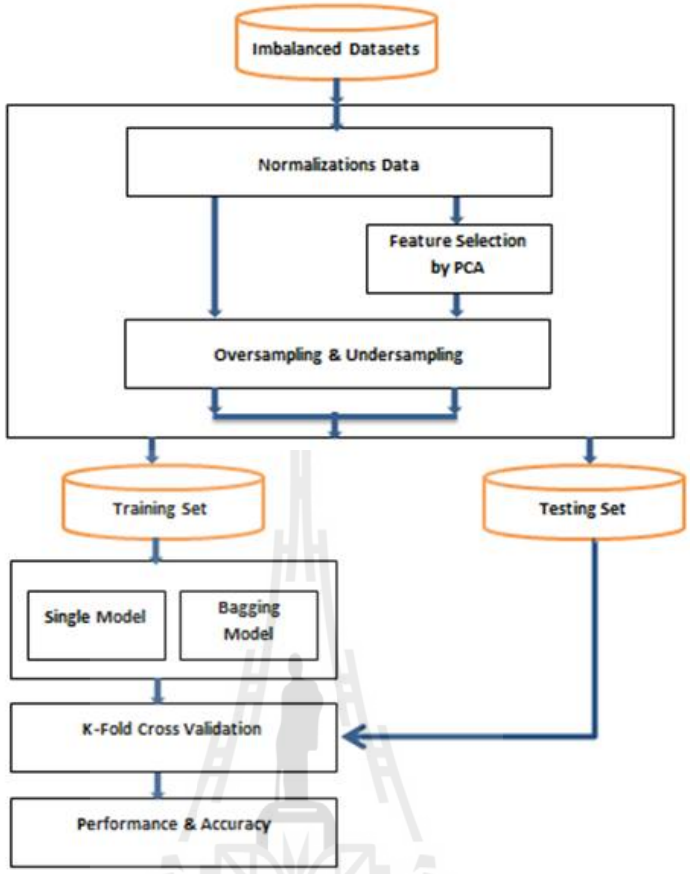
(Bayesian Classifiers)งานวิจัยนี้ได้ทำการทดลองกับสี่ชุดข้อมูลซึ่งประกอบด้วยชุดข้อมูลที่สังเคราะห์เอง ชุดข้อมูลมะเร็งเต้านมจาก UCI (Wisconsin) และสองชุดข้อมูลจากอุตสาหกรรมโลหะในการทดลองได้ทำการแบ่งชุดข้อมูลที่ไม่สมดุลออกเป็น 75% สำหรับชุดข้อมูลฝึกสอน และ 25% สำหรับชุดข้อมูลทดสอบ ผลที่ได้ปรากฏว่า วิธีที่นำเสนอสามารถทำการจำแนกข้อมูลที่ไม่สมดุลได้อย่างมีประสิทธิภาพ

M. Galarและคณะ [8] ได้นำเสนอเทคนิควิธีการเรียนรู้ร่วมกัน(Ensemble Learning) ที่ชื่อว่า EUSBoost เพื่อนำมาใช้แก้ปัญข้อมูลที่ไม่สมดุล โดยใช้อัลกอริทึม C4.5 ร่วมกับเทคนิคการสุ่มลดและการเรียนรู้ร่วมกันแบบบูสต์ติง (Boosting)โดยได้ทำการทดสอบประสิทธิภาพกับข้อมูลจากKeelผลที่ได้ปรากฏว่า เทคนิคดังกล่าวให้ประสิทธิภาพที่ดีที่สุด

จากงานวิจัยที่ได้กล่าวมาแล้วนั้นจะพบว่า นักวิจัยได้ให้ความสำคัญกับการแก้ปัญหาการจำแนกข้อมูลที่ไม่สมดุลด้วยเทคนิควิธีต่าง ๆ ก่อนที่จะนำไปสร้างโมเดล ซึ่งในงานวิจัยนี้ได้นำขั้นตอนการเตรียมข้อมูลตั้งแต่การปรับให้ข้อมูลให้อยู่ในรูปแบบปกติ คัดเลือกคุณลักษณะที่เหมาะสม และทำการเพิ่มและลดจำนวนข้อมูลจนกระทั่งข้อมูลมีลักษณะที่สมดุล หลังจากนั้นนำข้อมูลดังกล่าวไปสร้างโมเดลที่สามารถทำการจำแนกข้อมูลไม่สมดุลและสามารถทำนายข้อมูลที่อยู่ในคลาสส่วนน้อยได้อย่างถูกต้องและมีประสิทธิภาพ

2. กรอบแนวคิดและวิธีการดำเนินการวิจัย

ในงานวิจัยนี้ได้นำเทคนิควิธีดังกล่าวข้างต้นมาทำการจำแนกข้อมูลที่มีลักษณะที่ไม่สมดุล ซึ่งก่อนสร้างโมเดลนั้นได้ดำเนินการจัดการกับปัญหาของข้อมูลที่ไม่สมดุลโดยการคัดเลือกคุณลักษณะที่เหมาะสมด้วยเทคนิควิธี PCA จากนั้นทำให้กลุ่มของข้อมูลที่มีจำนวนน้อยมีจำนวนข้อมูลที่ใกล้เคียงกับกลุ่มที่มีจำนวนมากด้วยการเพิ่มจำนวนข้อมูลในกลุ่มที่มีข้อมูลน้อยด้วยเทคนิคการสุ่มเกิน และลดจำนวนข้อมูลในกลุ่มที่มีข้อมูลจำนวนมากด้วยเทคนิคการสุ่มลด ดังรูปที่ 1



รูปที่ 1 กรอบแนวคิดการสร้างตัวจำแนกข้อมูลของข้อมูลไม่สมดุล

จากรูปที่ 1 สามารถอธิบายขั้นตอนวิธีการดำเนินงานได้ดังต่อไปนี้

ขั้นตอนแรกการเตรียมข้อมูล (Preprocessing)

ในขั้นตอนนี้จะเริ่มต้นจากการแปลงข้อมูล (Normalization) ให้อยู่ในรูปแบบที่เป็นปกติ โดยจะมีค่าอยู่ในช่วง 0-1 หลังจากนั้นจะทำการเตรียมข้อมูลโดยแบ่งออกเป็น 2กลุ่มด้วยกัน ดังนี้

กลุ่มที่หนึ่งนำข้อมูลที่ผ่านการแปลงข้อมูลแล้วไปทำการเพิ่มและลดจำนวนข้อมูลให้มีลักษณะที่สมดุล โดยการทำให้กลุ่มของข้อมูลที่มีจำนวนน้อยมีจำนวนข้อมูลที่ใกล้เคียงกับกลุ่มที่มีจำนวนมากด้วยการเพิ่มจำนวนข้อมูลในกลุ่มที่มีข้อมูลน้อยด้วยเทคนิค Synthetic Minority Over-sampling Technique (SMOTE) และลดจำนวนข้อมูลในกลุ่มที่มีข้อมูลจำนวนมากด้วยเทคนิค Resampleหลังจากนั้นนำข้อมูลชุดนี้ไปสร้างโมเดล

กลุ่มที่สอง นำข้อมูลที่ผ่านการแปลงข้อมูลแล้วไปทำการคัดเลือกคุณลักษณะที่เหมาะสมด้วยเทคนิควิธี PCA ต่อจากนั้นทำการเพิ่มและลดจำนวนข้อมูลให้มีลักษณะที่สมดุล โดยการทำให้กลุ่มของข้อมูลที่มีจำนวนน้อยมีจำนวนข้อมูลที่ใกล้เคียงกับกลุ่มที่มีจำนวนมากด้วยการเพิ่มจำนวนข้อมูลในกลุ่มที่มีข้อมูลน้อยด้วย

เทคนิค Synthetic Minority Over-sampling Technique (SMOTE) และลดจำนวนข้อมูลในกลุ่มที่มีข้อมูลจำนวนมากด้วยเทคนิค Resampleหลังจากนั้นนำข้อมูลชุดนี้ไปสร้างโมเดล

ขั้นตอนที่สองการสร้างโมเดล

สำหรับโมเดลที่ใช้ในงานวิจัยนี้จะนำเทคนิควิธีการโครงข่ายประสาทเทียม ซัพพอร์ตเวกเตอร์แมชชีน (Linear) ต้นไม้ตัดสินใจ (J48) และกลุ่มของต้นไม้ตัดสินใจ (RandomForest)มาทำการสร้างแบบจำลองข้อมูลทั้งแบบเชิงเดี่ยวและแบบเชิงกลุ่ม ซึ่งแบบเชิงกลุ่มนั้นได้นำเทคนิควิธีการเรียนรู้ร่วมกันที่มีการตัดสินใจแบบแบ็กกิ้ง (Bagging)เข้ามาใช้

ในการทดลองนี้ผู้วิจัยได้ดำเนินการปรับพารามิเตอร์ของแต่ละโมเดล แต่ประสิทธิภาพของโมเดลที่ได้มีความแม่นยำที่ไม่แตกต่างจากโมเดลที่ไม่มีการปรับพารามิเตอร์ ดังนั้นพารามิเตอร์ที่ใช้จึงเป็นพารามิเตอร์ตามค่าตั้งต้น

ขั้นตอนที่สามการวัดประสิทธิภาพ

สำหรับการวัดประสิทธิภาพของโมเดลนั้นจะทำการทดสอบข้อมูลทั้งหมดด้วยการทดสอบแบบไขว้ข้าม (Cross Validation) จำนวน 10 Fold

3. ข้อมูลและเครื่องมือที่ใช้ในการวิจัย

งานวิจัยนี้ได้นำชุดข้อมูลที่ไม่สมดุลจาก Keel [9] มาทำการวัดประสิทธิภาพของโมเดลจำนวนทั้งหมด 6 ชุดข้อมูลโดยมีรายละเอียดดังตารางที่ 1

ตารางที่ 1แสดงรายละเอียดข้อมูลที่ไม่สมดุลและผ่านกระบวนการทำให้สมดุล

ชื่อชุดข้อมูล	แอตริบิวท์ (R/I/N)*	ข้อมูลไม่สมดุล				ข้อมูลสมดุลด้วยเทคนิค SMOTE		ข้อมูลสมดุลด้วยเทคนิค Resample	
		จำนวนข้อมูล	คลาสหลัก	คลาสรอง	IR**	คลาสหลัก	คลาสรอง	คลาสหลัก	คลาสรอง
ecoli4	7 (7/0/0)	336	316	20	15.8	316	316	20	20
glass-0*	9 (9/0/0)	205	188	17	11.06	188	187	17	17
new-thyroid1	5 (4/1/0)	215	180	35	5.14	180	184	35	35
vehicle0	18 (0/18/0)	846	647	199	3.25	647	656	199	199
yeast-0*	8 (8/0/0)	1004	905	99	9.14	905	914	99	99
yeast1	8 (8/0/0)	1484	1055	429	2.46	1055	1115	429	429

(R/I/N)* : R =Real, I = Integer, N = Nominal, IR** : อัตราการไม่สมดุล

glass-0* คือ glass-0-1-4-6_vs_2 , yeast-0* คือ yeast-0-2-5-6_vs_3-7-8-9

สำหรับการพัฒนาโมเดลนั้นผู้วิจัยได้นำโปรแกรมภาษา R [10] และPython 2.7 [11] มาเป็นเครื่องมือช่วยในการพัฒนาโมเดล

สำหรับการทดลองวัดความแม่นยำในการจำแนกข้อมูลที่ไม่สมดุลนั้น ผู้วิจัยได้นำโมเดลที่กล่าวมาแล้วข้างต้นมาทำการทดลอง รายละเอียดดังตารางที่ 2

ตารางที่ 2 รายละเอียดโมเดลที่ทำการทดลอง

โมเดล	รายละเอียดโมเดลที่ทดลอง	ชื่อย่อ	โมเดล	รายละเอียดโมเดลที่ทดลอง	ชื่อย่อ
1	ซัพพอร์ตเวกเตอร์แมชชีน+SMOTE	M01	17	ซัพพอร์ตเวกเตอร์แมชชีน+PCA+SMOTE	M17
2	โครงข่ายประสาทเทียม+SMOTE	M02	18	โครงข่ายประสาทเทียม+PCA +SMOTE	M18
3	ต้นไม้ตัดสินใจ+SMOTE	M03	19	ต้นไม้ตัดสินใจ+PCA +SMOTE	M19
4	กลุ่มของต้นไม้ตัดสินใจ+SMOTE	M04	20	กลุ่มของต้นไม้ตัดสินใจ+PCA +SMOTE	M20
5	ซัพพอร์ตเวกเตอร์แมชชีน+SMOTE+Bagging	M05	21	ซัพพอร์ตเวกเตอร์แมชชีน+PCA +SMOTE+Bagging	M21
6	โครงข่ายประสาทเทียม+SMOTE+Bagging	M06	22	โครงข่ายประสาทเทียม+PCA +SMOTE+Bagging	M22
7	ต้นไม้ตัดสินใจ+SMOTE+Bagging	M07	23	ต้นไม้ตัดสินใจ+PCA +SMOTE+Bagging	M23
8	กลุ่มของต้นไม้ตัดสินใจ+SMOTE+Bagging	M08	24	กลุ่มของต้นไม้ตัดสินใจ+SMOTE+Bagging	M24
9	ซัพพอร์ตเวกเตอร์แมชชีน+Undersampling	M09	25	ซัพพอร์ตเวกเตอร์แมชชีน+PCA +Undersampling	M25
10	โครงข่ายประสาทเทียม+ Undersampling	M10	26	โครงข่ายประสาทเทียม+PCA + Undersampling	M26
11	ต้นไม้ตัดสินใจ+ Undersampling	M11	27	ต้นไม้ตัดสินใจ+PCA + Undersampling	M27
12	กลุ่มของต้นไม้ตัดสินใจ+ Undersampling	M12	28	กลุ่มของต้นไม้ตัดสินใจ+PCA + Undersampling	M28
13	ซัพพอร์ตเวกเตอร์แมชชีน+Undersampling+Bagging	M13	29	ซัพพอร์ตเวกเตอร์แมชชีน+Undersampling+ Bagging	M29
14	โครงข่ายประสาทเทียม+ Undersampling+Bagging	M14	30	โครงข่ายประสาทเทียม+ Undersampling+Bagging	M30
15	ต้นไม้ตัดสินใจ+ Undersampling+Bagging	M15	31	ต้นไม้ตัดสินใจ+ Undersampling+Bagging	M31
16	กลุ่มของต้นไม้ตัดสินใจ+ Undersampling+Bagging	M16	32	กลุ่มของต้นไม้ตัดสินใจ+ Undersampling+Bagging	M32

การวัดประสิทธิภาพของการจำแนกข้อมูลของนั้นจะแสดงให้เห็นถึงความน่าเชื่อถือของโมเดล ซึ่งจะแสดงผลด้วยเมตริกซ์วัดประสิทธิภาพ (Confusion Matrix) ดังตารางที่ 3

ตารางที่ 3แสดงเมตริกซ์วัดประสิทธิภาพสำหรับการจำแนกข้อมูล 2 กลุ่ม

	Prediction Positive Class	Prediction NegativeClass
Actual Positive Class	True Positive (TP)	False Negative (FN)
Actual Negative Class	False Positive (FP)	True Negative (TN)

เมตริกซ์วัดประสิทธิภาพแสดงผลสรุปของการประเมินความสามารถในการจำแนกประเภทข้อมูลซึ่งทดสอบด้วยข้อมูลทดสอบ จากตารางที่ 3ค่าแต่ละตัวมีความหมายดังต่อไปนี้

TP คือ จำนวนข้อมูลทดสอบที่อยู่ในคลาส Positive และโมเดลจำแนกได้ถูกต้องว่าเป็น Positive

FN คือ จำนวนข้อมูลทดสอบที่อยู่ในคลาส Positive และโมเดลจำแนกผิดว่าเป็น Negative

FP คือ จำนวนข้อมูลทดสอบที่อยู่ในคลาส Negative และโมเดลจำแนกผิดว่าเป็น Positive

TN คือ จำนวนข้อมูลทดสอบที่อยู่ในคลาส Negative และโมเดลจำแนกได้ถูกต้องว่าเป็น Negative

ค่า TP, FP, TN และ FN นำมาใช้ในการคำนวณมาตรวัดต่าง ๆ เพื่อประเมินประสิทธิภาพการจำแนกประเภทข้อมูลของโมเดล ซึ่งในงานวิจัยนี้มีการใช้มาตรวัดดังต่อไปนี้

Accuracyสำหรับประเมินประสิทธิภาพการจำแนกประเภทข้อมูลโดยรวมของทุกคลาสในโมเดล ดังสมการที่ 1

$$Accuracy = \left(\frac{TP + TN}{TP + FN + TP + FP}\right) 100 \quad (1)$$

Sensitivity จะเป็นการวัดความสามารถในการค้นหาข้อมูลที่อยู่ในคลาส Positiveดังสมการที่ 2

$$Sensivitity = \left(\frac{TP}{TP + FN}\right) 100 \quad (2)$$

Specificityจะเป็นการวัดความสามารถในการค้นหาข้อมูลที่อยู่ในคลาส Negative ดังสมการที่ 3

$$Specificity = \left(\frac{TN}{TP + FP}\right) 100 \quad (3)$$

4. ผลการวิจัย

สำหรับการทดลองข้อมูลที่ไม่สมดุล 6ชุดข้อมูลด้วยโมเดลที่นำเสนอไปแล้วนั้น ผู้วิจัยได้ทำการทดลองออกเป็น 3 ส่วนด้วยกันคือ การทดลองโมเดลเชิงเดี่ยวและเชิงกลุ่มกับขุดข้อมูลพื้นฐานที่ไม่สมดุลการทดลองโมเดลเชิงเดี่ยวและเชิงกลุ่มกับชุดข้อมูลที่ผ่านกระบวนการทำให้สมดุลและการทดลองโมเดลเชิงเดี่ยวและเชิงกลุ่มกับชุดข้อมูลที่ผ่านการคัดเลือกคุณลักษณะและการทำให้สมดุลผลที่ได้แสดงดังตารางที่ 4ตารางที่ 5และตารางที่6ตามลำดับ

ตารางที่ 4แสดงค่าความถูกต้องของการทดสอบข้อมูลที่ไม่สมดุลกับโมเดลเชิงเดี่ยวและเชิงกลุ่ม

โมเดล	Single				Bagging				Class
ชื่อชุดข้อมูล	SVM	NN	J48	RF	SVM	NN	J48	RF	
ecoli4	0.94	0.97	0.96	0.97	0.94	0.96	0.97	0.98	Accuracy : Ac
	0.00	0.75	0.70	0.75	0.00	0.25	0.70	0.70	Sensitivity (Minority Class) : Se
	1.00	0.98	0.98	0.99	1.00	1.00	0.98	0.99	Specificity (Majority Class) : Sp
glass-0*	0.92	0.90	0.90	0.88	0.92	0.22	0.91	0.92	Accuracy : Ac
	0.00	0.00	0.24	0.06	0.00	0.94	0.06	0.06	Sensitivity (Minority Class) : Se
	1.00	0.98	0.96	0.96	1.00	0.15	0.99	0.99	Specificity (Majority Class) : Sp

ตารางที่ 4แสดงค่าความถูกต้องของการทดสอบข้อมูลที่ไม่สมดุลกับโมเดลเชิงเดี่ยวและเชิงกลุ่ม (ต่อ)

โมเดล	Single				Bagging				Class
ชื่อชุดข้อมูล	SVM	NN	J48	RF	SVM	NN	J48	RF	
new-thyroid1	0.92	0.98	0.98	0.98	0.92	0.94	0.97	0.99	Accuracy : Ac
	0.49	0.94	0.94	0.91	0.49	0.77	0.86	0.91	Sensitivity (Minority Class) : Se
	1.00	0.99	0.99	0.99	1.00	0.97	0.99	1.00	Specificity (Majority Class) : Sp
vehicle0	0.77	0.97	0.93	0.96	0.77	0.77	0.96	0.97	Accuracy : Ac
	0.03	0.96	0.87	0.95	0.03	0.00	0.92	0.96	Sensitivity (Minority Class) : Se
	1.00	0.98	0.95	0.96	1.00	1.00	0.97	0.97	Specificity (Majority Class) : Sp
yeast-0*	0.90	0.93	0.92	0.93	0.90	0.90	0.93	0.94	Accuracy : Ac
	0.01	0.54	0.35	0.52	0.01	0.02	0.39	0.51	Sensitivity (Minority Class) : Se
	0.99	0.97	0.99	0.98	0.99	1.00	0.99	0.98	Specificity (Majority Class) : Sp
yeast1	0.72	0.76	0.75	0.74	0.72	0.71	0.77	0.77	Accuracy : Ac
	0.07	0.46	0.46	0.51	0.07	0.00	0.49	0.47	Sensitivity (Minority Class) : Se
	0.98	0.88	0.87	0.83	0.99	1.00	0.88	0.89	Specificity (Majority Class) : Sp

ตารางที่ 5แสดงค่าความถูกต้องของการทดสอบข้อมูลที่ผ่านการทำให้สมดุลกับโมเดลเชิงเดี่ยวและเชิงกลุ่ม

โมเดล	SMOTE				SMOTE+Bagging			
ชื่อชุดข้อมูล	M01	M02	M03	M04	M05	M06	M07	M08
ecoli4 : Ac	0.98	0.98	0.98	0.98	0.98	0.81	0.98	0.99
Se	1.00	0.99	0.98	0.99	0.99	0.66	0.99	0.99
Sp	0.96	0.98	0.97	0.98	0.97	0.95	0.98	0.99
glass-0*: Ac	0.63	0.94	0.94	0.96	0.66	0.56	0.94	0.95
Se	1.00	0.97	0.95	0.98	0.99	1.00	0.96	0.98
Sp	0.27	0.92	0.93	0.93	0.32	0.12	0.92	0.92
thyroid1: Ac	0.99	0.99	0.98	0.99	0.99	0.88	0.99	0.99
Se	1.00	0.99	0.99	0.99	1.00	0.96	0.99	0.99
Sp	0.99	0.98	0.99	0.99	0.98	0.79	0.99	0.99
vehicle0 : Ac	0.87	0.98	0.96	0.97	0.97	0.50	0.97	0.98
Se	1.00	0.99	0.97	0.99	0.99	0.01	0.98	0.99
Sp	0.75	0.97	0.95	0.95	0.94	0.99	0.95	0.96
yeast-0* : Ac	0.81	0.85	0..92	0.95	0.81	0.56	0.95	0.96
Se	0.72	0.82	0..92	0.95	0.74	0.13	0.94	0.95
Sp	0.89	0.88	0..92	0.95	0.89	0.99	0.95	0.96
yeast1: Ac	0.71	0.74	0.79	0.85	0.71	0.60	0.84	0.86
Se	0.78	0.78	0.79	0.90	0.75	0.33	0.86	0.88
Sp	0.65	0.71	0.79	0.80	0.67	0.86	0.82	0.85

ตารางที่ 5แสดงค่าความถูกต้องของการทดสอบข้อมูลที่ผ่านการทำให้สมดุลกับโมเดลเชิงเดี่ยวและเชิงกลุ่ม (ต่อ)

โมเดล	Undersampling				Undersampling +Bagging			
ชื่อชุดข้อมูล	M09	M10	M09	M10	M09	M10	M09	M10
ecoli4 : Ac	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
Se	1.00	0.95	1.00	0.95	1.00	0.95	1.00	0.95
Sp	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
glass-0*: Ac	0.44	0.56	0.44	0.56	0.44	0.56	0.44	0.56
Se	0.65	0.82	0.65	0.82	0.65	0.82	0.65	0.82
Sp	0.24	0.29	0.24	0.29	0.24	0.29	0.24	0.29
thyroid1: Ac	0.94	0.97	0.94	0.97	0.94	0.97	0.94	0.97
Se	0.89	0.97	0.89	0.97	0.89	0.97	0.89	0.97
Sp	1.00	0.97	1.00	0.97	1.00	0.97	1.00	0.97
vehicle0 : Ac	0.95	0.98	0.95	0.98	0.95	0.98	0.95	0.98
Se	1.00	0.99	1.00	0.99	1.00	0.99	1.00	0.99
Sp	0.89	0.97	0.89	0.97	0.89	0.97	0.89	0.97
yeast-0* : Ac	0.81	0.75	0.81	0.75	0.81	0.75	0.81	0.75
Se	0.74	0.69	0.74	0.69	0.74	0.69	0.74	0.69
Sp	0.88	0.82	0.88	0.82	0.88	0.82	0.88	0.82
yeast1: Ac	0.69	0.70	0.69	0.70	0.69	0.70	0.69	0.70
Se	0.73	0.75	0.73	0.75	0.73	0.75	0.73	0.75
Sp	0.65	0.65	0.65	0.65	0.65	0.65	0.65	0.65

ตารางที่ 6แสดงค่าความถูกต้องของการทดสอบข้อมูลที่ผ่านการคัดเลือกคุณลักษณะและการทำให้สมดุลกับโมเดลเชิงเดี่ยวและเชิงกลุ่ม

โมเดล	PCA+SMOTE				PCA+SMOTE+Bagging			
ชื่อชุดข้อมูล	M17	M18	M17	M18	M17	M18	M17	M18
ecoli4 : Ac	0.99	0.98	0.99	0.98	0.99	0.98	0.99	0.98
Se	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
Sp	0.99	0.98	0.99	0.98	0.99	0.98	0.99	0.98
glass-0*: Ac	0.78	0.85	0.78	0.85	0.78	0.85	0.78	0.85
Se	0.93	0.92	0.93	0.92	0.93	0.92	0.93	0.92
Sp	0.63	0.78	0.63	0.78	0.63	0.78	0.63	0.78
thyroid1: Ac	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
Se	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Sp	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99

ตารางที่ 6แสดงค่าความถูกต้องของการทดสอบข้อมูลที่ผ่านการคัดเลือกคุณลักษณะและ การทำให้สมดุลกับโมเดลเชิงเดี่ยวและเชิงกลุ่ม (ต่อ)

โมเดล	PCA+SMOTE				PCA+SMOTE+Bagging			
ชื่อชุดข้อมูล	M17	M18	M17	M18	M17	M18	M17	M18
vehicle0 : Ac	0.97	0.96	0.97	0.96	0.97	0.96	0.97	0.96
Se	0.99	0.98	0.99	0.98	0.99	0.98	0.99	0.98
Sp	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94
yeast-0* : Ac	0.95	0.99	0.95	0.99	0.95	0.99	0.95	0.99
Se	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
Sp	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
yeast1: Ac	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
Se	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
Sp	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99

ตารางที่ 6แสดงค่าความถูกต้องของการทดสอบข้อมูลที่ผ่านการคัดเลือกคุณลักษณะและ การทำให้สมดุลกับโมเดลเชิงเดี่ยวและเชิงกลุ่ม (ต่อ)

โมเดล	PCA+Undersampling				PCA+Undersampling +Bagging			
ชื่อชุดข้อมูล	M25	M26	M25	M26	M25	M26	M25	M26
ecoli4 : Ac	0.86	0.86	0.86	0.86	0.86	0.86	0.86	0.86
Se	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75
Sp	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
glass-0*: Ac	0.50	0.58	0.50	0.58	0.50	0.58	0.50	0.58
Se	0.43	0.43	0.43	0.43	0.43	0.43	0.43	0.43
Sp	0.60	0.80	0.60	0.80	0.60	0.80	0.60	0.80
thyroid1: Ac	0.92	0.96	0.92	0.96	0.92	0.96	0.92	0.96
Se	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Sp	0.82	0.91	0.82	0.91	0.82	0.91	0.82	0.91
vehicle0 : Ac	0.90	0.91	0.90	0.91	0.90	0.91	0.90	0.91
Se	0.94	0.95	0.94	0.95	0.94	0.95	0.94	0.95
Sp	0.86	0.86	0.86	0.86	0.86	0.86	0.86	0.86
yeast-0* : Ac	0.71	0.72	0.71	0.72	0.71	0.72	0.71	0.72
Se	0.76	0.67	0.76	0.67	0.76	0.67	0.76	0.67
Sp	0.66	0.76	0.66	0.76	0.66	0.76	0.66	0.76
yeast1: Ac	0.82	0.78	0.82	0.78	0.82	0.78	0.82	0.78
Se	0.74	0.72	0.74	0.72	0.74	0.72	0.74	0.72
Sp	0.93	0.86	0.93	0.86	0.93	0.86	0.93	0.86

จากตารางที่ 4พบว่า โมเดลทุกโมเดลจะให้ค่าความแม่นยำในการจำแนกคลาสที่ใกล้เคียงกัน เนื่องจากแต่ละโมเดลสามารถจำแนกคลาสที่มีจำนวนข้อมูลมากได้อย่างถูกต้อง แต่ในขณะเดียวกันก็จะไม่สามารถจำแนกคลาสที่มีจำนวนข้อมูลน้อยได้อย่างถูกต้อง ซึ่งจะเห็นได้จากชุดข้อมูล yeast-0* yeast1ที่ไม่มีโมเดลใดสามารถจำแนกข้อมูลคลาสส่วนน้อยได้อย่างถูกต้อง และชุดข้อมูล glass-0* ที่มีเพียงโมเดลโครงข่ายประสาทเทียมแบบเชิงกลุ่มเพียงโมเดลเดียวเท่านั้นที่สามารถจำแนกคลาสที่มีจำนวนข้อมูลน้อยได้อย่างถูกต้อง ในขณะที่โมเดลที่เหลือไม่สามารถจำแนกคลาสได้อย่างถูกต้อง

จากตารางที่ 5พบว่าทั้งโมเดลเชิงเดี่ยวและเชิงกลุ่มของกลุ่มของต้นไม้ตัดสินใจ คือ M03 M04 M07 M08 ที่ผ่านขั้นตอนการทำให้สมดุลด้วยเทคนิควิธี SMOTE นั้นให้ผลการจำแนกที่มีความแม่นยำสูงกับข้อมูลเกือบทุกชุดข้อมูลในขณะที่การทำให้สมดุลด้วยวิธีการสุ่มลดนั้นทั้งโมเดลเชิงเดี่ยวและโมเดลเชิงกลุ่มนั้นจะให้ความแม่นยำกับบางชุดข้อมูลเท่านั้น

จากตารางที่ 6พบว่า โมเดล M24 ซึ่งเป็นโมเดลเชิงกลุ่มของกลุ่มของต้นไม้ตัดสินใจที่มีการทำงานบนข้อมูลที่ผ่านขั้นตอนการคัดเลือกคุณลักษณะและการทำให้สมดุลด้วยเทคนิควิธี SMOTE นั้นให้ผลการจำแนกที่มีความแม่นยำสูงกับข้อมูลเกือบทุกชุดข้อมูล ยกเว้นชุดข้อมูล glass-0*ที่ M20 สามารถจำแนกข้อมูลส่วนน้อยได้แม่นยำกว่าที่ 0.96

จากตารางที่ 5และ 6นั้น เมื่อได้ทำการเปรียบเทียบประสิทธิภาพในการจำแนกของแต่ละโมเดลแล้วนั้น จะพบว่า โมเดลเชิงกลุ่มที่มีการทำงานบนข้อมูลที่ผ่านขั้นตอนการคัดเลือกคุณลักษณะและการทำให้สมดุลด้วยเทคนิควิธี SMOTE นั้นจะให้ค่าความแม่นยำที่สูงกว่าโมเดลกลุ่มอื่น ๆ ซึ่งโมเดลเชิงกลุ่มที่สามารถจำแนกกลุ่มข้อมูลที่ไม่สมดุลได้อย่างมีประสิทธิภาพคือ โมเดลกลุ่มของต้นไม้ตัดสินใจ

จากผลการทดลองที่ได้พบว่า แนวทางของการเตรียมข้อมูลก่อนการทำงานของโมเดลโดยเฉพาะในส่วนของการคัดเลือกคุณลักษณะที่เหมาะสมพร้อมทั้งการสุ่มเพิ่มจำนวนข้อมูลในกลุ่มที่มีจำนวนน้อยให้มีจำนวนคลาสที่ใกล้เคียงกับคลาสที่มีจำนวนมากนั้นจะช่วยเพิ่มประสิทธิภาพความแม่นยำในการจำแนกของโมเดลเชิงกลุ่มได้มากกว่าโมเดลเชิงเดี่ยว อันเนื่องมาจากการใช้เทคนิคการรวมกลุ่มนั้นสามารถช่วยในการป้องกันปัญหาความลำเอียง (Bias) ที่มักจะเกิดขึ้นจากการกำหนดกลุ่มของข้อมูลที่ใช้ในการเรียนรู้

5. สรุปผล

การวิจัยนี้เป็นการสร้างโมเดลที่มีประสิทธิภาพสำหรับใช้ในการจำแนกข้อมูลที่ไม่สมดุลซึ่งเป็นข้อมูลที่ส่งผลกระทบต่อประสิทธิภาพของการจำแนก ซึ่งในงานวิจัยนี้ได้มองเห็นถึงความสำคัญของขั้นตอนการเตรียมข้อมูลโดยได้นำเทคนิควิธีการคัดเลือกคุณลักษณะที่สำคัญของข้อมูลด้วยขั้นตอนวิธีของ PCA หลังจากนั้นทำการสุ่มเพิ่มจำนวนข้อมูลของคลาสที่มีจำนวนน้อยให้มีจำนวนคลาสที่ใกล้เคียงกับคลาสที่มีจำนวนมากด้วยเทคนิควิธี SMOTE ผลการวิจัยพบว่า เมื่อนำโมเดลเชิงกลุ่มของกลุ่มของต้นไม้ตัดสินใจมาใช้ร่วมกับข้อมูลที่ผ่านขั้นตอนการเตรียมข้อมูลด้วยวิธีที่กล่าวมาแล้วนั้น พบว่าเทคนิควิธีของโมเดลเชิงกลุ่มของต้นไม้ตัดสินใจให้ผลลัพธ์ที่ดีกว่าเทคนิควิธีของโมเดลเชิงเดี่ยว ทั้งนี้เพราะเทคนิคการรวมกลุ่มนั้นสามารถช่วยในการป้องกันปัญหาความโน้มเอียงของข้อมูลที่มักจะเกิดขึ้นได้ส่งผลให้โมเดลที่ได้มีประสิทธิภาพมากยิ่งขึ้น

6. เอกสารอ้างอิง

1 N.V. Chawla, N. Japkowicz and A. Kotcz, “Editorial: special issue on learning from imbalanced data sets,” ACM Sigkdd Explorations Newsletter, Vol.6, No.1, pp:1-6, 2004.

2 H. He and E.A. Garcia, “Learning from imbalanced data,” IEEE Transactions on Knowledge and Data Engineering, Vol.21, No.9, pp:1263–1284, 2009.

3 Y. Sun, A.K.C. Wong and M.S. Kamel, “Classification of imbalanced data: a review,” International Journal of Pattern Recognition and Artificial Intelligence, Vol.23, No.04, pp:687–719, 2009.

4 C. Elkan, “The foundations of cost–sensitive learning,” in: Proceedings of the 17th IEEE International Joint Conference on Artificial Intelligence (IJCAI’01),,,pp:973–978,2001.

5 I. Brown andM. Christophe, “An experimental comparison of classification algorithms for imbalanced credit scoring data sets,” Expert Systems with Applications, Vol.39, No.3, pp:3446-3453, 2012.

6 V. Lopez, A. Fernandez, J.G. Moreno-Torres, and F. Herrera, “Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics,” Expert Systems with Applications, Vol.39, No.7, pp:6585-6608, 2012.

7 S. Cateni, V. Colla, and M. Vannucci, “A method for resampling imbalanced datasets in binary classification tasks for real-world problems,” Neurocomputing, Vol.135,, pp:32-4,2014.

8 M. Galar, A. Fernandez, E. Barrenechea, and F. Herrera, “Eusboost: enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling,” Pattern Recognition, Vol.46, No.12, pp:3460-3471, 2013.

9 Keel Datasets , available from http://www.keel.es/datasets.php.

10 R Language, available from http://www.r-project.org/.

11 Python Anaconda, available from https://store.continuum.io/cshop/anaconda/

Ensemble Learning For Imbalanced Data Classification Problem

Pasapitch Chujai*, KittipongChomboon, PongsakornTeerarassamee, Nittaya Kerdprasop, Kittisak Kerdprasop

School of Computer Engineering, Institute of Engineering, Suranaree University of Technology, NakhornRatchasima 3000, Thailand

*Corresponding Author:pasapitchchujai@gmail.com

Abstract

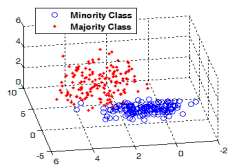
Imbalanced data is a kind of information that occurs in real life, such as medical diagnosis in which records of seriously ill patients outnumber by records of healthy ones. These imbalanced data affect the learning performance of algorithms in data mining. The boundary of decision in out of balance data chosen by most standard algorithms of machine learning tends to bias toward the majority class and hence misclassifies the minority class. Therefore, we present an approach for dealing with imbalanced data classification problem by applying the decision tree ensemble learning using both bagging and boosting techniques to build modelsthat compensate the misclassification with cost sensitive learning.In this research, we build the model templates from different characteristics of synthetic data.We have chosen an appropriate model template for the real data with different imbalanced rating and overlapping ratio. The results showed that the chosen model templatecan solve the imbalanced data classification problem efficiently. But there are some model templates that cannot classify correctly when imbalanced rate increases.

Keywords: Ensemble Learning, ImbalancedDatasets, Decision Tree Classification, Cost-Sensitive Learning, Visualization.

1. Introduction

Datamining[1]is a method that has been extensively used to retrieve the hidden knowledge from a large information repository. Data mining task has many categories depending on the purpose of application. The one of those categories is data classification that aims to learn patterns to make prediction about the class of some unknown data. Most standard algorithmsfor data classification can be applied very efficiently in terms of overall classification accuracy if data in each class are in equal proportion.However,these algorithms show poor learning performance when classifying the imbalanced data that have amount of instances in the group of interest less than those in the other groups[2].

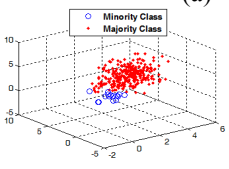
For example, we can demonstrate a comparison between classifying balanced data and unbalanced datawith300 instances and two classes. For the balanced dataset, the amounts of data in the two classes are equal; that is, 150 instances in each class. For the unbalanced dataset, there are 285 instances in the *class a*, whereas there areonly 15 instances in the *class b*. Take both datasets to be classified by decision tree induction. The results are shownin Fig. 1. Both datasets show good performance in terms of overall classification accuracy. When considering accuracy in each class, we found that the performance of classifying *class b*in the balanced data is more accurate than classifying *class b*in the imbalanced data. This classification accuracy drops drastically from 0.947 to 0.467.




```
   a   b   <-- classified as
 146   4 |   a = 1
   8 142 |   b = 2
```

Accuracy = 0.960
Pred. a = 0.973
Pred. b= 0.947

(a) Balanced data




```
   a   b   <-- classified as
 284   1 |   a = 1
   8   7 |   b = 2
```

Accuracy = 0.970
Pred. a = 0.996
Pred. b= 0.467

(b) Imbalanced data

Fig. 1.Comparisons of classification between the two datasets of balanced data and unbalanced data.

Thisexample indicates that usingimbalanced data in classification will affect the learning performance of algorithmsthattend to bias toward the group of majority and cause high misclassification rate over a group of minority.

The problem of classifying imbalanced data mentioned above has drawn attention from many researchers to propose various methods to solve this problem. The proposed methods focus on a more accurate classification over a minority group. Some important work thatproposed the methods mentioned above are as follows:

Brown and Mues[3] proposed the undersampling technique to deal five credit scoring imbalanced datasets.

Cateni et al.[4] proposed the oversampling and undersampling techniques to deal the benchmarks imbalanced datasets from the KEEL Repository.

Lopez et al.[5]studied the performance of classification with the hybrid techniques of SMOTE+ENN and SMOTE. They solved the problem by combining the techniques at data level approach and algorithm level approach and focusedon the cost-sensitive learning. They compared these techniques with their proposed hybrid techniques andperformed experiments with 66 datasets that were taken from the KEEL Repository.

Krawczyk et al.[6]proposed the decision tree ensemble data classification and cost-sensitive learning to deal with six binary benchmarks imbalanced datasets from the KEEL Repository. They proposed a technique to prune the decision tree using the novel algorithm and optimal $C_{minority}$ derived from the ROC analysis.They compared the proposed method with the other six methods. The result showed that their proposed method was efficient in some datasets.

Liao et al.[7]proposed the ensemble learning for binary classification. They used Support Vector Machine (SVM) for rebalancing data in the stage of preprocessing and then selected the featuresfor ensemble learning with Back-Propagation Neural Network (BPNN). The outputs from ensemble learning were taken to build new knowledge by using the rough set theory. They performed experiments with the listed electronics companies from 2005 to 2011: 63 financial crisiscorporations and 2680 non-financial crisis corporations.The result showed that their proposed method was more efficient than other methods.

We recognize the importanceof solvingimbalanced data classification problem with an effective method. Therefore,we present an approach for dealing with imbalanced datasets by applying the decision tree ensemble learning using both bagging and boosting techniques to build models and combine the compensation technique to handle misclassification with cost sensitive learning.

The rest of this research is organized as follows: Section 2 gives details of the background and relevant techniques. Section 3 presents details of our proposed method. The experimental results and analysis will be presented in Section 4. Finally, the research is concluded in Section 5.

2. Background

2.1 Imbalanced Data

Imbalanced data is the data that have the amount of instances in the group of interest much smaller than those in other classes[2].The group of data that has a larger number of instances is called the majority class or negative class, whereas the group of data that has a smallest number of instances is called the minority class or positive class[8].When classifying imbalanced data, the boundary of decision acceptable by standard algorithms tends to bias toward the majority class and misclassify in minority class as illustrated in Fig. 2.

Characteristics of imbalanced data that can influence the classification algorithms[9]are divided into three casesas follows:

(a) Imbalanced Ratio

Imbalanced data can be verified by the degree of imbalance, which represents the ratio between the number of data instances in majority class ($n_{majority}$) and the number of data instances in minority class ($n_{minority}$)[10]. The imbalanced ration can be defined by Eq. (1)

$$Imbalanced\ Ratio\ (IR) = \frac{n_{majority}}{n_{minority}} \quad (1)$$

(b) Lack of data

This problem occurs when the size of samples in the minority class is too small[9]. Because of small sample size will cause difficulty in finding the patterns.

(c) Overlapping ratio between classes

Overlap occurs when the data of each class has shared area. Overlap that occurs in conjunction with imbalanced data would result in the more complex situation for

classification[11].

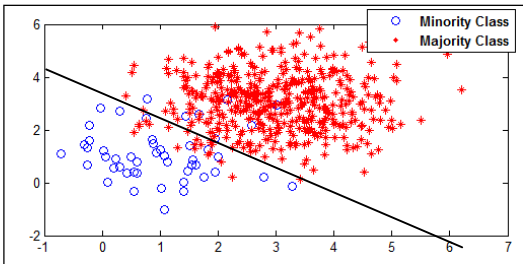


Fig. 2. Linear classification of imbalanced data whichbias towards majority class.

Maximum Fisher's Discriminant Ratio (F1) is one method that can be used for measurement the overlapping ratio.The F1 is defined by Eq. (2)

$$f_i = \frac{(\mu_{minority} - \mu_{majority})_i^2}{(\sigma^2_{minority} - \sigma^2_{majority})_i} \quad (2)$$

wheref_i is Fisher's Discriminant Ratio of feature *i*.

$\mu_{minority}$, $\mu_{majority}$are mean of minority class and majority class, respectively.

$\sigma^2_{minority}, \sigma^2_{majority}$ are variance of minority class and majority class, respectively.

The methods for solving imbalanced classification problem[5,12]can be divided into three approaches as follows:

(a) Data Level Approaches

This approach solves the problem in a pre-processing stage by rebalancingthe class distributionusing thesampling techniques: oversampling, undersampling, and a hybrid technique.

(b) Algorithm Level Approaches

This approach attempt to adapt existing algorithms by adjusting the parameters.

(c) Cost Sensitive Approaches

This approach uses both data level approaches by adding special cost to misclassification and algorithm level approaches by modification the algorithms to the possible classification that leads to less errors.

2.2 Ensemble Learning

The functions of single model have high classification performance but have a problem in terms of a fixed a set of parameters, which causes the bias. Reduction of such bias can be obtained through the ensemble learning.

The performance of ensemble learning depends on the precisionof classifiers. In ensemble classification learning, multiple classifiers are used to learn the original dataset

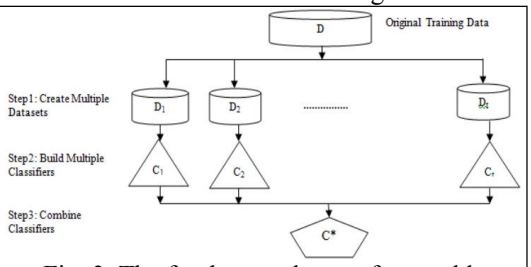


Fig. 3. The fundamental step of ensemble learning.

together. The results from learning will be combined and then used to classify the unknown data. The process of ensemble learning is given in Fig. 3[13].

Ensemble learning can be divided into three approaches as follows:

(a) Boosting method

Boosting method[14]is an ensemble classification such that each classifier has a weight which is derived from the precision of learning. The resultsmodels are used to predict the unknown data by the majority vote. The popular technique is AdaBoost.

(b) Bagging method

Bagging method[15]builds the models from the same learning algorithm but each algorithm learns from different instances. This method also uses majority vote for predictionof unknown data. The popular technique is Bag.

(c) Random subspace method

Random subspace method or Attribute Bagging[16]learns from the same dataset and then performs sampling without replacement over the features.The method also uses majority vote for the prediction of unknown data.

2.3 Cost Sensitive Learning

Cost-sensitive learning takes into account the cost of misclassification. Penalties of misclassification will be built as a cost matrix as shown in Table 1.

From Table 1, let *C(i,j)* be the cost of predicting the sample in class *i* as class *j*. *C(0,0)* and *C(1,1)* are the cost of correct classification which is set to be equal to 0. *C(0,1)* is the cost of misclassifying of majority class to be minority, and the cost is set to 1.*C(1,0)* is the cost of misclassifying of minority class to be majority. The cost is $C_{minority}$, which can be adjusted according to the specific algorithm.

The most important issue for solving the imbalanced data classification problem is recognizing correctly the positive instance (minority class) rather than the negative

instance (majority class). Therefore, the cost of misclassifying of minority class musthigher than the cost of misclassifying of majority class (*C(1,0)>C(0,1)*).

Table 1. Cost matrix C for binary classification.

	Predicted Class	
Actual Class	Majority Class (0)	Minority Class (1)
Majority Class (0)	*C(0,0)* = 0	*C(0,1)* = $C_{majority}$ = 1
Minority Class (1)	*C(1,0)* = $C_{minority}$	*C(1,1)* = 0

3. Methodology

In this research, our main objective is to find the classification model efficiently for solving the imbalanced data classification problem with highaccuracy and efficiency. Our concern is to improve the process of imbalanced classification at different imbalanced ratio and overlapping ratio between classes. We apply the decision tree ensemble learning using both bagging and boosting techniques to build models and compensate the misclassification with cost sensitive learning by building cost matrix and then take the values from cost matrix to adjust the parameters of ensemble learning.

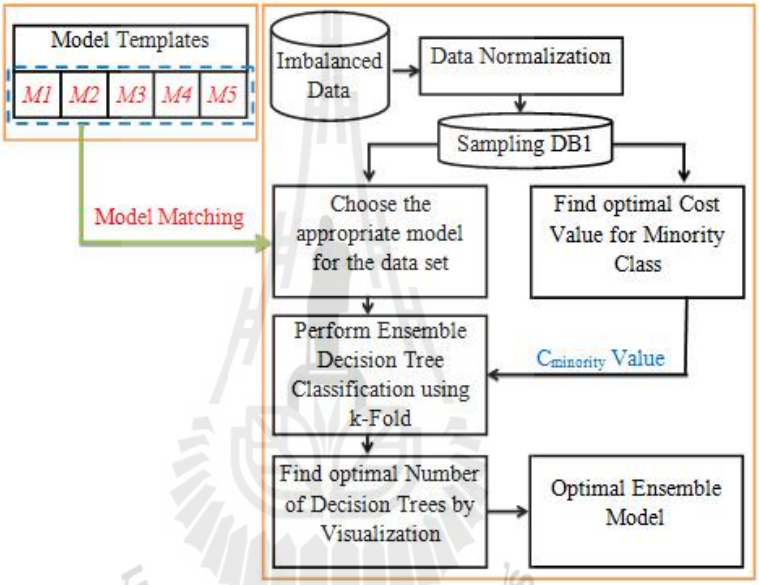


Fig. 4. The framework of steps for building theimbalanced classification models

We also find the optimal number of trees by visualization. The framework and the steps are shown in Fig. 4, we can explain in detail of each stage as follows:

(a) Building the Model Templates

This stage is for building the model templates with the following two steps:

1. Generate the numeric synthetic data, 1000 instances with normal distribution, slightlyoverlapped with imbalance ratiosof 10%.
2. Build model templates from the synthetic data using five algorithms: AdaBoostM1, Bag, TotalBoost, LogitBoost and RUSBoost. Name of the model templates are M1, M2, M3, M4 and M5, respectively.

(b) Building the System Model

This stage is for building the system model with the following six steps:

1. Normalization the features to zero mean and set standard deviation equal to 1.
2. Sampling data by using stratified sampling to draw samples from the imbalanced data with several of imbalance ratios. We call the sampled data DB1.
3. Find optimal cost value for minority class by generate cost matrix for misclassification cost. We are setting IR of DB1 to be the $C_{minority}$. Other values in the cost matrix will be determined by a constant: $C_{majority}$equals 1, C(0,0) and C(1,1) equals 0.
4. Model matching by analyzes the characteristics of DB1 and then choose the appropriate modelfrom the model templates.
5. Building the imbalanced classification model using the model templates and the best value of $C_{minority}$. We initialize the number of decision trees to be equal to 200 and test the model by k-fold cross validation, k=5.
6. Find optimal number of decision trees, weemploy visualization to reduce the number of decision trees obtained from

ensembles learning to achieve a number of suitable decision trees. The visualization will show the test of classification error for each decision tree and we will choose the top-10 decision trees with theminimum error.

(c) Test Performance of the Ensemble Model

The optimal ensemble model will be tested to evaluate its performance by the evaluation measure.

4. Experimental Results

4.1 Datasets

For our experiment, the proposed ensemble models have been developed and applied for binary classification on the following datasets:

(a) Synthetic datasets

The synthetic datasets have been created usingMatlab. We created five datasets whichhave a slight overlap with an initial imbalanced ratio of 10%, two classes and three features.Details of the synthetic datasets and the optimal model templates are given in Table 2and one of the synthetic dataset is shown in Fig. 5.

(b) KEEL datasets

For our experiments, we have taken six binary imbalanced datasets from the KEEL Repository[17]. Details of the employed datasets are given in Table 3.

Table 2.Details of the synthetic datasets and optimal model templates.

Data Set	Characteristics	Model Template
D1	M=[0 0 0; 2 2 2]S1=[0.2 0.0 0.0; 0.0 0.1 0.0; 0.0 0.0 0.0] S2=[1.1 0.3 0.2; 0.3 1.1 0.7; 0.2 0.7 1.2]	AdaBoostM1
D2	M=[0.65 -0.09 -2.59; -0.05 0.01 0.19]S1=[1.09,0.03,-0.13; 0.03,0.73, 0; -0.13,0, 0.05] S2=[0.96,-0.02,0.57; -0.02,1.02,0; 0.57,0, 0.55]	Bag
D3	M=[0.18 0.45 1.07; -0.02 -0.05 -0.12]S1=[1.87,1.46,2.42;1.46,8.58,-0.81;2.42,-0.81,6.33] S2=[0.90,0.26,0.31;0.26, 0.11,0.06;0.31,0.06,0.25]	TotalBoost
D4	M=[3 1 3;0 0 0]S1=[0.77,-1.17,-0.98;-1.17,5.82,5.17;-0.98,5.17,4.6] S2=[0.23,-0.21,-0.09;-0.21,0.49,0.2; -0.09,0.2,0.14]	LogitBoost
D5	M=[0 0 0; 1 1 1]S1=[0.8,-0.2,-0.2;-0.2,1.8,-0.1;-0.2,-0.1,0.3] S2=[0.8,0.1,-0.01; 0.1, 0.8,0.2; -0.01,0.2, 0.8]	RUSBoost
M is mean of minority class and majority class. S1 and S2 are variance of minority class and majority class, respectively.		

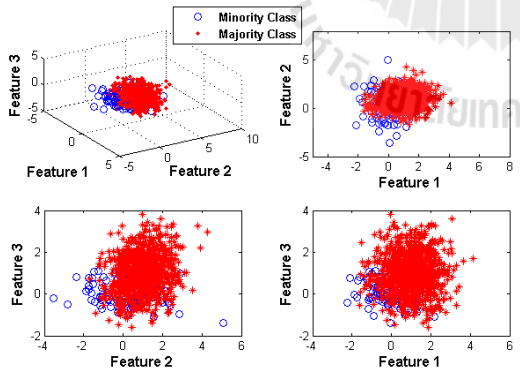


Fig. 5.An overview of the synthetic dataset: D5.

Table 3. Details of the datasets used in the experiments.

Name	Attribute (R/I/N)*	Instance	Majority Class	Minority Class	IR
page-blocks	10 (4/6/0)	5472	4913	559	8.79
pima	8 (8/0/0)	768	500	268	1.87
segment	18 (18/0/0)	2308	1979	329	6.02
shuttle	9 (0/9/0)	1829	1706	123	13.87
vehicle	18 (0/18/0)	846	628	218	2.88
yeast	8 (8/0/0)	1484	1321	163	8.10

(R/I/N)* : Real/Integer/Nominal Valued

4.2 Evaluation Metrics

In order to evaluate the effectiveness of the proposed method we used confusion matrix to show the accuracy of the classification and the reliability of the model. Detail of confusion matrix is given in Table 4.

From Table 4, row of the matrix shows the number of actual instances of each class and column shows the number of predicted of each class. It is divided into the following four cases:

Table 4. Confusion Matrix for Binary classification.

	Positive Prediction	Negative Prediction
Actual Positive Class	True Positive (TP)	False Negative (FN)
Actual Negative Class	False Positive (FP)	True Negative (TN)

TP: number of instances that are correctly classified as positive class.

TN:number of instances that are correctly classified as negative class.

FP: number of instances that are negative class incorrectly classified as positive class.

FN: number of instances that are positive class incorrectly classified as negative class.

(a) Sensitivity or True Positive Rate (TPRate) or Recall

This measure shows the ability of the model to classify positive class, which is defined as the ratio between the number of correctly classified positive class and the total number of the actual positive class. The sensitivity is defined by Eq. (3)

$$Recall = Sensitivity = \frac{TP}{(TP + FN)} \quad (3)$$

(b) Specificity or True Negative Rate (TNRate)

This measure shows the precision of the classification model in classifying negative class, which is defined as the ratio between the number of correctly classified negative class and the total number of the actual negative class. The specificity is defined by Eq. (4)

$$Specificity = TNrate = \frac{TN}{(TN + FP)} \quad (4)$$

(c) Accuracy

This measure shows evaluation of the overall performance of all classes in the classification of a model.The accuracy is defined by Eq. (5)

$$Accuracy = \left(\frac{TP + TN}{TP + FN + TP + FP}\right) * 100 \quad (5)$$

4.3 Results and Analyses

In this section, we present the results from the evaluation of our proposed model using six binary benchmarks imbalanced datasets. We perform stratified sampling to draw samples from imbalanced datasets at different imbalanced ratios and then analyze the characteristics of data to find the suitable model from the model templates. The results of the suitable model templates are given in Table 5.

We derive the $C_{minority}$ from imbalanced ratio (IR) and then use it in the step of building the ensemble model. The initial number of decision trees is 200 and then reduce this number to get the optimal number of decision trees using visualization.

Table 5. Optimal model templates for benchmark datasets.

Dataset	Model Template
page-blocks	TotalBoost
pima	RUSBoost
segment	AdaBoostM1, LogitBoostandTotalBoost
shuttle	Bag and LogitBoost
vehicle	LogitBoost
yeast	RUSBoost

Fig. 6 shown the decrease in the number of decision trees of the yeast dataset adjusted with the smallest error rate. The optimal number of decision trees equal 30.

From the obtained optimal model templates shown in Table 5, we performed the experiments over the benchmark datasets with these models. We also performed the experiments with the rest of model templates as a base line for comparisons. We experiments with the three imbalanced ratios: 1:10, 1:25 and 1:50. Each imbalanced ratio is equal $C_{minority}$. The performance of such models in terms of sensitivity (SE), specificity (SP) and accuracy (Acc) are given in Table 6, Table 7 and Table 8, respectively.

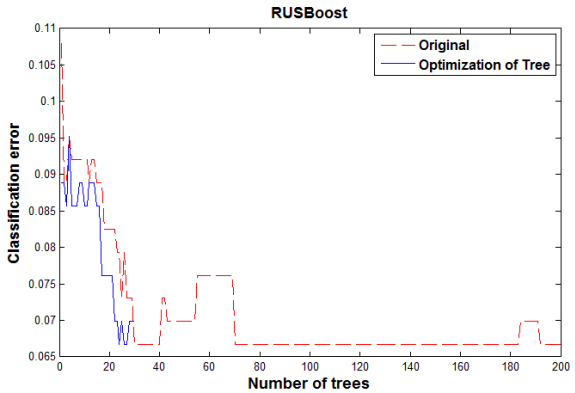


Fig. 6.Shows the error of classification of yeast dataset with varying number of decision trees.

Table 6. Classification results for benchmark datasets with imbalanced ratios 1:10, the best value of $C_{minority}$ is 10.

Model	AdaBoostM1			Bag			TotalBoost			LogitBoost			RUSBoost		
Dataset	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc
page-blocks	66.67	99.39	96.30	40.83	99.83	94.25	***82.50***	***98.00***	***96.54***	81.67	99.04	97.40	79.17	95.74	94.17
pima	20.00	95.33	88.48	6.67	100.00	91.52	33.33	94.00	88.48	26.67	98.67	92.12	***80.00***	***76.00***	***76.36***
segment	***97.50***	***99.50***	***99.32***	57.50	100.00	96.14	95.00	99.25	98.86	95.00	99.50	99.09	85.00	99.50	98.18
shuttle	0.00	100.00	90.24	***100.00***	***100.00***	***100.00***	0.00	100.00	90.24	***100.00***	***100.00***	***100.00***	80.00	100.00	98.05
vehicle	70.00	99.00	96.36	0.00	100.00	90.91	70.00	98.00	95.45	75.00	99.00	96.82	***85.00***	***90.00***	***89.55***
yeast	66.67	98.95	95.87	3.33	100.00	90.79	86.67	94.39	93.65	76.67	98.60	96.51	***93.33***	***93.68***	***93.65***

Table 7. Classification results for benchmark datasets with imbalanced ratios 1:25, the best value of $C_{minority}$ is 25.

Model	AdaBoostM1			Bag			TotalBoost			LogitBoost			RUSBoost		
Dataset	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc
page-blocks	54.17	99.93	98.17	17.50	99.93	96.76	***80.00***	***99.27***	***98.53***	77.50	99.53	98.69	79.17	96.77	96.09
pima	13.33	99.47	96.15	0.00	100.00	96.15	40.00	96.00	93.85	26.67	97.33	94.62	***80.00***	***80.80***	***80.77***
segment	***97.50***	***100.00***	***99.90***	15.00	100.00	96.73	***97.50***	***99.90***	***99.81***	***97.50***	***100.00***	***99.90***	75.00	99.80	98.85
shuttle	0.00	100.00	96.15	75.00	100.00	99.04	0.00	100.00	96.15	***100.00***	***100.00***	***100.00***	***100.00***	***99.70***	***99.71***
vehicle	50.00	99.80	97.88	5.00	100.00	96.35	65.00	88.40	87.50	65.00	99.60	98.27	***85.00***	***90.60***	***90.38***
yeast	36.67	99.47	97.05	3.33	100.00	96.28	80.00	93.47	92.95	66.67	99.60	98.33	***86.67***	***94.27***	***93.97***

Table 8. Classification results for benchmark datasets with imbalanced ratios 1:50, the best value of $C_{minority}$ is 50.

Model	AdaBoostM1			Bag			TotalBoost			LogitBoost			RUSBoost		
Dataset	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc
page-blocks	35.83	99.98	98.73	12.50	100.00	98.28	***73.33***	***99.62***	***99.10***	70.83	99.82	99.25	***73.33***	***97.20***	***96.73***
pima	0.00	100.00	98.04	0.00	100.00	98.04	26.67	99.07	97.65	13.33	100.00	98.30	***73.33***	***73.73***	***73.73***
segment	***92.50***	***100.00***	***99.85***	15.00	99.85	98.19	***92.50***	***99.95***	***99.80***	***92.50***	***100.00***	***99.85***	82.50	99.55	99.22
shuttle	0.00	100.00	98.04	35.00	100.00	98.73	0.00	100.00	98.04	***100.00***	***100.00***	***100.00***	***100.00***	***99.85***	***99.85***

Table 8. Classification results for benchmark datasets with imbalanced ratios 1:50, the best value of $C_{minority}$ is 50 (con't).

Model	AdaBoostM1			Bag			TotalBoost			LogitBoost			RUSBoost		
Dataset	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc
vehicle	40.00	99.90	98.73	5.00	100.00	98.14	60.00	99.50	98.73	50.00	99.90	98.92	***90.00***	***89.70***	***89.71***
yeast	3.33	99.93	98.04	0.00	100.00	98.04	80.00	86.07	85.95	50.00	99.60	98.63	***83.33***	***95.93***	***95.69***

Table 9. Comparison between the best results from our proposed and themethod with proposed byKrawczyk et al.[6]

IR	1:10				1:25				1:50			
Dataset	Krawczyk et al.[6]		Proposed		Krawczyk et al.[6]		Proposed		Krawczyk et al.[6]		Proposed	
	SE	SP	SE	SP	SE	SP		SE	SP	SE	SP	SE
page-blocks	***82.95***	***80.23***	82.50	98.00	70.76	81.23	***80.00***	***99.27***	***88.04***	***87.64***	73.33	99.62
pima	***85.23***	***97.10***	80.00	76.00	77.13	94.50	***80.00***	***80.80***	71.43	92.67	***73.33***	***73.73***
segment	75.24	82.94	***97.50***	***99.50***	72.89	90.11	***97.50***	***100.00***	70.02	93.43	***92.50***	***100.00***
shuttle	92.31	89.23	***100.00***	***100.00***	86.28	91.05	***100.00***	***100.00***	75.98	92.98	***100.00***	***100.00***
vehicle	***88.23***	***89.23***	85.00	90.00	77.67	93.45	***85.00***	***90.60***	67.37	90.98	***90.00***	***89.70***
yeast	70.25	97.23	***93.33***	***93.68***	67.78	98.11	***86.67***	***94.27***	60.34	96.22	***83.33***	***95.93***

The results of the experiments in Table 6, Table 7 and Table 8, show that there are chosen model templates could classify efficiently, such as RUSBoost model for yeast and pima datasets and TotalBoost model for page-blocks dataset. For vehicle dataset, we choose LogitBoost model templates which are inappropriate, RUSBoost model could classify efficiently. The rest of datasets demonstrate that part of model template can classify efficiently and some model templates without chosen; RUSBoost model for shuttle and page-blocks datasets; are able to classify efficiently.

The model template is disabling to classify efficiently when increasing the imbalanced ratio. The examples are Bag model for segment dataset can classify correctly at imbalanced ratio at 1:10, and AdaBoostM1, TotalBoost and LogitBoost model for segment dataset can classify correctly at imbalanced ratio at 1:25. For increasing imbalanced ratios of shuttle dataset, our proposed method could improve the classification of imbalanced data significantly with chosen model template: LogitBoost model.

As for comparison with the results proposed by Krawczyk et al.[6] is shown in Table 9. The proposed method shows the performance is quite satisfactory; especially the imbalanced datasets are imbalance ratio of 1:25 and 1:50. For an imbalance ratio of 1:10, the proposed method was statistically better on three of imbalanced datasets, while the competition method was statistically better on the rest of imbalanced datasets. For an imbalance ratio of 1:25, the proposed method was statistically better on all of six imbalanced datasets. For an imbalance ratio of 1:50, the proposed method was statistically better on five of imbalanced datasets, while the competition method was statistically better on the rest of imbalanced datasets.

5. Conclusions

Imbalanced data classification isa significant challenge for standard algorithms of machine learning. In this research, we propose the method to deal with imbalanced data

classification problems with the main focus to improve recognition of the minority class. We combine the cost-sensitive learning with ensemble decision tree classification using bagging and boosting techniques. The numbers of decision trees are also decreased to an optimal number of decision trees by visualization. We create normally distributed synthetic data with binary classes, three features and 1000 instances. Then, we build the model templates from five algorithms: AdaBoostM1, Bag, TotalBoost, LogitBoost and RUSBoost. We analyze the standard datasets and selected the best model template by considering the overlapping between classes, mean and standard variation of minority class and majority class with closed to model templates. The experiments show that the overlapping ratio between classes has an effect to the performance of proposed model. If datasets has overlapping between classes, the model can classify correctly at imbalanced ratio not over 25. The appropriate ensemble technique is boosting such as RUSBoost, LogitBoost, TotalBoost, and AdaBoostM1. For Bag model, it classifies correctly at imbalanced ratio not over 10. The best model is RUSBoost that can classify an imbalanced data which have overlapping between classes and high imbalanced ratio.

Acknowledgment

Financial support for Pasapitch Chujai has been provided by a scholarship from Office of the Higher Education Commission, Thailand.

References

(1) JiaweiHan, and MichelineKamber: “Data mining: concepts and techniques” Amsterdam; Boston: Elsevier, 2006
(2) Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer: “SMOTE: synthetic minority over-sampling technique”,Journal of Artificial Intelligence Research, Vol. 16, pp. 321–357, 2002
(3) Iain Brown, andChristopheMues: “An experimental comparison of classification algorithms for imbalanced credit scoring data sets”, Expert Systems with Applications, Vol. 39, No. 3, pp. 3446-3453, 2012
(4) Silvia Cateni, ValentinaColla, and Marco Vannucci: “A method for resampling imbalanced datasets in binary classification tasks for real-world problems”, Neurocomputing, Vol. 135, pp.32-41, 2014
(5) Victoria López, Alberto Fernández, Jose G. Moreno-Torres, and Francisco Herrera: “Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. Open problems on intrinsic data characteristics”, Expert Systems with Applications, Vol. 39, No. 7, pp. 6585-6608, 2012
(6) BartoszKrawczyk, Michal Wozniak, and Gerald Schaefer: “Cost-sensitive decision tree ensembles for effective imbalanced classification”, Applied Soft Computing, Vol. 14, pp. 554-562, 2014
(7) Jui-Jung Liao, Ching-HuiShih, Tai-Feng Chen, and Ming-Fu Hsu: “An ensemble-based model for two-class imbalanced financial problem”,Economic Modeling, Vol. 37, pp. 175-183, 2014
(8) M.A.H. Farquad, and Indranil Bose: “Preprocessing unbalanced data using support vector machine”, Decision Support Systems, Vol. 53, No. 1, pp. 226-233, 2012
(9) Son Lam Phung, AbdesselamBouzerdoum, and Giang Hoang Nguyen: “Learning pattern classification tasks with imbalanced data sets”, In P . Yin (Eds.), Pattern recognition, Vukovar, Croatia: In-Teh,pp. 193-208, 2009
(10) Albert Orriols-Puig, and Ester Bernadó-Mansilla: “Evolutionary rule-based systems for imbalanced data sets”, Soft Computing,Vol. 13, No. 3, pp. 213-225, 2009
(11) Julian Luengo, Alberto Fernández, and Francisco Herrera: “Addressing data-complexity for imbalanced data-sets: A preliminary study on the use of preprocessing for c4.5”, In Intelligent Systems Design and Applications, 2009. ISDA'09. Ninth International Conference on IEEE,pp. 523-528, 2009
(12) MikelGalar, Alberto Fernandez, EdurneBarrenechea, HumbertoBustince, and Francisco Herrera: “A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches”, IEEE transactions on systems, man, and cybernetics—part C: Applications and reviews, Vol. 42, No. 4, 2012
(13) Meenakshi A. Thalor, and S. T. Patil: “Review of Ensemble Based Classification Algorithms for Nonstationary and Imbalanced Data”,IOSR Journal of Computer Engineering (IOSR-JCE), Vol.

16, No. 6, pp. 103-107, 2014
(14) YoavFreund, and Robert E. Schapire: "Experiments with a new boosting algorithm", Proceedings 13th International Conference on Machine Learning, Vol. 96, pp. 148-156, 1996
(15) Leo Breiman: "Bagging predictors", Machine learning, Vol. 24, No. 2, pp. 123-140, 1996
(16) Robert Bryll, Ricardo Gutierrez-Osuna, and Francis Quek: "Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets", Pattern Recognition, Vol. 30, No. 6, pp. 1291–1302, 2003
(17) Keel Datasets, available from http://www.keel.es/datasets.php.

ประวัติผู้เขียน

นางสาวภาสพิชญ์ ชูใจ เกิดเมื่อวันที่ 3 มิถุนายน 2518 ที่อำเภอฉวาง จังหวัดนครศรีธรรมราช เริ่มเข้าศึกษาระดับชั้นประถมศึกษาปีที่ 1 ถึง 6 ที่โรงเรียนบ้านโคกมะขาม อำเภอฉวาง จังหวัดนครศรีธรรมราช จากนั้นศึกษาต่อในระดับมัธยมตอนต้นและตอนปลายที่โรงเรียนฉวางรัชดาภิเษก อำเภอฉวาง จังหวัดนครศรีธรรมราช ในปีการศึกษา 2537 ได้เข้าศึกษาต่อระดับปริญญาตรีในสาขาวิชาวิทยาการคอมพิวเตอร์ ที่มหาวิทยาลัยรามคำแหง และสำเร็จการศึกษาในปีการศึกษา 2543 หลังจากสำเร็จการศึกษาได้เข้าทำงานในบริษัทวิสดอมอินฟอร์เมชั่น จำกัด ตำแหน่งโปรแกรมเมอร์เป็นเวลา 5 ปีในช่วงเวลานั้นได้เข้าศึกษาต่อระดับปริญญาโทสาขาวิชาคอมพิวเตอร์และเทคโนโลยีสารสนเทศ มหาวิทยาลัยพระจอมเกล้าธนบุรีเมื่อปีพ.ศ. 2545 และสำเร็จการศึกษาในปีการศึกษา 2547 หลังจากสำเร็จการศึกษาได้เข้าทำงานในบริษัทบางกอกมอเตอร์เวอคส์ จำกัด ตำแหน่งโปรแกรมเมอร์ ในปีพ.ศ. 2549 ทำงานที่บริษัทโอโร่เจมส์แมนูแฟคเจอริ่งจำกัด ตำแหน่งโปรแกรมเมอร์ ต่อจากนั้น ปีพ.ศ. 2552 ได้ทำงานทางสายงานวิชาการ ตำแหน่งอาจารย์ วิทยาลัยเทคโนโลยีอรรถวิทย์พณิชยการเป็นเวลา 1 ปี ในขณะเดียวกันก็เป็นที่ปรึกษาและดูแลโปรแกรมระบบ JMax ให้กับบริษัทโอโร่เจมส์แมนูแฟคเจอริ่งจำกัดจนถึงปัจจุบัน และในปีการศึกษา 2555 ได้เศึกษาต่อระดับปริญญาเอกในสาขาวิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ในระหว่างการศึกษาได้รับความอนุเคราะห์อย่างดีจากอาจารย์ที่ปรึกษาและอาจารย์ประจำวิชาต่าง ๆ และได้รับความไว้วางใจให้เป็นผู้ช่วยสอนปฏิบัติการในรายวิชา Knowledge Discovery and Data Miningหลังจากนั้นได้รับการตีพิมพ์เผยแพร่บทความวิจัย รายละเอียดดังภาคผนวก ข